Mathematical Foundations for a Compositional Account of the Bayesian Brain

Toby St Clere Smithe

St Edmund Hall

University of Oxford

A thesis submitted for the degree of

Doctor of Philosophy

Trinity 2023

Acknowledgements

Abstract

This dissertation reports some first steps towards a compositional account of active inference and the Bayesian brain. Specifically, we use the tools of contemporary applied category theory to supply *functorial semantics* for approximate inference. To do so, we define on the 'syntactic' side the new notion of *Bayesian lens* and show that Bayesian updating composes according to the compositional lens pattern. Using Bayesian lenses, and inspired by compositional game theory, we define fibrations of *statistical games* and classify various problems of statistical inference as corresponding sections: the chain rule of the relative entropy is formalized as a strict section, while maximum likelihood estimation and the free energy give lax sections. In the process, we introduce a new notion of 'copy-composition'.

On the 'semantic' side, we present a new formalization of general open dynamical systems (particularly: deterministic, stochastic, and random; and discrete- and continuous-time) as certain coalgebras of polynomial functors, which we show collect into monoidal opindexed categories (or, alternatively, into algebras for multicategories of generalized polynomial functors). We use these opindexed categories to define monoidal bicategories of *cilia*: dynamical systems which control lenses, and which supply the target for our functorial semantics. Accordingly, we construct functors which explain the bidirectional compositional structure of predictive coding neural circuits under the free energy principle, thereby giving a formal mathematical underpinning to the bidirectionality observed in the cortex. Along the way, we explain how to compose rate-coded neural circuits using an algebra for a multicategory of *linear circuit diagrams*, showing subsequently that this is subsumed by lenses and polynomial functors.

Because category theory is unfamiliar to many computational neuroscientists and cognitive scientists, we have made a particular effort to give clear, detailed, and approachable expositions of all the category-theoretic structures and results of which we make use. We hope that this dissertation will prove helpful in establishing a new "well-typed" science of life and mind, and in facilitating interdisciplinary communication.

Contents

1. Introduction

The work of which this dissertation is a report began as a project to understand the brain's "cognitive map", its internal representation of the structure of the world. Little of that work is reported here, for it rapidly became clear at the outset that there was no coherent framework in which such a project should most profitably be undertaken. This is not to say that no progress on understanding the cognitive map can be made, a claim which would be easily contradicted by the evidence. Rather, each research group has its own language and its own research questions, and it is not always evident how to translate concepts from one group, or even one moment in time, faithfully to another; what translation is done is performed at best highly informally.

If the aim of science[1] is to tell just-so stories, or if the aim is only to answer one's own research questions in isolation, then this state of affairs may be perfectly satisfactory. But the brain and the behaviours that brains produce are so marvellous and so complex, and the implications of a finer understanding so monumental, that one cannot but hope that science could do better. Of course, of late, science has not been doing better, with disciplines as socially important as psychology [201] and medicine [23, 135, 188] and machine learning [134, 149] struck by crises of reproducibility. At the same time, as broadband internet has spread across the globe, the sheer amount of output produced by scientists and other researchers has ballooned, contributing to the impossibility of verification and the aforementioned translational difficulties, at least if one desires to do other than simply following the herd. In some sense, although scientists all now speak English, science still lacks a *lingua franca*, or at least a sufficiently precise one.

As luck would have it, while mainstream science has been suffering from this loss of faith, the first phrases of a potentially adequate precise new language have begun to spread, with the coalescence of a new community of researchers in *applied category theory*[2]. One part of the present

---

[1]Or indeed, "if the aim of scientists", as science itself may not have volition of its own.

[2]The first major interdisciplinary meeting of applied category theorists (or at least the first meeting sufficiently confident to take *Applied Category Theory* as its name) was held in 2018 in Leiden, although categorical methods have for some time been used in computer science [210] and physics [16], and especially at their nexus [2, 68, 69]. More sporadically, category theory had shown up elsewhere, such as in biology [86, 220], network theory [93–95], game theory [3, 89, 119], cognitive science [37, 85, 183, 209] and linguistics [67, 70, 130], and in 2014 a workshop was held at Dagstuhl bringing together some of these researchers [4], in what was to be a precursor to the Applied Category

difficulty of scientific translation is that each research group has not only its own language, but also its own perspective; and another part of the difficulty is that these languages and perspectives are not well connected, with the English language a very lossy medium through which to make these connections. Fortunately, the language of category theory—being a mathematical rather than a natural language—resolves both of these difficulties.

Category theory is the mathematics of pattern, composition, connection, and interaction; its concepts are as crisp and clear as the water of a mountain pool; its simplicity lends it great power. Categories describe how objects can be constructed from parts, and such compositional descriptions extend to categories themselves: as a result, the language of category theory is 'homoiconic', and can be used to translate constructions between contexts. One is able to abstract away from irrelevant details, and show precisely how structures give rise to phenomena; and by choosing the abstractions carefully, it becomes possible to see that, sometimes, important constructions are 'universal', able to be performed in any relevant context. As a result, category theory resolves both problems of scientific translation indicated above: concepts expressed categorically are inevitably expressed in context, and not in isolation; and these contexts are naturally interconnected as if by a categorical web (with the connections also expressed categorically). Moreover, not being English, categorical definitions tend to be extremely concise and information-dense; and since the basic concepts of category theory are themselves simple, concepts so expressed are not biased by geography or geopolitics.

From the middle of the 20th century, the concepts of category theory began to revolutionize much of mathematics[3], and applied category theorists such as the present author believe that the time is nigh for this revolution to spread throughout the sciences and alleviate some of their struggles. Just as the internet constitutes physical infrastructure that fundamentally accelerates human communications, we expect category theory to constitute conceptual infrastructure of similar catalytic consequence. This thesis is a contribution to building this infrastructure, in the specific domain of computational neuroscience and the general domain of (what was once, and will be again, called) cybernetics[4]. In particular, we show that a prominent theory of brain function—*predictive*

Theory meetings; many of those researchers still work in this new interdisciplinary field.

[3]The basic concepts of category theory were originally written down by Eilenberg and Mac Lane in order to formalize processes of translation, and so clarify structures in the ways indicated in the main text above, in the field of algebraic topology. This occurred at the end of the first half of the 20th century, in 1945 [87]. The ideas soon spread beyond algebraic topology, gathering momentum rapidly from the 1950s, in which Cartan defined the concept of sheaf [56, 57] and Grothendieck reconceived the foundations of algebraic geometry [121]. By the mid-1960s, and especially through the work of Lawvere on logic [165] and set theory [166], it was clear that category theory would be able to supply supple but sturdy new foundations for all of mathematics.

[4]Owing to its affinity for pattern and abstraction, it is hard to do interesting domain-specific work in category theory without there being at least some more general results to be found, and indeed this is the case here: what began as

coding—has a clear compositional structure, that explains the bidirectional circuitry observed in the brain [21], and that renders precise connections to the structure of statistical and machine learning systems [187, 221, 278], as well as to the structure of much larger scale adaptive systems traditionally modelled by economic game theory [119].

Predictive coding models were originally developed in the neuroscience of vision to explain observations that neural activity might decrease as signals became less surprising [216] (rather than increase as signals became more 'preferred'), as well as to explain the robustness of sensory processing to noise [246] and as a source of metabolic efficiency [32][5]. The typical form of these models involves a neuron or neural ensemble representing the system's current prediction of (or expectation about) its input, alongside another neuron or ensemble representing the difference between this prediction and the actual input (*i.e.*, representing the prediction error). We can think of the former ensemble as directed from within the brain towards the sensory interface (such as the retina), and the latter ensemble as carrying information from the world into the brain: this is the aforementioned bidirectionality.

Another important observation about visual processing in the brain is that its circuitry seems to be roughly hierarchical [179], with regions of cortex further from the retina being involved in increasingly abstract representation [212]. Given a model of predictive coding at the level of a single circuit, accompanied by models of how sensory circuits are coupled (and their representations transformed), a natural next step is to construct hierarchical predictive coding models, in an attempt to extend the benefits of the single circuit to a whole system; and indeed such hierarchical circuits were prominently proposed in the literature [104, 216].

This hierarchical structure is a hint of compositionality, and thus a sign that a categorical approach may be helpful and enlightening. This impression is strengthened when one considers a particularly influential class of predictive coding models, obtained in the context of the "free energy principle" [100, 104, 107], where the underlying equations themselves exhibit a form of compositionality which is (more or less explicitly) used to obtain the hierarchical models[6]. Despite this hint of compositionality, the equations of motion for these hierarchical systems are typically derived from scratch each time [21, 48, 76, 108, 148, 264, 265], a redundant effort that would not be

---

a project in theoretical neuroscience swiftly became a study of adaptive and cybernetic systems more broadly, of which the brain is of course the prime exemplar.

[5]If the prediction is good, then communicating the difference between prediction and actuality can be done much more efficiently than transmitting the whole incoming signal, which would contain much redundant information. This is the principle underlying most data compression algorithms.

[6]That is to say, the dynamics of each level of hierarchy i are governed by a quantity $\mathcal{F}_i$, and the dynamics of two adjacent levels i and $i + 1$ are governed by $\mathcal{F}_i + \mathcal{F}_{i+1}$; see Buckley et al. [48, Eq. 72].

required had a compositional formalism such as category theory been used from the start. This thesis supplies such a categorical formalism and exemplifies it with hierarchical predictive coding under the free energy principle.

The "free energy" framework not only underpins a modern understanding of predictive coding, but has more broadly been proposed as a unified theory of brain function [100], and latterly of all adaptive or living systems [38, 102, 159, 204]. In the neuroscientific context, it constitutes a theory of the *Bayesian brain*, by which most or all brain function can be understood as implementing approximate Bayesian inference [160]; in the more broadly biological (or even metaphysical) contexts, this claim is generalized to state that all life can be understood in this way. However, despite these claims to universality, these proposals have to date been quite informally specified, leading to confusion [28, 103] and charges of unfalsifiability [38, 71, 281]. As we will see, category theory has a rich formal vocabulary for precisely describing universal constructions, and so not only does a categorical formulation of the free energy framework promise to clarify the current confusions, but it may be expected also to shed light on its potential universality. In particular, as we discuss in Chapter 8, we will be able to make precise the questions of whether any dynamical system of the appropriate type can universally be seen as performing approximate inference (in our language, "playing a statistical game"), and of whether any cybernetic system (such as an economic game player) can be expressed as an active inference system.

The notion of *active inference* is closely related to the free energy framework: an active inference model of a system describes both the processes by which it updates its internal states on the basis of incoming signals, and the processes by which it chooses how to act, using approximate Bayesian inference. In this thesis, we do not get as far as a completely general formulation of active inference, but we hope that our development of *statistical games* and their "dynamical semantics" in *approximate inference doctrines* will provide a useful starting point for such a formulation, and in our final chapter (8) we sketch how we might expect this formulation to go. Because active inference models, and the free energy framework more broadly, are descriptions of systems that are 'open' to an environment, interacting with it, and therefore situated "in context", they are particularly suited to a category-theoretic reformulation. Likewise, Bayesianism and the free energy framework lend themselves to a subjectivist metaphysics [102, 114, 115], which is itself in alignment with the unavoidable perspective-taking of categorical models, and which is not dissimilar from the emerging 'biosemiotic' reconceptualization of biological information-processing [20]. As we have indicated, categorical tools help us to draw connections between concepts, and we see our efforts

as a contribution to this endeavour.

It is through these connections that we hope eventually to make contact again with the cognitive map. As noted above, the state of the art is fragmented, but there exist current models that are expressed in the language of approximate (variational) inference [279], models expressed in the language of reinforcement learning [257], and models that attempt to combine the two [185]. We will see throughout the thesis that reinforcement learning (and its cousin, game theory) is closely related to approximate inference, and so we expect that the foundations developed here, along with the extensions proposed in §8.1.3, will help us unify these accounts. The key observation that we expect to drive such a development is that learning a cognitive map (alternatively, learning a "world model") means internalizing a representation of the structure of the environment; and comparing and translating structures is category theory's forte.

Of course, even if the theory that we develop is sufficient to unify these computational-phenomenological models, this is not to say it will satisfy all neuroscientists, many of which may be expected to desire more biologically detailed models. In the contemporary undergraduate neuroscience curriculum, one is taught informally to relate models at a high 'computational' level to lower level models concerned with biological 'implementation', following Marr's "three levels of explanation" [179]. As we discuss in §2.2.3, this story is a shadow of the categorical notion of *functorial semantics*, by which structures are translated precisely between contexts formalized as categories. Although we concentrate on the more abstract computational level in this thesis, our discussion of functorial semantics foreshadows the introduction of formal algebraic tools for building biologically plausible neural circuit models (§3.3).

Our treatment of cognitive and neural systems is not the first to adopt categorical methods, but we do believe that it is the first to do so in a comprehensively integrated and wide-ranging way, taking functorial semantics seriously. Categorical concepts have been variously proposed in biology as early as 1958 [220], and in cognitive science (with one eye toward the brain) since at least 1987 [84, 85]; more recently, category theory has been used to study classic cognitive-science concepts such as systematicity [209]. While inspirational, these studies do not make the most of the translational power of categories, using only some concepts or methods in isolation. Moreover, by working almost purely categorically, these works were invariably rather abstract, and did not make direct contact with the tools and concepts of mainstream mathematical science. As a result, they did not have the unifying impact or adoption that we hope the new wave of applied category theoretical developments to have.

Our primary motivation in writing this thesis is to lay the groundwork for *well-typed* cognitive science and computational neuroscience. 'Types' are what render categorical concepts so precise, and what allow categorical models to be so cleanly compositional: two systems can only "plug together" if their interface types match. Because every concept in category theory has a type (*i.e.*, every object is an object of some category), categorical thinking is forced to be very clear. As we will sketch in §2.3.4, the "type theories" (or "internal languages") of categories can be very richly structured, but still the requirement to express concepts with types is necessarily burdensome. But this burden is only the burden of thinking clearly: if one is not able to supply a detailed type, one can resort to abstraction. And, to avoid the violence of declaring some object to be identified as of some type[7], it is necessary to understand the relationships between types; fortunately, as we will soon make clear, and as we have attempted to emphasize, category theory is fundamentally the mathematics of relationship.

Contemporary science is unavoidably computational, and the notion of 'type' that we invoke here is closely related to (though not identical with) the informal notion of type that is used in computer programming. Just as one of the strategies adopted to overcome the crises of modern science that we invoked at the opening of this introduction is the making available of the code and data that underlie scientific studies, we can envisage a near future in which accompanying these is a formal specification of the types of the concepts that each study is about[8]. Some work along these lines has already begun, particularly with the development of the Algebraic Julia ecosystem [122].

The free energy framework, like the structurally adjacent framework of compositional game theory, has a strong flavour of teleology (that follows directly from its mathematics): systems act in order to make their predictions come true. We therefore hope that, although we do not quite get as far as a full compositional theory of active inference, the contributions reported in this dissertation may in some small way help to make this particular prediction (of a well-typed science) come true, and thereby help to overcome some of the aforenoted crises of scientific faith—as well as to shed light not only on the form and function of 'Bayesian' brains, but also other complex adaptive systems, such as the whole scientific community itself.

---

[7]A perspective for which we must thank Brendan Fong.

[8]One might think of this specification as akin to a scientifically elaborated version of the notion of *header file* in programming languages such as C or C++: these files specify the types of functions and data structures, typically without instantiating these types with detailed implementations. We can thus think of category theory as a very rich metaprogramming language for the mathematical sciences (and this analogy goes quite far, as categorical proofs are typically 'constructive' and hence correspond to computable functions, as we also sketch in §2.3.4).

1.1. Overview of the dissertation

Category theory being quite alien to most researchers in computational neuroscience (and the cognitive sciences more broadly), we begin the work of this dissertation in Chapter 2 with a comprehensive review of the concepts and results needed to understand our mathematical contributions. Using three hopefully familiar examples, we introduce categories as contrapuntal to graphs, which are more familiar to scientists, but which lack important features of categories such as composition and, somehow, dynamism. We then explain how enriched categories allow us to "connect the connections" of categories, and attach extra data to them, and we exemplify these concepts with the 2-category of categories, functors, and natural transformations—as well as a more formal discussion of functorial 'translation' and semantics. The remainder of Chapter 2 is dedicated to introducing the remaining key concepts of basic category theory: universal constructions, and the Yoneda Lemma (categories' fundamental theorem). All of these ideas are very well known to category theorists.

In Chapter 3, we begin to reapproach neural modelling, and more generally the 'algebraic' modelling of the structure of interacting systems. We explain how 'monoidal' categories allow us to consider processes "in parallel" (as well as just sequentially), and how this gives us a formal account of the concept of 'parameterized' system. We then change the perspective a little, and introduce our first piece of original work: an account of how to connect neural circuits into larger-scale systems, using 'multicategorical' algebra. The remainder of the chapter is dedicated to developing the theory of such algebra to the point needed later in the thesis, ending with the introduction of *polynomial functors* which will supply a rich syntax for the interaction of systems, as well as a language in which to express their dynamical semantics.

Chapter 4 presents our first main result, that Bayesian updating composes according to the categorical 'lens' pattern. This result is abstractly stated, and so applies to whichever compositional model of probability one might be interested in—but because we are later interested in concrete models, we spend much of the chapter recapitulating compositional probability theory using the tools introduced in Chapters 2 and 3 and instantiating it in discrete and continuous settings. We also introduce and contextualize the lens pattern, in order to define our new notion of *Bayesian lens*, which provides a mathematical formalization of the bidirectionality of predictive coding circuits.

Our main aim in this thesis is to formalize predictive coding through functorial semantics, and Bayesian lenses will provide an important part of the 'syntax' of statistical models that we need. But the Bayesian lenses that satisfy the main result of Chapter 4 are 'exact', while natural systems are

inherently approximate. In order to measure the performance of such approximate systems, Chapter 5 introduces our next new notion, the concept of *statistical game*, which attaches loss functions to lenses. These statistical games collect into a categorical structure known as a fibration (a kind of categorified fibre bundle), and we can use the sections of this fibration to classify well-behaved systems of approximate inference into *loss models*. These loss models include well-known quantities such as the relative entropy, (maximum) likelihood, the free energy, and the Laplace approximation of the latter. However, in order to make this classification work, we first introduce a new kind of categorical composition, which we call *copy-composition*, and which seems to cleave the basic process of composition in categories of stochastic channels, which typically proceeds first by copying and then by marginalization ('discarding').

Having developed the syntactic side of predictive coding, we turn in Chapter 6 to the semantics, which is found in a new abstract formalization of the concept of *open dynamical system*. We make much use here of the language of polynomial functors: these will represent the interfaces of interacting systems, and the dynamical systems themselves will be defined as particular classes of morphisms of polynomials. We extend the traditional notion of polynomial functor to a setting which allows for non-determinism, and thereby obtain new categories of open Markov process and random dynamical system, both in discrete and continuous time. We then synthesize these developments with the algebraic structures of Chapter 3, to define monoidal bicategories of 'hierarchical' cybernetic systems that we call *cilia*, as they control lenses.

Connecting these pieces together, Chapter 7 presents our functorial formalization of predictive coding, using a new notion of *approximate inference doctrine*, by which statistical models are translated into dynamical systems. This formalizes the process by which research in active inference turns the abstract specification of a "generative model" into a dynamical system that can be simulated and whose behaviours can then be compared with experimentally observed data. We explain how this functorial process is decomposed into stages, and then exhibit them in two ways: first, with the basic 'Laplacian' form of predictive coding; and then by introducing 'Hebbian' plasticity.

Finally, Chapter 8 reviews the prospects for future work, from the mathematics of the cognitive map (a programme that we call *compositional cognitive cartography*), to the composition of multi-agent systems and ecosystems and the connections with compositional game theory, categorical cybernetics, and categorical systems theory. We close with some speculation on a new mathematics of life, along with associated developments of fundamental theory.

1.2. Contributions

The main individual contribution of this thesis is the formalization of models of predictive coding circuits as functorial semantics, and the associated development and exemplification of fibrations of statistical games, as well as the introduction of Bayesian lenses and the proof that Bayesian updates compose optically. We believe our presentation of general open dynamical systems as certain polynomial coalgebras also to be novel, along with the concept of *cilia* and their associated monoidal bicategories. The categories of statistical games (and of Bayesian lenses) supply the syntax, and the monoidal bicategories of cilia the semantics, for our functorial treatment of predictive coding, and hence the basis for our compositional active inference framework. Each of these structures is to our knowledge new, although of course inspired by much work that has gone before, and by interactions with the beneficent community of researchers of which this author finds himself a member.

Each of these strands of work has in some way been exhibited through publication, principally as refereed presentations at the conference on Applied Category Theory (ACT) in 2020 [251], 2021 [252], and 2022 [254] (each published in the conference proceedings); but also in preliminary form at the NeurIPS 2019 *Context and Compositionality* workshop [247], through a number of more informal invited talks (*e.g.* [249]), as one main theme of a full-day workshop at the 2022 Cognitive Science Society conference [12], and our ongoing series of preprints on compositional active inference [250, 253]. Our work on Bayesian lenses, in collaboration with Dylan Braithwaite and Jules Hedges [42][9], has been accepted for publication at MFCS 2023; and we are presently preparing for journal publication an account of our compositional framework for predictive coding aimed explicitly at computational neuroscientists.

Besides these specific novel contributions, we hope that this dissertation contributes to a renaissance of cognitive and computational (neuro)science through the adoption of categorical methods; it is for this reason that we have been so diligent in our exposition of the basic theory. We hope that this exposition proves itself a useful contribution for interested researchers, and that its cognitive-neuroscientific framing is sufficiently novel to be interesting.

Some work performed during the author's DPhil studies is not included in this dissertation. In particular, there has unfortunately not been the scope to include our simulation results on a fragment of the circuitry underlying the cognitive map—a study on the development of place and head-direction cells, which was published as [255]—although this did motivate our algebra of

---

[9]See Remark 4.3.1 for the scholarly history.

rate-coded neural circuits (§3.3), which is to the best of our knowledge novel (though much inspired by earlier work on wiring-diagram algebras [237, 282]). We have also not exhibited our work on Bayesian optics (as an alternative to Bayesian lenses) [248], as this would require a digression through some unnecessarily complicated theory; and we have not presented in detail the examples of "polynomial life" presented at ACT 2021 [252].

A first draft of this thesis was produced in December 2022, at which point the author intended to submit it. However, shortly before submission, the author realized that the then-current treatment of statistical games could be much improved. This led to the present fibrational account, and the new notion of loss model (which formalizes the chain rule of the relative entropy), but which also demanded a corresponding revision of the treatment of predictive coding. At the cost of some higher-categorical machinery, we believe these changes amount to a substantial improvement, worth the delay in submission. The new account of statistical games has been accepted as a proceedings paper at ACT 2023.

2. Basic category theory for computational and cognitive (neuro)scientists

This chapter constitutes a comprehensive review of the concepts and results from basic category theory that scaffold the rest of the thesis, written for the computational neuroscientist or cognitive scientist who has noticed the 'network' structure of complex systems like the brain and who wonders how this structure relates to the systems' function. Category theory gives us a mathematical framework in which precise answers to such questions can be formulated, and reveals the interconnectedness of scientific ideas. After introducing the notions of category and diagram (§2.1), we swiftly introduce the notions of enriched category, functor, and adjunction (§2.2), with which we can translate and compare mathematical concepts. We then explain how category theory formalizes pattern as well as translation, using the concept of universal construction (§2.3), which we exemplify with many common and important patterns. Finally, we introduce the fundamental theorem of category theory, the Yoneda Lemma, which tells us that to understand a thing is to see it from all perspectives (§2.4).

Category theory is well established in the foundations of mathematics, but not yet explicitly in the foundations of science. As a result, although the only slightly original part of this chapter is its presentation, we have given proofs of most results and plentiful examples, in order to familiarize the reader with thinking categorically.

2.1. Categories, graphs, and networks

We begin by motivating the use of category theory by considering what is missing from a purely graph-theoretic understanding of complex computational systems. Later in the thesis, we will see how each of the diagrams depicted below can be formalized categorically, incorporating all the scientifically salient information into coherent mathematical objects.

2.1.1. Three examples

2.1.1.1. Neural circuits: dynamical networks of neurons

In computational and theoretical neuroscience, it is not unusual to encounter diagrams depicting proposed architectures for neural circuits, such as on the left or right below:
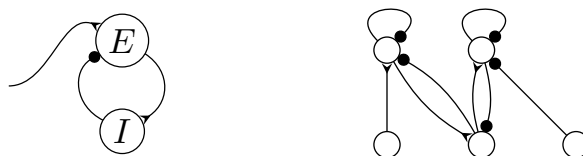
On the left, we have depicted a standard "excitatory-inhibitory circuit" motif, in which one neuron or ensemble of neurons E receives input from an external source as well as from a counterposed inhibitory circuit I which itself is driven solely by E. On the right, we have reproduced a figure depicting a "predictive coding" circuit from Bogacz [33], and we see that the E-I circuit is indeed motivic, being recapitulated twice: we could say that the predictive coding circuit is *composed* from interconnected E-I motifs, in a sense similarly to the composition of the E-I circuit from the subnetworks E and I of neurons.

Both circuits have evident graphical structure — the nodes are the white circles, and the edges the black wires between them — but of course there is more to neural circuits than these graphs: not only do graphs so defined omit the decorations on the wires (indicating whether a connection is excitatory or inhibitory), but they miss perhaps the more important detail, that these are circuits *of dynamical systems*, which have their own rich structure and behaviours. Moreover, mere graphs miss the aforementioned *compositionality* of neural circuits: we can fill in the white circles with neurons or ensembles or other circuits and we can wire circuits together, and at the end of doing so we have another 'composite' neural circuit.

Working only with graphs means we have to treat the decorations, the dynamics, and the compositionality informally, or at least in some other data structure, thereby increasing the overhead of this accounting.

2.1.1.2. Bayesian networks: belief and dependence

In computational statistics, one often begins by constructing a model of the causal dependence between events, which can then be interrogated for the purposes of inference or belief-updating. Such models are typically graphical, with representations as shown below; the nodes are again the

circles, and the dashed edge implies the repetition of the depicted motif:

On the left, the graph represents a model of an event with two possible antecedents; on the right, a set of events (or an event, repeated) with a recurrent cause. Although these graphical models — otherwise known as *Bayesian networks* — may encode useful information about causal structure, in themselves they do not encode the information about *how* events are caused; this is data that must be accounted for separately. And once again, mere graphs are unlike causality in that they are non-compositional: the structure does not explain how, given the causal dependence of B on A and A' and of C on B, one might model the dependence of C on A.

2.1.1.3. Computations: sets and functions

In a similar way, pure computations — in the sense of transformations between sets of data — are often depicted graphically:
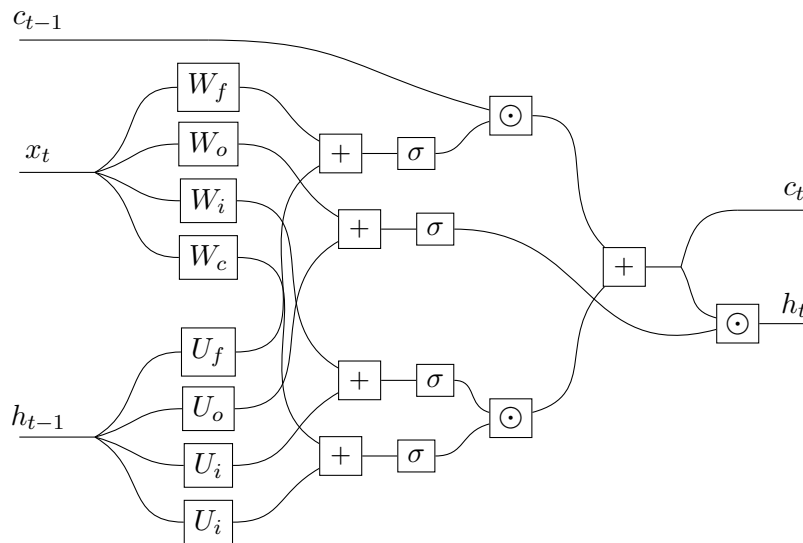
Here, we have depicted a single 'cell' from a long short-term memory network [132]: a function that ingests three variables (c_{t-1}, an internal state; x_t, an external input; and h_{t-1}, an internal 'memory'), and emits two (c_t, a new internal state; and h_t, an updated memory). This function is itself composed from other functions, depicted above as boxes. (One typically takes the variables c_t, x_t, h_t as vectors of given dimension for all t, so that the domain and codomain of the function are products of vector spaces; the boxes W_i and U_i represent matrices which act on these vectors; the boxes $+$ and $\odot$ denote elementwise sum and product; the box σ represents the elementwise

application of a logisitic function; and the splitting of wires represents the copying of data.) The nodes of the graph in this instance are the functions (boxes), and the edges encode the flow of information. Once more, however, a purely graphical model does not account for the compositional structure of the computation: we could fill in the boxes with other graphs (representing elaborations of the computations implied), and we could adjoin another such diagram beside and connect the wires where the types match. To account for this compositionality — here and in the examples above — we will need to add something to the structure: we need to move from graphs to categories.

2.1.2. From graphs to categories

A category is a directed graph in which edges can be composed: whenever the target of an edge f is the source of another edge g, then there must be a composite edge denoted $g \circ f$ whose source is the source of f and whose target is the target of g, as in the following diagram.
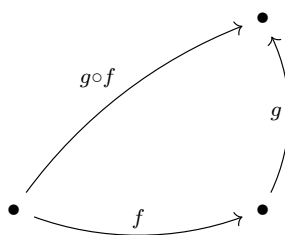
This composition rule incorporates into the structure a way to allow systems with compatible interfaces to connect to each other, and for the resulting composite system also to be a system of the same 'type'; but as we will see, it has some other important consequences. Firstly, every ('small') category has an underlying directed graph: but because of the composition rule, this underlying graph typically has more edges than the graphs of the examples above, in order to account for the existence of composites. Secondly, it is the edges, which in a categorical context we will call *morphisms*, that compose: the nodes, which we will call *objects*, represent something like the 'interfaces' at which composition is possible. This means that we cannot just interpret a circuit diagram "as a category", whose objects are ensembles of neurons and whose morphisms are their axons: as we will see in §3.3, we need to do something a bit more sophisticated.

Before we get to that, however, we must first define categories precisely. We will take a graphical approach, with a view to interpreting the above examples categorically, starting with the diagram demonstrating the composition of $g \circ f$: how should we interpret this in a category? To answer this question, we first need to specify exactly what we mean by 'graph'.

Definition 2.1.1. A *directed graph* $\mathcal{G}$ is a set $\mathcal{G}_0$ of *nodes* along with a set $\mathcal{G}(a, b)$ of *edges* from a to b for each pair $a, b : \mathcal{G}_0$ of nodes. We will sometimes write $\mathcal{G}_1$ to denote the disjoint union

of the sets of edges, $\mathcal{G}_1 := \sum_{a,b} \mathcal{G}(a,b)$. If $e : \mathcal{G}(a,b)$ is an edge from a to b, we will write this as $e : a \to b$ and call a its *source* or *domain* and b its *target* or *codomain*. This assignment of domain and codomain induces a pair of functions, $\mathrm{dom}, \mathrm{cod} : \mathcal{G}_1 \to \mathcal{G}_0$ respectively, such that for $e : a \to b$ we have $\mathrm{dom}(e) = a$ and $\mathrm{cod}(e) = b$.

A category is a graph whose edges can be 'associatively' composed together, and where every node has a special edge from itself to itself called its 'identity'.

Definition 2.1.2. A *(small) category* $\mathcal{C}$ is a directed graph whose nodes are each assigned a corresponding *identity* edge and whose edges are equipped with a *composition* operation $\circ$ that is *associative* and *unital* with respect to the identities. In the context of categories, we call the nodes $\mathcal{C}_0$ the *objects* or *0-cells*, and the edges $\mathcal{C}_1$ the *morphisms* or *1-cells*.

Identities are assigned by a function $\mathrm{id} : \mathcal{C}_0 \to \mathcal{C}_1$ satisfying $\mathrm{dom}(\mathrm{id}_a) = a = \mathrm{cod}(\mathrm{id}_a)$ for every object a. The composition operation is a family of functions $\circ_{a,b,c} : \mathcal{C}(b,c) \times \mathcal{C}(a,b) \to \mathcal{C}(a,c)$ for each triple of objects a, b, c. The notation $\mathcal{C}(a,b)$ indicates the set of all morphisms $a \to b$, for each pair of objects a and b; we call this set the *hom set* from a to b.

Given morphisms $f : a \to b$ and $g : b \to c$, their composite $a \xrightarrow{f} b \xrightarrow{g} c$ is written $g \circ f$, which we can read as "g after f".

Associativity means that $h \circ (g \circ f) = (h \circ g) \circ f$, and so we can omit the parentheses to write $h \circ g \circ f$ without ambiguity. Unitality means that, for every morphism $f : a \to b$, we have $\mathrm{id}_b \circ f = f = f \circ \mathrm{id}_a$.

Remark 2.1.3. We say *small* category to mean that both the collection of objects $\mathcal{C}_0$ and the collection of morphisms $\mathcal{C}_1$ is a true set, rather than a proper class. We will say a category is *locally small* if, for every pair a, b of objects in $\mathcal{C}$, the hom set $\mathcal{C}(a,b)$ is a set (rather than a proper class); this allows for the collection of objects still to be a proper class, while letting us avoid "size issues" such as Russell's paradox in the course of normal reasoning.

More precisely, we can fix a 'universe' of sets, of size assumed to be smaller than a hypothesized (and typically inaccessible) cardinal $\aleph_i$. Then we say that a category is locally small with respect to $\aleph_i$ if every hom set is within this universe, or small if both $\mathcal{C}_0$ and $\mathcal{C}_1$ are. We say that a category is *large* if it is not small, but note that the 'set' of objects or morphisms of a large category may still be a 'set', just in a larger universe: a universe whose sets are of cardinality at most $\aleph_{i+1} > \aleph_i$.

In the remainder of this thesis, we will typically assume categories to be locally small with respect to a given (but unspecified) universe.

Our first example of a category is in some sense the foundation of basic category theory, and supplies a classic illustration of size issues.

Example 2.1.4. The category $\mathbf{Set}$ has sets as objects and functions as morphisms. The identity on a set A is the identity function $\mathsf{id}_A : A \to A : a \mapsto a$. Composition of morphisms in $\mathbf{Set}$ is function composition: given $f : A \to B$ and $g : B \to C$, their composite is the function $g \circ f : A \to C$ defined for each $a : A$ by $(g \circ f)(a) = g(f(a))$; it is easy to check that function composition is associative.

Note that $\mathbf{Set}$ is a large category: the set $\mathbf{Set}_0$ of all sets of at most size $\aleph_i$ must live in a larger universe.

Not all categories are large, of course. Some are quite small, as the following examples demonstrate.

Example 2.1.5. There is a category with only two objects 0 and 1 and four morphisms: the identities $\mathsf{id}_0 : 0 \to 0$ and $\mathsf{id}_1 : 1 \to 1$, and two non-identity morphisms $s, t : 0 \to 1$, as in the following diagram:

$$0 \underset{t}{\overset{s}{\rightrightarrows}} 1$$
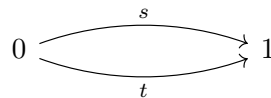
When depicting categories graphically, we often omit identity morphisms as they are implied by the objects.

Example 2.1.6. There is a category, denoted $\mathbf{1}$, with a single object $*$ and a single morphism, its identity.

Example 2.1.7. The natural numbers $\mathbb{N}$ form the morphisms of another category with a single object $*$: here, composition is addition and the identity morphism $\mathsf{id}_* : * \to *$ is the number 0. Since addition is associative and unital with respect to 0, this is a well-defined category.

Since a category is a directed graph equipped with a composition operation, we can 'forget' the latter to recover the underlying graph on its own.

Proposition 2.1.8. Given a category $\mathcal{C}$, we can obtain a directed graph $(\mathcal{C}_0, \mathcal{C}_1)$ by keeping the objects $\mathcal{C}_0$ and morphisms $\mathcal{C}_1$ and forgetting the composition and identity functions.

Proof. Take the objects to be the nodes and the morphisms to be the edges. $\qquad\square$

However, in the absence of other data, obtaining a category from a given graph is a little more laborious, as we must ensure the existence of well-defined composite edges. The following proposition tells us how we can do this.

Proposition 2.1.9. Given a directed graph $\mathcal{G}$, we can construct the *free category* generated by $\mathcal{G}$, denoted $F\mathcal{G}$, as follows. The objects of $F\mathcal{G}$ are the nodes $\mathcal{G}_0$ of $\mathcal{G}$. The morphisms $F\mathcal{G}(a, b)$ from a to b are the *paths* in $\mathcal{G}$ from a to b: finite lists (e, f, g) of edges in which the domain of the first edge is a, the codomain of any edge equals the domain of its successor (if any), and the codomain of the last edge is b. Composition is by concatenation of lists, and the identity morphism for any node is the empty list $()$.

Proof. Let $f := (f_1, \ldots, f_l) : a \to b$, $g := (g_1, \ldots, g_m) : b \to c$, and $h := (h_1, \ldots, h_n) : c \to d$ be paths. Then

$$
\begin{aligned}
h \circ (g \circ f) &= (h_1, \ldots, h_n) \circ (f_1, \ldots, f_l, g_1, \ldots, g_m) \\
&= (f_1, \ldots, f_l, g_1, \ldots, g_m, h_1, \ldots, h_n) \\
&= (g_1, \ldots, g_m, h_1, \ldots, h_n) \circ (f_1, \ldots, f_l) = (h \circ g) \circ f
\end{aligned}
$$

so concatenation of lists is associative. Concatenation is trivially unital on both right and left: $() \circ (f_1, \ldots, f_l) = (f_1, \ldots, f_l) = (f_1, \ldots, f_l) \circ ()$. So the free category as defined is a well-defined category. □

Remark 2.1.10. Observe that the underlying graph of $F\mathcal{G}$ is not in general the same as the original graph $\mathcal{G}$: because the edges of $\mathcal{G}$ have no composition information (even if, given a pair of edges $a \to b$ and $b \to c$, there is an edge $a \to c$), we needed a canonical method to generate such information, without any extra data. Since there is a notion of path in any graph, and since paths are naturally composable, this gives us the canonical method we seek.

We begin to see some important differences between categories and graphs, as foreshadowed above. Categories are somehow more 'dynamical' objects, more concerned with movement and change than graphs; later in Chapter 6, we will even see how a general definition of *dynamical system* emerges simply from some of the examples we have already seen.

At this point, to emphasize that categories allow us to study not just individual structures themselves but also the relationships and transformations between structures, we note that directed graphs themselves form a category.

Example 2.1.11. Directed graphs $(\mathcal{G}_0, \mathcal{G}_1, \mathrm{dom}_{\mathcal{G}}, \mathrm{cod}_{\mathcal{G}})$ are the objects of a category, denoted **Graph**. Given directed graphs $\mathcal{G} := (\mathcal{G}_0, \mathcal{G}_1, \mathrm{dom}_{\mathcal{G}}, \mathrm{cod}_{\mathcal{G}})$ and $\mathcal{H} := (\mathcal{H}_0, \mathcal{H}_1, \mathrm{dom}_{\mathcal{H}}, \mathrm{cod}_{\mathcal{H}})$, a morphism $f : \mathcal{G} \to \mathcal{H}$ is a *graph homomorphism* from $\mathcal{G}$ to $\mathcal{H}$: a pair of functions $f_0 : \mathcal{G}_0 \to \mathcal{G}_0$ and $f_1 : \mathcal{G}_1 \to \mathcal{H}_1$ that preserve the graphical structure in the sense that for every edge e in $\mathcal{G}$, $f_0(\mathrm{dom}_{\mathcal{G}}(e)) = \mathrm{dom}_{\mathcal{H}}(f_1(e))$ and $f_0(\mathrm{cod}_{\mathcal{G}}(e)) = \mathrm{cod}_{\mathcal{H}}(f_1(e))$. Since graph homomorphisms are pairs of functions, they compose as functions, and the identity morphism on a graph $\mathcal{G}$ is the pair $(\mathrm{id}_{\mathcal{G}_0}, \mathrm{id}_{\mathcal{G}_1})$ of identity functions on its sets of nodes and edges.

In large part, the power of category theory derives from its elevation of relationship and transformation to mathematical prominence: objects are represented and studied in context, and one we gain the ability to compare patterns of relationships across contexts. By expressing these patterns categorically, we are able to abstract away irrelevant detail, and focus on the fundamental structures that drive phenomena of interest; and since these patterns and abstract structures are again expressed in the same language, we can continue to apply these techniques, to study phenomena from diverse perspectives. Indeed, as we will soon see, category theory is 'homoiconic', able to speak in its language about itself.

Accordingly, it is often helpful to apply graphical or diagrammatic methods to reason about categories: for example, to say that two (or more) morphisms are actually equal. We can illustrate this using the category **Graph**: the definition of graph homomorphism requires two equalities to be satisfied. These equalities say that two (composite) pairs of functions are equal; since functions are morphisms in **Set**, this is the same as saying that they are equal as morphisms there. Using the fact that **Set** has an underlying graph, we can represent these morphisms graphically, as in the following two diagrams:

$$
\begin{array}{ccc}
\mathcal{G}_1 \xrightarrow{\;f_1\;} \mathcal{H}_1 & \qquad & \mathcal{G}_1 \xrightarrow{\;f_1\;} \mathcal{H}_1 \\
\Big\downarrow{\scriptstyle\mathrm{dom}_{\mathcal{G}}} \quad \Big\downarrow{\scriptstyle\mathrm{dom}_{\mathcal{H}}} & & \Big\downarrow{\scriptstyle\mathrm{cod}_{\mathcal{G}}} \quad \Big\downarrow{\scriptstyle\mathrm{cod}_{\mathcal{H}}} \\
\mathcal{G}_0 \xrightarrow[\;f_0\;]{} \mathcal{H}_0 & & \mathcal{G}_0 \xrightarrow[\;f_0\;]{} \mathcal{H}_0
\end{array}
\tag{2.1}
$$

Then to say that $f_0 \circ \mathrm{dom}_{\mathcal{G}} = \mathrm{dom}_{\mathcal{H}} \circ f_1$ and $f_0 \circ \mathrm{cod}_{\mathcal{G}} = \mathrm{cod}_{\mathcal{H}} \circ f_1$ is to say that these diagrams *commute*.

Definition 2.1.12. We say that two paths in a graph are *parallel* if they have the same start and end nodes. We say that a diagram in a category $\mathcal{C}$ *commutes* when every pair of parallel paths in the diagram corresponds to a pair of morphisms in $\mathcal{C}$ that are equal.

To clarify this definition, we can use category theory to formalize the concept of diagram, which will have the useful side-effect of simultaneously rendering it more general and more precise.

2.1.2.1. Diagrams in a category, functorially

The richness of categorical structure is reflected in the variety of diagrammatic practice, and in this thesis we will encounter a number of formal diagram types. Nonetheless, there is one type that is perhaps more basic than the rest, which we have already begun to call *diagrams in a category*: these are the categorical analogue of equations in algebra. Often in category theory, we will be interested in the relationships between more than two morphisms at once, and expressing such relationships by equations quickly becomes cumbersome; instead, one typically starts with a directed graph and interprets its nodes as objects and its edges as morphisms in one's category of interest.

Formally, this interpretation is performed by taking the category generated by the graph and mapping it 'functorially' into the category of interest. However, in order to account for relationships such as equality between the morphisms represented in the graph, the domain of this mapping cannot be as 'free' as in Proposition 2.1.9, as it needs to encode these relationships. To do this, we can quotient the free category by the given relationships, as we now show.

Proposition 2.1.13 (Mac Lane [175, Prop. II.8.1])**.** Let $\mathcal{G}$ be a directed graph, and suppose we are given a relation $\sim_{a,b}$ on each set $F\mathcal{G}(a,b)$ of paths $a \to b$; write $\sim$ for the whole family of relations, and call it a *relation on the category* $\mathcal{C}$. Then there is a category $F\mathcal{G}/\sim$, the *quotient* of the free category $F\mathcal{G}$ by $\sim$, which we call *the category generated by $\mathcal{G}$ with relations* $\sim$ or simply *generated by $(\mathcal{G}, \sim)$*.

The objects of $F\mathcal{G}/\sim$ are again the nodes $\mathcal{G}_0$. The morphisms are equivalence classes of paths according to $\sim$, extended to a congruence: suppose $p \sim_{a,b} p'$; then they both belong to the same equivalence class $[p]$, and correspond to the same morphism $[p] : a \to b$ in $F\mathcal{G}/\sim$.

Before we can make sense of and prove this proposition, and thus establish that composition in $F\mathcal{G}/\sim$ does what we hope, we need to define *congruence*.

Definition 2.1.14. Suppose $\sim$ is a relation on the category $\mathcal{C}$. We call $\sim$ a *congruence* when its constituent relations $\sim_{a,b}$ are equivalence relations compatible with the compositional structure of $\mathcal{C}$. This means that

1. if $f \sim_{a,b} f' : a \to b$ and $g \sim_{b,c} g' : b \to c$, then $g \circ f \sim_{a,c} g' \circ f'$; and

2. for each pair of objects $a, b : \mathcal{C}$, $\sim_{a,b}$ is a symmetric, reflexive, transitive relation.

The notion of congruence is what allows us to extend the family of relations $\sim$ to composites of morphisms and thus ensure that it is compatible with the categorical structure; constructing the most parsimonious congruence from $\sim$ is the key to the following proof.

Proof sketch for Proposition 2.1.13. First of all, we extend $\sim$ to a congruence; we choose the smallest congruence containing $\sim$, and denote it by $\cong$. Explicitly, we can construct $\cong$ in two steps. First, define an intermediate relation $\simeq$ as the symmetric, reflexive, transitive closure of $\sim$. This means that if $f \simeq f'$, then either $f \sim f'$, or $f' \sim f$ (symmetry), or $f = f'$ (reflexivity), or there exists some $\phi : a \to c$ such that $f \sim \phi$ and $\phi \sim f'$ (transitivity). Next, define $\cong$ as the closure of $\simeq$ under composition. This means that if $\varphi \cong \varphi' : a \to c$, then either $\varphi \simeq \varphi'$, or there exist composable pairs $f, f' : a \to b$ and $g, g' : b \to c$ such that $f \simeq f'$ and $g \simeq g'$, and such that $\varphi = g \circ f$ and $\varphi' = g' \circ f'$. To see that $\cong$ is the least congruence on $F\mathcal{G}$, observe that every congruence must contain it by definition.

Having constructed the congruence $\cong$, we can form the quotient of $F\mathcal{G}$ by it, which we denote by $F\mathcal{G}/\sim$ in reference to the generating relation $\sim$. As in the statement of the proposition, the objects of $F\mathcal{G}/\sim$ are the nodes of $\mathcal{G}$ and the morphisms are equivalence classes of paths, according to $\cong$; since $\cong$ is by definition an equivalence relation, these equivalence classes are well-defined. Moreover, the composite of two equivalence classes of morphisms $[f] : a \to b$ and $[g] : b \to c$ coincides with the equivalence class $[g \circ f]$. $\qquad\square$

Example 2.1.15. To exemplify the notion of category generated with relations, let $\mathcal{J}$ denote the following directed graph

$$
\begin{array}{ccc}
G_1 & \xrightarrow{\ \varphi_1\ } & H_1 \\
\Big\downarrow{\scriptstyle \delta_G} & & \Big\downarrow{\scriptstyle \delta_H} \\
G_0 & \xrightarrow[\ \varphi_0\ ]{} & H_0
\end{array}
$$

and let $\sim$ be the relation $\varphi_0 \circ \delta_G \sim \delta_H \circ \varphi_1$. Then the category $F\mathcal{J}/\sim$ generated by $(\mathcal{J}, \sim)$ has four objects (G_1, G_0, H_1, H_0) and nine morphisms: an identity for each of the four objects; the morphisms $\varphi_0 : G_0 \to H_0$, $\varphi_1 : G_1 \to H_1$, $\delta_G : G_1 \to G_0$, and $\delta_H : H_1 \to H_0$; and a single morphism $G_1 \to H_0$, the equivalence class consisting of $\varphi_0 \circ \delta_G$ and $\delta_H \circ \varphi_1$.

The category $F\mathcal{J}/\sim$ generated in this example expresses the commutativity of one of the diagrams defining graph homomorphisms, but as things stand, it is simply a category standing alone: to say that any particular pair of functions (f_0, f_1) satisfies the property requires us to interpret the morphisms φ_0 and φ_1 accordingly *as those functions*. That is, to interpret the diagram,

we need to translate it, by mapping $FJ/\sim$ into **Set**. Such a mapping of categories is known as a *functor*.

Definition 2.1.16. A *functor* $F : \mathcal{C} \to \mathcal{D}$ from the category $\mathcal{C}$ to the category $\mathcal{D}$ is a pair of functions $F_0 : \mathcal{C}_0 \to \mathcal{D}_0$ and $F_1 : \mathcal{C}_1 \to \mathcal{D}_1$ between the sets of objects and morphisms that preserve domains, codomains, identities and composition, meaning that $F_0(\mathrm{dom}_{\mathcal{C}}(f)) = \mathrm{dom}_{\mathcal{D}}(F_1(f))$ and $F_0(\mathrm{cod}_{\mathcal{C}}(f)) = \mathrm{cod}_{\mathcal{D}}(F_1(f))$ for all morphisms f, $F_1(\mathsf{id}_a) = \mathsf{id}_{F(a)}$ for all objects a, and $F_1(g \circ f) = F_1(g) \circ F_1(f)$ for all composites $g \circ f$ in $\mathcal{C}$.

Remark 2.1.17. Note that we could equivalently say that a functor $\mathcal{C} \to \mathcal{D}$ is a homomorphism from the underlying graph of $\mathcal{C}$ to that of $\mathcal{D}$ that is additionally *functorial*, meaning that it preserves identities and composites.

Notation 2.1.18. Although a functor F consists of a pair of functions (F_0, F_1), we will typically write just F whether it is applied to an object or a morphism, since the distinction will usually be clear from the context. Since function composition (and hence application) is associative, we will also often omit brackets, writing Fa for $F(a)$, except where it is helpful to leave them in.

For each object c in a category $\mathcal{C}$, there are two very important functors, the *hom* functors, which exhibit $\mathcal{C}$ in **Set** "from the perspective" of c by returning the hom sets out of and into c.

Definition 2.1.19. Given an object $c : \mathcal{C}$, its *covariant hom functor* $\mathcal{C}(c, -) : \mathcal{C} \to$ **Set** is defined on objects x by returning the hom sets $\mathcal{C}(c, x)$ and on morphisms $g : x \to y$ by returning the *postcomposition* function $\mathcal{C}(c, g) : \mathcal{C}(c, x) \to \mathcal{C}(c, y)$ defined by mapping morphisms $f : c \to x$ in the set $\mathcal{C}(c, x)$ to the composites $g \circ f : c \to y$ in $\mathcal{C}(c, y)$. To emphasize the action of $\mathcal{C}(c, g)$ by postcomposition, we will sometimes write it simply as $g \circ (-)$. (That $\mathcal{C}(c, -)$ is a well-defined functor follows immediately from the unitality and associativity of composition in $\mathcal{C}$.)

The covariant hom functor $\mathcal{C}(c, -)$ "looks forward" along morphisms emanating *out of* c, in the direction that these morphisms point, and therefore in the direction of composition in $\mathcal{C}$: it is for this reason that we say it is *covariant*. Dually, it is of course possible to "look backward" at morphisms pointing *into* c. Since this means looking contrary to the direction of composition in $\mathcal{C}$, we say that the resulting backwards-looking hom functor is *contravariant*. To define it as a functor in the sense of Definition 2.1.16, we perform the trick of swapping the direction of composition in $\mathcal{C}$ around and then defining a covariant functor accordingly.

Definition 2.1.20. For any category $\mathcal{C}$ there is a corresponding *opposite category* $\mathcal{C}^{\mathrm{op}}$ with the same objects as $\mathcal{C}$ and where the hom set $\mathcal{C}^{\mathrm{op}}(a, b)$ is defined to be the 'opposite' hom set in $\mathcal{C}$, namely $\mathcal{C}(b, a)$. Identity morphisms are the same in $\mathcal{C}^{\mathrm{op}}$ as in $\mathcal{C}$, but composition is also reversed. If we write $\circ$ for composition in $\mathcal{C}$ and $\circ^{\mathrm{op}}$ for composition in $\mathcal{C}^{\mathrm{op}}$, then, given morphisms $g : c \to b$ and $f : b \to a$ in $\mathcal{C}^{\mathrm{op}}$ corresponding to morphisms $g : b \to c$ and $f : a \to b$ in $\mathcal{C}$, their composite $f \circ^{\mathrm{op}} g : c \to a$ in $\mathcal{C}^{\mathrm{op}}$ is the morphism $g \circ f : a \to c$ in $\mathcal{C}$. (Observe that this makes $\mathcal{C}^{\mathrm{op}}$ a well-defined category whenever $\mathcal{C}$ is.)

Remark 2.1.21. Because we can always form opposite categories in this way, categorical constructions often come in two forms: one in $\mathcal{C}$, and a 'dual' one in $\mathcal{C}^{\mathrm{op}}$. Typically, we use the prefix *co-* to indicate such a dual construction: so if we have a construction in $\mathcal{C}$, then its dual in $\mathcal{C}^{\mathrm{op}}$ would be called a *co*construction.

The dual of the covariant hom functor $\mathcal{C}(c, -) : \mathcal{C} \to \mathbf{Set}$ is the contravariant hom functor.

Definition 2.1.22. Given an object $c : \mathcal{C}$, its *contravariant hom functor* $\mathcal{C}(-, c) : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ is defined on objects x by returning the hom sets $\mathcal{C}(x, c)$. Given a morphism $f : x \to y$ in $\mathcal{C}$, we define the *precomposition* function $\mathcal{C}(f, c) : \mathcal{C}(y, c) \to \mathcal{C}(x, c)$ by mapping morphisms $g : y \to c$ in the set $\mathcal{C}(y, c)$ to the composites $g \circ f : x \to c$ in $\mathcal{C}(x, c)$. To emphasize the action of $\mathcal{C}(f, c)$ by precomposition, we will sometimes write it simply as $(-) \circ f$. (That $\mathcal{C}(-, c)$ is a well-defined functor again follows from the unitality and associativity of composition in $\mathcal{C}$ and hence in $\mathcal{C}^{\mathrm{op}}$.)

Remark 2.1.23. A *contravariant* functor on $\mathcal{C}$ is a (covariant) functor on $\mathcal{C}^{\mathrm{op}}$.

Notation 2.1.24. In line with other mathematical literature, we will also occasionally write the precomposition function $(-) \circ f$ as f^*; dually, we can write the postcomposition function $g \circ (-)$ as g_*. In these forms, the former action f^* is also known as *pullback* along f, as it "pulls back" morphisms along f, and the latter action g_* is also known as *pushforward* along g, as it "pushes forward" morphisms along g. There is a close relationship between the pulling-back described here and the universal construction also known as pullback (Example 2.3.43): $f^*(-)$ defines a functor which acts by the universal construction on objects and by precomposition on morphisms, which we spell out in Definition 4.2.28.

Functors are the homomorphisms of categories, and just as graphs and their homomorphisms form a category, so do categories and functors.

Example 2.1.25. The category **Cat** has categories for objects and functors for morphisms. The identity functor $\mathrm{id}_{\mathcal{C}}$ on a category $\mathcal{C}$ is the pair $(\mathrm{id}_{\mathcal{C}_0}, \mathrm{id}_{\mathcal{C}_1})$ of identity functions on the sets of objects and morphisms. Since functors are pairs of functions, functor composition is by function composition, which is immediately associative and unital with respect to the identity functors so defined. Note that, without a restriction on size, **Cat** is a large category, like **Set**.

As an example, we observe that the construction of the category $F\mathcal{G}/\sim$ generated by $(\mathcal{G}, \sim)$ from the free category $F\mathcal{G}$ is functorial.

Example 2.1.26. There is a 'projection' functor $[\cdot] : F\mathcal{G} \to F\mathcal{G}/\sim$. It maps every object to itself, and every morphism to the corresponding equivalence class. The proof of Proposition 2.1.13 demonstrated the functoriality: identities are preserved by definition, and we have $[g \circ f] = [g] \circ [f]$ by construction.

With the notion of functor to hand, we can formalize the concept of diagram simply as follows.

Definition 2.1.27. A *J-shaped diagram* in a category $\mathcal{C}$ is a functor $D : J \to \mathcal{C}$. Typically, J is a small category generated from a graph with some given relations, and the functor D interprets J in $\mathcal{C}$.

Example 2.1.28. The diagrams expressing the commutativity conditions for a graph homomorphism (2.1) are therefore witnessed by a pair of functors $F\mathcal{J}/\sim \to \mathbf{Set}$ from the category $F\mathcal{J}/\sim$ generated in Example 2.1.15 into **Set**: each functor interprets φ_0 and φ_1 as f_0 and f_1 respectively, while one functor interprets δ_G as $\mathrm{dom}_{\mathcal{G}}$ and δ_H as $\mathrm{dom}_{\mathcal{H}}$ and the other interprets δ_G as $\mathrm{cod}_{\mathcal{G}}$ and δ_H as $\mathrm{cod}_{\mathcal{H}}$. The fact that there is only a single morphism $G_1 \to H_0$ in $F\mathcal{J}/\sim$ (even though there are two in $F\mathcal{J}$) encodes the requirements that $f_0 \circ \mathrm{dom}_{\mathcal{G}} = \mathrm{dom}_{\mathcal{H}} \circ f_1$ and $f_0 \circ \mathrm{cod}_{\mathcal{G}} = \mathrm{cod}_{\mathcal{H}} \circ f_1$.

Throughout this thesis, we will see the utility of diagrams as in Definition 2.1.27: not only will they be useful in reasoning explicitly about categorical constructions, but in §2.3.3 they will also be used to formalize 'universal constructions', another concept which exhibits the power of category theory.

Despite this, 'mere' categories and their diagrams are in some ways not expressive enough: often we will want to encode looser relationships than strict equality, or to compose diagrams together by 'pasting' them along common edges; we may even want to consider morphisms between morphisms! For this we will need to 'enrich' our notion of category accordingly.

2.2. Connecting the connections

As we have indicated, basic category theory is not sufficient if we want to encode information about the relationships between morphisms into the formal structure. In this section, we will see how to enrich the notion of category by letting the morphisms collect into more than just sets, and how this leads naturally to higher category theory, where we have morphisms between the morphisms, and from there to the notion of adjunction, with which we can translate concepts faithfully back and forth between contexts. Amidst the development, we discuss the concept of "functorial semantics" from a scientific perspective, considering how categorical tools let us supply rich semantics for structured models of complex systems such as the brain.

2.2.1. Enriched categories

We can think of the condition that a diagram commutes — or equivalently the specification of an equivalence relation on its paths — as a 'filling-in' of the diagram with some extra data. For example, we can 'fill' the diagram depicting the graph homomorphism condition $f_0 \circ \mathrm{dom}_{\mathcal{G}} = \mathrm{dom}_{\mathcal{H}} \circ f_1$ with some annotation or data witnessing this relation, as follows:

$$
\begin{array}{ccc}
\mathcal{G}_1 & \xrightarrow{\ f_1\ } & \mathcal{H}_1 \\
{\scriptstyle \mathrm{dom}_{\mathcal{G}}}\Big\downarrow & & \Big\downarrow{\scriptstyle \mathrm{dom}_{\mathcal{H}}} \\
\mathcal{G}_0 & \xrightarrow[\ f_0\ ]{} & \mathcal{H}_0
\end{array}
$$

If we have a composite graph homomorphism $g \circ f : \mathcal{G} \to \mathcal{I}$, we should be able to paste the commuting diagrams of the factors together and fill them in accordingly:

$$
\begin{array}{ccccc}
\mathcal{G}_1 & \xrightarrow{\ f_1\ } & \mathcal{H}_1 & \xrightarrow{\ g_1\ } & \mathcal{I}_1 \\
{\scriptstyle \mathrm{dom}_{\mathcal{G}}}\Big\downarrow & & {\scriptstyle \mathrm{dom}_{\mathcal{H}}}\Big\downarrow & & \Big\downarrow{\scriptstyle \mathrm{dom}_{\mathcal{I}}} \\
\mathcal{G}_0 & \xrightarrow[\ f_0\ ]{} & \mathcal{H}_0 & \xrightarrow[\ g_0\ ]{} & \mathcal{I}_0
\end{array}
$$

and we should be able to 'compose' the filler equalities to obtain the diagram for the composite:

$$
\begin{array}{ccccc}
\mathcal{G}_1 & \xrightarrow{\ f_1\ } & \mathcal{H}_1 & \xrightarrow{\ g_1\ } & \mathcal{I}_1 \\
{\scriptstyle \mathrm{dom}_{\mathcal{G}}}\Big\downarrow & & & & \Big\downarrow{\scriptstyle \mathrm{dom}_{\mathcal{I}}} \\
\mathcal{G}_0 & \xrightarrow[\ f_0\ ]{} & \mathcal{H}_0 & \xrightarrow[\ g_0\ ]{} & \mathcal{I}_0
\end{array} \ .
$$

The extra data with which we have filled these diagrams sits 'between' the morphisms, and so if we wish to incorporate it into the categorical structure, we must move beyond mere sets, for sets are just collections of elements, with nothing "in between". What we will do is allow the hom sets of a category to be no longer sets, but objects of another 'enriching' category. Now, observe that, in pasting the two diagrams above together, we had to place them side by side: this means that any suitable enriching category must come equipped with an operation that allows us to place its objects side by side; in the basic case, where our categories just have hom sets, the enriching category is **Set**, and this side-by-side operation is the product of sets.

Definition 2.2.1. Given sets A and B, their *product* is the set $A \times B$ whose elements are pairs (a, b) of an element $a : A$ with an element $b : B$.

We have already made use of the product of sets above, when we defined the composition operation for (small) categories in Definition 2.1.2. In general, however, we don't need precisely a product; only something weaker, which we call *tensor*. In order to define it, we need the notion of *isomorphism*.

Definition 2.2.2. A morphism $l : c \to d$ in a 1-category is an *isomorphism* if there is a morphism $r : d \to c$ such that $l \circ r = \mathsf{id}_d$ and $\mathsf{id}_c = r \circ l$. We say that l and r are mutually *inverse*.

Definition 2.2.3. We will say that a category $\mathcal{C}$ has a *tensor product* if it is equipped with a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ along with an object $I : \mathcal{C}$ called the *tensor unit* and three families of isomorphisms:

1. *associator* isomorphisms $\alpha_{a,b,c} : (a \otimes b) \otimes c \xrightarrow{\sim} a \otimes (b \otimes c)$ for each triple of objects a, b, c;

2. *left unitor* isomorphisms $\lambda_a : I \otimes a \xrightarrow{\sim} a$ for each object a; and

3. *right unitor* isomorphisms $\rho_a : a \otimes I \xrightarrow{\sim} a$ for each object a.

Remark 2.2.4. The notion of tensor product forms part of the definition of *monoidal category*, which we will come to in §3.1.2. Beyond having a tensor product, a monoidal category must have structure isomorphisms that are *coherent* with respect to the ambient categorical structure, which itself satisfies properties of associativity and unitality; this is an echo of the *microcosm principle* which we discuss in Remark 3.4.7. However, to give the full definition the notion of monoidal category requires us to introduce the notion of *natural transformation*, which we otherwise do not need until Definition 2.2.17; moreover, questions of coherence of tensor products will not yet arise.

Unsurprisingly, the product of sets gives us our first example of a tensor product structure.

Example 2.2.5. The product of sets gives us a tensor product $\times : \mathbf{Set} \times \mathbf{Set} \to \mathbf{Set}$. To see that it is functorial, observe that, given a product of sets $A \times B$ and a function $f : A \to A'$, we naturally obtain a function $f \times B : A \times B \to A \times A'$ by applying f only to the A-components of the elements of the product $A \times B$; likewise given a function $g : B \to B'$. The unit of the tensor product structure is the set 1 with a single element $*$. The associator and unitors are almost trivial: for associativity, map $((a, b), c)$ to $(a, (b, c))$.

Using the tensor product to put morphisms side by side, we can define the notion of *enriched category*.

Definition 2.2.6. Suppose $(\mathcal{E}, \otimes, I, \alpha, \lambda, \rho)$ is a category equipped with a tensor product. An $\mathcal{E}$-*category* $\mathcal{C}$, or *category* $\mathcal{C}$ *enriched in* $\mathcal{E}$, constitutes

1. a set $\mathcal{C}_0$ of *objects*;

2. for each pair (a, b) of $\mathcal{C}$-objects, an $\mathcal{E}$-object $\mathcal{C}(a, b)$ of *morphisms* from a to b;

3. for each object a in $\mathcal{C}$, an $\mathcal{E}$-morphism $\mathrm{id}_a : I \to \mathcal{C}(a, a)$ witnessing *identity*; and

4. for each triple (a, b, c) of $\mathcal{C}$-objects, an $\mathcal{E}$-morphism $\circ_{a,b,c} : \mathcal{C}(b, c) \otimes \mathcal{C}(a, b) \to \mathcal{C}(a, c)$ witnessing *composition*;

such that composition is unital, *i.e.* for all $a, b : \mathcal{C}$

$$
\begin{array}{ccccccc}
\mathcal{C}(a,b) \otimes I \xrightarrow{\mathcal{C}(a,b) \otimes \mathrm{id}_a} \mathcal{C}(a,b) \otimes \mathcal{C}(a,a) & & & \mathcal{C}(a,b) \otimes \mathcal{C}(a,a) \xleftarrow{\mathrm{id}_b \otimes \mathcal{C}(a,b)} I \otimes \mathcal{C}(a,b) \\
\searrow_{\rho_{\mathcal{C}(a,b)}} \quad \downarrow_{\circ_{a,a,b}} & \text{and} & \downarrow_{\circ_{a,b,b}} \quad \swarrow_{\lambda_{\mathcal{C}(a,b)}} \\
\mathcal{C}(a,b) & & \mathcal{C}(a,b)
\end{array}
$$

,

and associative, *i.e.* for all $a, b, c, d : \mathcal{C}$

$$
\begin{array}{ccc}
\big(\mathcal{C}(c,d) \otimes \mathcal{C}(b,c)\big) \otimes \mathcal{C}(a,b) & \xrightarrow{\alpha_{a,b,c,d}} & \mathcal{C}(c,d) \otimes \big(\mathcal{C}(b,c) \otimes \mathcal{C}(a,b)\big) \\
\downarrow_{\circ_{b,c,d} \otimes \mathcal{C}(a,b)} & & \downarrow_{\mathcal{C}(c,d) \otimes \circ_{a,b,c}} \\
\mathcal{C}(b,d) \otimes \mathcal{C}(a,b) & & \mathcal{C}(c,d) \otimes \mathcal{C}(a,c) \\
\searrow_{\circ_{a,b,d}} & & \swarrow_{\circ_{a,c,d}} \\
& \mathcal{C}(a,d) &
\end{array}
$$

.

Our first example of enriched categories validates the definition.

Example 2.2.7. A locally small category is a category enriched in $(\mathbf{Set}, \times, 1)$.

Remark 2.2.8. In $\mathbf{Set}$, morphisms $1 \to A$ out of the unit set 1 correspond to elements of A: each such morphism is a function mapping the unique element $* : 1$ to its corresponding element of A. This is why identities in enriched category theory are given by morphisms $I \to \mathcal{C}(a, a)$, and it is also why we will call morphisms out of a tensor unit *generalized elements*. (Even more generally, we might say that morphisms $X \to A$ are generalized elements *of shape X*, reflecting our use of the word 'shape' to describe the domain of a diagram.)

To incorporate nontrivial fillers into our diagrams, we move instead to enrichment in *pro*sets.

Example 2.2.9. A *preordered set* or *proset* is a category where there is at most one morphism between any two objects. The objects of such a 'thin' category are the points of the proset, and the morphisms encode the (partial) ordering of the points; as a result, they are often written $a \leqslant a'$. Functors between prosets are functions that preserve the ordering, and the restriction of $\mathbf{Cat}$ to prosets produces a category that we denote by $\mathbf{Pro}$. The product of sets extends to prosets as follows: if A and B are prosets, then their product is the proset $A \times B$ whose points are the points of the product set $A \times B$ and a morphism $(a, b) \leqslant (a', b')$ whenever there are morphisms $a \leqslant a'$ and $b \leqslant b'$ in A and B respectively.

A category enriched in $\mathbf{Pro}$ is therefore a category whose hom sets are (pre)ordered and whose composition operation preserves this ordering, which we can illustrate as follows:

We can see how enrichment in $\mathbf{Pro}$ generalizes the situation with which we introduced this section, where we considered filling diagrams with data witnessing the equality of morphisms: here we have *inequality* data, and it is not hard to see how enriched composition encompasses the pasting-and-composing discussed there (just replace the cells here by the squares above).

In order to make these filled diagrams precise, we need to extend the notion of functor to the enriched setting; and so we make the following definition.

Definition 2.2.10. Suppose $\mathcal{C}$ and $\mathcal{D}$ are $\mathcal{E}$-categories. Then an $\mathcal{E}$-*functor F* constitutes

1. a function $F_0 : \mathcal{C}_0 \to \mathcal{D}_0$ between the sets of objects; and

2. for each pair $(a, b) : \mathcal{C}_0 \times \mathcal{C}_0$ of objects in $\mathcal{C}$, an $\mathcal{E}$-morphism $F_{a,b} : \mathcal{C}(a, b) \to \mathcal{D}(F_0 a, F_0 b)$

which preserve identities

$$
\begin{array}{ccc}
& I & \\
{}^{\mathrm{id}_a}\swarrow & & \searrow^{\mathrm{id}_{F_0 a}} \\
\mathcal{C}(a,a) & \xrightarrow[\quad F_{a,a} \quad]{} & \mathcal{D}(F_0 a, F_0 a)
\end{array}
$$

and composition

$$
\begin{array}{ccc}
\mathcal{C}(b,c) \otimes \mathcal{C}(a,b) & \xrightarrow{\ \circ_{a,b,c}\ } & \mathcal{C}(a,c) \\
{\scriptstyle F_{b,c} \otimes F_{a,b}}\Big\downarrow & & \Big\downarrow{\scriptstyle F_{a,c}} \\
\mathcal{D}(F_0 b, F_0 c) \otimes \mathcal{D}(F_0 a, F_0 b) & \xrightarrow[\circ_{F_0 a, F_0 b, F_0 c}]{} & \mathcal{D}(F_0 a, F_0 c)
\end{array} \quad .
$$

A diagram in an $\mathcal{E}$-enriched category $\mathcal{C}$ is therefore a choice of $\mathcal{E}$-enriched category J (the diagram's shape) and an $\mathcal{E}$-functor $J \to \mathcal{C}$. J encodes the objects, morphisms and relationships of the diagram, and the functor interprets it in $\mathcal{C}$. In this enriched setting, we need not quotient parallel paths in the shape of a diagram (which destroys their individuality); instead, we have extra data (the fillers) encoding their relationships.

2.2.2. 2-categories

We have seen that filling the cells of a diagram with inequalities pushes us to consider enrichment in **Pro**. Since **Pro** is the category of categories with at most one morphism (*i.e.*, the inequality) between each pair of objects, a natural generalization is to allow a broader choice of filler: that is, to allow there to be morphisms between morphisms. This means moving from enrichment in **Pro** to enrichment in **Cat**, and hence to the notion of *2-category*. We therefore make the following definition.

Definition 2.2.11. A *strict 2-category* is a category enriched in the 1-category **Cat**. This means that, instead of hom sets, a 2-category has hom *categories*: the objects of these hom categories are the *1-cells* of the 2-category, and the morphisms of the hom categories are the *2-cells*; the *0-cells* of the 2-category are its objects. To distinguish between the composition defined by the enriched category structure from the composition within the hom categories, we will sometimes call the former *horizontal* and the latter *vertical* composition.

Remark 2.2.12. We say *1-category* above to refer to the '1-dimensional' notion of category defined in Definition 2.1.2.

Remark 2.2.13. We say *strict* to mean that the associativity and unitality of composition hold up to equality; later, it will be helpful to weaken this so that associativity and unitality only hold up to "coherent isomorphism", meaning that instead of asking the diagrams in Definition 2.2.6 simply to commute (and thus be filled by equalities), we ask for them to be filled with 'coherently' defined isomorphism. Weakening 2-categorical composition in this way leads to the notion of *bicategory* (§3.1.4).

In order to give a well-defined notion of enrichment in **Cat**, we need to equip it with a suitable tensor product structure; for this, we can extend the product of sets to categories, as follows.

Proposition 2.2.14. Given categories $\mathcal{C}$ and $\mathcal{D}$, we can form the *product* category $\mathcal{C} \times \mathcal{D}$. Its set of objects $(\mathcal{C} \times \mathcal{D})_0$ is the product set $\mathcal{C}_0 \times \mathcal{D}_0$. Similarly, a morphism $(c, d) \to (c', d')$ is a pair (f, g) of a morphism $f : c \to c'$ in $\mathcal{C}$ with a morphism $g : d \to d'$ in $\mathcal{D}$; hence $(\mathcal{C} \times \mathcal{D})_1 = \mathcal{C}_1 \times \mathcal{D}_1$. Composition is given by composing pairwise in $\mathcal{C}$ and $\mathcal{D}$: $(f', g') \circ (f, g) := (f' \circ f, g' \circ g)$.

Proof. That composition is associative and unital in $\mathcal{C} \times \mathcal{D}$ follows immediately from those properties in the underlying categories $\mathcal{C}$ and $\mathcal{D}$. $\square$

Remark 2.2.15. Using the product of categories, we can gather the co- and contravariant families of hom functors $\mathcal{C}(c, -)$ and $\mathcal{C}(-, c)$ into a single *hom functor* $\mathcal{C}(-, =) : \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathbf{Set}$, mapping $(x, y) : \mathcal{C}^{\mathrm{op}} \times \mathcal{C}$ to $\mathcal{C}(x, y)$.

Proposition 2.2.16. The product of categories extends to a functor $\times : \mathbf{Cat} \times \mathbf{Cat} \to \mathbf{Cat}$. Given functors $F : \mathcal{C} \to \mathcal{C}'$ and $G : \mathcal{D} \to \mathcal{D}'$, we obtain a functor $F \times G$ by applying F to the left factor of the product $\mathcal{C} \times \mathcal{D}$ and G to the right.

Proof. Sufficiently obvious that we omit it. $\square$

The archetypal 2-category is **Cat** itself, as we will now see: morphisms between functors are called *natural transformation*, and they will play an important rôle throughout this thesis.

Definition 2.2.17. Suppose F and G are functors $\mathcal{C} \to \mathcal{D}$. A *natural transformation* $\alpha : F \Rightarrow G$ is a family of morphisms $\alpha_c : F(c) \to G(c)$ in $\mathcal{D}$ and indexed by objects c of $\mathcal{C}$, such that for any morphism $f : c \to c'$ in $\mathcal{C}$, the following diagram — called a *naturality square* for α — commutes:

$$
\begin{array}{ccc}
Fc & \xrightarrow{\ \alpha_c\ } & Gc \\
{\scriptstyle Ff}\big\downarrow & & \big\downarrow{\scriptstyle Gf} \\
Fc' & \xrightarrow[\ \alpha_{c'}\ ]{} & Gc'
\end{array}\ .
$$

When the component 1-cells of a natural transformation α are all isomorphisms, then we call α a *natural isomorphism*.

Example 2.2.18. Every morphism $f : a \to b$ in a category $\mathcal{C}$ induces a (contravariant) natural transformation $\mathcal{C}(f, -) : \mathcal{C}(b, -) \Rightarrow \mathcal{C}(a, -)$ between covariant hom functors, acting by precomposition. Dually, every morphism $h : c \to d$ induces a (covariant) natural transformation $\mathcal{C}(-, h) : \mathcal{C}(-, c) \Rightarrow \mathcal{C}(-, d)$ between contravariant hom functors, acting by postcomposition. To see that these two families are natural, observe that the square below left must commute for all objects $a, b, c : \mathcal{C}$ and morphisms $f : a \to b$ and $h : c \to d$, by the associativity of composition in $\mathcal{C}$ (as illustrated on the right)

$$
\begin{array}{ccc}
\mathcal{C}(b, c) & \xrightarrow{\mathcal{C}(f, c)} & \mathcal{C}(a, c) \\
{\scriptstyle \mathcal{C}(b, h)}\big\downarrow & & \big\downarrow{\scriptstyle \mathcal{C}(a, h)} \\
\mathcal{C}(b, d) & \xrightarrow[\mathcal{C}(f, d)]{} & \mathcal{C}(a, d)
\end{array}
\qquad\qquad
\begin{array}{ccc}
g & \longmapsto & g \circ f \\
\big\downarrow & & \big\downarrow \\
h \circ g & \longmapsto & h \circ g \circ f
\end{array}
$$

and that it therefore constitutes a naturality square for both $\mathcal{C}(f, -)$ and $\mathcal{C}(-, h)$. Note also that we can take either path through this square as a definition of the function $\mathcal{C}(f, h) : \mathcal{C}(b, c) \to \mathcal{C}(a, d)$ which thus acts by mapping $g : b \to c$ to $h \circ g \circ f : a \to d$.

Remark 2.2.19. We will see in §3.1.2 that the families of structure morphisms for a tensor product (and hence used in the definition of enriched category) are more properly required to be natural transformations.

The existence of morphisms between functors implies that the collection of functors between any pair of categories itself forms a category, which we now define.

Proposition 2.2.20. The functors between two categories $\mathcal{C}$ and $\mathcal{D}$ constitute the objects of a category, called the *functor category* and denoted by $\mathbf{Cat}(\mathcal{C}, \mathcal{D})$ or $\mathcal{D}^{\mathcal{C}}$, whose morphisms are the natural transformations between those functors. The identity natural transformation on a functor is the natural transformation whose components are all identity morphisms.

Proof. First, observe that the identity natural transformation is well-defined, as the following diagram commutes for any morphism $f : c \to c'$:

$$
\begin{array}{ccc}
Fc & \overset{\mathrm{id}_{Fc}}{=\!=\!=} & Fc \\
{\scriptstyle Ff}\big\downarrow & & \big\downarrow{\scriptstyle Ff} \\
Fc' & \underset{\mathrm{id}_{Fc'}}{=\!=\!=} & Fc'
\end{array}
$$

(Note that in general, we will depict an identity morphism in a diagram as an elongated equality symbol, as above.) Given two natural transformations $\alpha : F \Rightarrow G$ and $\beta : G \Rightarrow H$, their composite is the natural transformation defined by composing the component functions: $(\beta \circ \alpha)_c := \beta_c \circ \alpha_c$. We can see that this gives a well-defined natural transformation by pasting the component naturality squares:

$$
\begin{array}{ccccc}
Fc & \xrightarrow{\ \alpha_c\ } & Gc & \xrightarrow{\ \beta_c\ } & Hc \\
\downarrow{\scriptstyle Ff} & & \downarrow{\scriptstyle Gf} & & \downarrow{\scriptstyle Hf} \\
Fc' & \xrightarrow[\ \alpha_{c'}\ ]{} & Gc' & \xrightarrow[\ \beta_{c'}\ ]{} & Hc'
\end{array}
$$

Since the two inner squares commute, so must the outer square. And since the composition of natural transformations reduces to the composition of functions, and the identity natural transformation has identity function components, the composition of natural transformations inherits strict associativity and unitality from composition in **Set**. $\qquad\square$

This gives us our a first nontrivial example of a 2-category.

Example 2.2.21. Functor categories constitute the hom categories of the strict 2-category **Cat**, and henceforth we will write $\mathbf{Cat}_1$ to denote the 1-category of categories and functors; we can therefore say that **Cat** is enriched in $\mathbf{Cat}_1$. The 0-cells of **Cat** are categories, the 1-cells are functors, and the 2-cells are natural transformations. If α is a natural transformation $F \Rightarrow G$, with F and G functors $\mathcal{C} \to \mathcal{D}$, then we can depict it as filling the cell between the functors:

$$
\mathcal{C} \underset{G}{\overset{F}{\rightrightarrows}} \Downarrow\alpha\ \mathcal{D}
$$

(More generally, we will depict 2-cells in this way, interpreting such depictions as diagrams of enriched categories in the sense discussed above.)

Since **Cat** is a 2-category, it has both vertical composition (composition within hom-categories) and horizontal (composition between them). In Proposition 2.2.20, we introduced the vertical composition, so let us now consider the horizontal, which we will denote by $\diamond$ to avoid ambiguity. The horizontal composition of 1-cells is the composition of functors (as morphisms in $\mathbf{Cat}_1$), but by the definition of enriched category, it must also extend to the 2-cells (here, the natural transformations). Suppose then that we have natural transformations φ and γ as in the following

diagram:

The horizontal composite $\gamma \diamond \varphi$ is the natural transformation $GF \Rightarrow G'F'$ with components

$$GFb \xrightarrow{G\varphi_b} GF'b \xrightarrow{\gamma_{F'b}} G'F'b \ .$$

Notation 2.2.22 (Whiskering). It is often useful to consider the horizontal composite of a natural transformation $\alpha : F \Rightarrow G$ with (the identity natural transformation on) a functor, as in the following diagrams, with precomposition on the left and postcomposition on the right:

We will often write the left composite $\alpha \diamond L : FL \Rightarrow GL$ as α_L, since its components are $\alpha_{Ld} : FLd \to GLd$ for all $d : \mathcal{D}$; and we will often write the right composite $R \diamond \alpha : RF \Rightarrow RG$ as $R\alpha$, since its components are $R\alpha_c : RFc \to RGc$ for all $c : \mathcal{C}$. This use of notation is called *whiskering*.

2.2.3. On functorial semantics

At this point, we pause to consider category theory from the general perspective of our motivating examples, to reflect on how category theory might surprise us: as we indicated in §2.1.2, categories are more 'dynamical' than graphs, more preoccupied with change, and so behave differently; in fact, they have a much richer variety of behaviours, and just as categories can often be very well-behaved, they can also be quite unruly. Through its homoiconicity—its ability to describe itself—the use of category theory impels us to consider not only how individual systems are constructed, nor only how systems of a given type can be compared, but also how to compare different classes of system. In this way, category theory rapidly concerns itself with notions not only of connection and composition, but also of pattern and translation.

Scientifically, this is very useful: in the computational, cognitive, or otherwise cybernetic sciences, we are often concerned with questions about when and how natural systems 'compute'. Such questions amount to questions of translation, between the abstract realm of computation to the more concrete realms inhabited by the systems of interest and the data that they generate; one often asks how natural structures might correspond to 'algorithmic' details, or whether the behaviours of

systems correspond to computational processes. It is for this reason that we chose our motivating examples, which exhibited (abstract) natural structure as well as two kinds of informational or computational structure: a central question in contemporary neuroscience is the extent to which neural circuits can be understood as performing computation (particularly of the form now established in machine learning). This question is in some way at the heart of this thesis, which aims to establish a compositional framework in which the theories of predictive coding and active inference may be studied.

The dynamism of categories is a hint that it is possible to describe both the structure of systems and their function categorically, with a 'syntax' for systems on the one hand and 'semantics' on the other. This is the notion of *functorial semantics* [165], by which we translate syntactic structures in one category to another category which supplies semantics: the use of functors means that this translation preserves basic compositional structure, and we often ask for these functors to preserve other structures, too; a typical choice, that we will adopt in Chapter 3 is to use *lax monoidal* functors, which preserve composition in two dimensions, allowing us to place systems "side by side" as well as "end to end".

Of course, the particular choices of syntactic and semantic category will depend upon the subject at hand—in this thesis we will be particularly interested in supplying dynamical semantics for approximate inference problems—but typically the syntactic category will have some 'nice' algebraic structure that is then preserved and interpreted by the functorial semantics. This is, for instance, how functorial semantics lets us understand processes that "happen on graphs", and as a simple example, we can consider diagrams in **Set**: the shape of the diagram tells us how to compose the parts of a system together, while the diagram functor gives us, for each abstract part, a set of possible components that have a compatible interface, as well as functions realizing their interconnection.

In categorical 'process' theory, and the more general categorical theory of systems, one therefore often considers the objects of the 'syntactic' category as representing the shapes or interfaces of systems and the morphisms as representing how the different shapes can plug together. This is an algebraic approach to systems design: mathematically, the syntactic structure is encoded as a *monad*, and the functorial semantics corresponds to a *monad algebra*, as we explain in Chapter 3; and the desire for composition richer than merely end-to-end is another motivation for venturing into higher category theory. In Chapter 6, we will 'unfold' a combination of these ideas, to construct bicategories whose objects represent interfaces, whose 1-cells are processes 'between' the interfaces that can be composed both sequentially and in parallel, and whose 2-cells are homomorphisms of

processes. This bicategory will then in Chapter 7 supply the semantics for models of predictive coding.

In science, there is rarely only one way to study a phenomenon, and our collective understanding of phenomena is therefore a patchwork of perspectives. At the end of this chapter, we will discuss the Yoneda Lemma, which formalizes this observation that to understand a thing is to see it from all perspectives, and it is for this reason that we expect category theory to supply a *lingua franca* for the mathematical sciences. In computational neuroscience specifically, an influential methodological theory is David Marr's "three levels of explanation" [179], in which complex cognitive systems are profitably studied at the levels of 'computation', 'algorithm', and 'implementation'. These levels are only very informally defined, and the relationships between them not at all clear. We hope that functorial semantics and other categorical approaches can replace such methodologies so that instead of a simplistic hierarchical understanding of systems, we can progressively and clearly expose the web of relationships between models.

2.2.4. Adjunction and equivalence

We discussed above the use of functors to translate between mathematical contexts. Often, we are interested not only in translation in one direction, but also in translating back again. When we have a pair of functors—or 1-cells more generally—in opposite directions and when the two translations are somehow reversible, we often find ourselves with an *adjunction*; for example, the functorial mappings of graphs to categories and back are adjoint (Example 2.2.25 below), and we conjecture in Chapter 8 that the mapping of "statistical games" to dynamical systems forms part of an adjunction, too. Adjunctions are particularly well-behaved 'dual' translations, and they will therefore be of much use throughout this thesis. For its conceptual elegance, we begin with an abstract definition, which exhibits the fundamental essence.

Definition 2.2.23. Suppose $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ are 1-cells of a 2-category. We say that they are *adjoint* or form an *adjunction*, denoted $L \dashv R$, if there are 2-cells $\eta : \mathsf{id}_\mathcal{C} \Rightarrow RL$ and $\epsilon : LR \Rightarrow \mathsf{id}_\mathcal{D}$, called respectively the *unit* and *counit* of the adjunction, which satisfy the *triangle* equalities $\epsilon_L \circ L\eta = \mathsf{id}_L$ and $R\epsilon \circ \eta_R = \mathsf{id}_R$, so called owing to their diagrammatic depictions:

$$
\begin{array}{ccc}
L \xrightarrow{\;L\eta\;} LRL & & R \xrightarrow{\;\eta_R\;} RLR \\
\Big\| \qquad \downarrow{\scriptstyle \epsilon_L} & \text{and} & \Big\| \qquad \downarrow{\scriptstyle R\epsilon} \\
L & & R
\end{array}
$$

The unit and counit of the adjunction measure 'how far' the round-trip composite functors $LR : \mathcal{C} \to \mathcal{C}$ and $RL : \mathcal{D} \to \mathcal{D}$ leave us from our starting place, as indicated in the following diagrams:

$$
\begin{array}{ccc}
& \mathcal{D} & \\
L \nearrow \ \Uparrow\eta\ \searrow R & & \\
\mathcal{C} \xlongequal{\quad} \mathcal{C} & \text{and} & \\
\mathrm{id}_{\mathcal{C}} & &
\end{array}
\qquad
\begin{array}{c}
\mathcal{C} \\
R \nearrow \ \Downarrow\epsilon\ \searrow L \\
\mathcal{D} \xrightarrow{\ \mathrm{id}_{\mathcal{D}}\ } \mathcal{D}
\end{array}
$$

The triangle identities then ensure that the round-trips have an isomorphic 'core', so that it is possible to translate morphisms on one side to the other losslessly (which we will exemplify in Proposition 2.2.26), and that the adjunction has a natural 'algebraic' interpretation (which we will encounter in Proposition 3.4.13).

In the specific case of the 2-category **Cat**, we can make the following alternative characterization of adjunctions. Here we see that the "isomorphic core" of the adjunction can be characterized by saying that morphisms into objects in $\mathcal{C}$ that come from $\mathcal{D}$ via R are in bijection with morphisms out of objects in $\mathcal{D}$ that come from $\mathcal{C}$ via L.

Definition 2.2.24. Suppose $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ are functors between categories $\mathcal{C}$ and $\mathcal{D}$. We say that they are *adjoint functors* when there is an isomorphism between the hom-sets $\mathcal{D}(Lc, d) \cong \mathcal{C}(c, Rd)$ that is natural in $c : \mathcal{C}$ and $d : \mathcal{D}$.

Given a morphism $f : Lc \to d$ in $\mathcal{D}$, we denote its *(right) adjunct* in $\mathcal{C}$ by $f^\sharp : c \to Rd$. Inversely, given a morphism $g : c \to Rd$ in $\mathcal{C}$, we denote its *(left) adjunct* in $\mathcal{D}$ by $g^\flat : Lc \to d$. The existence of the isomorphism means that $f^{\sharp\flat} = f$ and $g = g^{\flat\sharp}$.

Example 2.2.25. The functor $F : \mathbf{Graph} \to \mathbf{Cat}$ mapping a graph to the corresponding free category (Proposition 2.1.9) is left adjoint to the forgetful functor $U : \mathbf{Cat} \to \mathbf{Graph}$ mapping a category to its underlying graph (Proposition 2.1.8). To see this, we need to find a natural isomorphism $\mathbf{Cat}(F\mathcal{G}, \mathcal{C}) \cong \mathbf{Graph}(\mathcal{G}, U\mathcal{C})$. A graph homomorphism $\mathcal{G} \to U\mathcal{C}$ is a mapping of the nodes of $\mathcal{G}$ to the objects of $\mathcal{C}$ and of the edges of $\mathcal{G}$ to the morphisms of $\mathcal{C}$ that preserves sources (domains) and targets (codomains). A functor $F\mathcal{G} \to \mathcal{C}$ is a mapping of the nodes of $\mathcal{G}$ to the objects of $\mathcal{C}$ along with a mapping of paths in $\mathcal{G}$ to morphisms in $\mathcal{C}$ that preserves domains, codomains, identities and composites. A path in $\mathcal{G}$ is a list of 'composable' edges, with the identity path being the empty list, so such a mapping of paths is entirely determined by a mapping of edges to morphisms that preserves domains and codomains. That is to say, a functor $F\mathcal{G} \to \mathcal{C}$ is determined by, and determines, a graph homomorphism $\mathcal{G} \to U\mathcal{C}$, and so the two sets are isomorphic: in some sense, functors between free categories *are* graph homomorphisms. To see that

the isomorphism is natural, observe that it doesn't matter if we precompose a graph homomorphism $\mathcal{G}' \to \mathcal{G}$ (treated as a functor between free categories) or postcompose a functor $\mathcal{C} \to \mathcal{C}'$ (treated as a graph homomorphism): because graph homomorphisms compose preserving the graph structure, we would still have an isomorphism $\mathbf{Cat}(F\mathcal{G}', \mathcal{C}') \cong \mathbf{Graph}(\mathcal{G}', U\mathcal{C}')$.

Before we can properly say that adjoint functors form an adjunction, we need to prove it. As the following proof shows, the mappings $(-)^\sharp$ and $(-)^\flat$ define and are defined by the unit and counit of the adjunction.

Proposition 2.2.26. Functors that form an adjunction in $\mathbf{Cat}$ are exactly adjoint functors.

Proof. We need to show that functors that form an adjunction are adjoint, and that adjoint functors form an adjunction; that is, we need to show that any pair of functors $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ satisfying the definition of adjunction in Definition 2.2.23 necessarily constitute adjoint functors according to Definition 2.2.24, and that if L and R are adjoint according to Definition 2.2.24 then they form an adjunction according to Definition 2.2.23: *i.e.*, the two definitions are equivalent.

We begin by showing that if $L \dashv R$, then L and R are adjoint functors. This means we need to exhibit a natural isomorphism $\mathcal{D}(Lc, d) \cong \mathcal{C}(c, Rd)$. We define a function $(-)^\sharp : \mathcal{D}(Lc, d) \to \mathcal{C}(c, Rd)$ by setting

$$f^\sharp := c \xrightarrow{\eta_c} RLc \xrightarrow{Rf} Rd$$

and a function $(-)^\flat : \mathcal{C}(c, Rd) \to \mathcal{D}(Lc, d)$ by setting

$$g^\flat := Lc \xrightarrow{Lg} LRd \xrightarrow{\epsilon_d} d.$$

We then use naturality and the triangle equalities to show that $f^{\sharp\flat} = f$ and $g^{\flat\sharp} = g$:

$$f^{\sharp\flat} = Lc \xrightarrow{Lf^\sharp} LRd \xrightarrow{\epsilon_d} d \qquad\qquad g^{\flat\sharp} = c \xrightarrow{\eta_c} RLc \xrightarrow{Rg^\flat} Rd$$
$$= Lc \xrightarrow{L\eta_c} LRLc \xrightarrow{LRf} LRd \xrightarrow{\epsilon_d} d \qquad = c \xrightarrow{\eta_c} RLc \xrightarrow{RLc} RLRd \xrightarrow{R\epsilon_d} Rd$$
$$= Lc \xrightarrow{L\eta_c} LRLc \xrightarrow{\epsilon_{Lc}} Lc \xrightarrow{f} d \qquad = c \xrightarrow{g} Rd \xrightarrow{\eta_{Rd}} RLRd \xrightarrow{R\epsilon_d} Rd$$
$$= Lc \xrightarrow{f} d \qquad\qquad\qquad\qquad = c \xrightarrow{g} Rd$$

In each case the first two lines follow by definition, the third by naturality, and the fourth by the triangle equality; hence we have an isomorphism $\mathcal{D}(Lc, d) \cong \mathcal{C}(c, Rd)$. The naturality of this isomorphism follows from the naturality of η and ϵ. We first check that the isomorphisms $(-)^\sharp$ are

natural in c, which means that the following squares commute for all $\phi : c' \to c$ in $\mathcal{C}$:

$$
\begin{array}{ccc}
\mathcal{D}(Lc, d) & \xrightarrow{(-)^{\sharp}_{c,d}} & \mathcal{C}(c, Rd) \\
\downarrow \scriptstyle{\mathcal{D}(L\phi,d)} & & \downarrow \scriptstyle{\mathcal{C}(\phi,Rd)} \\
\mathcal{D}(Lc', d) & \xrightarrow[(-)^{\sharp}_{c',d}]{} & \mathcal{C}(c', Rd)
\end{array}
$$

This requires in turn that $(f \circ L\phi)^{\sharp} = f^{\sharp} \circ \phi$, which we can check as follows:

$$
\begin{aligned}
(f \circ L\phi)^{\sharp} &= c' \xrightarrow{\eta_{c'}} RLc' \xrightarrow{RL\phi} RLc \xrightarrow{Rf} Rd \\
&= c' \xrightarrow{\phi} c \xrightarrow{\eta_c} RLc \xrightarrow{Rf} Rd \\
&= c' \xrightarrow{\phi} c \xrightarrow{f^{\sharp}} Rd
\end{aligned}
$$

where the second equality holds by the naturality of η. The naturality of $(-)^{\sharp}$ in d requires that $(\phi' \circ f)^{\sharp} = R\phi' \circ f^{\sharp}$ for all $\phi' : d \to d'$, which can be checked almost immediately:

$$
\begin{aligned}
(\phi' \circ f)^{\sharp} &= c \xrightarrow{\eta_c} RLc \xrightarrow{Rf} Rd \xrightarrow{R\phi'} Rd' \\
&= c \xrightarrow{f^{\sharp}} Rd \xrightarrow{R\phi'} Rd'
\end{aligned}
$$

Dually, the naturality of $(-)^{\flat} : \mathcal{C}(c, Rd) \to \mathcal{D}(Lc, d)$ in d requires that $(R\phi' \circ g)^{\flat} = \phi' \circ g^{\flat}$ for all $\phi' : d \to d'$, which obtains by the naturality of ϵ:

$$
\begin{aligned}
(R\phi' \circ g)^{\flat} &= Lc \xrightarrow{Lg} LRd \xrightarrow{LR\phi'} LRd' \xrightarrow{\epsilon_{d'}} d' \\
&= Lc \xrightarrow{Lg} LRd \xrightarrow{\epsilon_d} d \xrightarrow{\phi'} d' \\
&= Lc \xrightarrow{g^{\flat}} d \xrightarrow{\phi'} d'
\end{aligned}
$$

The naturality of $(-)^{\flat}$ in c, which requires that $(g \circ \phi)^{\flat} = g^{\flat} \circ L\phi$, obtains similarly immediately:

$$
\begin{aligned}
(g \circ \phi)^{\flat} &= Lc' \xrightarrow{L\phi} Lc \xrightarrow{Lg} LRd \xrightarrow{\epsilon_d} d \\
&= Lc' \xrightarrow{L\phi} Lc \xrightarrow{g^{\flat}} d
\end{aligned}
$$

Thus $(-)^{\sharp}$ and $(-)^{\flat}$ are both natural in c and d, and hence L and R are adjoint functors.

To show the converse, that if $L : \mathcal{C} \to \mathcal{D}$ and $R : \mathcal{D} \to \mathcal{C}$ are adjoint functors then $L \dashv R$, we need to establish natural transformations $\eta : \mathrm{id}_{\mathcal{C}} \Rightarrow RL$ and $\epsilon : LR \Rightarrow \mathrm{id}_{\mathcal{D}}$ from the natural isomorphisms $(-)^{\sharp}$ and $(-)^{\flat}$, such that the triangle equalities $\epsilon_L \circ L\eta = \mathrm{id}_L$ and $R\epsilon \circ \eta_R = \mathrm{id}_R$ are satisfied. We first define η componentwise, by observing that η_c must have the type $c \to RLc$, and

that the image of $\mathsf{id}_{Lc} : Lc \to Lc$ under $(-)^\sharp$ is of this type, and therefore defining $\eta_c := (\mathsf{id}_{Lc})^\sharp$.

Dually, we define ϵ by observing that ϵ_d must have the type $LRd \to d$, and that the image of id_{Rd} under $(-)^\flat$ has this type. We therefore define $\epsilon_d := (\mathsf{id}_{Rd})^\flat$. To see that these definitions constitute natural transformations, observe that they are themselves composed from natural transformations. Explicitly, the naturality of η means that for any $f : c \to c'$, we must have $RLf \circ \eta_c = \eta_{c'} \circ f$, and the naturality of ϵ means that for any $g : d \to d'$, we must have $g \circ \epsilon_d = \epsilon_{d'} \circ LRg$. These obtain as follows:

$$
\begin{aligned}
RLf \circ \eta_c &= c \xrightarrow{(\mathsf{id}_{Lc})^\sharp} RLc \xrightarrow{RLf} RLc' \\
&= c \xrightarrow{(Lf \circ \mathsf{id}_{Lc})^\sharp} RLc' \\
&= c \xrightarrow{(\mathsf{id}_{Lc'} \circ Lf)^\sharp} RLc' \\
&= c \xrightarrow{f} c' \xrightarrow{(\mathsf{id}_{Lc'})^\sharp} RLc' \\
&= \eta_{c'} \circ f
\end{aligned}
\qquad
\begin{aligned}
g \circ \epsilon_d &= LRd \xrightarrow{(\mathsf{id}_{Rd})^\flat} d \xrightarrow{g} d' \\
&= LRd \xrightarrow{(Rg \circ \mathsf{id}_{Rd})^\flat} d' \\
&= LRd \xrightarrow{(\mathsf{id}_{Rd'} \circ Rg)^\flat} d' \\
&= LRd \xrightarrow{LRg} LRd' \xrightarrow{(\mathsf{id}_{Rd'})^\flat} d' \\
&= \epsilon_{d'} \circ LRg
\end{aligned}
$$

In each case, the first equality holds by definition, the second by naturality of $(-)^\sharp$ and $(-)^\flat$ (left and right, respectively) in d, the third by naturality of id, the fourth by naturality in c, and the last by definition. It remains to check that η and ϵ so defined satisfy the triangle equalities. Expressed componentwise, we demonstrate that $\epsilon_{Lc} \circ L\eta_c = \mathsf{id}_{Lc}$ and that $R\epsilon_d \circ \eta_{Rd} = \mathsf{id}_{Rd}$ as follows:

$$
\begin{aligned}
\epsilon_{Lc} \circ L\eta_c &= Lc \xrightarrow{L(\mathsf{id}_{Lc})^\sharp} LRLc \xrightarrow{(\mathsf{id}_{RLc})^\flat} Lc \\
&= Lc \xrightarrow{(\mathsf{id}_{RLc} \circ (\mathsf{id}_{Lc})^\sharp)^\flat} Lc \\
&= Lc \xrightarrow{(\mathsf{id}_{Lc})^{\sharp\flat}} Lc \\
&= Lc \xrightarrow{\mathsf{id}_{Lc}} Lc
\end{aligned}
\qquad
\begin{aligned}
R\epsilon_d \circ \eta_{Rd} &= Rd \xrightarrow{(\mathsf{id}_{LRd})^\sharp} RLRd \xrightarrow{R(\mathsf{id}_{Rd})^\flat} Rd \\
&= Rd \xrightarrow{((\mathsf{id}_{Rd})^\flat \circ \mathsf{id}_{LRd})^\sharp} Rd \\
&= Rd \xrightarrow{(\mathsf{id}_{Rd})^{\flat\sharp}} Rd \\
&= Rd \xrightarrow{\mathsf{id}_{Rd}} Rd
\end{aligned}
$$

The first equality (on each side) holds by definition, the second (on the left) by naturality of $(-)^\flat$ in c and (on the right) by naturality of $(-)^\sharp$ in d, the third by unitality of composition, and the fourth by the $\sharp/\flat$ isomorphism. This establishes that $L \dashv R$, and hence the result. $\qquad\square$

Sometimes, the 'distances' measured by the unit and counit are so small that the categories $\mathcal{C}$ and $\mathcal{D}$ are actually 'equivalent': this happens when the unit and counit are natural isomorphisms, meaning that the isomorphic core of the adjunction extends to the whole of $\mathcal{C}$ and $\mathcal{D}$. This gives us the following definition.

Definition 2.2.27. Suppose $L \dashv R$ in a 2-category. When the unit and counit of the adjunction are additionally isomorphisms, we say that L and R form an *adjoint equivalence*.

Remark 2.2.28. More generally, an *equivalence* of categories is a pair of functors connected by natural isomorphisms of the form of the unit and counit of an adjunction, but which may not necessarily satisfy the triangle identities; however, given any such equivalence, it is possible to modify the unit or counit so as to upgrade it to an adjoint equivalence. Henceforth, we will have no need to distinguish equivalences from adjoint equivalences, so we will say simply 'equivalence' for both. If there is an equivalence between a pair of categories, then we will say that the two categories are *equivalent*.

Note that the notion of equivalence of categories can be generalized to *equivalence in a 2-category*, by replacing the categories by 0-cells, the functors by 1-cells, and the natural isomorphisms by invertible 2-cells.

The structure of an equivalence of categories can alternatively be specified as properties of the functors concerned, which in some situations can be easier to verify.

Definition 2.2.29. We say that a functor $F : \mathcal{C} \to \mathcal{D}$ is

1. *full* when it is surjective on hom sets, in the sense that the functions $F_{a,b} : \mathcal{C}(a,b) \to \mathcal{D}(Fa, Fb)$ are surjections;

2. *faithful* when it is injective on hom sets, in the sense that the functions $F_{a,b}$ are injections;

3. *fully faithful* when it is both full and faithful (*i.e.*, isomorphic on hom sets); and

4. *essentially surjective* when it is surjective on objects up to isomorphism, in the sense that for every object $d : \mathcal{D}$ there is an object $c : \mathcal{C}$ such that $Fc \cong d$.

Proposition 2.2.30. Categories $\mathcal{C}$ and $\mathcal{D}$ are equivalent if and only if there is a functor $F : \mathcal{C} \to \mathcal{D}$ that is fully faithful and essentially surjective.

Proof [213, Lemma 9.4.5]. First, we show that if $F \dashv G : \mathcal{D} \to \mathcal{C}$ is an equivalence of categories, then $F : \mathcal{C} \to \mathcal{D}$ is fully faithful and essentially surjective. For the latter, observe that G gives us, for any $d : \mathcal{D}$, an object $Gd : \mathcal{C}$ and ϵ_d is by definition an isomorphism $FGd \xrightarrow{\sim} d$; hence F is essentially surjective. To show that F is fully faithful means showing that each function $F_{a,b} : \mathcal{C}(a,b) \to \mathcal{D}(Fa, Fb)$ is an isomorphism; we can define the inverse $F_{a,b}^{-1}$ as the following composite:

$$\mathcal{D}(Fa, Fb) \xrightarrow{G_{Fa,Fb}} \mathcal{C}(GFa, GFb) \xrightarrow{\mathcal{C}(\eta_a, \eta_b^{-1})} \mathcal{C}(a,b)$$
$$g \qquad\qquad \mapsto \qquad\qquad Gg \qquad\qquad \mapsto \qquad \left(a \xrightarrow{\eta_a} GFa \xrightarrow{Gg} GFb \xrightarrow{\eta_b^{-1}} b\right)$$

Here, the function $\mathcal{C}(\eta_a, \eta_b^{-1})$ is the function $f \mapsto \eta_b^{-1} \circ f \circ \eta_a$ obtained from the hom functor (Remark 2.2.15). Hence $F_{a,b}^{-1}(g) := \eta_b^{-1} \circ Gg \circ \eta_a$. To see that this is indeed the desired inverse, consider applying the functor F to the morphism $F_{a,b}^{-1}(g)$; we have the following equalities:

$$Fa \xrightarrow{F\eta_a} FGFa \xrightarrow{FGg} FGFb \xrightarrow{F\eta_b^{-1}} Fb$$
$$= Fa \xrightarrow{g} Fb \xrightarrow{F\eta_b} FGFb \xrightarrow{F\eta_b^{-1}} Fb$$
$$= Fa \xrightarrow{g} Fb$$

where the first equality holds by the naturality of η and the second equality holds since η_b is an isomorphism. Since F is therefore isomorphic on hom sets, it is fully faithful.

Next, we show that if $F : \mathcal{C} \to \mathcal{D}$ is fully faithful and essentially surjective, then there is a functor $G : \mathcal{D} \to \mathcal{C}$ and natural isomorphisms $\eta : \mathrm{id}_{\mathcal{C}} \Rightarrow GF$ and $\epsilon : FG \Rightarrow \mathrm{id}_{\mathcal{D}}$. On objects $d : \mathcal{D}$, we can define $Gd : \mathcal{C}$ as any choice of object such that $FGd \xrightarrow{\sim} d$: such an object must exist since F is essentially surjective. We then define ϵ_d to be the associated isomorphism $FGd \to d$; it is easy to verify that ϵ so defined is natural. On morphisms, let the functions $G_{d,e}$ be defined as the composite functions

$$
\begin{array}{ccccc}
\mathcal{D}(d,e) & \xrightarrow{\mathcal{D}(\epsilon_d, \epsilon_e^{-1})} & \mathcal{D}(FGd, FGe) & \xrightarrow{F_{Gd,Ge}^{-1}} & \mathcal{C}(Gd, Ge) \\
g & \mapsto & \left(FGd \xrightarrow{\epsilon_d} d \xrightarrow{g} e \xrightarrow{\epsilon_e^{-1}} FGe\right) & \mapsto & F_{Gd,Ge}^{-1}\left(\epsilon_e^{-1} \circ g \circ \epsilon_d\right)
\end{array}.
$$

Since F is a fully faithful functor and ϵ is a natural isomorphism, this makes G a well-defined functor. Finally, we define η as having the components $\eta_c := F_{c,GFc}^{-1}\left(\epsilon_{Fc}^{-1}\right)$; since ϵ is a natural isomorphism, so is ϵ^{-1}, which is thus preserved as such by the inverse action of F in defining η. This establishes all the data of the equivalence.

(Note that we can actually prove a little more: it is not hard to verify additionally that the two constructions are inverse, so that equivalences are themselves equivalent to fully faithful essentially surjective functors.) $\qquad\square$

Remark 2.2.31. In the above proof, we assumed the axiom of choice, defining Gd as a choice of object such that $FGd \xrightarrow{\sim} d$. It is possible to avoid making this assumption, by asking for the surjection on objects $F_0 : \mathcal{C}_0 \to \mathcal{D}_0$ to be 'split' in the sense that it comes with a function $s : \mathcal{D}_0 \to \mathcal{C}_0$ such that $F_0(s(d)) \cong d$ in $\mathcal{D}$ for every object $d : \mathcal{D}$; then we just set $Gd := s(d)$.

2.3. Universal constructions

In the preceding sections, we have used diagrams to represent some patterns in a categorical context, and we have discussed how functors allow us to translate patterns and structures between contexts; indeed, we used functors to formalize diagrams themselves. But an important facet of the notion of pattern is replication across contexts, and in many important situations, we will encounter patterns that apply to all objects in a category. We call such patterns *universal*, and much of science is a search for such universal patterns: for example, much of physics, and by extension much of the theory of the Bayesian brain, is a study of the universal principle of stationary action. In this section, we introduce the formal characterization of universality and exemplify it with some examples that will be particularly important later on — as well as some examples that we have encountered already.

2.3.1. The universality of common patterns

We begin with some basic examples of universal patterns.

2.3.1.1. Disjunctions, or coproducts

Our first example of a universal pattern is the *coproduct*, which captures the essence of the following examples — situations like disjunction, where there is an element of choice between alternatives.

Example 2.3.1. Given two propositions, such as $\mathsf{P}_1 := $ " $-$ is flat" and $\mathsf{P}_2 := $ " $-$ is sharp", we can form their disjunction $\mathsf{P}_1 \vee \mathsf{P}_2$ (meaning " $-$ is flat or sharp"). Similarly, given two subsets $P_1, P_2 \subseteq X$, we can form their *join* or *union*, $P_1 \cup P_2$: an element x is an element of $P_1 \cup P_2$ if (and only if) it is an element of P_1 or an element of P_2.

Example 2.3.2. Given two numbers, we can form their *sum*; for instance, $1 + 2 = 3$. More generally, given two sets A and B, we can form their *disjoint union*, denoted $A + B$. The elements of $A + B$ are pairs (i, x) where x is an element of A or of B and i indicates which set x is drawn from (this ensures that if an element x of A is the same as an element of B, it is added twice to the disjoint union). Therefore, if A has 1 element and B has 2, then $A + B$ has 3 elements.

Remark 2.3.3. The preceding example illustrates how we can think of numbers equivalently as sets of the indicated cardinality. Many operations on sets are generalizations of familiar operations on numbers in this way.

Example 2.3.4. Given two graphs, G and G', we can form the sum graph $G + G'$, whose set of nodes is $G_0 + G'_0$ and whose set of edges is $G_1 + G'_1$.

Example 2.3.5. Given two vector spaces V and V', we can form their *direct sum* $V \oplus V'$, whose vectors are linear combinations of vectors either in V or in V'.

Each of these is an example of a coproduct, which we now define.

Definition 2.3.6. Given objects A and B in a category $\mathcal{C}$, their *coproduct* (if it exists) is an object, canonically denoted $A + B$, equipped with two morphisms $\mathrm{inj}_A : A \to A + B$ and $\mathrm{inj}_B : B \to A + B$ such that, for any object Q equipped with morphisms $f : A \to Q$ and $g : B \to Q$, there is a unique morphism $u : A + B \to Q$ such that $f = u \circ \mathrm{inj}_A$ and $g = u \circ \mathrm{inj}_B$. The morphisms inj_A and inj_B are called *injections*, and the morphism u is called the *copairing* and often denoted by $[f, g]$.

Example 2.3.7. Morphisms of subsets are inclusions, so given subsets $P_1, P_2 \subseteq X$, there are evident inclusions $P_1 \subseteq P_1 \cup P_2$ and $P_2 \subseteq P_1 \cup P_2$. Moreover, given a subset Q such that $P_1 \subseteq Q$ and $P_2 \subseteq Q$, it is clearly the case that $P_1 \subseteq P_1 + P_2 \subseteq Q$ and $P_2 \subseteq P_1 + P_2 \subseteq Q$.

Similarly, morphisms of propositions are implications, so given P_1 and P_2 such that $\mathsf{P}_1 \to Q$ and $\mathsf{P}_2 \to Q$, then it is necessarily the case that $\mathsf{P}_1 \to \mathsf{P}_1 \vee \mathsf{P}_2 \to Q$ and $\mathsf{P}_2 \to \mathsf{P}_1 \vee \mathsf{P}_2 \to Q$: clearly, both P_1 and P_2 imply $\mathsf{P}_1 \vee \mathsf{P}_2$ by definition, and if both P_1 and P_2 imply Q, then so does $\mathsf{P}_1 \vee \mathsf{P}_2$.

Example 2.3.8. Given sets A and B, the injections $\mathrm{inj}_A : A \to A + B$ and $\mathrm{inj}_B : B \to A + B$ are the corresponding inclusions: inj_A maps a to $(1, a)$ and inj_B maps b to $(2, b)$, where 1 tags an element as coming from A, and 2 tags an element as coming from B. Given $f : A \to Q$ and $g : B \to Q$ the copairing $[f, g] : A + B \to Q$ is the function that takes an element (i, x) and returns $f(x)$ if $i = 1$ or $g(x)$ if $i = 2$; it is from this that the 'choice' interpretation arises for the coproduct.

Example 2.3.9. Morphisms of vectors spaces are linear maps, and if the spaces are finite-dimensional, then we can represent these maps as matrices: if V is n-dimensional and W is m-dimensional, then a morphism $V \to W$ is a matrix of shape (m, n); writing the elements of V and W as column vectors, such a matrix has m rows and n columns. Moreover, in this case, the direct sum $V \oplus W$ is $(n + m)$-dimensional.

Therefore suppose that V, V' and W have respective dimensions n, n' and m, and suppose we have linear maps $f : V \to W$ and $g : V' \to W$. The injection $V \to V \oplus V'$ is the block matrix $\begin{pmatrix} 1_n \\ 0_{n'} \end{pmatrix}$ where 1_n is the n-by-n identity matrix and $0_{n'}$ is the n'-by-n' zero matrix; similarly, the

injection $V' \to V \oplus V'$ is the block matrix $\begin{pmatrix} 0_n \\ 1_{n'} \end{pmatrix}$. And the copairing $[f, g] : V \oplus V' \to W$ is the block matrix $\begin{pmatrix} f & g \end{pmatrix}$.

Remark 2.3.10. The coproducts we have considered so far have all been *binary*, being coproducts of only two objects. More generally, we can often consider coproducts of more objects, by repeating the binary coproduct operation; typically, there is an isomorphism $(A + B) + C \cong A + (B + C)$. We can extend this further to finite (and, often, infinite) collections of objects. Suppose then that $\{A_i\}$ is a collection of objects indexed by $i : I$, where I is a set, and form the iterated coproduct $\sum_{i:I} A_i$; we will call this object a *dependent sum*, because the summands A_i *depend on* $i : I$. In the case where the objects A_i are all sets, the dependent sum $\sum_i A_i$ is again a set, whose elements are pairs (i, x) where i is an element of I and x is an element of A_i. In other categories $\mathcal{C}$, typically the name *dependent sum* is reserved for the case when all of the objects A_i and the object I are objects of $\mathcal{C}$. But when I remains a set, we may still be able to form the I-indexed coproduct $\sum_i A_i$ in $\mathcal{C}$.

2.3.1.2. Conjunctions, products, and sections

Our next example of a universal pattern is the *product*, which captures situations like conjunction, in which things come along in separable pairs of individuals.

Example 2.3.11. Given two propositions, such as $\mathsf{P}_1 := $ "$-$ is small" and $\mathsf{P}_2 := $ "$-$ is connected", we can form their conjunction $\mathsf{P}_1 \wedge \mathsf{P}_2$ (meaning "$-$ is small and connected"). Similarly, given two subsets $P_1, P_2 \subseteq X$, we can form their *meet* or *intersection*, $P_1 \cap P_2$: an element x is an element of $P_1 \cap P_2$ if (and only if) it is an element of P_1 and an element of P_2.

Example 2.3.12. Given two numbers, we can form their *product*; for instance, $2 \times 3 = 6$. More generally, as we saw in Definition 2.2.1, we can form the product of any two sets A and B, denoted $A \times B$. The elements of $A \times B$ are pairs (a, b) where a is an element of A and b is an element of B. Therefore, if A has 2 elements and B has 3, then $A \times B$ has 6 elements.

Remark 2.3.13. When all the summands of a dependent sum are the same set or object A regardless of their associated index $i : I$, then the object $\sum_{i:I} A$ is isomorphic to the product $I \times A$: this is simply a categorification of the fact that "multiplication is repeated addition".

Example 2.3.14. Given vector spaces V and V' (of respective dimensions n and n'), their product is again the direct sum $V \oplus V'$. Since the direct sum of vector spaces is both a product and a coproduct, it is also said to be a *biproduct*.

Categorically, the product is the dual of the coproduct.

Definition 2.3.15. Given objects A and B in a category $\mathcal{C}$, their *product* (if it exists) is an object, canonically denoted $A \times B$, equipped with two morphisms $\mathsf{proj}_A : A \times B \to A$ and $\mathsf{proj}_B : A \times B \to B$ such that, for any object Q equipped with morphisms $f : Q \to A$ and $g : Q \to B$, there is a unique morphism $u : Q \to A \times B$ such that $f = \mathsf{proj}_A \circ u$ and $g = \mathsf{proj}_B \circ u$. The morphisms proj_A and proj_B are called *projections*, and the morphism u is called the *pairing* and often denoted by (f, g).

Example 2.3.16. Given subjects $P_1, P_2 \subseteq X$, there are evident inclusions $P_1 \cap P_2 \subseteq P_1$ and $P_1 \cap P_2 \subseteq P_2$. Moreover, given a subset Q such that $Q \subseteq P_1$ and $Q \subseteq P_2$, it is clearly then the case that $Q \subseteq P_1 \cap P_2 \subseteq P_1$ and $Q \subseteq P_1 \cap P_2 \subseteq P_2$.

Similarly, given propositions P_1 and P_2 such that $Q \to \mathsf{P}_1$ and $Q \to \mathsf{P}_2$, it is (by the definition of "and") the case that $Q \to \mathsf{P}_1 \wedge \mathsf{P}_2 \to \mathsf{P}_1$ and $Q \to \mathsf{P}_1 \wedge \mathsf{P}_2 \to \mathsf{P}_2$.

Example 2.3.17. Given sets A and B, the projections $\mathsf{proj}_A : A \times B \to A$ and $\mathsf{proj}_B : A \times B \to B$ are the functions $(a, b) \mapsto a$ and $(a, b) \mapsto b$ respectively. Given $f : Q \to A$ and $g : Q \to B$, the pairing $(f, g) : Q \to A \times B$ is the function $x \mapsto \big(f(x), g(x)\big)$; note that this involves 'copying' x, which will be relevant when we come to consider copy-discard categories in §3.1.

Remark 2.3.18. Above, we observed that a coproduct with constant summands A is equivalently a product $I \times A$ of the indexing object I with A; we therefore have a projection $\mathsf{proj}_I : I \times A \to I$. More generally, for any dependent sum $\sum_{i:I} A_i$, there is a projection $\sum_{i:I} A_i \to I$; in the case of dependent sums in **Set**, this is unsurprisingly the function $(i, x) \mapsto i$.

Example 2.3.19. Suppose we have vector spaces V, V' and W of respective dimensions n, n' and m. The projection $V \oplus V' \to V$ is the block matrix $\begin{pmatrix} 1_n & 0_{n'} \end{pmatrix}$, and the projection $V \oplus V' \to V'$ is the block matrix $\begin{pmatrix} 0_n & 1_{n'} \end{pmatrix}$. Given linear maps $f : W \to V$ and $g : W \to V'$, the pairing $(f, g) : W \to V \oplus V'$ is the block matrix $\begin{pmatrix} f \\ g \end{pmatrix}$. Note that, in a sign of the duality between products and coproducts, the projections and the pairing are respectively the injections and the copairing transposed.

Remark 2.3.20. Just as in the case of coproducts, we can also consider products of more than two objects, by repeating the product operation; there is again typically an isomorphism $(A \times B) \times C \cong A \times (B \times C)$. If $\{A_i\}$ is a collection of objects indexed by $i : I$ (with I again a set), we can form

the *dependent product*[1] $\prod_{i:I} A_i$. In the case where I is finite and the objects A_i are all sets, the dependent product $\prod_{i:I} A_i$ is again a set, whose elements can equivalently be seen as lists $(a_i, \dots)$ indexed by i with each element a_i drawn from the corresponding set A_i or as functions s with domain I and codomain the dependent sum $\sum_{i:I} A_i$ such that each $s(i)$ is tagged by i. This situation is summarized by the commutativity of the diagram

$$I \xrightarrow{\ \ s\ \ } \sum_{i:I} A_i \downarrow \pi I$$

where π is the projection and which therefore requires that $\pi \circ s = \mathrm{id}_I$. Such a function s is known as a *section* of p, and we can think of sections therefore as *dependent functions*, since the types of their output values (*i.e.*, A_i) may depend on the input values i.

The notion of section is important enough to warrant a general definition.

Definition 2.3.21. Suppose $p : E \to B$ is a morphism. A *section* of p is a morphism $s : B \to E$ such that $p \circ s = \mathrm{id}_B$.

2.3.1.3. Subobjects and equalizers

Our next examples of universal patterns do not involve pairing or grouping objects together to make new ones. For instance, in many situations, it is of interest to restrict our attention to 'subobjects' (of a single given object) that satisfy a certain property, which may not extend to the whole object at hand.

Example 2.3.22. In examples above, we saw that subsets and propositions behave similarly with respect to disjunctions and conjunctions. More broadly, there is a correspondence between subsets and propositions, if we think of propositions on a set X as functions $X \to 2$, where 2 is the 2-element set $\{\bot, \top\}$ of truth values (where we interpret $\bot$ as 'false' and $\top$ as 'true'). Every subset $A \subseteq X$ has an associated injection, $A \hookrightarrow X$, and there is a correspondence between such injections and propositions $\mathsf{P}_A : X \to 2$, where $\mathsf{P}_A(x)$ is true whenever x is an element of A. This situation

---

[1]This set-indexed product is also known as an *indexed product*, to emphasize that the factors are indexed by the set I; since I has elements, we can properly think of these as indices, which may not be true for other kinds of object. We will see in Definition 2.3.63 how to use categorical structure to abstract away the requirement that I be a set.

can be summarized by the commutativity of the diagram

$$
\begin{array}{ccc}
A & \xrightarrow{\ !\ } & 1 \\
\downarrow & & \downarrow{\scriptstyle \top} \\
X & \xrightarrow{\ \mathsf{P}_A\ } & 2
\end{array}
$$

where 1 is the singleton set $\{*\}$, $!$ is the unique function sending every element of A to $*$, and $\top$ is the function $* \mapsto \top$ picking the truth value $\top$. If, in a category $\mathcal{C}$, there is an object such that, for any subobject $A \hookrightarrow X$, there is a unique morphism $X \to 2$ such that the above diagram commutes (and moreover defines a *pullback square* in the sense of Example 2.3.43 below), then we say that the object 2 is a *subobject classifier* in $\mathcal{C}$; in this case, we interpret 1 as the 'terminal' object in $\mathcal{C}$ (introduced below, in Example 2.3.41).

A pattern that will be particularly common is that in which we care about a subset of elements of a set that make two functions equal. This can be generalized to arbitrary categories using the following notion.

Definition 2.3.23. Suppose f and g are both morphisms $X \to Y$. Their *equalizer* is an object E equipped with a function $e : E \to X$ such that $f \circ e = g \circ e$ (so that e is said to *equalize* f and g) as in the commuting diagram

$$
E \xrightarrow{\ e\ } X \underset{g}{\overset{f}{\rightrightarrows}} Y
$$

and such that, for any $d : D \to X$ equalizing f and g, there is a unique morphism $u : D \to E$ such that $d = e \circ u$, as in the diagram

$$
D \overset{u}{\dashrightarrow} E \xrightarrow{\ e\ } X \underset{g}{\overset{f}{\rightrightarrows}} Y \quad .
$$

Example 2.3.24. Via the correspondence between subsets and propositions, we can express the conjunction of propositions as an equalizer. Suppose have have two propositions $\mathsf{P}_A : X \to 2$ and $\mathsf{P}_B : X \to 2$, corresponding to subsets $A \hookrightarrow X$ and $B \hookrightarrow X$, whose inclusions we denote by ι_A and ι_B respectively. The equalizer of $\quad A \times B \underset{\iota_B \circ \mathsf{proj}_B}{\overset{\iota_A \circ \mathsf{proj}_A}{\rightrightarrows}} X \quad$ is the subset of $A \times B$ whose elements are pairs (a, b) in which $a = b$ in X. This subset is isomorphic to the meet $A \cap B$, which corresponds as a proposition to the conjunction $\mathsf{P}_A \wedge \mathsf{P}_B$.

2.3.1.4. Coequalizers and quotients

We can also make objects 'smaller' by dividing them into equivalence classes, as we did when quotienting free categories by given relations (*cf.* Proposition 2.1.13). In general, this pattern is captured by the notion of *coequalizer*, which is dual to the notion of equalizer in the same way that coproducts are dual to products.

Definition 2.3.25. Suppose f and g are both morphisms $X \to Y$. Their *coequalizer* is an object P equipped with a function $p : Y \to P$ such that $p \circ f = p \circ g$ (with p said to *coequalize* f and g) as in the commuting diagram

$$X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{\ p\ } P$$

and such that, for any $q : Y \to Q$ coequalizing f and g, there is a unique morphism $u : P \to Q$ such that $q = u \circ p$, as in the diagram

$$X \underset{g}{\overset{f}{\rightrightarrows}} Y \xrightarrow{\ p\ } P \dashrightarrow^{u} Q \ .$$

Example 2.3.26. A relation $\sim$ on a set X is a proposition on $X \times X$, and thus equivalently a subset $R \hookrightarrow X \times X$; let ι denote the inclusion. The coequalizer of $R \underset{\mathsf{proj}_2 \circ \iota}{\overset{\mathsf{proj}_1 \circ \iota}{\rightrightarrows}} X$ is the set of equivalence classes of X according to $\sim$, which is precisely the quotient $X/\sim$.

2.3.2. The pattern of universality

There is a common pattern to the common patterns above: in each case, we described an object U equipped with some morphisms, such that, given any object X with morphisms of a similar shape, there was a unique morphism u relating X and U. The existence of such a unique morphism for any comparable X makes the object U a *universal* representative of the situation at hand and has a number of powerful consequences: in particular, it entirely characterizes the object U up to isomorphism. Much of the power of category theory comes from the use of universal properties to classify, compare, and reason about situations of interest — for the general notion of universality itself can be characterized categorically.

Definition 2.3.27. Suppose $F : \mathcal{C} \to \mathcal{D}$ is a functor and $X : \mathcal{D}$ an object. We define two dual universal constructions. A *universal morphism from X to F* is a morphism $u : X \to FU$ for a corresponding *universal object* $U : \mathcal{C}$ such that for any $f : X \to FV$ in $\mathcal{D}$ there exists a unique $e : U \to V$ such that $f = X \xrightarrow{u} FU \xrightarrow{Fe} FV$.

Dually, a *universal morphism from F to X* is a morphism $u : FU \to X$ for a given $U : \mathcal{C}$ such that for any $f : FV \to X$ in $\mathcal{D}$ there exists a unique $e : V \to U$ such that $f = FV \xrightarrow{Fe} FU \xrightarrow{u} X$.

We can now formalize the universal properties of the examples we met above, beginning with the coproduct.

Example 2.3.28. Let $\Delta : \mathcal{C} \to \mathcal{C} \times \mathcal{C}$ denote the functor $X \mapsto (X, X)$. A coproduct of X and Y in $\mathcal{C}$ is a universal morphism from the object (X, Y) in $\mathcal{C} \times \mathcal{C}$ to Δ: that is, an object $X + Y$ in $\mathcal{C}$ and a morphism $(\mathsf{inj}_X, \mathsf{inj}_Y) : (X, Y) \to (X + Y, X + Y)$ in $\mathcal{C} \times \mathcal{C}$ such that, for any $(f, g) : (X, Y) \to (Q, Q)$ in $\mathcal{C} \times \mathcal{C}$, the copairing $[f, g] : X + Y \to Q$ uniquely satisfies the equation $(f, g) = (X, Y) \xrightarrow{(\mathsf{inj}_X, \mathsf{inj}_Y)} (X + Y, X + Y) \xrightarrow{([f,g],[f,g])} (Q, Q)$.

Example 2.3.29. Again let $\Delta : \mathcal{C} \to \mathcal{C} \times \mathcal{C}$ denote the functor $X \mapsto (X, X)$. A product of X and Y in $\mathcal{C}$ is a universal morphism from the object $(X, Y) : \mathcal{C} \times \mathcal{C}$ to Δ: that is, an object $X \times Y$ in $\mathcal{C}$ and a morphism $(\mathsf{proj}_X, \mathsf{proj}_Y) : (X \times Y, X \times Y) \to (X, Y)$ in $\mathcal{C} \times \mathcal{C}$ such that, for any $(f, g) : (Q, Q) \to (X, Y)$ in $\mathcal{C} \times \mathcal{C}$, the pairing $(f, g) : Q \to X \times Y$ uniquely satisfies the equation $(f, g) = (Q, Q) \xrightarrow{((f,g),(f,g))} (X \times Y, X \times Y) \xrightarrow{(\mathsf{proj}_X, \mathsf{proj}_Y)} (X, Y)$.

Remark 2.3.30. If we let $\mathbf{2}$ denote the two-object discrete category $\{\bullet \ \bullet\}$, then there is an equivalence $\mathcal{C} \times \mathcal{C} \cong \mathcal{C}^{\mathbf{2}}$ and so a pair of morphisms in $\mathcal{C}$ is equivalently a natural transformation in $\mathcal{C}^{\mathbf{2}}$. (This is a categorification of the familiar fact that "exponentiation is repeated multiplication", which we will explore in §2.3.4.)

Consequently, the functor Δ from the preceding examples can equivalently be defined as a functor $\mathcal{C} \to \mathcal{C}^{\mathbf{2}}$. Letting the exponent take a more general shape, we obtain a family of constant functors.

Proposition 2.3.31. Suppose $\mathcal{C}$ and $\mathcal{D}$ are categories, and $d : \mathcal{D}$ is an object. Then there is a *constant functor* on d, denoted $\Delta_d : \mathcal{C} \to \mathcal{D}$, which takes each object $c : \mathcal{C}$ to $d : \mathcal{D}$ and each morphism $f : c \to c'$ to id_d; note that $Fc = d = Fc'$. The assignment $d \mapsto \Delta_d$ is itself trivially functorial, giving a functor $\Delta : \mathcal{D} \to \mathcal{D}^{\mathcal{C}}$ which we call the *constant functor functor*.

Example 2.3.32. Let $\mathcal{J}$ be the category with two objects, 1 and 2, and two non-identity morphisms $\alpha, \beta : 1 \to 2$, as in the diagram $1 \xrightarrow[\beta]{\alpha} 2$, and let Δ be the constant functor functor $\mathcal{C} \to \mathcal{C}^{\mathcal{J}}$. Now suppose f and g are two morphisms $X \to Y$ in $\mathcal{C}$. To construct their equalizer as a universal morphism, let D be the diagram $\mathcal{J} \to \mathcal{C}$ mapping $\alpha \mapsto f$ and $\beta \mapsto g$. Then an equalizer of f and g is a universal morphism from Δ to D (with D being an object of the functor category

$\mathcal{C}^{\mathcal{J}}$): that is, an object $E : \mathcal{C}$ equipped with a natural transformation $\epsilon : \Delta_E \Rightarrow D$ satisfying the universal property that, for any $\varphi : \Delta_F \Rightarrow D$ there exists a unique morphism $u : F \to E$ such that $\varphi = \epsilon \circ \Delta_u$.

Unraveling this definition, we find that such a natural transformation ϵ consists of a pair of morphisms $\epsilon_1 : E \to X$ and $\epsilon_2 : E \to Y$ making the following naturality squares commute:

$$
\begin{array}{ccc}
E & \xrightarrow{\epsilon_1} & X \\
\| & & \downarrow{\scriptstyle f} \\
E & \xrightarrow{\epsilon_2} & Y
\end{array}
\qquad\qquad
\begin{array}{ccc}
E & \xrightarrow{\epsilon_1} & X \\
\| & & \downarrow{\scriptstyle g} \\
E & \xrightarrow{\epsilon_2} & Y
\end{array}
$$

We can therefore set $\epsilon_1 = e$, where e is the equalizing morphism $E \to X$. The commutativity of the naturality squares enforces that $f \circ e = \epsilon_2 = g \circ e$ and hence that $f \circ e = g \circ e$, which is the first condition defining the equalizer. Unwinding the universal property as expressed here shows that the morphisms φ_1 and u correspond exactly to the morphisms d and u of Definition 2.3.23.

Example 2.3.33. The case of a coequalizer is precisely dual to that of an equalizer. Therefore, let $\mathcal{J}, \Delta$, and D be defined as above. A coequalizer of $f, g : X \to Y$ is then a universal morphism from D to Δ.

Example 2.3.34. In Proposition 2.1.13, we constructed a category generated with relations $F\mathcal{G}/\sim$ as a quotient of a free category on a graph $F\mathcal{G}$. Since this category $F\mathcal{G}/\sim$ is a quotient and quotients are coequalizers (by Example 2.3.26), the projection functor $F\mathcal{G} \to F\mathcal{G}/\sim$ (Example 2.1.26) constitutes the associated universal morphism, in the sense dual to the morphism ϵ_1 of Example 2.3.32.

Example 2.3.35. The free category construction itself (Proposition 2.1.9) exhibits a universal property, as a consequence of the free-forgetful adjunction (Example 2.2.25): given a graph $\mathcal{G}$ and a category $\mathcal{C}$, any functor $F\mathcal{G} \to \mathcal{C}$ is uniquely determined by a graph homomorphism $\mathcal{G} \to U\mathcal{C}$ from $\mathcal{G}$ to the underlying graph of $\mathcal{C}$. More precisely, there is a universal morphism from the singleton set 1 to the functor $\mathbf{Cat}(F-, \mathcal{C})$ for every category $\mathcal{C}$. This means that, for any graph $\mathcal{G}$, every functor $f : F\mathcal{G} \to \mathcal{C}$ factors as $F\mathcal{G} \xrightarrow{Fh} FU\mathcal{C} \xrightarrow{u} \mathcal{C}$ where u is the universal morphism and h is the unique graph homomorphism. This universal property follows abstractly from facts that we will soon encounter: that adjoint functors are 'representable' (Proposition 2.4.16); and that representable functors are universal (Proposition 2.4.23). We hinted at this property in Example 2.2.25, where we observed that functors between free categories 'are' graph homomorphisms: the homomorphism h here is the graph homomorphism corresponding to the functor f, and u renders it as a functor into $\mathcal{C}$.

When an object satisfies a universal property, then this property characterizes the object uniquely: as a result, universal properties are powerful constructions, telling us that for certain questions, there can be only one possible answer. Characterizing an object by a universal property abstracts away from contextually irrelevant details (for example, the particular elements making up a set), and crystallizes its essence.

The uniqueness of universal morphisms is formalized by the following proposition.

Proposition 2.3.36 (Universal constructions are unique up to unique isomorphism). Suppose $u : X \to FU$ and $u' : X \to FU'$ are both universal morphisms from $X : \mathcal{C}$ to $F : \mathcal{C} \to \mathcal{D}$. Then there is a unique isomorphism $i : U \to U'$.

To prove this, we need to know that functors preserve isomorphisms.

Proposition 2.3.37. If $F : \mathcal{C} \to \mathcal{D}$ is a functor and $f : x \to y$ is an isomorphism in $\mathcal{C}$, then $Ff : Fx \to Fy$ is an isomorphism in $\mathcal{D}$.

Proof. For f to be an isomorphism, there must be a morphism $f^{-1} : y \to x$ such that $f^{-1} \circ f = \mathsf{id}_x$ and $f \circ f^{-1} = \mathsf{id}_y$. We have $\mathsf{id}_{Fx} = F(\mathsf{id}_x) = F(f^{-1} \circ f) = Ff^{-1} \circ Ff$, where the first and third equations hold by the functoriality of F and the second equation holds *ex hypothesi*. Similarly, $\mathsf{id}_{Fy} = F(\mathsf{id}_y) = F(f \circ f^{-1}) = Ff \circ Ff^{-1}$. Therefore Ff^{-1} is both a right and left inverse for Ff, and so Ff is an isomorphism. $\square$

Proof of Proposition 2.3.36. Since u' is a morphism from X to F, the universal property of u says that there exists a unique morphism $i : U \to U'$ such that $u' = Fi \circ u$. Similarly, the universal property of u' stipulates that there exists a unique morphism $i' : U' \to U$ such that $u = Fi' \circ u'$. We can substitute the latter into the former and the former into the latter:

$$
\begin{aligned}
u' &= X \xrightarrow{u'} FU' \xrightarrow{i'} FU \xrightarrow{Fi} FU' & u &= X \xrightarrow{u} FU \xrightarrow{Fi} FU' \xrightarrow{Fi'} FU \\
&= X \xrightarrow{u'} FU' \xrightarrow{F(i \circ i')} FU' & &= X \xrightarrow{u} FU \xrightarrow{F(i' \circ i)} FU \\
&= X \xrightarrow{u'} FU' \xrightarrow{F\,\mathsf{id}_{U'}} FU' & &= X \xrightarrow{u} FU \xrightarrow{F\,\mathsf{id}_U} FU
\end{aligned}
$$

and since functors preserve isomorphism, we have $i \circ i' = \mathsf{id}_{U'}$ and $i' \circ i = \mathsf{id}_U$. Therefore, i is an isomorphism which is unique by definition. $\square$

2.3.3. Limits and colimits: mapping in to and out of diagrams

Many of the universal constructions above[2] fall into their own general pattern, in which a diagram of objects and morphisms is specified, and a universal morphism is produced which encodes the data of mapping into or out of that diagram, in a sufficiently parsimonious way that any other way of mapping into or out of the diagram factors through it. In the case of the (co)product, the diagram is simple: simply a pair of objects, with no morphisms between them. In the case of the (co)equalizer, the diagram is a little more complex, being a 'fork' of the form $1 \underset{\beta}{\overset{\alpha}{\rightrightarrows}} 2$. We can generalize these examples further, to consider the most parsimonious ways of mapping into or out of diagrams of arbitrary shape: these universal constructions are called *colimits* and *limits* respectively, and to formalize them, we need to define what it means to map into or out of a diagram. For this purpose, we use the following notion of *cone* over a diagram.

Definition 2.3.38. A *cone* over the J-shaped diagram D in $\mathcal{C}$ is a natural transformation $\Delta_c \Rightarrow D$ for a given object $c : \mathcal{C}$ which we call its *apex*. Dually, a *cocone* under D with apex c is a natural transformation $D \Rightarrow \Delta_c$. We say that J is the *shape* of the cone.

With this definition to hand, the notions of limit and colimit are easy to define.

Definition 2.3.39. A *limit* is a universal cone, and a *colimit* is a universal cocone. More explicitly, if D is a J-shaped diagram in $\mathcal{C}$, then the limit of D is a universal morphism from the constant diagram functor functor $\Delta : \mathcal{C} \to \mathcal{C}^J$ to D (considered as an object of the functor category), and the colimit of D is a universal morphism from D to Δ; alternatively, a colimit in $\mathcal{C}$ is a limit in $\mathcal{C}^{\mathrm{op}}$. In both cases, the apex of the cone is the universal object of the construction, which in the case of the limit of D we denote by $\lim D$, and in the case of the colimit, $\mathrm{colim}\, D$.

Note that we will often say '(co)limit' to refer to the apex of the universal (co)cone, even though the (co)limit is properly the whole universal construction. We are entitled to say "*the* (co)limit" thanks to the uniqueness of universal constructions.

We will often denote a universal cone by proj and call its component morphisms *projections*; dually, we will often denote a universal cocone by inj and call its morphisms *injections*.

Example 2.3.40. We can now exemplify the pattern of the limiting examples above. We will draw diagrams to depict the shape categories, with each symbol • indicating a distinct object and each arrow → indicating a distinct non-identity morphism.

---

[2]Products, coproducts, equalizers, and coequalizers.

1. A coproduct is a colimit of shape $\{ \; \bullet \qquad \bullet \; \}$;

2. a product is a limit of shape $\{ \; \bullet \qquad \bullet \; \}$;

3. an equalizer is a limit of shape $\{ \; \bullet \rightrightarrows \bullet \; \}$; and

4. a coequalizer is a colimit of shape $\{ \; \bullet \rightrightarrows \bullet \; \}$.

Of course, these are not the only possible shapes of limits and colimits. Some others will be particularly important, too.

Example 2.3.41. Let $\mathbf{0}$ denote the category with no objects or morphisms. A limit of shape $\mathbf{0}$ is known as a *terminal object*. This is an object 1 such that, for every object X, there is a unique morphism $! : X \to 1$. The terminal object in **Set** is a singleton set $\{*\}$.

Dually, a colimit of shape $\mathbf{0}$ is known as an *initial object*: an object 0 such that, for every object X, there is a unique morphism $\mathsf{i} : 0 \to X$. The initial object in **Set** is the empty set.

Remark 2.3.42. In Remark 2.2.8, we noted that morphisms $1 \to A$ in **Set** correspond to elements of A. In general categories $\mathcal{C}$ with a terminal object, one sometimes calls morphisms out of the terminal object *global elements*. The word 'global' emphasizes the special position of the terminal object in a category, which has a unique view of every object.

Example 2.3.43. A *pullback* is a limit of shape $\{ \; \bullet \longrightarrow \bullet \longleftarrow \bullet \; \}$. That is, given morphisms $f : A \to X$ and $g : B \to X$, their pullback is an object P and morphisms $\mathsf{proj}_A : P \to A$ and $\mathsf{proj}_B : P \to B$ making the following diagram commute

$$
\begin{array}{ccc}
P & \xrightarrow{\;\mathsf{proj}_B\;} & B \\
{\scriptstyle \mathsf{proj}_A}\big\downarrow & \lrcorner & \big\downarrow{\scriptstyle g} \\
A & \xrightarrow[\;f\;]{} & X
\end{array}
$$

in the universal sense that, for any object Q and morphisms $\pi_A : Q \to A$ and $\pi_B : Q \to B$ such that $f \circ \pi_A = g \circ \pi_B$, then there is a unique morphism $u : Q \to P$ such that $\pi_A = \mathsf{proj}_A \circ u$ and $\pi_B = \mathsf{proj}_B \circ u$. We indicate a *pullback square* using the symbol $\lrcorner$ as in the diagram above, and will variously denote the limiting object P by $A \times_X B$, f^*B, or g^*A, depending on the context.

The interpretation of the pullback is something like a generalized equation: in the category **Set**, the pullback $A \times_X B$ is the subset of the product $A \times B$ consisting of elements (a, b) for which $f(a) = g(b)$. Alternatively, it can be understood as a kind of generalized intersection: given two objects A and B and "ways of assigning them properties in X" f and g, the pullback $A \times_X B$ is

the generalized intersection of A and B according to these X-properties. In fact, we already saw this latter interpretation in Example 2.3.24, where we exhibited an intersection as an equalizer; now we can see that that equalizer was 'secretly' a pullback.

Remark 2.3.44. Dually, a colimit of shape $\{\ \bullet \longrightarrow \bullet \longleftarrow \bullet\ \}$ is known as a *pushout*. Whereas a pullback has an interpretation as a subobject of a product, a pushout has an interpration as a quotient of a coproduct. In this work, we will make far more use of pullbacks than pushouts.

The observation that pullbacks can be interpreted as subobjects of products (and dually that pushouts can be interpreted as quotients of coproducts) is a consequence of the more general result that all limits can be expressed using products and equalizers (and hence dually that colimits can be expressed using coproducts and coequalizers).

Proposition 2.3.45. Let $D : J \to \mathcal{C}$ be a diagram in $\mathcal{C}$, and suppose the products $\prod_{j:J_0} D(j)$ and $\prod_{f:J_1} D(\operatorname{cod} f)$ exist. Then, if it exists, the equalizer of

$$
\prod_{j:J_0} D(j) \xrightarrow[\prod_{f:J_1} \operatorname{proj}_{\operatorname{cod} f}]{\prod_{f:J_1} \left(Df \circ \operatorname{proj}_{\operatorname{dom} f}\right)} \prod_{f:J_1} D(\operatorname{cod} f)
$$

is the limit of D.

Proof sketch. Observe that the equalizer of the diagram above is an object L such that, for every morphism $f : j \to j'$ in J, the diagram

commutes, and such that any cone over D factors through it. This is precisely the universal property of the limit of D, and so by Proposition 2.3.36, (L, proj) is the limit. $\square$

Remark 2.3.46. As we indicated above, a dual result holds expressing colimits using coequalizers and coproducts. Because results obtained for limits in $\mathcal{C}$ will hold for colimits in $\mathcal{C}^{\operatorname{op}}$, we will henceforth not always give explicit dualizations.

2.3.3.1. Functoriality of taking limits

In the statement of Proposition 2.3.45, we used the fact that taking products extends to morphisms, too: a fact that was exemplified concretely in Example 2.2.5, and which follows from the fact that a pair of morphisms in $\mathcal{C}$ is equivalently a morphism in $\mathcal{C} \times \mathcal{C}$. We then saw in Remark 2.3.30 that $\mathcal{C} \times \mathcal{C} \cong \mathcal{C}^{\mathbf{2}}$. By letting the exponent again vary, the functoriality of taking products generalizes to the functoriality of taking limits, as long as $\mathcal{C}$ has all limits of the relevant shape.

Proposition 2.3.47 (Taking limits is functorial)**.** Suppose $\mathcal{C}$ has all limits of shape J (*i.e.*, for any diagram $D : J \to \mathcal{C}$, the limit $\lim D$ exists in $\mathcal{C}$). Then $\lim$ defines a functor $\mathbf{Cat}(J, \mathcal{C}) \to \mathcal{C}$.

Proof. We only need to check the assignment is well-defined on morphisms and functorial. Suppose D and D' are two diagrams $J \to \mathcal{C}$ with corresponding limiting cones $u : \Delta_{\lim D} \Rightarrow D$ and $u' : \Delta_{\lim D'} \Rightarrow D'$, and suppose $\delta : D \Rightarrow D'$ is a natural transformation. Observe that the composite natural transformation $\Delta_{\lim D} \overset{u}{\Rightarrow} D \overset{\delta}{\Rightarrow} D'$ is a cone on D', and that cones on D' are in bijection with morphisms in $\mathcal{C}$ into the apex object $\lim D'$. Therefore, by the universal property of the limit, there is a unique morphism $d : \lim D \to \lim D'$ such that $\delta \circ u = u' \circ \Delta_d$. This situation is summarized by the commutativity of the following diagram, where the dashed arrow indicates the uniqueness of Δ_d:

$$
\begin{array}{ccc}
\Delta_{\lim D} & \overset{\Delta_d}{=\!=\!=\!=\!=\!\Rightarrow} & \Delta_{\lim D'} \\
{\big\Downarrow}{\scriptstyle u} & & {\big\Downarrow}{\scriptstyle u'} \\
D & \overset{\delta}{\longrightarrow} & D'
\end{array}
$$

We define the action of the functor $\lim : \mathbf{Cat}(J, \mathcal{C}) \to \mathcal{C}$ on the natural transformation δ by this unique morphism d, setting $\lim \delta := d : \lim D \to \lim D'$.

It therefore only remains to check that this assignment is functorial (*i.e.*, that it preserves identities and composites). To see that $\lim$ preserves identities, just take $\delta = \mathsf{id}_D$ in the situation above; clearly, by the uniqueness of d, we must have $\lim \mathsf{id}_D = \mathsf{id}_{\lim D}$. Now suppose $\delta' : D' \to D''$ is another natural transformation. To see that $\lim(\delta' \circ \delta) = \lim \delta' \circ \lim \delta$, consider the pasting of the associated diagrams:

$$
\begin{array}{ccccc}
& & \overset{\Delta_{d'd}}{\overparen{=\!=\!=\!=\!=\!=\!=\!=\!=\!=\!=\!=\!=}} & & \\
\Delta_{\lim D} & \overset{\Delta_d}{=\!=\!=\!\Rightarrow} & \Delta_{\lim D'} & \overset{\Delta_{d'}}{=\!=\!=\!\Rightarrow} & \Delta_{\lim D''} \\
{\big\Downarrow}{\scriptstyle u} & & {\big\Downarrow}{\scriptstyle u'} & & {\big\Downarrow}{\scriptstyle u''} \\
D & \overset{\delta}{\longrightarrow} & D' & \overset{\delta'}{\longrightarrow} & D''
\end{array}
$$

We have $\lim(\delta' \circ \delta) = d'd$, which is unique by definition. Therefore we must have $d'd = d' \circ d = \lim(\delta') \circ \lim(\delta)$, and hence $\lim(\delta' \circ \delta) = \lim(\delta') \circ \lim(\delta)$ as required. $\square$

2.3.3.2. (Co)limits as adjoints

Since taking limits is functorial, it makes sense to ask if the functor $\lim$ has an adjoint, and indeed it does, in a familiar form.

Proposition 2.3.48. The functor $\lim : \mathcal{C}^J \to \mathcal{C}$ is right adjoint to the constant diagram functor functor $\Delta : \mathcal{C} \to \mathcal{C}^J$, *i.e.* $\Delta \dashv \lim$.

Proof. We need to show that $\mathcal{C}^J(\Delta_c, D) \cong \mathcal{C}(c, \lim D)$ naturally in $c : \mathcal{C}$ and $D : J \to \mathcal{C}$. It is sufficient to demonstrate naturality in each argument separately, by the universal property of the product in **Cat**. We have already established naturality in $c : \mathcal{C}$ in Lemma 2.3.51 and shown that taking limits is functorial (Proposition 2.3.47). So it only remains to show that this extends to naturality in $D : J \to \mathcal{C}$, which requires the commutativity of the following diagram for any $\delta : D \to D'$, where we write α_D for the isomorphism $\mathcal{C}(c, \lim D) \xrightarrow{\sim} \mathcal{C}^J(\Delta_c, D)$:

$$
\begin{array}{ccc}
\mathcal{C}(c, \lim D) & \xrightarrow{\alpha_D} & \mathcal{C}^J(\Delta_c, D) \\
\downarrow {\scriptstyle \mathcal{C}(c, \lim \delta)} & & \downarrow {\scriptstyle \mathcal{C}^J(\Delta_c, \delta)} \\
\mathcal{C}(c, \lim D') & \xrightarrow{\alpha_{D'}} & \mathcal{C}^J(\Delta_c, D')
\end{array}
$$

Chasing a morphism $\beta : c \to \lim D$ around this diagram, we find that its commutativity amounts to the commutativity of the following diagram of cones for all $\varphi : i \to j$ in J, where by definition $\alpha_D(\beta)_i = \pi_i \circ \beta$ and $\alpha_{D'}(\lim \delta \circ \beta)_i = \pi'_i \circ \lim \delta \circ \beta$:

This diagram commutes by definition, so the isomorphism is natural in D, which therefore establishes the desired adjunction. $\square$

Remark 2.3.49. Dually, if all colimits of shape J exist in $\mathcal{C}$, then colim is left adjoint to Δ.

Later, we will see that every adjoint functor exhibits a universal property (Propositions 2.4.16 and 2.4.23, results that we've already seen exemplified in Example 2.3.35), and this therefore gives us another perspective on the universality of limits.

2.3.3.3. Hom preserves limits

We end this section with a useful result on the interaction between the covariant hom functors $\mathcal{C}(c, -) : \mathcal{C} \to \mathbf{Set}$ and taking limits.

Proposition 2.3.50 (Hom functor preserves limits). Suppose $D : J \to \mathcal{C}$ is a diagram in the category $\mathcal{C}$. There is an isomorphism $\mathcal{C}(c, \lim D) \cong \lim \mathcal{C}(c, D(-))$ which is natural in $c : \mathcal{C}$.

To prove this proposition, it helps to have the following lemma, which establishes a natural isomorphism between the set of morphisms into a limit and the set of cones on the corresponding diagram.

Lemma 2.3.51. $\mathcal{C}(c, \lim D) \cong \mathcal{C}^J(\Delta_c, D)$, naturally in $c : \mathcal{C}$.

Proof. For a given $c : \mathcal{C}$, the isomorphism $\mathcal{C}(c, \lim D) \cong \mathcal{C}^J(\Delta_c, D)$ follows directly from the universal property of the limit: morphisms from c into the limiting object $\lim D$ are in bijection with cones $\Delta_c \Rightarrow D$. So it only remains to show that this isomorphism is natural in $c : \mathcal{C}$. Writing $\alpha : \mathcal{C}(-, \lim D) \Rightarrow \mathcal{C}^J(\Delta_{(-)}, D)$ for the natural transformation that takes each morphism into the limit to the corresponding cone on D, naturality amounts to the commutativity of the following square for each $f : c' \to c$ in $\mathcal{C}$:

$$
\begin{array}{ccc}
\mathcal{C}(c, \lim D) & \xrightarrow{\ \alpha_c\ } & \mathcal{C}^J(\Delta_c, D) \\
{\scriptstyle \mathcal{C}(f, \lim D)} \downarrow & & \downarrow {\scriptstyle \mathcal{C}^J(\Delta_f, D)} \\
\mathcal{C}(c', \lim D) & \xrightarrow{\ \alpha_{c'}\ } & \mathcal{C}^J(\Delta_{c'}, D)
\end{array}
$$

Commutativity of this naturality square witnesses the fact that, given a morphism $g : c \to \lim D$, you can either take the corresponding cone $\alpha_c(g)$ and pull it back along Δ_f (at its apex) to obtain the cone $\alpha_c(g) \circ \Delta_f$, or you can form the composite morphism $g \circ f$ and take its cone $\alpha_{c'}(g \circ f)$, and you'll have the same cone: $\alpha_c(g) \circ \Delta_f = \alpha_{c'}(g \circ f)$. This is illustrated by the commutativity of the following diagram, which shows fragments of the limiting cone denoted π, the cone $\alpha_c(g)$,

and the cone $\alpha_{c'}(g \circ f)$, for a morphism $\varphi : i \to j$ in J:
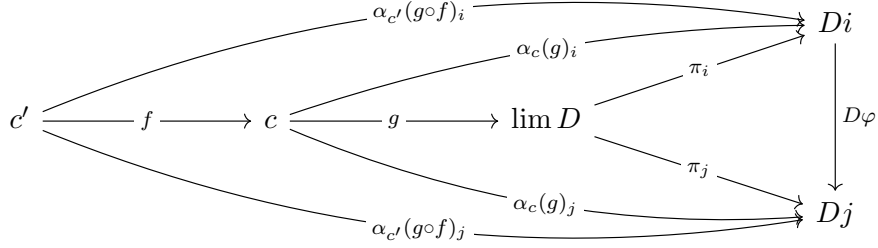
By the universal property of the limit, we must have $\alpha_{c'}(g \circ f)_i = \alpha_c(g)_i \circ f = \pi_i \circ g \circ f$ naturally in i, and hence $\alpha_{c'}(g \circ f) = \alpha_c(g) \circ \Delta_f$. $\qquad\square$

Proof of Proposition 2.3.50. By Lemma 2.3.51, we have a natural isomorphism $\mathcal{C}^J(\Delta_c, D) \cong \lim \mathcal{C}(c, D(-))$, so it suffices to establish a natural isomorphism $\mathcal{C}^J(\Delta_c, D) \cong \lim \mathcal{C}(c, D(-))$. This says that cones on D with apex c are isomorphic to the limit of $\mathcal{C}(c, D(-)) : J \to \mathbf{Set}$, naturally in c. First, note that this limiting cone in $\mathbf{Set}$ is constituted by a family of functions $\{p_i : \lim \mathcal{C}(c, D(-)) \to \mathcal{C}(c, Di)\}_{i:J}$, as in the following commuting diagram:
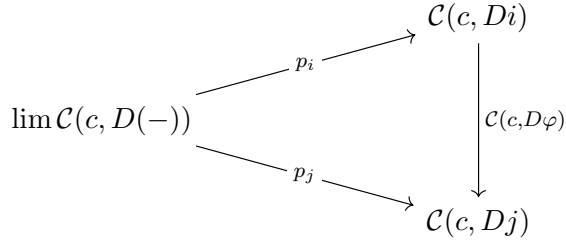
Next, note there is a bijection between cones $\Delta_c \Rightarrow D$ on D in $\mathcal{C}$ with apex c, as in the commuting diagram below-left, and cones $\Delta_1 \Rightarrow \mathcal{C}(c, D(-))$ in $\mathbf{Set}$, as in the commuting diagram below-right.
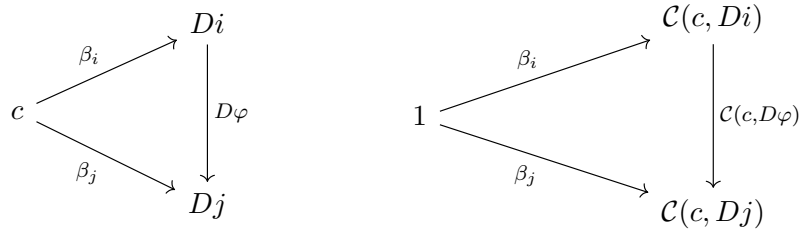
By the univeral property of the limit, any cone $\{\beta_i\}$ as on the right factors uniquely through $\lim \mathcal{C}(c, D(-))$, as in the following commuting diagram. Similarly, any element β of $\lim \mathcal{C}(c, D(-))$ induces a corresponding cone $\{p_i(\beta)\}$, by composition with the limiting cone p. To see that this correspondence is an isomorphism, observe that the element of the set $\lim \mathcal{C}(c, D(-))$ assigned to the cone $\{p_i(\beta)\}$ must be exactly β, since the universal property of $\lim \mathcal{C}(c, D(-))$ ensures that β

is uniquely determined.

$$\begin{array}{ccc}
 & \xrightarrow{\beta_i} & \mathcal{C}(c, Di) \\
& \nearrow p_i & \\
* \dashrightarrow{\beta} \lim \mathcal{C}(c, D(-)) & & \downarrow \mathcal{C}(c, D\varphi) \\
& \searrow p_j & \\
& \xrightarrow{\beta_j} & \mathcal{C}(c, Dj)
\end{array}$$
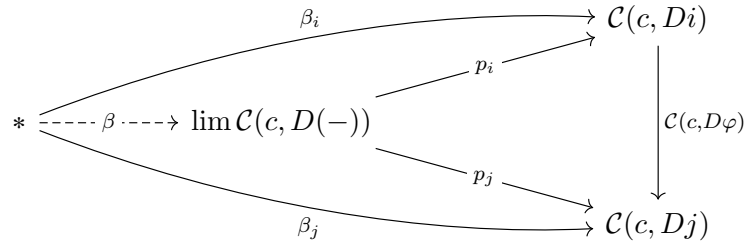
It only remains to check that this correspondence is natural in c, so suppose f is any morphism $c' \to c$ in $\mathcal{C}$. If we write $p_- : \lim \mathcal{C}(c, D(-)) \to \mathcal{C}^J(\Delta_c, D)$ to denote the function $\beta \mapsto \{p_i(\beta)\}$, and p'_- to denote the corresponding function for c', naturality requires the following square to commute:

$$\begin{array}{ccc}
\lim \mathcal{C}(c, D(-)) & \xrightarrow{\ p_- \ } & \mathcal{C}^J(\Delta_c, D) \\
\lim \mathcal{C}(f, D(-)) \downarrow & & \downarrow \mathcal{C}^J(\Delta_f, D) \\
\lim \mathcal{C}(c', D(-)) & \xrightarrow{\ p'_- \ } & \mathcal{C}^J(\Delta_{c'}, D)
\end{array}$$

The commutativity of this square in turn corresponds to the commutativity of the following diagram in **Set**, for any cone β:

$$\begin{array}{ccccc}
& \mathcal{C}(c, Di) & \xrightarrow{\mathcal{C}(f, Di)} & \mathcal{C}(c', Di) \\
& \nearrow p_i & & \nearrow p'_i & \\
1 \xrightarrow{\beta} \lim \mathcal{C}(c, D(-)) & \downarrow \mathcal{C}(c, D\varphi) \xrightarrow{\lim \mathcal{C}(f, D(-))} \lim \mathcal{C}(c', D(-)) & & \downarrow \mathcal{C}(c', D\varphi) \\
& \searrow p_j & & \searrow p'_j & \\
& \mathcal{C}(c, Dj) & \xrightarrow{\mathcal{C}(f, Dj)} & \mathcal{C}(c', Dj)
\end{array}$$
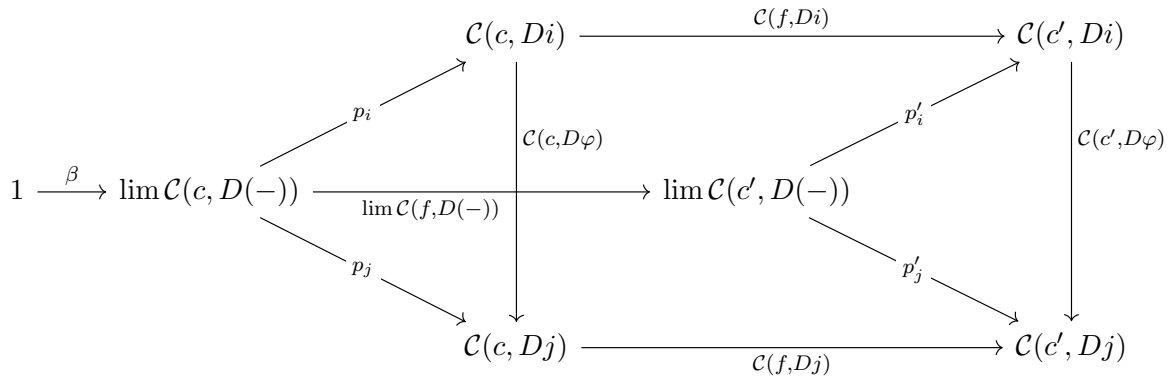
By the correspondence between cones $\Delta_c \Rightarrow D$ in $\mathcal{C}$ and cones $\Delta_1 \Rightarrow \mathcal{C}(c, D(-))$ in **Set**, this diagram commutes if and only if the following diagram commutes:

$$\begin{array}{ccc}
& \xrightarrow{\beta_i \circ f} & Di \\
& \nearrow \beta_i & \downarrow D\varphi \\
c' \xrightarrow{f} c & & \\
& \searrow \beta_j & \\
& \xrightarrow{\beta_j \circ f} & Dj
\end{array}$$
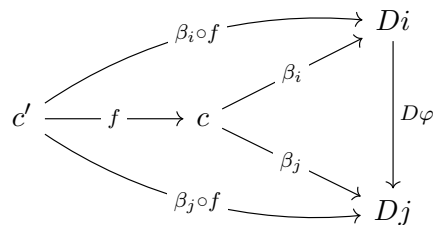
This diagram commutes by the definition of β and of the composites $\{\beta_i \circ f\}$, thereby establishing the naturality of the isomorphism $\lim \mathcal{C}(c, D(-)) \cong \mathcal{C}^J(\Delta_c, D)$. Since we also have a natural isomorphism $\mathcal{C}^J(\Delta_c, D) \cong \mathcal{C}(c, \lim D)$, we have established the result. $\qquad\square$

The preceding proof established more than just the hom functor's preservation of limits: it gave us another useful natural isomorphism, this time betwen the set of cones $\Delta_c \Rightarrow D$ in $\mathcal{C}$ and the set of cones $\Delta_1 \Rightarrow \mathcal{C}(c, D)$ on the diagram $\mathcal{C}(c, D) : J \to \mathbf{Set}$ with apex the terminal set 1.

Corollary 2.3.52. There is an isomorphism $\mathcal{C}^J(\Delta_c, D) \cong \mathbf{Set}^J(\Delta_1, \mathcal{C}(c, D))$, natural in $c : \mathcal{C}$.

Remark 2.3.53. Since limits in $\mathcal{C}^{\mathrm{op}}$ are colimits in $\mathcal{C}$, Proposition 2.3.50 implies that the contravariant hom functors $\mathcal{C}(-, c)$ turn limits into colimits; *i.e.* $\mathcal{C}(\lim D, c) \cong \operatorname{colim} \mathcal{C}(D(-), c)$.

2.3.4. Closed categories and exponential objects

A distinguishing feature of adaptive systems such as the brain is that they contain processes which themselves control other processes, and so it is useful to be able to formalize this situation compositionally. When a category contains objects which themselves represent the morphisms of the category, we say that the category is *closed*: in such categories, we may have processes whose outputs are again processes, and we may think of the latter as controlled by the former.

A basic instance of this mathematical situation is found amidst the natural numbers, where repeated multiplication coincides with exponentiation, as in $2 \times 2 \times 2 = 2^3$. If we think of numbers as sets of the corresponding size, and let 2^3 denote the set of functions $3 \to 2$, then it is not hard to see that there are 8 such distinct functions. If we generalize this situation from numbers to arbitrary objects, and from functions to morphisms, we obtain the following general definition of exponentiation.

Definition 2.3.54. Let $\times$ denote the product in a category $\mathcal{C}$. When there is an object $e : \mathcal{C}$ such that $\mathcal{C}(x, e) \cong \mathcal{C}(x \times y, z)$ naturally in x, we say that e is an *exponential object* and denote it by z^y. The image of id_{z^y} under the isomorphism is called the *evaluation map* and is written $\mathsf{ev}_{y,z} : z^y \times y \to z$.

Example 2.3.55. In $\mathbf{Set}$, given sets A and B, the exponential object B^A is the set of functions $A \to B$. Given a function $f : A \to B$, the evaluation map $\mathsf{ev}_{B,A}$ acts by applying f to elements of A: *i.e.*, $\mathsf{ev}_{B,A}(f, a) = f(a)$.

Typically, we are most interested in situations where every pair of objects is naturally exponentiable, which induces the following adjunction, formalizing the idea that exponentiation is repeated multiplication.

Proposition 2.3.56. When the isomorphism $\mathcal{C}(x \times y, z) \cong \mathcal{C}(x, z^y)$ is additionally natural in z, we obtain an adjunction $(-) \times y \dashv (-)^y$ called the *product-exponential adjunction*, and this uniquely determines a functor $\mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathcal{C} : (y, z) \mapsto z^y$ that we call the *internal hom* for $\mathcal{C}$.

Proof. That the natural isomorphism induces an adjunction is immediate from Proposition 2.2.26; the counit of this adjunction is the family of evaluation maps $\mathrm{ev} : (-)^y \times y \Rightarrow \mathrm{id}_\mathcal{C}$. The uniqueness of the internal hom follows from the uniqueness of adjoint functors (which we will establish in Corollary 2.4.18). □

Definition 2.3.57. A category $\mathcal{C}$ in which every pair of objects has a product is called *Cartesian*. A Cartesian category $\mathcal{C}$ with a corresponding internal hom is called *Cartesian closed*.

Example 2.3.58. We've already seen that **Set** is Cartesian closed. So is **Cat**: the internal hom $\mathcal{C}^{\mathcal{B}}$ is the category of functors $\mathcal{B} \to \mathcal{C}$.

Example 2.3.59 (A non-example). The category **Meas** of measurable spaces with measurable functions between them is Cartesian but not Cartesian closed: the evaluation function is not always measurable [15]. In this context, we will introduce *quasi-Borel spaces* (originally due to Heunen et al. [131]) in §4.1.5, in order to construct stochastic processes which emit functions.

It is not hard to prove the following result, which says that Cartesian closed categories can "reason about themselves".

Proposition 2.3.60. A Cartesian closed category is enriched in itself.

This 'internalization' is witnessed by the hom functors, which in the case of a Cartesian closed enriching category $\mathcal{E}$ become $\mathcal{E}$-functors.

Proposition 2.3.61. Suppose $\mathcal{C}$ is an $\mathcal{E}$-category where $\mathcal{E}$ is Cartesian closed. Then the hom functor $\mathcal{C}(-, =)$ is an $\mathcal{E}$-functor $\mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathcal{E}$. On objects (c, d), the hom functor returns the object $\mathcal{C}(c, d)$ in $\mathcal{E}$ of morphisms $c \to d$. Then, for each quadruple (b, c, a, d) of objects in $\mathcal{C}$, we define an $\mathcal{E}$-morphism $\mathcal{C}^{\mathrm{op}}(b, a) \times \mathcal{C}(c, d) \to \mathcal{E}\big(\mathcal{C}(b, c), \mathcal{C}(a, d)\big)$ as the image of the composite

$$\big(\mathcal{C}(a, b) \times \mathcal{C}(c, d)\big) \times \mathcal{C}(b, c) \xrightarrow{\alpha} \mathcal{C}(a, b) \times \big(\mathcal{C}(c, d) \times \mathcal{C}(b, c)\big) \cdots$$

$$\cdots \xrightarrow{\mathcal{C}(a,b) \times \circ_{b,c,d}} \mathcal{C}(a, b) \times \mathcal{C}(b, d) \xrightarrow{\sigma} \mathcal{C}(b, d) \times \mathcal{C}(a, b) \xrightarrow{\circ_{a,b,d}} \mathcal{C}(a, d)$$

under the product-exponential isomorphism

$$\mathcal{E}\big(\mathcal{C}(a, b) \times \mathcal{C}(c, d), \mathcal{C}(a, d)^{\mathcal{C}(b,c)}\big) \cong \mathcal{E}\Big(\big(\mathcal{C}(a, b) \times \mathcal{C}(c, d)\big) \times \mathcal{C}(b, c), \mathcal{C}(a, d)\Big)$$

where α is the associativty of the product and σ is its symmetry $X \times Y \cong Y \times X$, and where we have used that $\mathcal{C}^{\,\mathrm{op}}(b, a) = \mathcal{C}(a, b)$.

Remark 2.3.62. The rôle of the symmetry here is testament to the fact that we can read a composite morphism $g \circ f$ as either "g after f" or "f before g".

Proof sketch. To give an $\mathcal{E}$-functor (Definition 2.2.10) is to give a function on objects and a family of $\mathcal{E}$-morphisms (corresponding to the hom objects of $\mathcal{C}$) such that identities and composites are preserved. We have given such a function and such a family in the statement of the proposition, and so it remains to check the axioms: these follow by the unitality and associativity of composition in an $\mathcal{E}$-category (Definition 2.2.6). $\qquad\square$

When $\mathcal{E}$ is Cartesian closed, then as a corollary its hom functor $\mathcal{E}(-, =)$ is an $\mathcal{E}$-functor.

When a diagram commutes, every parallel path is equal when interpreted as a morphism. If a diagram commutes up to some 2-cell or 2-cells, then parallel paths can be transformed into each other using the 2-cell(s). Much categorical reasoning therefore consists in using morphisms in the base of enrichment to translate between different hom objects; the simplest such of course being pre- and post-composition. In the next section, we will see many explicit examples of this kind of reasoning when we prove the Yoneda Lemma—which says that the hom objects contain all the data of the category—but we have already seen examples of it above, when we considered adjunctions: after all, adjunctions are families of isomorphisms between hom objects.

When a category is Cartesian closed, it is its own base of enrichment, and so one does not have to move to an external perspective to reason categorically about it: one can do so using its 'internal language'. We have already seen a correspondence between the language of logic and that of sets, in which we can think of elements of sets as witnesses to the proof of propositions represented by those sets, and where logical operations such as conjunction and disjunction correspond to operations on sets. This correspondence extends to Cartesian closed categories generally: universal constructions such as those we have introduced above can be interpreted as encoding the logic of the internal language.

More precisely, Cartesian closed categories are said to provide the semantics for *dependent type theory*: a higher-order logic in which propositions are generalized by 'types'[3]. One can construct a 'syntactic' category representing the logic of the type theory, and then interpret it functorially

---

[3]A type is something like a proposition in which we're 'allowed' to distinguish between its witnesses, which we call *terms* of the given type.

in a Cartesian closed category. This correspondence is known as the *Curry-Howard-Lambek correspondence*, which says that logical proofs correspond to morphisms in a Cartesian closed category, and that such morphisms can equally be seen as representing the functions computed by deterministic computer programs. (In general, the correspondence is an adjoint one: dually, one can construct from a given category a 'syntactic' category encoding the logic of its internal language.)

When a category moreover has (internal) dependent sums and products, then it can be interpreted as a model of *dependent type theory*, in which types themselves may depend on values; for instance, one might expect that the type of a weather forecast should depend on whether one is on land or at sea. We will not say much more about dependent type theory, although we will make implicit use of some of its ideas later in the thesis. Therefore, before moving on to the Yoneda Lemma, we will say just enough to define the notion of dependent product 'universally', without reference to sets.

2.3.4.1. Dependent products

In Remark 2.3.20, we discussed products where the factors were indexed by an arbitrary set and explained how they correspond to sets of generalized 'dependent' functions, where the codomain type may vary with the input. In that case, we were restricted to considering products indexed by sets, but with the machinery of limits at hand, we can 'internalize' the definition to other Cartesian closed categories.

Definition 2.3.63. Suppose $\mathcal{C}$ is Cartesian closed and has all limits, and suppose $p : E \to B$ is a morphism in $\mathcal{C}$. The *dependent product* of p along B is the pullback object $\prod_B p$ as in the diagram

$$
\begin{array}{ccc}
\prod_B p & \longrightarrow & E^B \\
\downarrow & & \downarrow{\scriptstyle p^B} \\
1 & \xrightarrow{\ \mathsf{id}_B\ } & B^B
\end{array}
$$

where 1 is the terminal object, id_B is the element picking the identity morphism $B \to B$, and p^B is the postcomposition morphism induced by the functoriality of exponentiation.

Remark 2.3.64. When p is the projection $\sum_{b:B} P_b \to B$ out of a dependent sum, we will write its dependent product as $\prod_{b:B} P_b$. Since a product $B \times C$ is isomorphic to the dependent sum $\sum_{b:B} C$, note that this means we can alternatively write the exponential object C^B as $\prod_{b:B} C$.

To understand how Definition 2.3.63 generalizes Remark 2.3.20, we can interpret the former in **Set** and see that the two constructions coincide. The set E^B is the set of functions $s : B \to E$,

and p^B acts by $s \mapsto p \circ s$. The indicated pullback therefore selects the subset of E^B such that $p \circ s = \mathsf{id}_B$. This is precisely the set of sections of p, which is in turn the dependent product of p in **Set**.

Remark 2.3.65. Definition 2.3.63 is entirely *internal* to $\mathcal{C}$: it depends only on structure that is available within $\mathcal{C}$ itself, and not on 'external' structures (such as indexing sets) or knowledge (such as knowledge of the make-up of the objects of $\mathcal{C}$). It is epistemically parsimonious: a purely categorical definition, stated entirely in terms of universal constructions.

Remark 2.3.66. Under the Curry-Howard-Lambek correspondence, exponential objects represent the propositions that one proposition implies another; in type theory, they represent the type of functions from one type to another. As dependent exponential objects, dependent products could therefore be seen as representing 'dependent' implications; as we have already seen, they do represent the type of dependent functions. However, dependent products and sums have another kind of logical interpretation: as *quantifiers*. That is, the logical proposition represented by $\prod_{b:B} P_b$ is $\forall b : B.P(b)$: an element of $\prod_{b:B} P_b$ is a proof that, for all $b : B$, the proposition $P(b)$ is satisfied. Dually, the proposition represented by $\sum_{b:B} P_b$ is $\exists b : B.P(b)$: an element of $\sum_{b:B} P_b$ is a pair (b, x) of a witness to B and a witness x of the satisfaction of $P(b)$.

2.4. The Yoneda Lemma: a human perspective

We end this chapter by introducing the fundamental theorem of category theory, the Yoneda Lemma, which expresses mathematically the idea that to know how a thing is related to other things is to know the identity of the thing itself. The notion of relational identity is recognized throughout human endeavour. In linguistics, it underlies the observation of Firth [92] that "you shall know a word by the company it keeps!", which in turn is the foundation of distributional semantics and thus much of contemporary natural language processing in machine learning. In culture, it is illustrated by the ancient parable of the blind men and the elephant, in which the identity of the creature is only known by stitching together evidence from many perspectives. In society, it is reflected in the South African philosophy of *ubuntu* (meaning "I am because we are") and the Māori notion of *whanaungatanga* (in which personal identity is developed through kinship), and the observation that "actions speak louder than words". Finally, the Yoneda Lemma is manifest in science, where our understanding of phenomena derives from the accumulation across contexts of results and their interpretation and translation: no single individual understands the totality of any subject, and no subject or phenomenon is understood in isolation.

2.4.1. Formalizing categorical reasoning via the Yoneda embedding

In §2.3.4, we saw how Cartesian closed categories allow us to internalize categorical reasoning. The category $\mathbf{Set}$ is the archetypal Cartesian closed category, and constitutes the base of enrichment for all locally small categories. The Yoneda embedding allows us to move from reasoning about the objects in any given category $\mathcal{C}$ to reasoning about the morphisms between its hom sets: the natural transformations between hom functors. In this context, the hom functors constitute special examples of functors into the base of enrichment, which we call 'presheaves' (contravariantly) and 'copresheaves' (covariantly), and which can be thought of as $\mathcal{C}$-shaped diagrams in $\mathbf{Set}$.

Definition 2.4.1. Let $\mathcal{C}$ be a category. A *presheaf* on $\mathcal{C}$ is a functor $\mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$. Dually, a *copresheaf* is a functor $\mathcal{C} \to \mathbf{Set}$. The corresponding functor categories are the categories of *(co)presheaves* on $\mathcal{C}$.

Remark 2.4.2. In the enriched setting, when $\mathcal{C}$ is enriched in $\mathcal{E}$, an $\mathcal{E}$-*presheaf* is an $\mathcal{E}$-functor $\mathcal{C}^{\mathrm{op}} \to \mathcal{E}$ and an $\mathcal{E}$-*copresheaf* is an $\mathcal{E}$-functor $\mathcal{C} \to \mathcal{E}$.

As a first example of a presheaf, we have an alternative definition of the notion of directed graph.

Example 2.4.3. Let $\mathbb{G}$ denote the category of Example 2.1.5 containing two objects 0 and 1 and two morphisms $s, t : 0 \to 1$. Then a *directed graph* is a presheaf on $\mathbb{G}$.

This definition is justified by the following proposition.

Proposition 2.4.4. There is an equivalence of categories $\mathbf{Graph} \cong \mathbf{Set}^{\mathbb{G}^{\mathrm{op}}}$, where $\mathbf{Graph}$ is the category of directed graphs introduced in Example 2.1.11.

Proof. To each graph $\mathcal{G}$ we can associate a presheaf $G : \mathbb{G}^{\mathrm{op}} \to \mathbf{Set}$ by defining $G(0) := \mathcal{G}_0$, $G(1) := \mathcal{G}_1$, $G(s) := \mathrm{dom}_{\mathcal{G}}$ and $G(t) := \mathrm{cod}_{\mathcal{G}}$; and to each presheaf we can likewise associate a graph, so that we have defined a bijection on objects. It therefore only remains to show that there is a bijection between graph homomorphisms and natural transformations accordingly: but this is easy to see once we have observed that the graph homomorphism axioms are precisely the law of naturality, as illustrated diagrammatically in (2.1). $\qquad\square$

Taking again a general perspective, the Yoneda embedding is the embedding of a category $\mathcal{C}$ into its presheaf category, obtained by mapping $c : \mathcal{C}$ to the presheaf $\mathcal{C}(-, c)$; and there is of course a dual 'coYoneda' embedding.

Remark 2.4.5. We say 'embedding' to mean a functor that is injective on objects and faithful (injective on hom sets). The Yoneda embedding will turn out to be fully faithful (bijective on hom sets), as a consequence of the Yoneda lemma.

Owing to its importance, we make a formal definition of the Yoneda embedding.

Definition 2.4.6. Let $\mathcal{C}$ be a category. By applying the product-exponential adjunction in **Cat** to the hom functor $\mathcal{C}(-,=) : \mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathbf{Set}$, we obtain a functor $\sout : \mathcal{C} \to \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}} : c \mapsto \mathcal{C}(-, c)$ of $\mathcal{C}$ into its presheaf category, and dually a functor $\nsout : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}^{\mathcal{C}} : c \mapsto \mathcal{C}(c, =)$ into the copresheaf category. We call the former functor the *Yoneda embedding* and the latter the *coYoneda embedding*. When $\mathcal{C}$ is an $\mathcal{E}$-category and $\mathcal{E}$ is Cartesian closed, then the Yoneda embedding is instead an $\mathcal{E}$-functor $\mathcal{C} \to \mathcal{E}^{\mathcal{C}^{\mathrm{op}}}$ (and likewise for the coYoneda embedding).

Remark 2.4.7. This abstract definition does not make explicit how $\sout$ acts on morphisms. However, we have already seen this action, when we first exemplified natural transformations in Example 2.2.18.

As we discussed in §2.3.4, much categorical reasoning corresponds to following morphisms between hom objects, and often the reasoning is agnostic either to where one starts, or to where one ends up. The Yoneda embedding witnesses such proofs as morphisms in the (co)presheaf categories. As an example, consider the proof of Proposition 2.4.20 below: each step corresponds to the application of a natural transformation.

Remark 2.4.8. It also so happens that every (co)presheaf category is very richly structured, inheriting its structure from the base of enrichment. For example, this means that the presheaf category $\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}$ has all limits, is Cartesian closed, has a subobject classifier, and dependent sums and products, even when $\mathcal{C}$ has none of these. (Interestingly, this means that the category of directed graphs is accordingly richly structured, being a presheaf category by Proposition 2.4.4.) As a result, (co)presheaf categories are very powerful places to do categorical reasoning.

2.4.2. Knowing a thing by its relationships

The Yoneda lemma says that every (co)presheaf on $\mathcal{C}$ is determined by "how it looks from $\mathcal{C}$". Since under the (co)Yoneda embedding every object gives rise to a (co)presheaf, a corollary of the Yoneda lemma is that every object can be identified by its relationships.

Remark 2.4.9. If the base of enrichment of a category is Cartesian closed, then one can prove an analogous enriched version of the Yoneda lemma. We will only prove the standard **Set**-enriched case here.

We will also only prove the Yoneda lemma for presheaves; there is of course a dual coYoneda lemma for copresheaves, which follows simply by swapping $\mathcal{C}$ for $\mathcal{C}^{\mathrm{op}}$.

Theorem 2.4.10 (Yoneda lemma). Let $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ be a presheaf on $\mathcal{C}$. Then for each $c : \mathcal{C}$, there is an isomorphism $Fc \cong \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c), F)$. Moreover, this isomorphism is natural in both $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ and $c : \mathcal{C}$.

Proof. We first define a mapping $\gamma : Fc \to \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c), F)$ as follows. Given $h : Fc$, we define the natural transformation $\gamma(h) : \mathcal{C}(-,c) \Rightarrow F$ to have components $\gamma(h)_b : \mathcal{C}(b,c) \to Fb : f \mapsto Ff(h)$; note that since $h : Fc$ and $f : b \to c$, we have $Ff : Fc \to Fb$ and hence $Ff(h) : Fb$. To check that this definition makes $\gamma(h)$ into a natural transformation, suppose $g : a \to b$. We need to check $Fg \circ \gamma(h)_b = \gamma(h)_a \circ \mathcal{C}(g,c)$. Since $\mathcal{C}(g,c)(f) = f \circ g$, this means verifying $Fg \circ Ff(h) = F(f \circ g)(h)$. But F is a contravariant functor, so $F(f \circ g) = Fg \circ Ff$, thereby establishing naturality.

Conversely, we define a mapping $\gamma' : \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c), F) \to Fc$ as follows. Suppose α is a natural transformation $\mathcal{C}(-,c) \Rightarrow F$, so that its component at c is the function $\alpha_c : \mathcal{C}(c,c) \to Fc$. We define $\gamma'(\alpha) := \alpha_c(\mathrm{id}_c)$.

Next, we need to establish that γ and γ' are mutually inverse. First, we check that $\gamma' \circ \gamma = \mathrm{id}_{Fc}$. Given $h : Fc$, we have

$$\gamma'(\gamma(h)) = \gamma(h)_c(\mathrm{id}_c) = F(\mathrm{id}_c)(h) = \mathrm{id}_{Fc}(h) = h$$

as required. We now check that $\gamma \circ \gamma' = \mathrm{id}_{\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c),F)}$. Given $\alpha : \mathcal{C}(-,c) \Rightarrow F$, we have $\gamma'(\alpha) = \alpha_c(\mathrm{id}_c)$ by definition. Hence $\gamma(\gamma'(\alpha)) : \mathcal{C}(-,c) \Rightarrow F$ has components $\gamma(\gamma'(\alpha))_b : \mathcal{C}(b,c) \to Fb$ which act by $f \mapsto Ff(\alpha_c(\mathrm{id}_c))$. So we need to show that $Ff(\alpha_c(\mathrm{id}_c)) = \alpha_b(f)$. This follows directly from the naturality of α. The commutativity of the naturality square on the left in particular holds at $\mathrm{id}_c : \mathcal{C}(c,c)$ as on the right:

$$
\begin{array}{ccc}
\mathcal{C}(c,c) \xrightarrow{\;\alpha_c\;} Fc & \qquad & \mathrm{id}_c \longmapsto \alpha_c(\mathrm{id}_c) \\
\Big\downarrow{\scriptstyle \mathcal{C}(f,c)} \qquad \Big\downarrow{\scriptstyle Ff} & & \Big\downarrow \qquad\qquad \Big\downarrow \\
\mathcal{C}(b,c) \xrightarrow{\;\alpha_b\;} Fb & & f \longmapsto \alpha_b(f) = Ff(\alpha_c(\mathrm{id}_c))
\end{array}
$$

Note that $\mathcal{C}(f,c)(\mathsf{id}_c) = \mathsf{id}_c \circ f = f$. This establishes that $\gamma \circ \gamma' = \mathsf{id}_{\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c),F)}$, and since $\gamma' \circ \gamma = \mathsf{id}_{Fc}$, we have $Fc \cong \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c), F)$.

It remains to verify that this isomorphism is natural in F and c. Suppose $\varphi : F \Rightarrow F'$ is a natural transformation, and write γ'_{Fc} for the function γ' defined above, and $\gamma'_{F'c}$ for the corresponding function for F'. Naturality in F means that the diagram on the left below commutes, which we can see by chasing the natural transformation α as on the right:

$$
\begin{array}{ccc}
\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c),F) & \xrightarrow{\;\gamma'_{Fc}\;} & Fc \\
{\scriptstyle \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c),\varphi)}\Big\downarrow & & \Big\downarrow{\scriptstyle \varphi_c} \\
\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c),F') & \xrightarrow{\;\gamma'_{F'c}\;} & F'c
\end{array}
\qquad
\begin{array}{ccc}
\alpha & \longmapsto & \gamma'_{Fc}(\alpha) \\
\Big\downarrow & & \Big\downarrow \\
\varphi \circ \alpha & \longmapsto & \gamma'_{F'c}(\varphi \circ \alpha) = \varphi_c \circ \gamma'_{Fc}(\alpha)
\end{array}
$$

Since $\gamma'_{Fc}(\alpha)_c := \alpha_c(\mathsf{id}_c)$ and $\gamma'_{F'c}(\varphi \circ \alpha)_c := \varphi_c \circ \alpha_c(\mathsf{id}_c)$, the equation $\gamma'_{F'c}(\varphi \circ \alpha) = \varphi_c \circ \gamma'_{Fc}(\alpha)$ holds by definition, thereby establishing naturality in F. Finally, suppose $f : b \to c$ in $\mathcal{C}$, and write γ_{Fc} for the function γ defined above and γ_{Fb} for the corresponding function for $b : \mathcal{C}$. Naturality in c means the commutativity of the following diagram:

$$
\begin{array}{ccc}
Fc & \xrightarrow{\;\gamma_{Fc}\;} & \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,c),F) \\
{\scriptstyle Ff}\Big\downarrow & & \Big\downarrow{\scriptstyle \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,f),F)} \\
Fb & \xrightarrow{\;\gamma_{Fb}\;} & \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,b),F)
\end{array}
$$

Suppose $h : Fc$. The component of $\gamma_{Fc}(h)$ at $a : \mathcal{C}$ is the function $\gamma_{Fc}(h)_a : \mathcal{C}(a,c) \to Fa$ defined by $g \mapsto Fg(h)$. The component of $\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,f),F) \circ \gamma_{Fc}(h)$ at $a : \mathcal{C}$ is thus the function $\gamma_c(h)_a \circ \mathcal{C}(a,f) : \mathcal{C}(a,b) \to Fa$ taking $g : a \to b$ to $F(f \circ g)(h)$. On the other hand, the component of $\gamma_{Fb}(Ff(h))$ at $a : \mathcal{C}$ is the function $\gamma_{Fb}(Ff(h))_a : \mathcal{C}(a,b) \to Fa$ taking g to $Fg(Ff(h))$. Since F is a contravariant functor, we have $F(f \circ g)(h) = Fg(Ff(h))$. This establishes the commutativity of the naturality square, and thus naturality in c as well as F. $\qquad \square$

The identification of an object with its collection of hom sets is formalized by the following corollary.

Corollary 2.4.11 (Representables are unique up to isomorphism). Suppose there is an isomorphism of presheaves $\mathcal{C}(-,a) \cong \mathcal{C}(-,b)$. Then $a \cong b$ in $\mathcal{C}$.

This corollary follows from the next one, which expresses that the image of the Yoneda embedding is isomorphic with $\mathcal{C}$ itself.

Corollary 2.4.12. The Yoneda embedding is fully faithful.

Proof. The Yoneda embedding defines a family of functions on the hom sets of $\mathcal{C}$:

$$よ_{b,c} : \mathcal{C}(b,c) \to \mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,b), \mathcal{C}(-,c))$$

$$f \mapsto \mathcal{C}(-,f)$$

By the Yoneda lemma, we immediately have $\mathbf{Set}^{\mathcal{C}^{\mathrm{op}}}(\mathcal{C}(-,b), \mathcal{C}(-,c)) \cong \mathcal{C}(b,c)$, which is the required isomorphism of hom sets. $\square$

Next, we have the following fact, that fully faithful functors transport isomorphisms in their codomain to their domain (they 'reflect' them).

Proposition 2.4.13 (Fully faithful functors reflect isomorphisms)**.** Suppose $F : \mathcal{C} \to \mathcal{D}$ is a fully faithful functor. If $f : a \to b$ is a morphism in $\mathcal{C}$ such that Ff is an isomorphism in $\mathcal{D}$, then f is an isomorphism in $\mathcal{C}$.

Proof. $Ff : Fa \to Fb$ being an isomorphism means that there is a morphism $g' : Fb \to Fa$ in $\mathcal{D}$ such that $g' \circ Ff = \mathrm{id}_{Fa}$ and $Ff \circ g' = \mathrm{id}_{Fb}$. By the functoriality of F, we have $\mathrm{id}_{Fa} = F\,\mathrm{id}_a$ and $\mathrm{id}_{Fb} = F\,\mathrm{id}_b$. Hence $g' \circ Ff = F\,\mathrm{id}_a$ and $Ff \circ g' = F\,\mathrm{id}_b$. Since F is isomorphic on hom sets, there is a unique $g : b \to a$ such that $g' = Fg$. Hence $Fg \circ Ff = F\,\mathrm{id}_a$ and $Ff \circ Fg = F\,\mathrm{id}_b$. By the functoriality of F, we have $Fg \circ Ff = F(g \circ f)$ and $Ff \circ Fg = F(f \circ g)$. Hence $F(g \circ f) = F\,\mathrm{id}_a$ and $F(f \circ g) = F\,\mathrm{id}_b$. Finally, since F is isomorphic on hom sets, we must have $g \circ f = \mathrm{id}_a$ and $f \circ g = \mathrm{id}_b$, and hence f is an isomorphism in $\mathcal{C}$. $\square$

And this gives us the proof we seek:

Proof of Corollary 2.4.11. Since the Yoneda embedding is fully faithful (Corollary 2.4.12), it reflects isomorphisms by Proposition 2.4.13. $\square$

Presheaves in the image of the Yoneda embedding consequently play a special rôle in category theory: to show that an arbitrary presheaf F is isomorphic to $\mathcal{C}(-,c)$ is to identify it with the object c itself, and in this case, we can say that F is *represented* by c. We therefore make the following definition.

Definition 2.4.14. Suppose F is a presheaf on $\mathcal{C}$. We say that it is *representable* if there is a natural isomorphism $F \cong \mathcal{C}(-,c)$ for some object $c : \mathcal{C}$ which we call its *representing object*; we call the natural isomorphism $\mathcal{C}(-,c) \Rightarrow F$ its *representation*. Dually, if F is instead a copresheaf, we call

it *corepresentable* if there is a natural isomorphism $F \cong \mathcal{C}(c, =)$, with c being the *corepresenting object*; we call the natural isomorphism $\mathcal{C}(c, =) \Rightarrow F$ its *corepresentation*.

Remark 2.4.15. Corepresentable copresheaves will play an important rôle later in this thesis: their coproducts are called *polynomial functors* (§3.5), and these will be used to formalize the interfaces of interacting adaptive systems.

Via the uniqueness of representables, the Yoneda lemma underlies universal constructions, since knowing the morphisms into or out of an object is enough to identify that object. The definition of a limit, notably, is the statement that morphisms into it correspond to morphisms into a diagram; and this in turn is equivalently the statement that $\lim$ is right adjoint to Δ. Indeed, adjointness is itself a certain kind of representability: the definition of adjoint functor (2.2.24) is precisely a natural characterization of morphisms into and out of objects, as related by the adjunction!

Proposition 2.4.16 (Adjoints are representable)**.** Suppose $R : \mathcal{D} \to \mathcal{C}$ is right adjoint to L. Then for every $d : \mathcal{D}$, the presheaf $\mathcal{D}(L-, d) : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ is represented by the object $Rd : \mathcal{C}$. Dually, the copresheaf $\mathcal{C}(c, R-) : \mathcal{D} \to \mathbf{Set}$ is corepresented by the object $Lc : \mathcal{D}$.

Proof. Since $L \vdash R$, we have an isomorphism $\mathcal{D}(Lc, d) \cong \mathcal{C}(c, Rd)$ natural in c and d. Therefore in particular we have a natural isomorphism of presheaves $\mathcal{C}(-, Rd) \Rightarrow \mathcal{D}(L-, d)$ and a natural isomorphism of copresheaves $\mathcal{D}(Lc, -) \Rightarrow \mathcal{C}(c, R-)$; the former is a representation and the latter a corepresentation. $\square$

From this, we can formalize the representability of limits and colimits.

Corollary 2.4.17 (Limits are representations)**.** Suppose $D : J \to \mathcal{C}$ is a diagram in $\mathcal{C}$. A limit of D is a representation of $\mathcal{C}^J(\Delta_{(-)}, D) : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$, or equivalently of $\mathbf{Set}^J(\Delta_1, \mathcal{C}(-, D))$. Dually, a colimit of D is a corepresentation of $\mathcal{C}^J(D, \Delta_{(-)}) : \mathcal{C} \to \mathbf{Set}$, or equivalently of $\mathbf{Set}^J(\Delta_1, \mathcal{C}(D, -))$.

Proof. If $\mathcal{C}$ has all limits of shape J, then this follows directly from the facts that $\lim$ is right adjoint to Δ (Proposition 2.3.48) and that adjoints are representable (Proposition 2.4.16); the dual result follows similarly from the fact that colim is left adjoint to Δ.

Otherwise, the limit case follows immediately from Lemma 2.3.51 (or equivalently Corollary 2.3.52) and the definition of representation (2.4.14); the colimit case is formally dual. $\square$

Accordingly, we recover the uniqueness of universal constructions.

Corollary 2.4.18. Adjoint functors are unique up to unique isomorphism.

Corollary 2.4.19. Limits and colimits are unique up to unique isomorphism.

Using these ideas, we obtain the following useful result relating limits and adjoint functors.

Proposition 2.4.20 (Right adjoints preserve limits). Suppose $D : J \to \mathcal{D}$ is a diagram in $\mathcal{D}$ and $L \vdash R : \mathcal{D} \to \mathcal{C}$ is an adjunction. Then $R \lim D \cong \lim RD$ in $\mathcal{C}$.

Proof. We have the following chain of natural isomorphisms:

$$
\begin{aligned}
\mathcal{C}(c, R \lim D) &\cong \mathcal{D}(Lc, \lim D) && \text{since } R \text{ is right adjoint to } L \\
&\cong \lim \mathcal{D}(Lc, D) && \text{since hom preserves limits} \\
&\cong \lim \mathcal{C}(c, RD) && \text{since } R \text{ is right adjoint to } L \\
&\cong \mathcal{C}(c, \lim RD) && \text{since hom preserves limits}
\end{aligned}
$$

Since representables are unique up to isomorphism and we have established an isomorphism of presheaves $\mathcal{C}(-, R \lim D) \cong \mathcal{C}(-, \lim RD)$, we must have $R \lim D \cong \lim RD$ in $\mathcal{C}$. $\qquad\square$

Remark 2.4.21. There is of course a dual result that left adjoints preserve colimits.

Remark 2.4.22. One might speculate about the converse: is it the case that the preservation of limits by a functor is enough to guarantee the existence of its left adjoint? The answer to this question is, "under certain conditions" on the size and structure of the categories and functors involved, and a positive answer is called an *adjoint functor theorem*. The "certain conditions" hold quite generally, and so it is often sufficient just to check whether a functor preserves limits (or colimits) to see that it is a right (or left) adjoint.

We end this chapter by closing the loop between universality and representability.

Proposition 2.4.23 (Universality of representability). Representable presheaves $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Set}$ correspond bijectively to universal morphisms from $1 : \mathbf{Set}$ to F.

Proof. A representation of F is a choice of object $c : \mathcal{C}$ and a natural isomorphism $\upsilon : \mathcal{C}(-, c) \Rightarrow F$. We construct a bijection between the set of representations of F and the set of universal morphisms from 1 to F. Therefore suppose given a representation $\upsilon : \mathcal{C}(-, c) \Rightarrow F$ of F; its component at $c : \mathcal{C}$ is the isomorphism $\upsilon_c : \mathcal{C}(c, c) \to Fc$. The Yoneda lemma assigns to υ an element $\gamma'(\upsilon) : 1 \to Fc$ satisfying $\gamma'(\upsilon) = \upsilon_c(\mathsf{id}_c)$. We now show that this element $\upsilon_c(\mathsf{id}_c)$ satisfies the universal property

that for all $f : 1 \to Fb$ there exists a unique morphism $h : b \to c$ in $\mathcal{C}$ such that $f = Fh \circ v_c(\mathsf{id}_c)$. Therefore let f be any such element $1 \to Fb$. Since v is a natural isomorphism, it has an inverse component at $b : \mathcal{C}$, denoted $v_b' : Fb \to \mathcal{C}(b, c)$, and so we obtain by composition an element $h := 1 \xrightarrow{f} Fb \xrightarrow{v_b'} \mathcal{C}(b, c)$ of $\mathcal{C}(b, c)$. Such an element is precisely a morphism $h : b \to c$ in $\mathcal{C}$. Consider now the following diagram:

$$
\begin{array}{ccccc}
1 & \xrightarrow{\ \mathsf{id}_c\ } & \mathcal{C}(c, c) & \xrightarrow{\ \mathcal{C}(h,c)\ } & \mathcal{C}(b, c) \\
& \searrow^{v_c(\mathsf{id}_c)} & \downarrow^{v_c} & & \downarrow^{v_b} \\
& & Fc & \xrightarrow{\ Fh\ } & Fb
\end{array}
$$

The triangle on the left commutes by definition and the square on the right commutes by the naturality of v, so that the whole diagram commutes. The composite morphism $\mathcal{C}(h, c) \circ \mathsf{id}_c$ along the top of the diagram picks out the element $\mathsf{id}_c \circ h$ of $\mathcal{C}(b, c)$. By the unitality of composition, this element is equal to h itself, so we can rewrite the diagram as follows:

$$
\begin{array}{ccc}
1 & \xrightarrow{\quad\quad h \quad\quad} & \mathcal{C}(b, c) \\
\downarrow^{v_c(\mathsf{id}_c)} & & \downarrow^{v_b} \\
Fc & \xrightarrow{\quad Fh \quad} & Fb
\end{array}
$$

Next, we can substitute the definition $h := v_b' \circ f$, and observe that $v_b \circ v_b' = \mathsf{id}_{Fb}$ (since v_b is an isomorphism with v_b' its inverse):

$$
\begin{array}{ccccc}
1 & \xrightarrow{\ f\ } & Fb & \xrightarrow{\ v_b'\ } & \mathcal{C}(b, c) \\
\downarrow^{v_c(\mathsf{id}_c)} & & \Big\| & \searrow & \downarrow^{v_b} \\
Fc & & \xrightarrow{\quad Fh \quad} & & Fb
\end{array}
$$

The commutativity of this diagram means that $f = Fh \circ v_c(\mathsf{id}_c)$. Moreover, since $h = v_b' \circ f$ and v_b' is an isomorphism, h is unique for a given f. Therefore $v_c(\mathsf{id}_c) : 1 \to Fc$ is a universal morphism from 1 to F.

Next, suppose given a universal morphism $u : 1 \to Fc$. The Yoneda lemmea associates to this element a natural transformation $\gamma(u)$ whose component at b is the function $\gamma(u)_b : \mathcal{C}(b, c) \to Fb$ which acts by $f \mapsto Ff(u)$. We need to show that this function is an isomorphism for every $b : \mathcal{C}$, so that $\gamma(u) : \mathcal{C}(-, c) \Rightarrow F$ is a natural isomorphism and hence F is represented by c. We therefore need to define an inverse function $\varphi_b : Fb \to \mathcal{C}(b, c)$, which we do using the universal property

of u: for each element $f : 1 \to Fb$, we have a unique morphism $h : b \to c$ such that $f = Fh(u)$. This unique h is an element of $\mathcal{C}(b, c)$, and so we can simply define $\varphi_b(f) := h$. The uniqueness of h ensures that φ_b is an inverse of $\gamma(u)_b$: observe that $\gamma(u)_b \circ \varphi_b$ acts by $f \mapsto h \mapsto Fh(u)$ and $f = Fh(u)$ by definition; in the opposite direction, we necessarily have $f \mapsto Ff(u) \mapsto f$.

We have constructed mappings between the set of representations of F and universal morphisms from 1 to F, so it remains to show that these mappings are mutually inverse. This again follows directly from the Yoneda lemma: the mapping of representations to universal morphisms takes a representation υ to the element $\gamma'(\upsilon)$ induced by the Yoneda lemma; and the mapping of universal morphisms to representations takes a universal morphism u to the natural transformation $\gamma(u)$ induced by the Yoneda lemma. Since the functions γ and γ' are mutually inverse, so must these mappings be: $\gamma \circ \gamma'(\upsilon) = \upsilon$ and $\gamma' \circ \gamma(u) = u$. $\qquad\square$

Using the universality of representability, and the uniqueness of universal morphisms, and the representability of limits and adjoints, we therefore obtain alternative proofs of the uniqueness of those universal constructions.

3. Algebraic connectomics

In Chapter 2, we motivated applied category theory in the context of complex systems like brains by its abilities to relate structure and function, to translate between models and frameworks, and to distil phenomena to their essences. However, the focus in that chapter was on 'one-dimensional' morphisms, which can be understood as connecting one interface to another, with the composition of 1-cells representing something like the 'end-to-end' composition of processes; although we considered some higher-dimensional category theory, this was largely restricted to weakening equalities and thus comparing morphisms.

Because systems can be placed 'side-by-side' as well as end-to-end, and because two systems placed side by side may be nontrivially wired together, in this chapter we extend the higher-dimensional categorical language accordingly, with a particular focus once more on the graphical and diagrammatic representation of systems and processes. In line with the distinction made in §2.2.3 between syntax and semantics, our treatment here of the syntax of wiring—of *connectomics*—is largely 'algebraic'. Later, in Chapter 6, we will see how our semantic focus will be 'coalgebraic'.

We will begin therefore by introducing the graphical calculus of monoidal categories, which allow us to depict and reason about sequential and parallel composition simultaneously. We follow this with the formal underpinnings of the structure—to use the term from Chapter 2, a monoidal structure is a 'well-behaved' tensor product—before explaining how monoidal categories relate to the higher category theory of Chapter 2 using the notion of bicategory. We then make use of the extra freedom afforded by bicategories to consider parameterized systems, with which we can model systems that not only act but also learn.

By this point, we will find ourselves ready to apply our new toolkit, and so in §3.3, we use functorial semantics to define a graphical algebra for neural circuits, revisiting our first example from Chapter 2. This involves a change of perspective from the graphical calculus with which we begin the chapter: instead of using the composition of morphisms to encode the plugging-together of systems at the same 'scale' or "level of hierarchy", we use composition to encode the wiring of circuits at one level into systems at a higher level. Although formally closely related to monoidal

categories, this 'hierarchical' perspective is strictly speaking *multicategorical* and allows morphisms' domains to take very general shapes.

After this extended example, we return to algebra, explaining what makes monoidal categories *monoidal*, and using the related concept of *monad* to explain how we think of them as algebraic; monads will later prove to be of importance in categorical approaches to probability theory. Finally, we end the chapter by introducing the richly structured category of polynomial functors $\mathbf{Set} \to \mathbf{Set}$, which we will use in Chapter 6 both to formalize a wide variety of open dynamical systems as well as to specify the shapes of those systems' interfaces.

Excepting the extended example of §3.3, the content of this chapter is well known to category-theoreticians. However, since it is not well known to mathematical scientists, we have again endeavoured to supply detailed motivations for the concepts and results that we introduce.

3.1. Categories and calculi for process theories

In this section, we introduce an alternative way of depicting morphisms and their composites in categories equipped with notions of both sequential and parallel composition. Such categories are useful for representing processes in which information flows: we formalize the processes as morphisms, and consider the flow as from domain to codomain, even when the categories themselves are quite abstract and lack a notion of time with which to make sense of 'flow'. In such contexts, the categories are often not only *monoidal*, but also *copy-discard* categories, since a distinctive feature of classical information is that it can be copied and deleted. Monoidal categories will therefore be important not only in depicting composite computations (as indicated in §2.1.1.3), but also in depicting and manipulating the factorization of probabilistic models (as indicated in §2.1.1.2).

3.1.1. String diagrams

Rather than beginning with the formal definition of "monoidal category", we start with the associated graphical calculus of *string diagrams* and its intuition.

Sequential and parallel composition   Diagrams in the graphical calculus depict morphisms as boxes on strings: the strings are labelled with objects, and a string without a box on it can be interpreted as an identity morphism. Sequential composition is represented by connecting strings together, and parallel composition by placing diagrams adjacent to one another; sequential composition distributes over parallel, and so we can of course compose parallel boxes in sequence.

Because monoidal structures are "well-behaved tensor products", we will typically denote them using the same symbols that we adopted in Chapter 2, with sequential composition denoted by $\circ$ and parallel composition (tensor) denoted by $\otimes$. Diagrams will be read in the direction of information flow, which will be either bottom-to-top or left-to-right; we will adopt the former convention in this section.

In this way, $c : X \to Y$, $\mathsf{id}_X : X \to X$, $d \circ c : X \xrightarrow{c} Y \xrightarrow{d} Z$, and $f \otimes g : X \otimes Y \to A \otimes B$ are depicted respectively as:
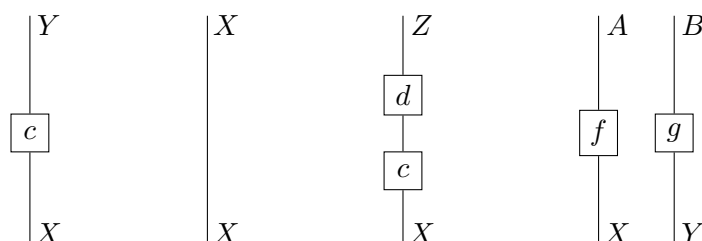
A monoidal structure comes with a *monoidal unit*, which we will also continue to call a *tensor unit*, and which will be not be depicted in diagrams, but rather left implicit. (Alternatively, it is depicted as the "empty diagram".) This is justified, as we will see, by the requirement that $I \otimes X \cong X \cong X \otimes I$ naturally in X.

States and costates   In Remark 2.2.8, we called a morphism $I \to X$ out of the tensor unit a *generalized element*, but owing to the many rôles they play, such morphisms go by many names. When we think of X as representing a system, we will also call such morphisms *states* of X. Dually, morphisms $X \to I$ can be called *costates*, or sometimes *effects*. When the unit object is the terminal object (such as when the monoidal structure is given by the categorical product), then these costates are trivial. In other categories, costates may be more effectful, and so carry more information: for example, in a category of vector spaces, states are vectors, costates are linear functionals, and so the composite of a state with a costate is an inner product.

Graphically, states $\eta : I \to X$ and costates $\epsilon : X \to I$ will be represented respectively as follows:

Discarding, marginalization, and causality   In a category with only trivial effects, we can think of these as witnessing the 'discarding' of information: in electronics terms, they "send the

signal to ground". For this reason, we will denote such trivial effects by the symbol $\bar{\bar{\top}}$, writing $\bar{\bar{\top}}_X : X \to I$ for each object X.

We can use discarding to depict *marginalization*. Given a 'joint' state (a state of a tensor product) $\omega : I \to X \otimes Y$, we can discard either Y or X to obtain 'marginal' states ω_1 of X and ω_2 of Y respectively, as in the following depiction:

We will see in Chapter 4 how this corresponds to the marginalization familiar from probability theory.

To make the notion of discarding more mathematically precise, we can use it to encode a causality condition: physically realistic processes should not be able to affect the past.

Definition 3.1.1. Whenever a morphism c satisfies the equation

we will say that c is *causal*: the equation says that, if you do c and throw away the result, the effect is of not having done c at all—and so c could not have had an anti-causal effect on its input.

Remark 3.1.2. If in a category every morphism is causal, then this is equivalently a statement of the naturality of family of discarding morphisms $\bar{\bar{\top}}_X : X \to I$, which implies that there is only one such morphism $X \to I$ for every object X, and which therefore means that I must be a terminal object.

Some categories of interest will have nontrivial costates, yet we will still need notions of discarding and marginalization. In these categories, it suffices to ask for each object X to be equipped with a 'comonoid' structure (to be elaborated in §3.4.1), of which one part is a 'counit' morphism $X \to I$ which can play a discarding rôle, and which we will therefore also denote by $\bar{\bar{\top}}_X$.

Copying   The other part of a comonoid structure on X is a 'copying' map $\blacktriangledown_X : X \to X \otimes X$, which has an intuitive graphical representation. As we will see in §3.4.1, the *comonoid laws* say

that copying must interact nicely with the discarding maps:
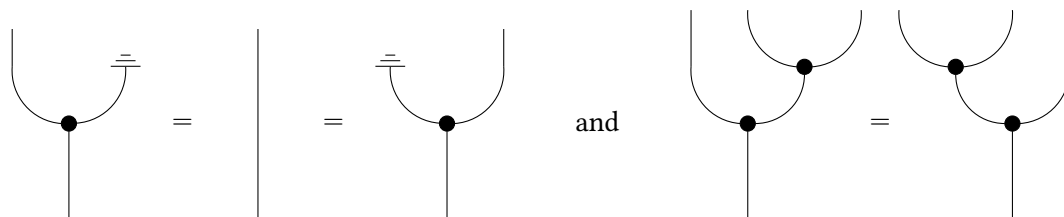
These equations say that making a copy and throwing it away is the same as not making a copy (left, *counitality*; and that in copying a copy, it doesn't matter which copy you copy (right, *coassociativity*).

Definition 3.1.3. A category with a comonoid structure $(\curlyvee_X, \bar{\top}_X)$ for every object X is called a *copy-discard category* [60].

Symmetry   In all our applications, the tensor product structure will be *symmetric*, meaning that $X \otimes Y$ can reversibly be turned into $Y \otimes X$ simply by swapping terms around. In the graphical calculus, we depict this by the swapping of wires, which we ask to satisfy the following equations:
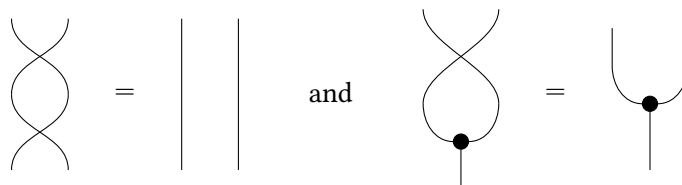
The equations say that swapping is self-inverse (on the left), and that copying is invariant under the symmetry (on the right). (Strictly speaking, the right equation is an axiom called *cocommutativity* that we additionally ask the comonoid structure to satisfy in the presence of a symmetric tensor.)

3.1.2. Monoidal categories

It being important to use tools appropriate for the jobs at hand, we will not always work just with the graphical calculus: we will need to translate between string diagrams and the symbolic algebra of Chapter 2. In the first instance, this means making mathematical sense of the graphical calculus itself, for which the key definition is that of the monoidal category.

Definition 3.1.4. We will call a category $\mathcal{C}$ *monoidal* if it is equipped with a functor $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ called the *tensor* or *monoidal product* along with an object $I : \mathcal{C}$ called the *monoidal unit* and three natural isomorphisms

1. an *associator* $\alpha : ((-) \otimes (-)) \otimes (-) \Rightarrow (-) \otimes ((-) \otimes (-))$;

2. a *left unitor* $\lambda : I \otimes (-) \Rightarrow (-)$; and

3. a *right unitor* $\rho : (-) \otimes I \Rightarrow (-)$

such that the unitors are compatible with the associator, *i.e.* for all $a, b : \mathcal{C}$ the diagram

$$(a \otimes I) \otimes b \xrightarrow{\ \alpha_{a,I,b}\ } a \otimes (I \otimes b)$$

with $\rho_a \otimes \mathsf{id}_b$ and $\mathsf{id}_a \otimes \lambda_b$ to $a \otimes b$

commutes, and such that the associativity is 'order-independent', *i.e.* for all $a, b, c, d : \mathcal{C}$ the diagram

$$(a \otimes (b \otimes c)) \otimes d \xrightarrow{\ \alpha_{a,b\otimes c,d}\ } a \otimes ((b \otimes c) \otimes d)$$

with $\alpha_{a,b,c} \otimes \mathsf{id}_d$, $\mathsf{id}_a \otimes \alpha_{b,c,d}$, $((a \otimes b) \otimes c) \otimes d$, $a \otimes (b \otimes (c \otimes d))$, $\alpha_{a\otimes b,c,d}$, $\alpha_{a,b,c\otimes d}$, $(a \otimes b) \otimes (c \otimes d)$

commutes.

We call $\mathcal{C}$ *strict monoidal* if the associator and unitors are equalities rather than isomorphisms; in this case, the diagrams above commute by definition.

Example 3.1.5. Any category equipped with a tensor product in the sense of Definition 2.2.3 where the structure isomorphisms are additionally natural and satisfy the axioms of compatibility and order-independence is a monoidal category.

Example 3.1.6. If $(\mathcal{C}, \otimes, I)$ is a monoidal category, then so is $(\mathcal{C}^{\mathrm{op}}, \otimes^{\mathrm{op}}, I)$, where $\otimes^{\mathrm{op}}$ is the induced opposite functor $\mathcal{C}^{\mathrm{op}} \times \mathcal{C}^{\mathrm{op}} \to \mathcal{C}^{\mathrm{op}}$.

The associativity of the tensor is what allows us to depict string diagrams "without brackets" indicating the order of tensoring, and the unitality is what allows us to omit the monoidal unit from the diagrams. Note that the functoriality of the tensor means that $\otimes$ distributes over $\circ$ as in $(f' \circ f) \otimes (g' \circ g) = (f' \otimes g') \circ (f \otimes g)$, both of which expressions are therefore depicted as
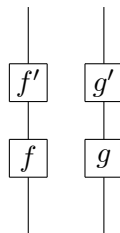
The symmetry of a monoidal structure is formalized as follows.

Definition 3.1.7. A *symmetric monoidal* category is a monoidal category $(\mathcal{C}, \otimes, I, \alpha, \lambda, \rho)$ that is additionally equipped with a natural isomorphism $\sigma : (-) \otimes (=) \Rightarrow (=) \otimes (-)$, called the *symmetry*, such that $\sigma_{b,a} \circ \sigma_{a,b} = \mathsf{id}_{a \otimes b}$ for all $a, b : \mathcal{C}$, and whose compatibility with the associator is witnessed by the commutativity of the following diagram:

$$
\begin{array}{ccccc}
(a \otimes b) \otimes c & \xrightarrow{\ \alpha_{a,b,c}\ } & a \otimes (b \otimes c) & \xrightarrow{\ \sigma_{a,b \otimes c}\ } & (b \otimes c) \otimes a \\
\Big\downarrow{\scriptstyle \sigma_{a,b} \otimes \mathsf{id}_c} & & & & \Big\downarrow{\scriptstyle \alpha_{b,c,a}} \\
(b \otimes a) \otimes c & \xrightarrow[\ \alpha_{b,a,c}\ ]{} & b \otimes (a \otimes c) & \xrightarrow[\ \mathsf{id}_b \otimes \sigma_{a,c}\ ]{} & b \otimes (c \otimes a)
\end{array}
$$

Here is a familiar family of examples of symmetric, but not strict, monoidal categories.

Example 3.1.8. Any category within which every pair of objects has a product is said to *have finite products*, and any category with finite products and a terminal object is a monoidal category. This includes the Cartesian products of sets (Definition 2.2.1 and Example 2.2.5) and of categories (Propositions 2.2.14 and 2.2.16).

To see that the Cartesian product of sets is not strictly associative, observe that the elements of $A \times (B \times C)$ are tuples $(a, (b, c))$ whereas the elements of $(A \times B) \times C$ are tuples $((a, b), c)$; evidently, these two sets are isomorphic, but not equal, and the same holds for the product of categories.

At the same time, it is easy to see that a Cartesian product is symmetric: we have $A \times B \cong B \times A$ by the mapping $(a, b) \leftrightarrow (b, a)$.

And here is a family of examples of strict, but not symmetric, monoidal categories.

Example 3.1.9. If $\mathcal{C}$ is any category, then the category $\mathcal{C}^{\mathcal{C}}$ of *endofunctors* $\mathcal{C} \to \mathcal{C}$ is a strict monoidal category, where the monoidal product is given by composition $\circ$ of endofunctors and the monoidal unit is the identity functor $\mathsf{id}_{\mathcal{C}}$ on $\mathcal{C}$. That the monoidal structure here is strict follows from the fact that composition in a category is strictly associative and unital.

In practice, we will tend to encounter strict monoidal categories only when the monoidal structure derives from the composition operator of a category, as in the preceding example. However, when we work with the graphical calculus, we are often implicitly working with strict monoidal structure, as a result of the following important theorem.

Theorem 3.1.10 (Mac Lane [175, Theorem XI.3.1]). Every monoidal category is strong monoidally equivalent to a strict monoidal one.

As a consequence of this *coherence theorem*, any two string diagrams where one can be transformed into the other by a purely topological transformation are equal, as in the following example (read from left to right):
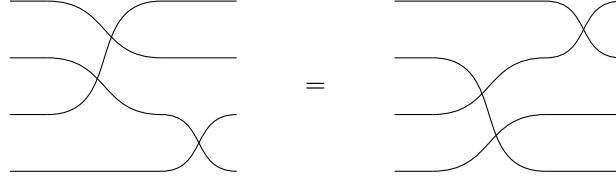
This follows because the coherence theorem renders parallel morphisms entirely constructed from identities, associators and unitors (and the symmetry, as long as it is strictly self-inverse) equal "on the nose"[1].

To make sense of the notion of *strong monoidal equivalence*, we need a notion of functor that preserves monoidal structure; we define the 'weak' case first.

Definition 3.1.11. Suppose $(\mathcal{C}, \otimes_{\mathcal{C}}, I_{\mathcal{C}})$ and $(\mathcal{D}, \otimes_{\mathcal{D}}, I_{\mathcal{D}})$ are monoidal categories. A *lax monoidal functor* $(\mathcal{C}, \otimes_{\mathcal{C}}, I_{\mathcal{C}}) \to (\mathcal{D}, \otimes_{\mathcal{D}}, I_{\mathcal{D}})$ is a triple of

1. a functor $F : \mathcal{C} \to \mathcal{D}$;

2. a state $\epsilon : I_{\mathcal{D}} \to F(I_{\mathcal{C}})$ called the *unit*; and

3. a natural transformation, the *laxator*, $\mu : F(-) \otimes_{\mathcal{D}} F(=) \Rightarrow F((-) \otimes_{\mathcal{C}} (=))$

satisfying the axioms of

(a) associativity, in that the following diagram commutes

$$
\begin{array}{ccc}
(F(a) \otimes_{\mathcal{D}} F(b)) \otimes_{\mathcal{D}} F(c) & \xrightarrow{\alpha^{\mathcal{D}}_{F(a),F(b),F(c)}} & F(a) \otimes_{\mathcal{D}} (F(b) \otimes_{\mathcal{D}} F(c)) \\
\downarrow{\scriptstyle \mu_{a,b} \otimes_{\mathcal{D}} F(c)} & & \downarrow{\scriptstyle F(a) \otimes_{\mathcal{D}} \mu_{b,c}} \\
F(a \otimes_{\mathcal{C}} b) \otimes_{\mathcal{D}} F(c) & & F(a) \otimes_{\mathcal{D}} F(b \otimes_{\mathcal{C}} c) \\
\downarrow{\scriptstyle \mu_{a \otimes_{\mathcal{C}} b, c}} & & \downarrow{\scriptstyle \mu_{F(a), b \otimes_{\mathcal{C}} c}} \\
F((a \otimes_{\mathcal{C}} b) \otimes_{\mathcal{C}} c) & \xrightarrow{F(\alpha^{\mathcal{C}}_{a,b,c})} & F(a \otimes_{\mathcal{C}} (b \otimes_{\mathcal{C}} c))
\end{array}
$$

where $\alpha^{\mathcal{C}}$ and $\alpha^{\mathcal{D}}$ are the associators of the respective monoidal structures on $\mathcal{C}$ and $\mathcal{D}$; and

---

[1] This process of turning natural isomorphisms into equalities is called *strictification*.

(b) (left and right) unitality, in that the following diagrams commute

$$
\begin{array}{ccc}
I_{\mathcal{D}} \otimes_{\mathcal{D}} F(a) & \xrightarrow{\ \epsilon \otimes_{\mathcal{D}} F(a)\ } & F(I_{\mathcal{C}}) \otimes_{\mathcal{D}} F(a) \\
\downarrow{\lambda^{\mathcal{D}}_{F(a)}} & & \downarrow{\mu_{I_{\mathcal{C}},a}} \\
F(a) & \xleftarrow{\ F(\lambda^{\mathcal{C}}_a)\ } & F(I_{\mathcal{C}} \otimes_{\mathcal{C}} a)
\end{array}
\quad \text{and} \quad
\begin{array}{ccc}
F(a) \otimes_{\mathcal{D}} I_{\mathcal{D}} & \xrightarrow{\ F(a) \otimes_{\mathcal{D}} \epsilon\ } & F(a) \otimes_{\mathcal{D}} F(I_{\mathcal{C}}) \\
\downarrow{\rho^{\mathcal{D}}_{F(a)}} & & \downarrow{\mu_{a,I_{\mathcal{C}}}} \\
F(a) & \xleftarrow{\ F(\rho^{\mathcal{C}}_a)\ } & F(a \otimes_{\mathcal{C}} I_{\mathcal{C}})
\end{array}
$$

where $\lambda^{\mathcal{C}}$ and $\lambda^{\mathcal{D}}$ are the left, and $\rho^{\mathcal{C}}$ and $\rho^{\mathcal{D}}$ the right, unitors of the respective monoidal structures on $\mathcal{C}$ and $\mathcal{D}$.

A *strong monoidal functor* is a lax monoidal functor for which the unit and laxator are isomorphisms. A strong monoidal equivalence is therefore an equivalence of categories in which the two functors are strong monoidal.

Remark 3.1.12. Laxness can be read as a sign of an "emergent property": if F is lax monoidal, then this means there are systems of type $F(X \otimes Y)$ that do not arise simply by placing a system of type $F(X)$ beside a system of type $F(Y)$ using $\otimes$; whereas if F is strong monoidal, then there are no such 'emergent' systems. More generally, we can think of emergence as an indication of higher-dimensional structure that is hidden when one restricts oneself to lower dimensions (and hence can appear mysterious). In this example, the higher-dimensional structure is the 2-cell of the laxator.

There is of course a notion of monoidal natural transformation, making monoidal categories, lax monoidal functors, and monoidal natural transformations into the constituents of a 2-category.

Definition 3.1.13. If (F, μ, ϵ) and (F', μ', ϵ') are lax monoidal functors $(\mathcal{C}, \otimes_{\mathcal{C}}, I_{\mathcal{C}}) \to (\mathcal{D}, \otimes_{\mathcal{D}}, I_{\mathcal{D}})$, then a *monoidal natural transformation* $\alpha : (F, \mu, \epsilon) \Rightarrow (F', \mu', \epsilon')$ is a natural transformation $\alpha : F \Rightarrow F'$ that is compatible with the unitors

$$
\begin{array}{ccc}
 & I_{\mathcal{D}} & \\
{\scriptstyle \epsilon}\swarrow & & \searrow{\scriptstyle \epsilon'} \\
F(I_{\mathcal{C}}) & \xrightarrow[\ \alpha_{I_{\mathcal{C}}}\ ]{} & F'(I_{\mathcal{C}})
\end{array}
$$

and the laxators

$$
\begin{array}{ccc}
Fa \otimes_{\mathcal{D}} Fb & \xrightarrow{\ \alpha_a \otimes_{\mathcal{D}} \alpha_b\ } & F'a \otimes_{\mathcal{C}} F'b \\
\downarrow{\mu_{a,b}} & & \downarrow{\mu'_{a,b}} \\
F(a \otimes_{\mathcal{C}} b) & \xrightarrow[\ \alpha_{a \otimes_{\mathcal{C}} b}\ ]{} & F'(a \otimes_{\mathcal{C}} b)
\end{array}
$$

for all $a, b : \mathcal{C}$.

Proposition 3.1.14. Monoidal categories, lax monoidal functors, and monoidal natural transformations form the 0-cells, 1-cells, and 2-cells of a 2-category, denoted **MonCat**.

Proof. Given composable lax monoidal functors $(F, \epsilon, \mu) : (\mathcal{C}, \otimes_{\mathcal{C}}, I_{\mathcal{C}}) \to (\mathcal{D}, \otimes_{\mathcal{D}}, I_{\mathcal{D}})$ and $(F', \epsilon', \mu') : (\mathcal{D}, \otimes_{\mathcal{D}}, I_{\mathcal{D}}) \to (\mathcal{E}, \otimes_{\mathcal{E}}, I_{\mathcal{E}})$, form their horizontal composite as follows. The functors compose as functors, $G \circ F$. The composite state is given by $I_{\mathcal{E}} \xrightarrow{\epsilon'} F'(I_{\mathcal{D}}) \xrightarrow{F'\epsilon} F'F(I_{\mathcal{C}})$. The laxator is given by

$$F'F(-) \otimes_{\mathcal{E}} F'F(=) \xRightarrow{\mu'_{F(-),F(=)}} F'(F(-) \otimes_{\mathcal{D}} F(=)) \xRightarrow{F'\mu_{a,b}} F'F((-) \otimes_{\mathcal{C}} (=)).$$

The identity lax monoidal functor on $\mathcal{C}$ is given by $(\mathsf{id}_{\mathcal{C}}, \mathsf{id}_{I_{\mathcal{C}}}, \mathsf{id}_{(-)\otimes_{\mathcal{C}}(=)})$. Unitality and associativity of composition of lax monoidal functors follow straightforwardly from unitality and associativity of composition of morphisms, functors, and natural transformations. Monoidal natural transformations compose vertically as natural transformations, and it is easy to see that the composites satisfy the compatibility conditions by pasting the relevant diagrams. $\qquad \square$

3.1.3. Closed monoidal categories

Since one source of monoidal structures is the generalization of the categorical product, it is no surprise that there is a corresponding generalization of exponentials: a 'tensor-hom' adjunction that induces a concept of closed monoidal category. Such categories will be important later in the thesis when we consider learning and adaptive systems: our compositional model of predictive coding, for example, will be built on a certain generalized exponential (see Remark 6.3.2).

Definition 3.1.15. Let $(\mathcal{C}, \otimes, I)$ be a monoidal category. When there is an object $e : \mathcal{C}$ such that $\mathcal{C}(x, e) \cong \mathcal{C}(x \otimes y, z)$ naturally in x, we say that e is an *internal hom object* and denote it by $[y, z]$. The image of $\mathsf{id}_{[y,z]}$ under the isomorphism is called the *evaluation map* and is written $\mathsf{ev}_{y,z} : [y, z] \otimes y \to z$.

Proposition 3.1.16. When the isomorphism $\mathcal{C}(x \otimes y, z) \cong \mathcal{C}(x, [y, z])$ is additionally natural in z, we obtain an adjunction $(-) \otimes y \dashv [y, -]$ called the *tensor-hom adjunction*, which uniquely determines a functor $\mathcal{C}^{\mathrm{op}} \times \mathcal{C} \to \mathcal{C} : (y, z) \mapsto [y, z]$ that we call the *internal hom* for $\mathcal{C}$.

Proof. A direct generalization of the Cartesian case (Proposition 2.3.56). $\qquad \square$

Definition 3.1.17. A monoidal category $\mathcal{C}$ with a corresponding internal hom is called *monoidal closed*.

Example 3.1.18. The category of finite-dimensional real vector spaces and linear maps between them is monoidal closed with respect to the tensor product of vector spaces, as each space of linear maps is again a vector space and the tensor is necessarily bilinear.

As in the Cartesian case, monoidal closed categories can reason about themselves.

Proposition 3.1.19. A monoidal closed category is enriched in itself.

And when a category is enriched in a symmetric monoidal category, then its hom functor is likewise enriched.

Proposition 3.1.20. Suppose $\mathcal{C}$ is an $\mathcal{E}$-category where $\mathcal{E}$ is symmetric monoidal closed. Then the hom functor $\mathcal{C}(-, =)$ is an $\mathcal{E}$-functor.

Proof. A direct generalization of Proposition 2.3.61. □

Remark 3.1.21. Since Cartesian closed categories have a rich internal logic, via the Curry-Howard-Lambek correspondence, one might wonder if there is an analogous situation for monoidal closed categories. To a certain intricate extent there is: the internal logic of monoidal closed categories is generally known as *linear logic*, and its corresponding language *linear type theory*. These are 'refinements' of intuitionistic logic and type theory which of course coincide in the Cartesian case, but which more generally clarify certain logical interactions; we shall say no more in this thesis, except that such logics find application in quantum mechanics, owing to the monoidal closed structure of vector spaces, where the linear structure constrains the use of resources (in relation, for example, to the famous quantum 'no-cloning' and 'no-deleting' theorems).

With respect to *dependent* types, the situation is a little more vexed, as the existence of well-behaved dependent sums and products classically depends on the existence of pullbacks and their coherence with products (and, for example, the tensor product of vector spaces is not a categorical product); this means that classical dependent data is somehow not resource-sensitive. Nonetheless, various proposals have been made to unify linear logic with dependent type theory[14, 111, 174, 182, 269]: the simplest of these proceed by requiring dependence to be somehow Cartesian, which is the approach we will take in Chapter 6 when we face a similar quandary in the context of defining a category of polynomial functors with non-deterministic feedback. (We will see in Chapter 4 that the property of Cartesianness is equally closely related to determinism.)

3.1.4. Bicategories

Monoidal categories are not the first two-dimensional categorical structures we have so far encountered, the other primary example being 2-categories. These two classes of examples are closely related: a strict monoidal category is a 2-category with one object; and so just as a monoidal category is a correspondingly weakened version, a bicategory is a 'weak 2-category'.

Definition 3.1.22. A *bicategory* $\mathcal{B}$ is constituted by

1. a set $\mathcal{B}_0$ of *objects* or *0-cells*;

2. for each pair (A, B) of $\mathcal{B}$-objects, a category $\mathcal{B}(A, B)$ called the *hom category*, the objects of which are the *morphisms* or *1-cells* from A to B, and the morphisms of which are the *2-cells* between those 1-cells;

3. for each 0-cell A, a 1-cell $\mathsf{id}_a : \mathcal{B}(A, A)$ witnessing *identity*; and

4. for each triple (A, B, C) of 0-cells, a functor $\diamond_{A,B,C} : \mathcal{B}(B, C) \times \mathcal{B}(A, B) \to \mathcal{B}(A, C)$ witnessing *horizontal composition* (with *vertical composition* referring to composition within each hom category);

5. for each pair (A, B) of 0-cells, natural isomorphisms $\rho_{A,B}$ (the *right unitor*) and $\lambda_{A,B}$ (the *left unitor*) witnessing the unitality of horizontal composition, as in the diagrams

where $\Lambda : \mathbf{1} \times (-) \Rightarrow (-)$ and $\mathrm{P} : (-) \times \mathbf{1} \Rightarrow (-)$ are the (almost trivial) left and right unitors of the product $\times$ on $\mathbf{Cat}$; and

6. for each quadruple (A, B, C, D) of 0-cells, a natural isomorphism $\alpha_{A,B,C,D}$ witnessing the

84

associativity of horizontal composition, as in the diagram

$$
\begin{array}{ccc}
\big(\mathcal{B}(C,D) \times \mathcal{B}(B,C)\big) \times \mathcal{B}(A,B) & \xrightarrow{\;\mathrm{A}_{\mathcal{B}(C,D),\mathcal{B}(B,C),\mathcal{B}(A,B)}\;} & \mathcal{B}(C,D) \times \big(\mathcal{B}(B,C) \times \mathcal{B}(A,B)\big)
\end{array}
$$

with vertical maps $\diamond_{B,C,D} \times \mathcal{B}(A,B)$ and $\mathcal{B}(C,D) \times \diamond_{A,B,C}$, the 2-cell $\alpha_{A,B,C,D}$, objects $\mathcal{B}(B,D) \times \mathcal{B}(A,B)$ and $\mathcal{B}(C,D) \times \mathcal{B}(A,C)$, diagonal maps $\diamond_{A,B,D}$ and $\diamond_{A,C,D}$ into $\mathcal{B}(A,D)$

where $\mathrm{A} : ((-) \times (-)) \times (-) \Rightarrow (-) \times ((-) \times (-))$ is the (almost trivial) associator of the product $\times$ on $\mathbf{Cat}$;

such that the unitors are compatible with the associator, *i.e.* for all 1-cells $a : \mathcal{B}(A,B)$ and $b : \mathcal{B}(B,C)$ the diagram

$$
\begin{array}{ccc}
(b \diamond \mathsf{id}_B) \diamond a & \xrightarrow{\;\alpha_{b,\mathsf{id}_B,a}\;} & b \diamond (\mathsf{id}_B \diamond a) \\
& & \\
\searrow^{\rho_b \diamond \mathsf{id}_a} & & \swarrow_{\mathsf{id}_b \diamond \lambda_a} \\
& b \diamond a &
\end{array}
$$

commutes (where we have omitted the subscripts indexing the 0-cells on α, ρ, and λ); and such that the associativity is 'order-independent', *i.e.* for all 1-cells $a : \mathcal{B}(A,B)$, $b : \mathcal{B}(B,C)$, $c : \mathcal{B}(C,D)$, and $d : \mathcal{B}(D,E)$ the diagram

$$
\begin{array}{ccc}
(a \diamond (b \diamond c)) \diamond d & \xrightarrow{\;\alpha_{a,b \diamond c,d}\;} & a \diamond ((b \diamond c) \diamond d) \\
\uparrow^{\alpha_{a,b,c} \diamond \mathsf{id}_d} & & \downarrow^{\mathsf{id}_a \diamond \alpha_{b,c,d}} \\
((a \diamond b) \diamond c) \diamond d & & a \diamond (b \diamond (c \diamond d)) \\
& \searrow_{\alpha_{a \diamond b,c,d}} \qquad \nearrow_{\alpha_{a,b,c \diamond d}} & \\
& (a \diamond b) \diamond (c \diamond d) &
\end{array}
$$

commutes (where we have again omitted the subscripts indexing the 0-cells on α).

Remark 3.1.23. Just as a 2-category is a category enriched in $\mathbf{Cat}$, a bicategory is a category *weakly enriched* in $\mathbf{Cat}$. This is easy to see by comparing Definition 3.1.22 with Definition 2.2.6: the former is obtained from the latter by taking $\mathcal{E}$ to be $\mathbf{Cat}$ and filling the unitality and associativity

diagrams with nontrivial fillers which are required to satisfy coherence laws generalizing those of the monoidal category structure (Definition 3.1.4). Conceptually, we can see this weakening in the context of our brief discussion of emergence above (Remark 3.1.12): we recognize the property of axiom-satisfaction as a shadow of a higher-dimensional structure (the fillers), which we categorify accordingly.

Bicategories will appear later in this thesis when we construct categories of dynamical hierarchical inference systems: the construction proceeds by using polynomial functors to "wire together" categories of dynamical systems, and the composition of polynomials distributes weakly but naturally over the categories of systems, thereby producing a category weakly enriched in **Cat**. Before then, we will encounter bicategories in the abstract context of general parameterized morphisms, where the 2-cells witness changes of parameter.

For now, our first examples of bicategories are induced by monoidal categories, which are equivalently single-object bicategories.

Proposition 3.1.24. Suppose $(\mathcal{C}, \otimes, I)$ is a monoidal category. Then there is a bicategory $\mathbf{B}\mathcal{C}$ with a single 0-cell, $*$, and whose category of 1-cells $\mathbf{B}\mathcal{C}(*, *)$ is $\mathcal{C}$. The identity 1-cell is I, and horizontal composition is given by the monoidal product $\mathcal{C}$; vertical composition is just the composition of morphisms in $\mathcal{C}$. The unitors and associator of the bicategory structure are the unitors and associator of the monoidal structure. We call $\mathbf{B}\mathcal{C}$ the *delooping* of $\mathcal{C}$.

Proof. The bicategory axioms are satisfied immediately, because the structure morphisms satisfy the (in this case identical) monoidal category axioms. $\square$

In the opposite direction, the equivalence is witnessed by the following proposition.

Proposition 3.1.25. Suppose $\mathcal{B}$ is a bicategory with a single 0-cell, $*$, and whose horizontal composition is denoted $\diamond$. Then $\big(\mathcal{B}(*, *), \diamond, \mathsf{id}_*\big)$ is a monoidal category.

Remark 3.1.26. It is possible to define a notion of monoidal bicategory, as something like a monoidal category weakly enriched in **Cat**, or as a one-object 'tricategory', and in many cases the bicategories considered below are likely to have such structure. We will say a little more about this in Remark 3.4.7 below, but will not define or make formal use of this higher structure in this thesis.

More generally, there are analogues of the other structures and results of basic category theory introduced both in this chapter and in Chapter 2 that are applicable to higher-dimensional categories such as bicategories, but they too will not play an important rôle in this thesis.

3.2. Parameterized systems

A category does not have to be monoidal closed for us to be able to talk about "controlled processes" in it: its being monoidal is sufficient, for we can consider morphisms of the form $P \otimes X \to Y$ and treat the object P as an object of adjustable parameters. Parameterized morphisms of this form can easily be made to compose: given another morphism $Q \otimes Y \to Z$, we can straightforwardly obtain a composite parameterized morphism $(Q \otimes P) \otimes X \to Z$, as we elaborate in §3.2.1 below.

Categories of such parameterized morphisms play a central rôle in the compositional modelling of cybernetic systems[54, 251], where we typically see the parameter as controlling the choice of process, and understand learning as a 'higher-order' process by which the choice of parameter is adjusted. More concretely, consider the synaptic strengths or weights of a neural network, which change as the system learns about the world, affecting the predictions it makes and actions it takes; or consider the process of Bayesian inference, where the posterior is dependent on a parameter that is typically called the 'prior'.

In this section, we introduce two related formal notions of parameterization: 'internal', where the parameter object constitutes a part of the domain of morphisms in a category; and 'external', where the parameters remain outside of the category being parameterized and the choice of morphism is implemented as a morphism in the base of enrichment. We will make use of both kinds of parameterization in this thesis.

Remark 3.2.1. Parameterization can be understood as introducing a new dimension into a category of processes. Consequently, the parameterization (either internal or external) of a category will produce a bicategory. When representing processes graphically, such as when using the string diagram calculus, this extra dimension becomes particularly explicit, and although we won't make use of graphical representations of parameterized processes in this thesis, they are typical in the applied-categorical literature, particularly in the literature on categorical cybernetics; for example, see Capucci et al. [54, Fig. 1], Cruttwell et al. [74, pp.1–2], and Capucci [51, Fig. 1].

3.2.1. Internal parameterization

Internal parameterization generalizes the case with which we opened this section, of morphisms $P \otimes X \to Y$, to a situation in which the parameterization may have different structure to the processes at hand, so that the parameterizing objects live in a different category. For this reason, we describe the 'actegorical' situation in which a category of parameters $\mathcal{M}$ *acts on* on a category of processes $\mathcal{C}$ to generate a category of parameterized processes. Nonetheless, even in this case, the

parameter ends up constituting part of the domain of the morphism representing the parameterized process.

The first concept we need is that of an 'actegory', which categorifies the better known mathematical notion of monoid action[2].

Definition 3.2.2 ($\mathcal{M}$-actegory). Suppose $\mathcal{M}$ is a monoidal category with tensor $\boxtimes$ and unit object I. We say that $\mathcal{C}$ is a *left $\mathcal{M}$-actegory* when there is a functor $\odot : \mathcal{M} \times \mathcal{C} \to \mathcal{C}$ called the *action* along with natural unitor and associator isomorphisms $\lambda_X^\odot : I \odot X \xrightarrow{\sim} X$ and $a_{M,N,X}^\odot : (M \boxtimes N) \odot X \xrightarrow{\sim} M \odot (N \odot X)$ compatible with the monoidal structure of $(\mathcal{M}, \boxtimes, I)$, in a sense analogous to the coherence data of a monoidal category (Definition 3.1.4). This means that the following triangle and pentagon diagrams must commute, where ρ and α are the right unitor and the associator of the monoidal structure on $\mathcal{M}$.

$$
\begin{array}{ccc}
(M \boxtimes I) \odot C & \xrightarrow{\quad a_{M,I,C}^\odot \quad} & M \odot (I \odot C) \\
& & \\
\rho_M \otimes \mathsf{id}_C \searrow & & \swarrow \mathsf{id}_M \odot \lambda_C^\odot \\
& M \odot C &
\end{array}
$$

$$
\begin{array}{ccc}
(K \boxtimes (M \boxtimes N)) \odot C & \xrightarrow{\quad a_{K,M\boxtimes N,C}^\odot \quad} & K \odot ((M \boxtimes N) \odot C) \\
\alpha_{K,M,N} \otimes \mathsf{id}_C \uparrow & & \downarrow \mathsf{id}_K \otimes a_{M,N,C}^\odot \\
((K \boxtimes M) \boxtimes N) \odot C & & K \odot (M \odot (N \odot C)) \\
a_{K\boxtimes M,N,C}^\odot \searrow & & \nearrow a_{K,M,N\odot C}^\odot \\
& (K \boxtimes M) \odot (N \odot C) &
\end{array}
$$

Given an actegory, we can define a category of correspondingly parameterized morphisms.

Proposition 3.2.3 (Capucci et al. [54]). Let $(\mathcal{C}, \odot, \lambda^\odot, a^\odot)$ be an $(\mathcal{M}, \boxtimes, I)$-actegory. Then there is a bicategory of $\mathcal{M}$-parameterized morphisms in $\mathcal{C}$, denoted $\mathbf{Para}(\odot)$. Its objects are those of $\mathcal{C}$. For each pair of objects X, Y, the set of 1-cells is defined as $\mathbf{Para}(\odot)(X, Y) := \sum_{M:\mathcal{M}} \mathcal{C}(M \odot X, Y)$; we denote an element (M, f) of this set by $f : X \xrightarrow{M} Y$. Given 1-cells $f : X \xrightarrow{M} Y$ and $g : Y \xrightarrow{N} Z$, their composite $g \circ f : X \xrightarrow{N\boxtimes M} Z$ is the following morphism in $\mathcal{C}$:

$$
(N \boxtimes M) \odot X \xrightarrow{a_{N,M,X}^\odot} N \odot (M \odot X) \xrightarrow{\mathsf{id}_N \odot f} N \odot Y \xrightarrow{g} Z
$$

---

[2]For a comprehensive reference on actegory theory, see Capucci and Gavranović [52].

Given 1-cells $f : X \xrightarrow{M} Y$ and $f' : X \xrightarrow{M'} Y$, a 2-cell $\alpha : f \Rightarrow f'$ is a morphism $\alpha : M \to M'$ in $\mathcal{M}$ such that $f = f' \circ (\alpha \odot \mathrm{id}_X)$ in $\mathcal{C}$; identities and composition of 2-cells are as in $\mathcal{C}$.

And when the action is 'strong' and the monoidal structure on $\mathcal{C}$ is symmetric, these parameterized categories inherit a monoidal structure.

Definition 3.2.4. Suppose $\mathcal{C}$ is a monoidal category and $F : \mathcal{C} \to \mathcal{C}$ is an endofunctor. A *right strength* for F is a natural transformation $\mathsf{str}^r_{X,Y} : FX \otimes Y \to F(X \otimes Y)$ making the following diagrams commute:

$$
\begin{array}{ccc}
FX \otimes (Y \otimes Z) & \xrightarrow{\quad \mathsf{str}^r_{X,Y\otimes Z} \quad} & F(X \otimes (Y \otimes Z)) \\
{\scriptstyle \alpha_{FX,Y,Z}} \big\uparrow & & \big\uparrow {\scriptstyle F(\alpha_{X,Y,Z})} \\
(FX \otimes Y) \otimes Z \xrightarrow{\mathsf{str}^r_{X,Y}\otimes\mathsf{id}_Z} F(X \otimes Y) \otimes Z & \xrightarrow{\mathsf{str}^r_{X\otimes Y,Z}} & F((X \otimes Y) \otimes Z)
\end{array}
$$

$$
\begin{array}{ccc}
FX \otimes I & \xrightarrow{\ \mathsf{str}^r_{X,I}\ } & F(X \otimes I) \\
& {\scriptstyle \rho_{FX}} \searrow \quad \swarrow {\scriptstyle F\rho_X} & \\
& FX &
\end{array}
$$

An action $\odot : \mathcal{M} \times \mathcal{C} \to \mathcal{C}$ induces a family of functors $M \odot (-) : \mathcal{C} \to \mathcal{C}$, natural in $M : \mathcal{M}$. If each of these is equipped with a right strength, also natural in $M : \mathcal{M}$, then we call the resulting transformation $\mathsf{str}^r_{M,X,Y}$ a right strength for $\odot$. Dually, there are notions of *left strength*, $\mathsf{str}^l_{X,Y} : X \otimes FY \to F(X \otimes Y)$ and *costrength*, with the latter obtained in the usual way as a strength in $\mathcal{C}^{\mathrm{op}}$ (reverse all the defining arrows).

Note that, if $\mathcal{C}$ is symmetric monoidal, then a left strength induces a right strength (by swapping) and likewise a right strength induces a left strength.

Proposition 3.2.5 (Capucci et al. [54, §2.1]). When $\mathcal{C}$ is equipped with both a symmetric monoidal structure $(\otimes, I)$ and an $(\mathcal{M}, \boxtimes, I)$-actegory structure $\odot$, and these are compatible in that the action $\odot$ has a strength isomorphism, the symmetric monoidal structure $(\otimes, I)$ lifts to $\mathbf{Para}(\odot)$.

The tensor of objects in $\mathbf{Para}(\odot)$ is then defined as the tensor of objects in $\mathcal{C}$, and the tensor of morphisms (1-cells) $f : X \xrightarrow{M} Y$ and $g : A \xrightarrow{N} B$ is given by the composite

$$f \otimes g : X \otimes A \xrightarrow{M \boxtimes N} Y \otimes B := (M \boxtimes N) \odot (X \otimes A) \xrightarrow{\iota_{M,N,X,A}} (M \odot A) \otimes (N \odot A) \xrightarrow{f \otimes g} Y \otimes B$$

where the *interchanger* $\iota_{M,N,X,A} : (M \boxtimes N) \odot (X \otimes A) \xrightarrow{\sim} (M \odot X) \otimes (N \odot A)$ is obtained using the associator of the actegory structure and the costrengths:

$$\iota_{M,N,X,A} := (M \boxtimes N) \odot (X \otimes A) \xrightarrow{a^{\odot}_{M,N,(X \otimes A)}} M \odot (N \odot (X \otimes A)) \cdots$$
$$\cdots \xrightarrow{M \odot \mathsf{costr}^l_{N,X,A}} M \odot (X \otimes (N \odot A)) \xrightarrow{\mathsf{costr}^r_{M,X,N \odot A}} (M \odot X) \otimes (N \odot A) .$$

(Note that the costrengths are obtained as the inverses of the strengths.)

We can see a monoidal product $\otimes : \mathcal{C} \times \mathcal{C} \to \mathcal{C}$ as an action of $\mathcal{C}$ on itself, and this induces the self-parameterization of $\mathcal{C}$.

Proposition 3.2.6 (Self-parameterization). If $(\mathcal{C}, \otimes, I)$ is a monoidal category, then it induces a parameterization $\mathbf{Para}(\otimes)$ on itself. For each $M, X, Y : \mathcal{C}$, the morphisms $X \xrightarrow{M} Y$ of $\mathbf{Para}(\otimes)$ are the morphisms $M \otimes X \to Y$ in $\mathcal{C}$.

Notation 3.2.7. When considering the self-paramterization induced by a monoidal category $(\mathcal{C}, \otimes, I)$, we will often write $\mathbf{Para}(\mathcal{C})$ instead of $\mathbf{Para}(\otimes)$.

It will frequently be the case that we do not in fact need the whole bicategory structure. The following proposition tells us that we can also just work 1-categorically, as long as we work with equivalence classes of isomorphically-parameterized maps, in order that composition is suffiently strictly associative.

Proposition 3.2.8. Each bicategory $\mathbf{Para}(\odot)$ induces a 1-category $\mathbf{Para}(\odot)_1$ by forgetting the bicategorical structure. The hom sets $\mathbf{Para}(\odot)_1(X, Y)$ are given by $U\mathbf{Para}(\odot)(X, Y)/\sim$ where U is the forgetful functor $U : \mathbf{Cat} \to \mathbf{Set}$ and $f \sim g$ if and only if there is some 2-cell $\alpha : f \Rightarrow g$ that is an isomorphism. We call $\mathbf{Para}(\odot)_1$ the *1-categorical truncation* of $\mathbf{Para}(\odot)$. When $\mathbf{Para}(\odot)$ is monoidal, so is $\mathbf{Para}(\odot)_1$.

Remark 3.2.9. We can understand the 1-categorical truncation of $\mathbf{Para}(\odot)$ as grouping the objects of each hom-category into their isomorphism-connected components.

3.2.2. External parameterization

In a monoidal closed category, morphisms $P \otimes X \to Y$ correspond bijectively to morphisms $P \to [X, Y]$. The fact that monoidal closed categories are enriched in themselves presents an opportunity for generalization in a different direction to the actegorical approach taken above:

that is, given a category of processes $\mathcal{C}$ enriched in $\mathcal{E}$, we can think of an externally parameterized process from X to Y as a morphism $P \to \mathcal{C}(X, Y)$ in $\mathcal{E}$.

This notion of external parameterization can be operationally more faithful to the structure of systems of interest, even though in the case of monoidal closed categories it is equivalent. For example, the improvement of the performance of a system of inference due to learning is often treated 'externally' to the inference process itself: the learning process might proceed by observing (but not interfering with) the inference process, and updating the parameters accordingly; and, if treated dynamically, the two processes might be assumed to exhibit a separation of timescales such that the parameters are stationary on the timescale of inference. We will make such assumptions when we formalize learning in Chapter 7, and so we will make use of external parameterization.

The definition of external parameterization is simplified by using the 'slice' construction.

Definition 3.2.10. Suppose X is an object of a category $\mathcal{E}$. We define the *slice* of $\mathcal{E}$ over X, denoted $\mathcal{E}/X$, as the category of 'bundles' over X in $\mathcal{E}$: its objects are morphisms $p : A \to X$ into X for any $A : \mathcal{E}$, which we call *bundles* over X and write as (A, p). The morphisms $f : (A, p) \to (B, q)$ in $\mathcal{E}/X$ are morphisms $f : A \to B$ in $\mathcal{E}$ such that $q \circ f = p$, as in the diagram

$$A \xrightarrow{\quad f \quad} B$$
$$p \searrow \qquad \swarrow q \qquad .$$
$$X$$

We therefore define external parameterization using slices over hom objects.

Definition 3.2.11. Given a category $\mathcal{C}$ enriched in $(\mathcal{E}, \times, 1)$, we define the *external parameterization* $\mathbf{P}\mathcal{C}$ of $\mathcal{C}$ in $\mathcal{E}$ as the following bicategory. 0-cells are the objects of $\mathcal{C}$, and each hom-category $\mathbf{P}\mathcal{C}(A, B)$ is given by the slice category $\mathcal{E}/\mathcal{C}(A, B)$. The composition of 1-cells is by composing in $\mathcal{C}$ after taking the product of parameters: given $f : \Theta \to \mathcal{C}(A, B)$ and $g : \Omega \to \mathcal{C}(B, C)$, their composite $g \circ f$ is

$$g \circ f := \Omega \times \Theta \xrightarrow{g \times f} \mathcal{C}(B, C) \times \mathcal{C}(A, B) \xrightarrow{\bullet} \mathcal{C}(A, C)$$

where $\bullet$ is the composition map for $\mathcal{C}$ in $\mathcal{E}$. The identity 1-cells are the points on the identity morphisms in $\mathcal{C}$. For instance, the identity 1-cell on A is the corresponding point $\mathrm{id}_A : 1 \to \mathcal{C}(A, A)$. We will denote 1-cells using our earlier notation for parameterized morphisms: for instance, $f : A \xrightarrow{\Theta} B$ and $\mathrm{id}_A : A \xrightarrow{1} A$. The horizontal composition of 2-cells is given by taking their product.

Remark 3.2.12. External parameterization is alternatively obtained as the change-of-enrichment induced by the *covariant self-indexing*, the functor $\mathcal{E}/(-) : \mathcal{E} \to \mathcal{E}\text{-}\mathbf{Cat}$, given on objects by $X \mapsto \mathcal{E}/X$ and on morphisms by the functor induced by post-composition[3]. A base of enrichment must *a fortiori* be a monoidal category, and in this case $\mathcal{E}/(-)$ is a lax monoidal functor. A lax monoidal functor out of the base of enrichment induces a corresponding change-of-enrichment pseudofunctor[4], and $\mathbf{P}$ is obtained precisely as the change-of-enrichment induced by $\mathcal{E}/(-)$.

One important consequence of this is that $\mathbf{P}$ defines a pseudofunctor $\mathbf{P} : \mathcal{E}\text{-}\mathbf{Cat} \to (\mathcal{E}\text{-}\mathbf{Cat})\text{-}\mathbf{Cat}$. Note that we take enrichment here to mean *weak* enrichment, in the sense indicated by Remark 2.2.13. In the case of locally small categories, where $\mathcal{E} = \mathbf{Set}$, this means that $\mathbf{P}$ has the type $\mathbf{Cat} \to \mathbf{Bicat}$, as suggested above. (We will discuss the definition of $\mathbf{Bicat}$ in §5.2.2, where we also define pseudofunctors between bicategories.)

Remark 3.2.13. In prior work, this external parameterization has been called 'proxying' [53]. We prefer the more explicit name 'external parameterization', reserving 'proxying' for a slightly different double-categorical construction to appear in future work by the present author.

Remark 3.2.14. Both internal and external parameterization are jointly generalized by the notion of *locally graded category* [172], which can be understood to mean "presheaf-enriched category". If $\mathcal{M}$ acts on $\mathcal{C}$ by $\odot$, then the hom category $\mathbf{Para}(\odot)(A, B)$ is the category of elements of the presheaf $\mathcal{C}(-\odot A, B) : \mathcal{M}^{\mathrm{op}} \to \mathbf{Set}$. Similarly, the hom category $\mathbf{P}\mathcal{C}(A, B)$ is the category of elements of the presheaf $\mathcal{E}\big(-, \mathcal{C}(A, B)\big) : \mathcal{E}^{\mathrm{op}} \to \mathbf{Set}$. We will see in §4.2.1 that the category of elements construction yields an equivalence between presheaves and categories-of-elements, and so we may as well consider $\mathbf{Para}(\odot)$ to be enriched in the presheaf category $[\mathcal{M}^{\mathrm{op}}, \mathbf{Set}]$ and $\mathbf{P}\mathcal{C}$ to be enriched in $[\mathcal{E}^{\mathrm{op}}, \mathbf{Set}]$. The phrase "locally graded" indicates that the 'hom sets' of $\mathbf{Para}(\odot)$ and $\mathbf{P}\mathcal{C}$ are 'graded' by the objects of $\mathcal{M}$ and $\mathcal{E}$ respectively. We learnt about locally graded categories from Dylan Braithwaite.

3.3. Systems from circuits

The dominant motivation for the use of monoidal categories so far has been in modelling the compositional structure of processes, on the basis of the observation that processes may generally

---

[3]Later, in Definition 4.2.28, we will encounter the *contravariant* self-indexing, which has the same action on objects but is given on morphisms by pullback. Whereas the covariant self-indexing is always well-defined, the contravariant self-indexing is therefore only well-defined in the more restricted situation where $\mathcal{E}$ has all pullbacks.

[4]A *pseudofunctor* is a kind of 'weakened' functor, for which functoriality only needs to hold up to isomorphism; see Definition 4.2.10.

be composed both sequentially and in parallel, and so 1-dimensional category theory alone is insufficient. The processes for which this kind of structure is most suited are those that exhibit a flow of information. For example, if we take the morphisms of the category **Set** as computable functions, then we see that the corresponding "process theory" is adequate for interpreting diagrams of the form of §2.1.1.3; and we will encounter in Chapter 4 a process-theoretic framework formalizing probabilistic graphical models of the kind discussed in §2.1.1.2.

In these monoidal categories, processes are represented by morphisms, with composition used to connect processes together: the composite of two processes is again a process. However, some morphisms are purely 'structural', implementing the plumbing of information flow—such as copying, discarding, and swapping—and so these categories somewhat blur the boundary between syntax and semantics. At the same time, it is strange to think of something like a neural circuit as a 'process': although it might reify some process in its behaviour, it is rather a *system*.

To sharpen the syntax-semantics boundary, one can show that every monoidal category arises as an algebra for a certain monad. We will make these notions precise in §3.4 below, and here it will suffice to provide some intuition: the monad defines the syntax, and the algebra supplies a compatible semantics. Algebra in this sense is a vast generalization of the abstract algebra of familiar mathematics, and typically involves defining symbolic operations and rules by which they can be combined, substituted, compared, and reduced.

In this section, although we do not explicitly make use of the technology of monads, we exemplify this approach with an example of compositional connectomics: on the syntactic side, we will introduce a 'multicategory' of *linear circuit diagrams* which govern patterns of neural connectivity; while on the semantic side, we will equip this multicategory with a functorial algebra of rate-coded neural circuits[5]. We will find that this more explicitly algebraic approach resolves the dilemma observed above between the compositional structure of processes and that of systems: algebraic syntax is in some sense about substitution, and so circuit diagrams will have 'holes' into which can be substituted other circuit diagrams. That is to say, a circuit diagram is a morphism which takes a given pattern of holes and connects them together into a single circuit, as in the following diagram,

---

[5]In the Appendix (§A.1), we sketch the connection between this multicategorical story and the monadic one.

which brings us back to our first motivating example from §2.1.1.1 and which we formalize below.
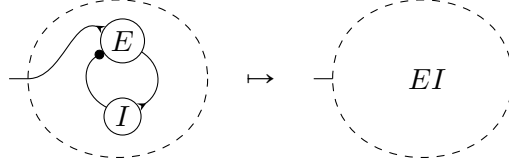
We will use a similar approach when we supply dynamical semantics for approximate inference, although there, for our general syntax of systems, we will use categories of polynomial functors, which we introduce in §3.5 at the end of this chapter. In any case, it will turn out that linear circuit diagrams embed naturally into polynomials, and so the circuits below can be understood as providing a sample of what is to come.

3.3.1. Multicategorical algebra for hierarchical systems

A multicategory is like a category, but where morphisms may have a 'complex' domain, such as a list of objects [168]. A morphism whose domain is an n-length list is called 'n-ary', and we can abstractly think of such morphisms as 'n-ary operations': for example, we will use them to model connecting n circuits together into a single system. Because these morphisms effect a kind of 'zooming-out', we can use them to construct hierarchical or 'nested' systems-of-systems.

Definition 3.3.1. A *multicategory* $\mathcal{O}$ consists of

1. a set $\mathcal{O}_0$ of objects;

2. a set $\mathcal{O}_1$ of morphisms, equipped with

   a) a codomain function $\mathrm{cod} : \mathcal{O}_1 \to \mathcal{O}_0$, and

   b) a domain function $\mathrm{dom} : \mathcal{O}_1 \to \mathsf{List}(\mathcal{O}_0)$, where $\mathsf{List}(\mathcal{O}_0)$ is the set of finite lists of objects $(o_1, \ldots, o_n)$,

   so that each n-*ary* morphism f has a list of n objects as its domain and a single object as its codomain, written $f : (o_1, \ldots, o_n) \to p$;

3. an *identity* function $\mathrm{id} : \mathcal{O}_0 \to \mathcal{O}_1$ such that $\mathrm{cod}(\mathrm{id}_o) = o$ and $\mathrm{dom}(\mathrm{id}_o) = (o)$, so that the identity on o is written $\mathrm{id}_o : o \to o$;

4. a family of *composition* functions

$$
\circ_{p,(o_i),(o_i^j)} : \mathcal{O}(o_1, \ldots, o_n; p) \times \mathcal{O}(o_1^1, \ldots, o_1^{k_1}; o_1) \times \cdots \times \mathcal{O}(o_n^1, \ldots, o_n^{k_n}; o_n)
$$
$$
\to \mathcal{O}(o_1^1, \ldots, o_1^{k_1}, \ldots, o_n^1, \ldots, o_n^{k_n}; p)
$$

written as

$$(f, f_1, \ldots, f_n) \mapsto f \circ (f_1, \ldots, f_n)$$

for each object p, n-ary list objects $(o_1, \ldots, o_n)$, and n k_i-ary lists of objects $(o_i^1, \ldots, o_i^{k_i})$;

satisfying the equations of *associativity*

$$f \circ \left(f_1 \circ (f_1^1, \ldots, f_1^{k_1}), \ldots, f_n \circ (f_n^1, \ldots, f_n^{k_n}) \right)$$
$$= \left(f \circ (f_1, \ldots, f_n) \right) \circ (f_1^1, \ldots, f_1^{k_1}, \ldots, f_n^1, \ldots, f_n^{k_n})$$

whenever such composites make sense, and *unitality*

$$f \circ (\mathsf{id}_{o_1}, \ldots, \mathsf{id}_{o_n}) = f = \mathsf{id}_p \circ f$$

for every $f : (o_1, \ldots, o_n) \to p$.

For our purposes, the order of objects in the lists will not matter, which we formalize with the notion of *symmetric* multicategory—analogous to the symmetric monoidal categories of §3.1.2.

Definition 3.3.2. Let S_n be the symmetric group on n elements. A *symmetric multicategory* $\mathcal{O}$ is a multicategory $\mathcal{O}$ which is additionally equipped, for each $n : \mathbb{N}$, with an action σ_n of S_n on the set $\mathcal{O}_1^n$ of n-ary morphisms

$$\sigma_n : S_n \times \mathcal{O}_1^n \to \mathcal{O}_1^n$$

such that composition $\circ$ preserves this action.

Remark 3.3.3. In other applied-category-theoretical contexts, multicategories of this kind are sometimes called *operads* (*cf. e.g.* [18, 97, 98, 171, 206, 222, 231, 237, 243, 244, 268, 282]). Traditionally, an operad is the same as a multicategory with one object[168]; sometimes therefore, multicategories are called *coloured* or *typed* operads[17, 59, 98, 168]. In order to avoid confusion, we will stick with 'multicategory'.

Although the multicategorical intuition—of hierarchically constructing complex systems—is valuable, the following fact means that there is a close connection between multicategories and monoidal categories, for in a monoidal category, we can interpret an n-ary tensor $x_1 \otimes \cdots \otimes x_n$ as an n-ary list of objects.

Proposition 3.3.4. Any monoidal category $(\mathcal{C}, \otimes, I)$ induces a corresponding multicategory $\mathcal{OC}$. The objects $\mathcal{OC}_0$ are the objects $\mathcal{C}_0$ of $\mathcal{C}$. The n-ary morphisms $(c_1, \ldots, c_n) \to d$ are the morphisms $c_1 \otimes \cdots \otimes c_n \to d$; *i.e.*, $\mathcal{OC}(c_1, \ldots, c_n; d) := \mathcal{C}(c_1 \otimes \cdots \otimes c_n, d)$. Identities are as in $\mathcal{C}$, and composition is defined by $(f, f_1, \ldots, f_n) \mapsto f \circ (f_1 \otimes \cdots \otimes f_n)$. When $\mathcal{C}$ is symmetric, so is $\mathcal{OC}$.

Example 3.3.5. An example that will soon become important is the operad $\mathbf{Sets}$ of sets and n-ary functions, which is obtained from the symmetric monoidal category $\mathbf{Set}$ by $\mathbf{Sets} := \mathcal{O}\,\mathbf{Set}$.

As we discussed above, we will consider multicategories as supplying a syntax for the composition of systems, and so actually to compose systems requires the extra data of what those systems are and how they can be composed according to the syntax. This extra semantic data is called an 'algebra' for the multicategory.

Definition 3.3.6. An *algebra for a multicategory* $\mathcal{M}$ is a *multifunctor* $\mathcal{M} \to \mathbf{Sets}$.

Multifunctors are the multicategorical analogues of functors; but fortunately (even though the definition is not a hard one), we will not need to define them, owing to the following result, which relates multifunctors and lax monoidal functors.

Proposition 3.3.7 (Leinster [168, Example 4.3.3, Definition 2.1.12])**.** If the multicategory $\mathcal{M}$ arises from a monoidal category $(\mathcal{C}, \otimes, I)$ as $\mathcal{M} = \mathcal{O}\mathcal{C}$, then an algebra for $\mathcal{M}$ is determined by a lax monoidal functor $(\mathcal{C}, \otimes, I) \to (\mathbf{Set}, \times, 1)$.

Remark 3.3.8. In §3.4, we will encounter the concept of "algebra for a monad", which is perhaps the more familiar concept in mathematics and computer science. One might therefore wonder what the relationship between the two notions of 'algebra' is: why do they both have this name? The answer is provided by Leinster [168]: every 'shape' of multicategory corresponds to a certain monad; and every multicategory algebra corresponds to an algebra for a monad derived (in the context of the particular multicategory at hand) from this multicategory-shape monad. For the interested reader, we review these results in the Appendix (§A.1). In §3.3.2, we will exemplify the notion of monad algebra with the more basic result that every small category corresponds to an algebra for a certain monad. Monad algebras will also prove useful later in the thesis in the context of compositional probability theory.

3.3.2. Linear circuit diagrams

Let us now exhibit the multicategory formalizing circuit diagrams of the type with which we opened this section. Although our motivation is multicategorical, for simplicity we will proceed by defining a symmetric monoidal category. Its objects will represent the 'output-input' dimensions of a circuit, written as pairs of numbers (n_o, n_i), and its morphisms $(n_o, n_i) \to (m_o, m_i)$ encode how to wire a circuit with n_o outputs and n_i inputs together to produce a circuit of m_o outputs and m_i inputs: this may involve connecting some of the n_o outputs to the m_o outputs; or connecting some

of the m_i inputs, or (to allow recurrence) the n_o outputs, to the n_i inputs. The definition may seem somewhat mysterious at first, but its form is owed to a more abstract structure (lenses) that we will define later, in §4.2.

Example 3.3.9. We define a symmetric monoidal category $\big(\mathbf{LinCirc}, +, (0,0)\big)$ of *linear circuit diagrams* and consider the induced multicategory $\mathcal{O}\mathbf{LinCirc}$. The objects of $\mathbf{LinCirc}$ are pairs (n_o, n_i) of natural numbers. A morphism $(n_o, n_i) \to (m_o, m_i)$ is a pair of real-valued matrices (A, B) with A of shape (m_o, n_o) and semi-orthogonal (*i.e.*, such that $AA^T = 1_{m_o}$) and B of shape $(n_i, n_o + m_i)$; equivalently, A is a semi-orthogonal linear map $\mathbb{R}^{n_o} \to \mathbb{R}^{m_o}$ and B is a linear map $\mathbb{R}^{n_o + m_i} \to \mathbb{R}^{n_i}$. The identity morphism $\mathrm{id}_{(n_o, n_i)}$ on (n_o, n_i) is the pair of matrices $(1_{n_o}, 01_{n_o})$ where 01_{n_o} is the block matrix $\begin{pmatrix} 0_{n_o} & 1_{n_o} \end{pmatrix}$. Given morphisms $(A, B) : (n_o, n_i) \to (m_o, m_i)$ and $(A', B') : (m_o, m_i) \to (k_o, k_i)$, their composite is the pair $(A'A, BB'_A)$ where $A'A$ is the usual matrix product and BB'_A is defined as the following block matrix multiplication:

$$BB'_A := B \begin{pmatrix} 1_{n_o} & 0 \\ 0 & B' \end{pmatrix} \begin{pmatrix} 1_{n_o} & 0 \\ A & 0 \\ 0 & 1_{k_i} \end{pmatrix}$$

Unitality and associativity of composition follow from those properties of matrix multiplication, and AA' is easily seen to be semi-orthogonal (by $AA'(AA')^T = AA'A'^T A^T = AA^T = 1_{m_o}$), so $\mathbf{LinCirc}$ is a well-defined category.

We now turn to the monoidal structure. The monoidal unit is the pair $(0,0)$; note that $\mathbb{R}^o \cong 1$. The monoidal product $+$ is defined on objects as the pointwise sum: $(n_o, n_i) + (m_o, m_i) := (n_o + m_o, n_i + m_i)$; note that $\mathbb{R}^{n_o + m_o} \cong \mathbb{R}^{n_o} \times \mathbb{R}^{m_o}$. Given morphisms $(A, B) : (n_o, n_i) \to (m_o, m_i)$ and $(A', B') : (n'_o, n'_i) \to (m'_o, m'_i)$, their monoidal product $(A, B) + (A', B')$ is defined as the pair $(A \oplus A', B \oplus B') : (n_o + n'_o, n_i + n'_i) \to (m_o + m'_o, m_i + m'_i)$ with

$$A \oplus A' := \begin{pmatrix} A & 0 \\ 0 & A' \end{pmatrix} \quad \text{and} \quad B \oplus B' := \begin{pmatrix} B & 0 \\ 0 & B' \end{pmatrix} \begin{pmatrix} 1_{n_o} & 0 & 0 & 0 \\ 0 & 0 & 1_{m_i} & 0 \\ 0 & 1_{n'_o} & 0 & 0 \\ 0 & 0 & 0 & 1_{m'_i} \end{pmatrix}.$$

For each pair of objects (n_o, n_i) and (m_o, m_i), the symmetry $\sigma_{(n_o, n_i),(m_o, m_i)} : (n_o, n_i) + (m_o, m_i) \to (m_o, m_i) + (n_o, n_i)$ is defined as the pair of matrices $(\sigma^o_{n,m}, \sigma^i_{n,m})$,

$$\sigma^o_{n,m} := \begin{pmatrix} 0 & 1_{m_o} \\ 1_{n_o} & 0 \end{pmatrix} \quad \text{and} \quad \sigma^i_{n,m} := \begin{pmatrix} 0 & 0 & 0 & 1_{n_i} \\ 0 & 0 & 1_{m_i} & 0 \end{pmatrix}.$$

That this definition produces a well-defined symmetric monoidal structure follows from more abstract considerations that we explain in Remark 4.2.33 and Corollary 4.2.36: $\mathbf{LinCirc}$ is a subcategory of Cartesian lenses, with the monoidal structure inherited accordingly.

The category of linear circuit diagrams is a syntactic category: on its own, it does not do anything. We need to equip it with semantics.

3.3.3. An algebra of rate-coded neural circuits

We begin by defining a notion of 'rate-coded' neural circuit.

Definition 3.3.10. An n_o-dimensional *rate-coded neural circuit* with n_i-dimensional input is an ordinary differential equation

$$\dot{x} = -\lambda \odot x + h\big(W(x \oplus i); \alpha, \beta, \gamma\big)$$

where $x, \lambda, \alpha, \beta, \gamma$ are real vectors of dimension n_o, i a real vector of dimension n_i, W a real matrix of shape $(n_o, n_o + n_i)$, $\odot$ elementwise multiplication, $\oplus$ the direct sum (so that $x \oplus i$ is the concatenation $\begin{pmatrix} x \\ i \end{pmatrix}$), and h the logistic function

$$h(x; \alpha, \beta, \gamma) = \frac{\gamma}{1 + \exp\big(-\beta(x - \alpha)\big)}$$

applied elementwise. We summarize the data of such a circuit as the tuple $(\lambda, \alpha, \beta, \gamma, W)$.

Remark 3.3.11. Rate-coded neural circuits are a coarse phenomenological model of neural dynamics. The state variable x represents the *firing rates* of an ensemble of neurons, either averaged over time or over subpopulations. Neural activity is of course not so simple: neurons communicate by the transmission of discrete 'action potentials' along their axons. The emission of an action potential is governed by the electrical potential of its cellular membrane: if this potential crosses a threshold, then the neuron 'fires' an action potential down its axon. The axon crosses the dendrites of other neurons at junctions called synapses, which modulate and transmit the activity accordingly: it is these afferent signals which in large part determine the neurons' membrane potentials.

There are of course detailed physiological models of this process (*cf. e.g.* [79, 133, 181, 226]), as well as many models which aim to capture its statistics and phenomenology in a more explicitly computational setting (*cf. e.g.* [81, 82, 117, 136, 146, 202, 227, 273]), but in some situations, one can simply model neural firing as an inhomogeneous Poisson process: in this case the variable x encodes the rate parameters of the processes. We expect there to be functorial connections between the different classes of models: in particular, we expect adjoint functors between certain spike-emission models and firing rate models of the class defined above; and in the specific case of

'efficient balanced' networks[32, 82], the relationships are expected to be quite simple. Nonetheless, we leave the exploration of such functors to future work.

The parameters of a rate-coded neural circuit—the terms $\lambda, \alpha, \beta, \gamma, W$— have a neurological interpretation, even though this dynamical model is not physiologically faithful. The term λ represents the 'leak' of voltage from the neuron's membrane, which has the effect of determining the timescale of its memory or signal-sensitivity (effectively, the voltage leak entails a process of filtering). The term α represents an abstraction of the neuron's firing threshold, and the term β its sensitivity (*i.e.*, how much its firing rate increases with incoming signals); the term γ determines the maximum firing rate of the neuron (and is typically normalized to 1). Finally, the matrix W records the strengths of the synaptic connections within the circuit: positive coefficients represent excitatory connections, while negative coefficients represent inhibitory connections.

Rate-coded neural circuits can be organized into complex 'hierarchical' systems using linear circuit diagrams: the linear connectivity of the diagrams is used to define the synaptic connection matrix of the complex, algebraically. The proof that the following construction does actually constitute an algebra ensures that composing systems from circuits using diagrams is predictably well-behaved, as we will subsequently exemplify.

Proposition 3.3.12 (Algebra of rate-coded neural circuits). There is a **LinCirc**-algebra (R, μ, ϵ) : $\big(\mathbf{LinCirc}, +, (0,0)\big) \to (\mathbf{Set}, \times, 1)$ of rate-coded neural circuits. On objects (n_o, n_i), define $R(n_o, n_i)$ to be the set of n_o-dimensional rate-coded neural circuits with n_i-dimensional input. Then, given a linear circuit diagram $(A, B) : (n_o, n_i) \to (m_o, m_i)$, define a function

$$R(A, B) : R(n_o, n_i) \to R(m_o, m_i)$$

$$(\lambda, \alpha, \beta, \gamma, W) \mapsto (A\lambda, A\alpha, A\beta, A\gamma, W_{AB})$$

where W_{AB} is the following block matrix product:

$$W_{AB} := AW \begin{pmatrix} 1_{n_o} & 0 \\ 0 & B \end{pmatrix} \begin{pmatrix} 1_{n_o} & 0 \\ 1_{n_o} & 0 \\ 0 & 1_{m_i} \end{pmatrix} \begin{pmatrix} A^T & 0 \\ 0 & 1_{m_i} \end{pmatrix} \ .$$

The laxator μ is defined componentwise as the family of functions

$$\mu_{(n_o, n_i),(m_o, m_i)} : R(n_o, n_i) \times R(m_o, m_i) \to R\big((n_o, n_i) + (m_o, m_i)\big)$$

taking a pair of circuits $(\lambda, \alpha, \beta, \gamma, W) : R(n_o, n_i)$ and $(\lambda', \alpha', \beta', \gamma', W') : R(m_o, m_i)$ to the circuit $(\lambda \oplus \lambda', \alpha \oplus \alpha', \beta \oplus \beta', \gamma \oplus \gamma', WW')$ where $x \oplus y$ is again the direct sum $\begin{pmatrix} x \\ y \end{pmatrix}$ and where

the matrix WW' is defined as

$$WW' := \begin{pmatrix} W & 0 \\ 0 & W' \end{pmatrix} \begin{pmatrix} 1_{n_o} & 0 & 0 & 0 \\ 0 & 0 & 1_{n_i} & 0 \\ 0 & 1_{m_o} & 0 & 0 \\ 0 & 0 & 0 & 1_{m_i} \end{pmatrix}.$$

The unitor ϵ is the isomorphism $\epsilon : 1 \xrightarrow{\sim} R(0,0)$.

Proof. We need to check that R is a lax monoidal functor, and begin by verifying functoriality. So suppose (A', B') is a linear circuit diagram $(m_o, m_i) \to (k_o, k_i)$. On the terms $\lambda, \alpha, \beta, \gamma$, the functoriality of R is immediate from matrix multiplication, so we concentrate on the action of R on W. We need to show that $R\big((A', B') \circ (A, B)\big)(W) = R(A', B') \circ R(A, B)(W)$, where $R(A, B)(W) = W_{AB}$ as defined above. Note that we can alternatively write W_{AB} as the following composite linear map

$$m_o + m_i \xrightarrow{A^T + m_i} n_o + m_i \xrightarrow{\curlyvee + m_i} n_o + n_o + m_i \xrightarrow{n_o + B} n_o + n_i \xrightarrow{W} n_o \xrightarrow{A} m_o .$$

We can therefore write $R(A', B')(W_{AB})$ as

$$k_o + k_i \xrightarrow{A'^T + k_i} m_o + k_i \xrightarrow{A^T + k_i} n_o + k_i \xrightarrow{\curlyvee + k_i} n_o + n_o + k_i \cdots$$

$$\cdots \xrightarrow{n_o + \curlyvee + k_i} n_o + n_o + n_o + k_i \xrightarrow{n_o + n_o + A + k_i} n_o + n_o + m_o + k_i \cdots$$

$$\cdots \xrightarrow{n_o + n_o + B'} n_o + n_o + m_i \xrightarrow{n_o + B} n_o + n_i \xrightarrow{W} n_o \xrightarrow{A} m_o \xrightarrow{A'} k_o$$

and $R\big((A', B') \circ (A, B)\big)(W)$ as

$$k_o + k_i \xrightarrow{A'^T + k_i} m_o + k_i \xrightarrow{\curlyvee + k_i} m_o + m_o + k_i \xrightarrow{m_o + B'} m_o + m_i \xrightarrow{A^T + m_i} \cdots$$

$$\cdots n_o + m_i \xrightarrow{\curlyvee + m_i} n_o + n_o + m_i \xrightarrow{n_o + B} n_o + n_i \xrightarrow{W} n_o \xrightarrow{A} m_o \xrightarrow{A'} k_o$$

so it suffices to check that

$$m_o + k_i \xrightarrow{A^T + k_i} n_o + k_i \xrightarrow{\curlyvee + k_i} n_o + n_o + k_i \xrightarrow{n_o + \curlyvee + k_i} n_o + n_o + n_o + k_i \cdots$$

$$\cdots \xrightarrow{n_o + n_o + A + k_i} n_o + n_o + m_o + k_i \xrightarrow{n_o + n_o + B'} n_o + n_o + m_i$$

$$=$$

$$m_o + k_i \xrightarrow{\curlyvee + k_i} m_o + m_o + k_i \xrightarrow{m_o + B'} m_o + m_i \xrightarrow{A^T + m_i} n_o + m_i \xrightarrow{\curlyvee + m_i} n_o + n_o + m_i$$

which we can do using the graphical calculus:
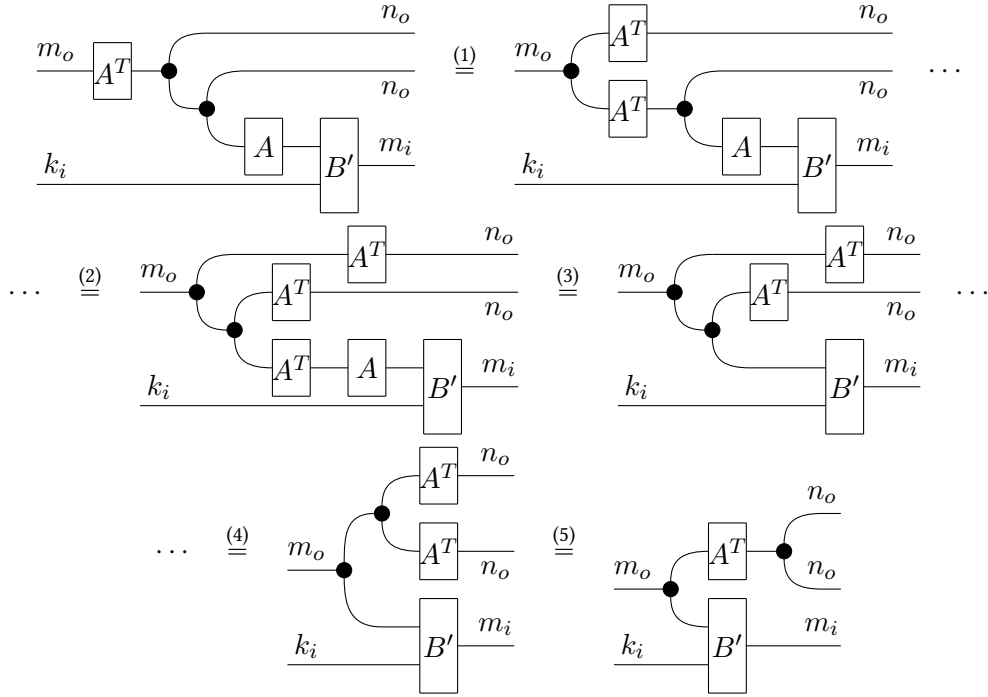
where the equality (1) holds because A^T is a comonoid morphism (Definition 3.4.23)[6], (2) likewise, (3) because A is semi-orthogonal, (4) by the coassociativity of copying, and (5) again because A^T is a comonoid morphism. Finally, we observe that the last string diagram depicts the linear map

$$m_o + k_i \xrightarrow{\;\curlyvee + k_i\;} m_o + m_o + k_i \xrightarrow{\;A^T + B'\;} n_o + m_i \xrightarrow{\;\curlyvee + m_i\;} n_o + n_o + m_i$$

which equals the required map

$$m_o + k_i \xrightarrow{\;\curlyvee + k_i\;} m_o + m_o + k_i \xrightarrow{\;m_o + B'\;} m_o + m_i \xrightarrow{\;A^T + m_i\;} n_o + m_i \xrightarrow{\;\curlyvee + m_i\;} n_o + n_o + m_i$$

by the unitality of composition. This establishes that R preserves composites; it remains to check that it preserves identities. Once again, this follows immediately on the terms $\lambda, \alpha, \beta, \gamma$, so we concentrate on the action on W. We have

$$R(1_{n_o}, 01_{n_o})(W) = 1_{n_o} W \begin{pmatrix} 1_{n_o} & 0 & 0 \\ 0 & 0 & 1_{n_o} \end{pmatrix} \begin{pmatrix} 1_{n_o} & 0 \\ 1_{n_o} & 0 \\ 0 & 1_{m_i} \end{pmatrix} \begin{pmatrix} 1_{n_o} & 0 \\ 0 & 1_{m_i} \end{pmatrix}$$

which is easily seen to be equal to W itself. Therefore R defines a functor.

We now need to verify that the unitor and laxator satisfy the unitality and associativity axioms of a lax monoidal functor. We begin by checking associativity, so suppose that we are given three circuits:

---

[6]This in turn because $\oplus$ is the Cartesian product, and so every morphism is a $\oplus$-comonoid morphism.

$(\lambda, \alpha, \beta, \gamma, W)$: $R(n_o, n_i)$, and $(\lambda', \alpha', \beta', \gamma', W')$: $R(m_o, m_i)$, and $(\lambda'', \alpha'', \beta'', \gamma'', W'')$: $R(k_o, k_i)$. Associativity on all the terms but W, W', W'' follows from the associativity of the direct sum $\oplus$, and so we just need to check that $\mu(W, \mu(W', W'')) = \mu(\mu(W, W'), W'')$ where $\mu(W, W') = WW'$ and $\mu(W', W'') = W'W''$, according to the definition above. Once more, we use the graphical calculus. Observe that we can depict WW' and $W'W''$ as
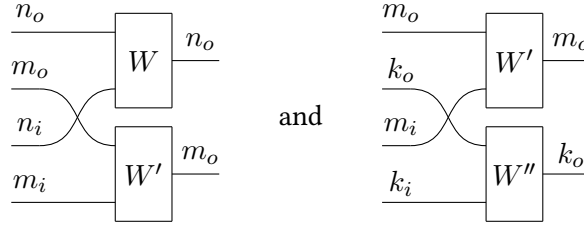
respectively. Hence $\mu(W, \mu(W', W''))$ satisfies the equality

$$\mu(W, \mu(W', W'')) \quad = \quad \cdots \quad = \quad \cdots$$
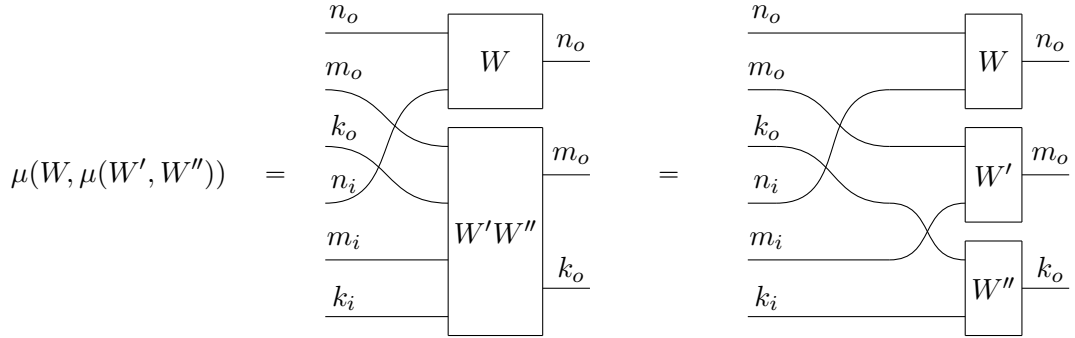
and likewise $\mu(\mu(W, W'), W'')$ satisfies

$$\mu(\mu(W, W'), W'') \quad = \quad \cdots \quad = \quad \cdots .$$
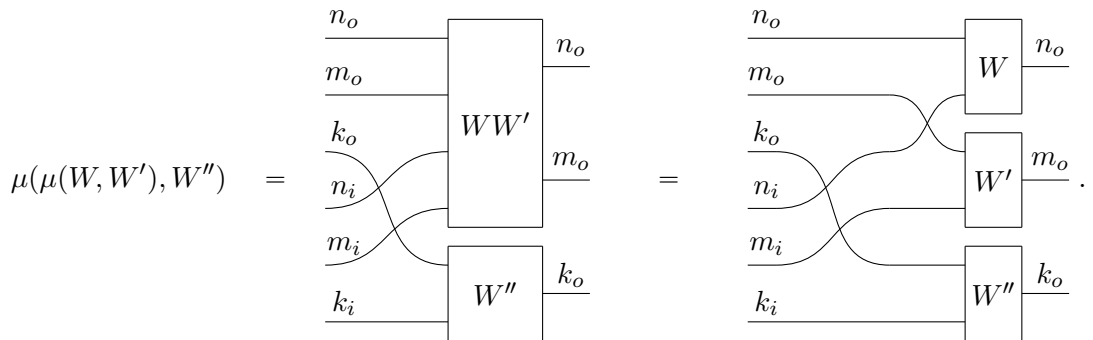
The two diagrams on the right hand side are equal up to a topological deformation, and so the depicted morphisms are equal by the coherence theorem for monoidal categories. This establishes the associativity of the laxator. It remains to establish unitality: but this follows immediately, because $R(0, 0) \cong \mathbb{R}^0$ and the 0-dimensional space is the unit for the direct sum $\oplus$. Hence (R, μ, ϵ) is a lax monoidal functor, and hence an algebra for $(\mathbf{LinCirc}, +, (0, 0))$. $\qquad \square$

Remark 3.3.13. At points in the preceding proof, we used the fact that a linear map is a *comonoid morphism*, which implies that it commutes with copying. We will define the notion of comonoid

morphism in §3.4.1 below; meanwhile, the fact that A^T is one follows from the fact that $\oplus$ is the categorical product of vector spaces, and so every linear map is a $\oplus$-comonoid morphism.

Remark 3.3.14. Let us return briefly to the distinction made at the beginning of this section between processes and systems, and their corresponding categorical incarnations. One might be tempted to try constructing a symmetric monoidal category of neural circuits using this algebra whose objects would be natural numbers and whose morphisms $i \to o$ would be circuits in $R(o, i)$, treated thus as 'processes'. But this won't work, because there is no neural circuit that will function as an identity morphism! Later in the thesis (§6.3), we will see one way around this problem, building monoidal categories of hierarchical dynamical systems that are in some sense analogous to these circuits (while being more general): there, we will use the rich structure of polynomial functors to define both the syntax of composition as well as the hom categories (for our construction will be bicategorical) of dynamical systems, and the extra generality will mean we will have identity systems (that compose appropriately unitally). Until then, we note that the moral of this observation might be that it affirms that the composition of neural circuits is multicategory-algebraic (formalizing a notion of hierarchy), rather than merely categorical.

The weight matrices resulting from the linear circuit algebra encode the pattern of connectivity specified by the diagram, as we now exemplify.

Example 3.3.15. Let us consider the circuit example from the beginning of this section, the wiring of an inhibitory circuit to an excitatory circuit, as in the diagram
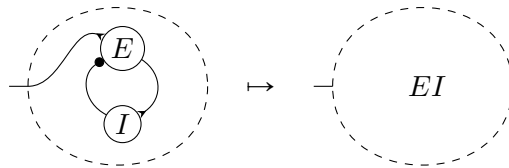
which depicts a linear circuit diagram $E + I \to EI$. In such a diagram, the input dimension of an object (such as E) must have dimension equalling the sum of the dimensions of the incoming wires. Dually, the dimension along a wire emanating from an object must have dimension equal to the output dimension of that object. To distinguish the source and target of a wire, we decorate the target ends: a filled circle denotes an inhibitory connection, interpreted in the linear circuit as the negative identity matrix -1 of the appropriate dimension; and an inverted arrowhead denotes an excitatory connection, interpreted as the positive identity 1 of the appropriate dimension. We will write the dimensions of the object E as (o_E, i_E), of I as (o_I, i_I), and of EI as (o_{EI}, i_{EI}). Therefore,

in this example, the following equalities must hold: $i_E = o_I + i_{EI}$; $i_I = o_E$; and $o_{EI} = o_E + o_I$. The last equation holds because the circuit EI is formed from the sum of the circuits E and I.

To give a circuit diagram $(A, B) : (o_E, i_E) + (o_I, i_I) \rightarrow (o_{EI}, i_{EI})$ is to give a semi-orthogonal real matrix A of shape $(o_{EI}, o_E + o_I)$ and a real matrix B of shape $(i_E + i_I, o_E + o_I + i_{EI})$. Using the preceding equalities, these are equivalently shaped as $(o_E + o_I, o_E + o_I)$ and $(o_I + i_{EI} + o_E, o_E + o_I + i_{EI})$, and we just choose the identity matrix $1_{o_E+o_I}$ for A. To define B, we read it off from the diagram as

$$B := \begin{pmatrix} 0 & -1_{o_I} & 0 \\ 0 & 0 & 1_{i_{EI}} \\ 1_{o_E} & 0 & 0 \end{pmatrix}.$$

Now suppose $(\lambda_E, \alpha_{,E}, \beta_E, \gamma_E, W_E)$ and $(\lambda_I, \alpha_{,I}, \beta_I, \gamma_I, W_I)$ are two rate-coded neural circuits, the former of type $R(o_E, i_E)$ and the latter of type $R(o_I, i_I)$. How does $R(A, B)$ act upon them to give our composite circuit?

On all the parameters but the weight matrices, $R(A, B)$ acts trivially (since A is just the identity matrix), and so we will concentrate on the action on W_E, W_I. Firstly, we need to form the monoidal product of the weight matrices, $\mu(W_E, W_I)$, which is defined by

$$\mu(W_E, W_I) = \begin{pmatrix} W_E & 0 \\ 0 & W_I \end{pmatrix} \begin{pmatrix} 1_{o_E} & 0 & 0 & 0 \\ 0 & 0 & 1_{i_E} & 0 \\ 0 & 1_{o_I} & 0 & 0 \\ 0 & 0 & 0 & 1_{i_I} \end{pmatrix}$$

$$= \begin{pmatrix} W_E & 0 \\ 0 & W_I \end{pmatrix} \begin{pmatrix} 1_{o_E} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1_{o_I} & 0 & 0 \\ 0 & 0 & 0 & 1_{i_{EI}} & 0 \\ 0 & 1_{o_I} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1_{o_E} \end{pmatrix}$$

where the second equality holds by applying the equalities between the dimensions defined above. The weight matrix $R(A, B)(\mu(W_E, W_I))$ is then defined as

$$A\mu(W_E, W_I) \begin{pmatrix} 1_{o_E+o_I} & 0 \\ 0 & B \end{pmatrix} \begin{pmatrix} 1_{o_E+o_I} & 0 \\ 1_{o_E+o_I} & 0 \\ 0 & 1_{i_{EI}} \end{pmatrix} \begin{pmatrix} A^T & 0 \\ 0 & 1_{i_{EI}} \end{pmatrix}.$$

Since $A = 1_{o_E+o_I}$, and by substituting the definition of $\mu(W_E, W_I)$, this reduces to

$$\begin{pmatrix} W_E & 0 \\ 0 & W_I \end{pmatrix} \begin{pmatrix} 1_{o_E} & 0 & 0 & 0 & 0 \\ 0 & 0 & 1_{o_I} & 0 & 0 \\ 0 & 0 & 0 & 1_{i_{EI}} & 0 \\ 0 & 1_{o_I} & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1_{o_E} \end{pmatrix} \begin{pmatrix} 1_{o_E+o_I} & 0 \\ 0 & B \end{pmatrix} \begin{pmatrix} 1_{o_E+o_I} & 0 \\ 1_{o_E+o_I} & 0 \\ 0 & 1_{i_{EI}} \end{pmatrix}.$$

Then, by substitution and matrix multiplication, we have the following equalities:

$$
\begin{pmatrix}
1_{o_E} & 0 & 0 & 0 & 0 \\
0 & 0 & 1_{o_I} & 0 & 0 \\
0 & 0 & 0 & 1_{i_{EI}} & 0 \\
0 & 1_{o_I} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1_{o_E}
\end{pmatrix}
\begin{pmatrix}
1_{o_E+o_I} & 0 \\
0 & B
\end{pmatrix}
\begin{pmatrix}
1_{o_E+o_I} & 0 \\
1_{o_E+o_I} & 0 \\
0 & 1_{i_{EI}}
\end{pmatrix}
$$

$$
=
\begin{pmatrix}
1_{o_E} & 0 & 0 & 0 & 0 \\
0 & 0 & 1_{o_I} & 0 & 0 \\
0 & 0 & 0 & 1_{i_{EI}} & 0 \\
0 & 1_{o_I} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1_{o_E}
\end{pmatrix}
\begin{pmatrix}
1_{o_E} & 0 & 0 & 0 & 0 \\
0 & 1_{o_I} & 0 & 0 & 0 \\
0 & 0 & 0 & -1_{o_I} & 0 \\
0 & 0 & 0 & 0 & 1_{i_{EI}} \\
0 & 0 & 1_{o_E} & 0 & 0
\end{pmatrix}
\begin{pmatrix}
1_{o_E} & 0 & 0 \\
0 & 1_{o_I} & 0 \\
1_{o_E} & 0 & 0 \\
0 & 1_{o_I} & 0 \\
0 & 0 & 1_{i_{EI}}
\end{pmatrix}
$$

$$
=
\begin{pmatrix}
1_{o_E} & 0 & 0 & 0 & 0 \\
0 & 0 & 1_{o_I} & 0 & 0 \\
0 & 0 & 0 & 1_{i_{EI}} & 0 \\
0 & 1_{o_I} & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 1_{o_E}
\end{pmatrix}
\begin{pmatrix}
1_{o_I} & 0 & 0 \\
0 & 1_{o_I} & 0 \\
0 & -1_{o_I} & 0 \\
0 & 0 & 1_{i_{EI}} \\
1_{o_E} & 0 & 0
\end{pmatrix}
$$

$$
=
\begin{pmatrix}
1_{o_E} & 0 & 0 \\
0 & -1_{o_I} & 0 \\
0 & 0 & 1_{i_{EI}} \\
0 & 1_{o_I} & 0 \\
1_{o_E} & 0 & 0
\end{pmatrix}
$$

so that the resulting weight matrix $R(A, B)(\mu(W_E, W_I))$ is

$$
\begin{pmatrix}
W_E & 0 \\
0 & W_I
\end{pmatrix}
\begin{pmatrix}
1_{o_E} & 0 & 0 \\
0 & -1_{o_I} & 0 \\
0 & 0 & 1_{i_{EI}} \\
0 & 1_{o_I} & 0 \\
1_{o_E} & 0 & 0
\end{pmatrix}.
$$

Reading off this weight matrix, we see that the E neurons receive external input plus recurrent excitatory input from themselves as well as inhibitory input from I, and that the I neurons receive only recurrent excitatory input plus excitatory input from E. This is exactly as it should be, given the diagram: by formalizing these computations, we render them mechanical (and hence computer-implementable). In particular, we can treat the resulting EI circuit as a "black box" and substitute it into other diagrams to construct still larger-scale systems.

Since linear circuit diagrams allow for any linear pattern of connectivity, we can of course generalize the picture above to allow for more subtle interconnections.

Example 3.3.16. Suppose that instead of incorporating only excitatory or inhibitory connections,

we sought something a little more complex, as in the following circuit diagram:
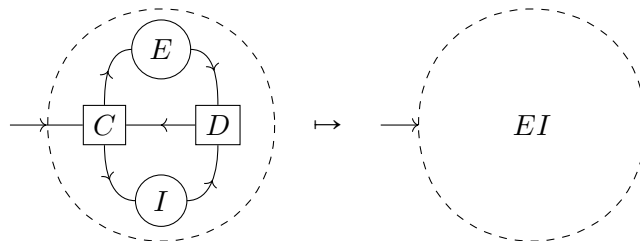
Now, we have decorated the wires with fleches, to indicate the flow of activity; and besides the circular boxes (representing circuits), we have incorporated square boxes (representing linear patterns of interconnection). Using the same notation for the dimensions of the circuits E,I and EI as in Example 3.3.15, this means that the boxes square boxes represent matrices C of shape $(i_E + i_I, n + i_{EI})$ and D of shape $(n, o_E + o_I)$, where n is the dimension of the D-C wire. If we again write (A, B) for the implied circuit diagram, and we can again set A to be the identity matrix, and read B from the diagram as the composite matrix

$$B := o_E + o_I + i_{EI} \xrightarrow{D \oplus 1_{i_{EI}}} n + i_{EI} \xrightarrow{C} i_E + i_I \ .$$

The rest of the calculation follows mechanically, just as before.

One feature missing from the construction in this section is synaptic plasticity: although we have shown how to compose circuits into systems, it is only the neural firing rates that are dynamical; the connection matrices remain fixed. In the preceding section, we motivated the introduction of parameterized categories by their application to learning problems, and indeed one could factorize the linear circuit algebra above by extracting the connection matrices into parameters; if one wanted to retain a choice of initial weight matrix, this could also be incorporated into a 'pointed' version of the structure.

This parameterized construction would be bicategorical, and so a faithful semantics for it would no longer land in **Set**, but rather in **Cat**: we would have categories of circuits related by reparameterizations of the weight matrices, and with the dynamics also incorporating plasticity[7].

With a sufficiently sophisticated algebra, it would even be possible to allow the circuit diagrams themselves to be dynamical and subject to learning. We will not pursue this line of enquiry further here, but we will return to it when we introduce plasticity into approximate inference doctrines: there, our structures will be sufficiently supple to incorporate all of the concepts sketched here.

---

[7]An even more faithful dynamical semantics would land in "bundle dynamical systems", of the form that we introduce in Chapter 6: two two levels of the bundle would witness the dynamics of the firing activity and the plasticity, and the bundles themselves would witness the timescale separation.

3.4. From monoids to monads

In order to reach the level of suppleness required by plastic dynamical approximate inference, it will help to understand the structures underlying the definitions and constructions introduced so far in this chapter—in particular, we will need a firm grasp of the concepts of monad and comonoid—and so at this point we return to an expository mode.

The fundamental concept underlying many of the structures we have seen so far is the *monoid*: an object equipped with two operations, one binary and one 'nullary', with the latter acting as a 'unit' for the former, and although the major operation is only binary, it can be chained in order to form n-ary operations. For this reason, monoids are fundamental to abstract algebra: categories themselves are "monoids with many objects" (in the same way that a multicategory is an operad with many objects). Both monads and comonoids can be defined using monoids.

Even though monoids are fundamental and intimately familiar to mathematicians and computer scientists, they remain underappreciated in computational and cognitive neuroscience. For this reason, we once again take a fairly pedagogical approach in this section.

Definition 3.4.1. Suppose $(\mathcal{C}, \otimes, I)$ is a monoidal category. A *monoid object* in $\mathcal{C}$ is an object m equipped with a *multiplication* morphism $\mu : m \otimes m \to m$ and a *unit* morphism $\eta : I \to m$, satisfying the axioms of (left and right) unitality
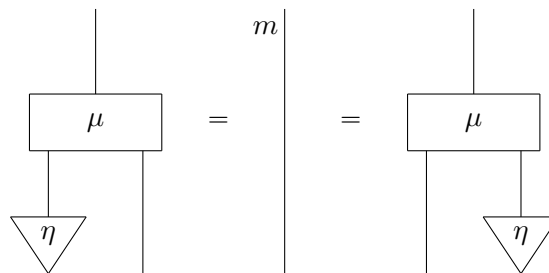
and associativity
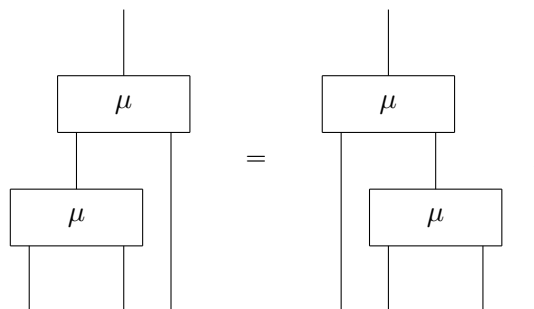
If $\mathcal{C}$ is symmetric monoidal then we say that the monoid (m, μ, η) is *commutative* if μ commutes

with the symmetry as in
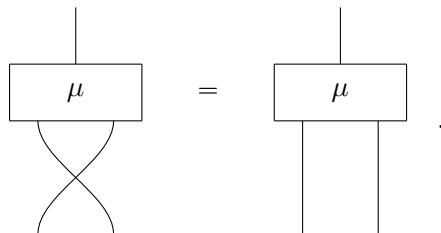
Since we are doing category theory, it is important to understand morphisms of monoids.

Definition 3.4.2. Suppose (m, μ, η) and (m', μ', η') are monoids in $(\mathcal{C}, \otimes, I)$. A *monoid morphism* $(m, \mu, \eta) \to (m', \mu', \eta')$ is a morphism $f : m \to m'$ in $\mathcal{C}$ that is compatible with the monoidal structures, *i.e.* by satisfying the axioms
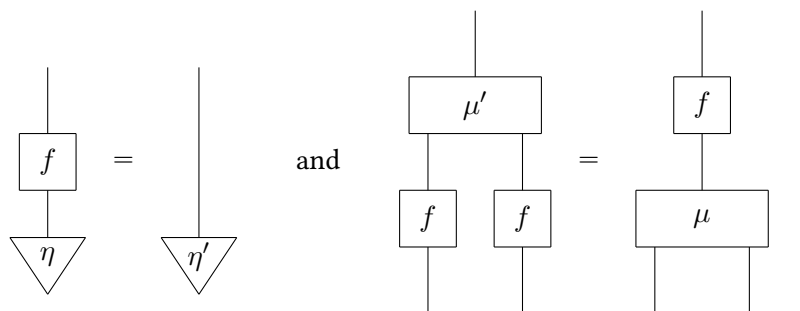
Monoids and their morphisms in $\mathcal{C}$ constitute a category $\mathbf{Mon}(\mathcal{C})$; composition and identities are as in $\mathcal{C}$, and it is easy to check that the composite of two monoid morphisms is again a monoid morphism.

If $\mathcal{C}$ is symmetric monoidal, then there is a subcategory $\mathbf{CMon}(\mathcal{C}) \hookrightarrow \mathbf{Mon}(\mathcal{C})$ of commutative monoids and their morphisms.

In the names $\mathbf{Mon}(\mathcal{C})$ and $\mathbf{CMon}(\mathcal{C})$, we leave the monoidal structure implicit; should it be necessary to be explicit, we write $\mathbf{Mon}_\otimes(\mathcal{C})$ and $\mathbf{CMon}_\otimes(\mathcal{C})$.

Let us consider some first examples of monoids in monoidal categories.

Example 3.4.3. The natural numbers $\mathbb{N}$ equipped with addition $+ : \mathbb{N} \times \mathbb{N} \to \mathbb{N}$ and zero 0 constitute a monoid in **Set**. (In fact, $(\mathbb{N}, +, 0)$ is the *free monoid* generated by a single element.)

Example 3.4.4. If A is a set, then there is a monoid $\big(\mathsf{List}(A), \circ, ()\big)$ of *lists of elements* of A: the elements of the set $\mathsf{List}(A)$ are finite lists $(a, b, \dots)$ of elements of A; the multiplication $\circ : \mathsf{List}(A) \times \mathsf{List}(A) \to \mathsf{List}(A)$ is given by concatenation of lists $(b_1, \dots) \circ (a_1, \dots) = (a_1, \dots, b_1, \dots)$; and the unit $1 \to \mathsf{List}(A)$ is given by then empty list $()$. We saw in the proof of Proposition 2.1.9 that list concatenation is associative and unital.

Example 3.4.5. A monoid $(m, \circ, *)$ in **Set** is a category with a single object, denoted $*$. We already saw an example of this, in Example 2.1.7: the monoid $(\mathbb{N}, +, 0)$, treated as a category. More generally, a monoid in a monoidal category $(\mathcal{C}, \otimes, I)$ is a $\mathcal{C}$-enriched category with a single object.

Example 3.4.6. A monoid $(\mathcal{C}, \otimes, I)$ in the monoidal category $(\mathbf{Cat}, \times, \mathbf{1})$ of categories and functors is a strict monoidal category: the tensor is the monoid multiplication, and its unit is the monoid unit. In fact, this explains the name "monoidal category": a (strict) monoidal category is a monoid object in **Cat**.

Remark 3.4.7. Non-strict monoidal categories are 'weak' in the same sense that bicategories are weak 2-categories; after all, a monoidal category is a one-object bicategory. In this way, we can also weaken the notion of monoid object in a bicategory, so that the axioms of unitality and associativity only hold up to 'coherent isomorphism': that is, up to isomorphisms that cohere with the weak unitality and associativity of the ambient bicategory. Such weak monoid objects are called *pseudomonoids*[8], and when interpreted in the monoidal 2-category $(\mathbf{Cat}, \times, \mathbf{1})$ their formal definition[77, §3] yields exactly the non-strict monoidal categories.

But note that to make sense in general of the notion of pseudomonoid, we first need to have a notion of monoidal bicategory. Abstractly, such a thing should be a one-object tricategory, but this often doesn't help: in those cases, we need something more concrete. Informally, then, a monoidal bicategory is a bicategory equipped with a monoidal structure that is coherent with the 2-cells, but as we have begun to see here, to specify all this coherence data quickly becomes quite verbose, and to prove their satisfaction by any given structure quite arduous, so we will only make use informally in this thesis of the notions of monoidal bicategory and pseudomonoid — and when we do, it will be by reference to the familiar structures on and in **Cat**: its Cartesian monoidal structure; and (non-strict) monoidal categories.

Finally, we note that the general phenomenon, of which we observe an instance here, wherein algebraic structures (such as monoids) may be defined internally to categories equipped with higher-dimensional analogues of that same structure is known as the *microcosm principle*[17].

In Example 3.1.9, we saw that categories of endofunctors are strict monoidal categories. Following Example 3.4.6, this means that endofunctor categories are equivalently monoid objects. In fact, since categories are monoids with many objects[9], this means we can consider any object of endomorphisms as an appropriately typed monoid object.

---

[8]One often uses the prefix 'pseudo-' in category theory to denote a weak structure.

[9]This pattern—of extending structures to "many objects"—is sometimes called *horizontal categorification*, to distinguish it from the 'vertical' categorification of adding an extra dimension of morphism.

Example 3.4.8. If $c : \mathcal{C}$ is any object in any category $\mathcal{C}$, then the hom-set $\mathcal{C}(c, c)$ is a monoid $\big(\mathcal{C}(c, c), \circ, \mathrm{id}_c\big)$ in **Set**. More generally, if $\mathcal{C}$ is enriched in $\mathcal{E}$, then $\big(\mathcal{C}(c, c), \circ, \mathrm{id}_c\big)$ is a monoid in $\mathcal{E}$. In each case, we call the monoid the *endomorphism monoid* on c.

In the case when the endomorphism objects are categories, as in the case of Example 3.1.9, the monoidal structure makes them into monoidal categories, and so we can consider monoids objects defined internally to them. More generally, we can do this inside any bicategory, and the resulting monoids will play an important rôle subsequently.

Remark 3.4.9. Just as a monoidal category is a bicategory with a single object, the hom-category $\mathcal{B}(b, b)$ for any 0-cell b in a bicategory $\mathcal{B}$ is a monoidal category: the objects are the 1-cells $b \to b$, the morphisms are the 2-cells between them, composed vertically; the tensor is horizontal composition of 1-cells, and its unit is the identity 1-cell id_b. We can therefore define a *monoid in a bicategory $\mathcal{B}$* to be a monoid in $\mathcal{B}(b, b)$ for some 0-cell $b : \mathcal{B}$, using this induced monoidal structure.

Since **Cat** is morally a 2-category (and *a fortiori* a bicategory), and thus to avoid confusion with monoid objects in $(\mathbf{Cat}, \times, 1)$ (*i.e.* strict monoidal categories), we will introduce a new term for monoids in the bicategory **Cat**.

Definition 3.4.10. A *monad* on the category $\mathcal{C}$ is a monoid object in the strict monoidal category $(\mathcal{C}^{\mathcal{C}}, \circ, \mathrm{id}_{\mathcal{C}})$.

Monads are often defined in a more explicit way, by expressing the monoid structures and axioms directly and diagrammatically.

Proposition 3.4.11. A monad on $\mathcal{C}$ is equivalently a triple (T, μ, η) of

1. a functor $T : \mathcal{C} \to \mathcal{C}$;

2. a natural transformation $\mu : TT \Rightarrow T$ called the *multiplication*; and

3. a natural transformation $\eta : \mathrm{id}_{\mathcal{C}} \Rightarrow T$ called the *unit*;

such that, for all $c : \mathcal{C}$, the following diagrams commute:

$$
\begin{array}{ccc}
TTTc & \xrightarrow{\;T\mu_c\;} & TTc \\
{\scriptstyle \mu_{Tc}}\big\downarrow & & \big\downarrow{\scriptstyle \mu_c} \\
TTc & \xrightarrow{\;\mu_c\;} & Tc
\end{array}
\qquad \text{and} \qquad
\begin{array}{ccccc}
Tc & \xrightarrow{\;\eta_{Tc}\;} & TTc & \xleftarrow{\;T\eta_c\;} & Tc \\
 & \searrow & \big\downarrow{\scriptstyle \mu_c} & \swarrow & \\
 & & Tc & &
\end{array}
$$

A monad is like a monoidal structure for composition: instead of taking two objects and constructing a single object representing their conjunction (like the tensor of a monoidal category), a monad takes two levels of nesting and composes them into a single level; this is the source of the connection between multicategory algebras and monad algebras.

Example 3.4.12. Recall the list monoid from Example 3.4.4. The mapping $A \mapsto \mathsf{List}(A)$ defines the functor part of a monad $\mathsf{List} : \mathbf{Set} \to \mathbf{Set}$; given a function $f : A \to B$, $\mathsf{List}(f) : \mathsf{List}(A) \to \mathsf{List}(B)$ is defined by applying f to each element of the lists: $(a_1, a_2, \dots) \mapsto (f(a_1), f(a_2), \dots)$. The monad multiplication $\mu : \mathsf{List}^2 \Rightarrow \mathsf{List}$ is given by "removing inner brackets" from lists of lists: $\mu_A\big((a_1^1, a_1^2, \dots), (a_2^1, \dots), \dots\big) = (a_1^1, a_1^2, \dots, a_2^1, \dots, \dots)$; equivalently, form the perspective of Example 3.4.4, this is the concatenation of the 'inner' lists into a single list. The monad unit $\eta : \mathsf{id}_{\mathbf{Set}} \Rightarrow \mathsf{List}$ is defined by returning 'singleton' lists: $\eta_A : A \to \mathsf{List}(A) : a \mapsto (a)$.

There is a close connection between monads and adjunctions: every adjunction induces a monad.

Proposition 3.4.13. Suppose $L \dashv R : \mathcal{D} \to \mathcal{C}$ is an adjunction, with unit $\eta : \mathsf{id}_{\mathcal{C}} \Rightarrow RL$ and counit $\epsilon : LR \Rightarrow \mathsf{id}_{\mathcal{D}}$. Then $(RL, R\epsilon_L, \eta)$ is a monad.

Proof. To see that the associativity axiom is satisfied, observe that $R\epsilon_{LRL} = RL\epsilon_{RL} = RLR\epsilon_L$ by naturality. Right unitality follows by the triangle identity $\epsilon_L \circ L\eta = \mathsf{id}_L$, which entails the required equation $R\epsilon_L \circ RL\eta = \mathsf{id}_{RL}$; and left unitality follows from right unitality by naturality, as $\eta_{RL} = RL\eta$. $\qquad\square$

It is also true that every monad arises from an adjunction: in fact, there are typically multiple adjunctions inducing the same monad, and we will exhibit one extremal case in §4.1.

Remark 3.4.14. This dual correspondence is itself an example of an adjunction—in the quite general bicategorical sense, following the definition of monad as a monoid in a bicategory—though we leave the demonstration of this to the reader.

Before we show in generality how every monad arises from an adjunction, we can exhibit the list monad as a classic special case.

Example 3.4.15 (Lists are free monoids). There is a forgetful functor $U : \mathbf{Mon}(\mathbf{Set}) \to \mathbf{Set}$, taking each monoid $(M, \circ, *)$ (or monoid morphism f) and forgetting the monoid structure to return just the set M (or the morphism f). This functor has a left adjoint $F : \mathbf{Set} \to \mathbf{Mon}(\mathbf{Set})$, which takes each set A to the *free monoid* on A; this free monoid $F(A)$ is precisely the monoid

111

$\bigl(\mathsf{List}(A), \circ, ()\bigr)$ of lists in A, equipped with concatenation as multiplication and the empty list as unit, as described in Example 3.4.4. The induced monad $(\mathsf{List}, \mu, \eta)$, described in Example 3.4.12, is then precisely the monad induced by this adjunction, with $\mathsf{List} = UF$.

At this point, with an example of a monad to hand, we can start to explore their connection to algebra.

Definition 3.4.16. Suppose (T, μ, η) is a monad on $\mathcal{C}$. A T-*algebra* is a choice of object $A : \mathcal{C}$ and a morphism $a : TA \to A$ such that the following diagrams commute:

$$
\begin{array}{ccc}
A \xrightarrow{\ \eta_A\ } TA & & TTA \xrightarrow{\ Ta\ } TA \\
\ \ \searrow \ \ \downarrow a & \text{and} & \mu_A \downarrow \qquad\qquad \downarrow a \\
\qquad A & & TA \xrightarrow{\ a\ } A
\end{array}
$$

Once again, this being category theory, we are interested less in individual T-algebras than in their category.

Definition 3.4.17. A *morphism of T-algebras* $(A, a) \to (B, b)$ is a morphism $f : A \to B$ that preserves the T-algebra structures, in the sense that the following diagram commutes:

$$
\begin{array}{ccc}
TA & \xrightarrow{\ Tf\ } & TB \\
a \downarrow & & \downarrow b \\
A & \xrightarrow{\ f\ } & B
\end{array}
$$

T-algebras and their morphisms constitute a category, denoted $\mathbf{Alg}(T)$ and called the *category of T-algebras* or the *Eilenberg-Moore category* for T. (Algebra morphisms compose by the composition of morphisms; a composite morphism of T-algebras is again a morphism of T-algebras by pasting. Identities are the usual identity morphisms in $\mathcal{C}$.)

We now demonstrate the 'algebra' of monad algebras using two familiar examples.

Example 3.4.18. The category of monoids in $(\mathbf{Set}, \times, 1)$ is equivalent to the category of List-algebras. A List-algebra is a pair of a set A and a function $a : \mathsf{List}(A) \to A$ satisfying the algebra axioms, which mean that a must map singleton lists to their corresponding elements, and that a must respect the ordering of elements in the list (so that it doesn't matter whether you apply a to the lists in a lists of lists, or to the collapsed list resulting from the monad multiplication). To obtain a monoid, we can simply take the set A. The monoid multiplication is given by the action of a on

2-element lists; and the monoid unit is given by the action of a on the empty list. Since a satisfies the monad algebra laws, the resulting multiplication and unit satisfy the monoid axioms: the monad laws are a categorification of the monoid axioms, and the algebra laws ensure compatibility with them.

Dually, given a monoid (A, m, e), we can construct a List algebra a by induction: on empty lists, return e; on singleton lists, return their elements; on 2-element lists, apply m; on lists of length n, apply m to the first two elements to obtain a list of length $n - 1$ repeatedly until reaching the 2-element case. The monoid laws then ensure that the monad axioms are satisfied.

Example 3.4.19. Recall from Proposition 2.1.8 that one can obtain from any category a directed graph by forgetting the compositional structure and retaining only the objects and morphisms as nodes and edges. Recall also from Proposition 2.1.9 that one can obtain from any directed graph $\mathcal{G}$ a category $F\mathcal{G}$, the free category on $\mathcal{G}$ whose objects are the nodes of $\mathcal{G}$ and whose morphisms are paths in $\mathcal{G}$. These two constructions form a free-forgetful adjunction, $F \dashv U : \mathbf{Cat} \to \mathbf{Graph}$, and the induced monad $UF : \mathbf{Graph} \to \mathbf{Graph}$ is called the *path monad*: on objects, it takes a graph $\mathcal{G}$ and returns a graph with the same nodes but whose edges are the paths in $\mathcal{G}$. The category $\mathbf{Alg}(UF)$ of algebras of UF is equivalent to the category $\mathbf{Cat}$ of (small) categories.

To see this, note that a UF-algebra is a graph homomorphism $UF\mathcal{G} \to \mathcal{G}$, for some graph $\mathcal{G}$: a mapping of nodes in $UF\mathcal{G}$ to nodes in $\mathcal{G}$, and a mapping of edges in $UF\mathcal{G}$ to edges in $\mathcal{G}$ that preserves domains and codomains. Since $UF\mathcal{G}$ and $\mathcal{G}$ have the same nodes, the simplest choice is to map each node to itself: we will consider the nodes as the objects of the resulting category. The mapping of paths to edges induces a composition operation on the edges of $\mathcal{G}$, which we henceforth think of as morphisms. The reasoning proceeds inductively, much like the List-algebra case: we take paths of length 0 to be identity morphisms; paths of length 1 are taken to their constituent morphisms; paths of length 2 are taken to their composites; and one obtains the composites of longer paths by induction. Associativity and unitality then follow easily from the monad algebra laws.

Remark 3.4.20. Both the preceding examples suggest a connection between monad algebras and inductive reasoning, and indeed one can formalize inductive reasoning (as *inductive types*) in terms of algebras. Dually, there is a close connection between 'coalgebras' and 'coinduction', which can be used to formalize the behaviours of systems that can be iterated, such as dynamical systems. As an informal example, the *coinductive type* corresponding to List is the type of "streams":

possibly infinite lists of the states or outputs of transition systems. In Chapter 6, we use coalgebra to formalize the compositional structure of 'open' (*i.e.*, interacting) dynamical systems quite generally.

In the Appendix (§A.1), we pursue the monad algebra story a little further, to demonstrate the connection with multicategory algebra. However, since that connection is not strictly germane to the rest of the thesis, and with the suggested notion of coalgebra to whet our appetite, we now turn to monoids' duals, comonoids.

3.4.1. Comonoids

We introduced comonoids graphically at the beginning of §3.1.1, as a structural manifestation of copying and discarding, but in the fullest of generality, comonoids are simply monoids in opposite categories.

Definition 3.4.21. A *comonoid* in $(\mathcal{C}, \otimes, I)$ is a monoid in $\mathcal{C}^{\mathrm{op}}$, when $\mathcal{C}^{\mathrm{op}}$ is equipped with the opposite monoidal structure induced by $(\otimes, I)$. Explicitly, this means an object $c : \mathcal{C}$ equipped with a *comultiplication* $\delta : c \to c \otimes c$ and *counit* $\epsilon : c \to I$, satisfying *counitality* and *coassociativity* laws formally dual to the corresponding unitality and associativity laws of monoids: read the diagrams of Definition 3.4.1 top-to-bottom, rather than bottom-to-top. Likewise, if $\mathcal{C}$ is symmetric monoidal, we say that a comonoid in $\mathcal{C}$ is *cocommutative* if its comultiplication commutes with the symmetry.

Example 3.4.22. Every object in a category with finite products $\times$ and a terminal object 1 is a comonoid with respect to the monoidal structure $(\times, 1)$. The comultiplications $\delta_X : X \to X \times X$ are defined by the pairing $(\mathrm{id}_X, \mathrm{id}_X)$ (recall Definition 2.3.15) and the counits $\epsilon_X : X \to 1$ are (necessarily) the unique morphisms into the terminal object.

Coassociativity follows because $(\mathrm{id}_X, (\mathrm{id}_X, \mathrm{id}_X)) = \alpha_{X,X,X} \circ ((\mathrm{id}_X, \mathrm{id}_X), \mathrm{id}_X)$, where α is the associator of the product. Counitality follows by the naturality of pairing, $(\mathrm{id}_X \times !) \circ (\mathrm{id}_X, \mathrm{id}_X) = (\mathrm{id}_X, !)$, and because $\mathrm{proj}_X \circ (\mathrm{id}_X, !) = \mathrm{id}_X$ by the universal property of the product; note that proj_X is the X component of the right unitor of the monoidal structure, and $(\mathrm{id}_X, !)$ is its inverse.

Instantiating this example in **Set**, we see that the comultiplication is given by copying, *i.e.*, $x \mapsto (x, x)$; and the counit is the unique map $x \mapsto *$ into the singleton set. This justifies our writing of the comonoid structure in copy-discard style as $(\mathbf{\Psi}_X, \bar{\bar{\top}}_X)$.

In general, when a comonoid structure is to be interpreted as a copy-discard structure, we will therefore write the struture morphisms as $(\mathbf{\Psi}, \bar{\bar{\top}})$ and depict them accordingly in the graphical calculus, rather than using the boxed forms of Definition 3.4.1. However, copy-discard structures

are not the only important comonoids that we will encounter. In the next section, we introduce the category of polynomial functors $\mathbf{Set} \to \mathbf{Set}$, and since these are endofunctors, their category inherits a monoidal structure given by functor composition. Comonoids for this monoidal structure in **Poly** give us another definition for a now multifariously familiar concept: they are again small categories, although their morphisms are not functors but rather *cofunctors*.

Of course, a morphism of comonoids is much like a morphism of monoids.

Definition 3.4.23. A *comonoid morphism* $f : (c, \delta, \epsilon) \to (c', \delta', \epsilon')$ in $(\mathcal{C}, \otimes, I)$ is a morphism $f : c \to c'$ that is compatible with the comonoid structures, in the sense of satisfying axioms dual to those of Definition 3.4.2. There is thus a category $\mathbf{Comon}(\mathcal{C})$ of comonoids in $\mathcal{C}$ and their morphisms, as well as a subcategory $\mathbf{CComon}(\mathcal{C}) \hookrightarrow \mathbf{Comon}(\mathcal{C})$ of commutatitve comonoids.

In the more familiar copy-discard setting, comonoid morphisms also play an important rôle. In the next chapter, we will see concretely that, in the context of stochastic maps, comonoid morphisms (with respect to the tensor product) correspond to the deterministic functions. This result is closely related to the following fact.

Proposition 3.4.24. If every morphism in the monoidal category $(\mathcal{C}, \otimes, I)$ is a comonoid morphism, then $a \otimes b$ satisfies the universal property of the product for every $a, b : \mathcal{C}$, and hence $\otimes$ is the categorical product and I the terminal object in $\mathcal{C}$ (up to isomorphism).

Proof. If every morphism is a comonoid morphism, then every object $a : \mathcal{C}$ carries a comonoid structure; assume a choice of comonoid structure $(\text{\Y}_a : a \to a \otimes a, \bar{\top}_a : a \to I)$ for every $a : \mathcal{C}$. The universal property of the product says that every morphism $f : x \to a \otimes b$ factors as

where $f_a : x \to a$ and $f_b : x \to b$ are uniquely defined as

Since f is *ex hypothesi* a comonoid morphism, we have

where the first equality holds by counitality, the second since f commutes with $\curlyvee_x$ *ex hypothesi*, and the third by definition. This establishes that $a \otimes b$ satisfies the universal property, and hence that $\otimes$ is the categorical product.

To see that I is the terminal object up to isomorphism, suppose that 1 is the terminal object. Since $\otimes$ is the categorical product, there is an isomorphism $a \xrightarrow{\sim} a \otimes 1$ for any $a : \mathcal{C}$, by the universal property. In particular, there is an isomorphism $I \xrightarrow{\sim} I \otimes 1$. But since I is the monoidal unit for $\otimes$, the component of the left unitor at 1 is an isomorphism $I \otimes 1 \xrightarrow{\sim} 1$. Hence we have a composite isomorphism $I \xrightarrow{\sim} I \otimes 1 \xrightarrow{\sim} 1$, and so $I \cong 1$. □

Remark 3.4.25. The preceding proposition gives us another way to look at comonoids: we can think of them as "products without the universal property". The reason for this is that, since products are characterized by their (universal) projections, we can use the counits to define projections for the monoidal product of comonoids: that is, if a and b are comonoids in $\mathcal{C}$, then we can define (non-universal) projections $a \xleftarrow{\mathsf{proj}_a} a \otimes b \xrightarrow{\mathsf{proj}_b} b$ by

$$a \xleftarrow{\rho_a} a \otimes I \xleftarrow{\mathsf{id}_a \otimes \bar{\bar{\top}}_b} a \otimes b \xrightarrow{\bar{\bar{\top}}_a \otimes \mathsf{id}_b} I \otimes b \xrightarrow{\lambda_b} b$$

where ρ and λ denote the right and left unitors of the monoidal structure respectively. The failure of universality means that the family of projections $\{\mathsf{proj}_a\}_{a:\mathcal{C}}$ in $\mathcal{C}$ does not constitute a natural transformation.

Remark 3.4.26. Abstractly, we can use naturality as a way to characterize deterministic morphisms: the naturality law for $\curlyvee$ requires that

$$a \xrightarrow{f} b \xrightarrow{\curlyvee_b} b \otimes b \; = \; a \xrightarrow{\curlyvee_a} a \otimes a \xrightarrow{f \otimes f} b \otimes b$$

and this says that first doing f and the copying its output is the same as copying the input and feeding each copy into f. If f were non-deterministic, then there would be a correlation between the copies in the former case but not in the latter, and so this equation would not hold. Therefore, we can think of those morphisms f for which copying *is* natural as the deterministic morphisms in $\mathcal{C}$. We will return to this perspective in Remark 4.1.19.

Finally, there is also a notion of comonad, dual to monad: a comonad is quite generally a comonoid in a bicategory, in the sense of Remark 3.4.9, or, less generally, a comonoid with respect to the composition product in a category of endofunctors. This means that the polynomial comonoids we discussed above are by definition comonads.

In Remark 3.4.20, we introduced the notion of 'coalgebra', and indeed there is a notion of comonad coalgebra that is dual to the notion of monad algebra; and indeed we will use coalgebras later to formalize dynamical systems. But although these coalgebras will be morphisms of the form $FX \to X$, for F an endofunctor and X an object, the endofunctor F will not necessarily have a comonad structure, and so the coalgebras will be more general than the algebras we considered above: there will be no comonad compatibility axioms to satisfy.

In many cases, the endofunctor F will be a polynomial functor, so let us now introduce these.

3.5. Polynomial functors

In order to be considered *adaptive*, a system must have something to adapt to. This 'something' is often what we call the system's *environment*, and we say that the system is *open* to its environment. The interface or boundary separating the system from its environment can be thought of as 'inhabited' by the system: the system is embodied by its interface of interaction; the interface is animated by the system. In this way, the system can affect the environment, by changing the shape or configuration of its interface[10]; through the coupling, these changes are propagated to the environment. In turn, the environment may impinge on the interface: its own changes, mediated by the coupling, arrive at the interface as immanent signals; and the type of signals to which the system is alive may depend on the system's configuration (as when an eye can only perceive if its lid is open). Thus, information flows across the interface.

The mathematical language capturing this kind of inhabited interaction is that of *polynomial functors*, which we adopt following Spivak and Niu [235]. We will see that this language—or rather, its category—is sufficiently richly structured to provide both a satisfactory syntax for the patterns of interaction of adaptive systems, generalizing the circuit diagrams of §3.3.2, as well as a home for the dynamical semantics that we will seek.

Polynomial functors are so named because they are a categorification of polynomial functions: functions built from sums, products, and exponentials, of the form $y \mapsto \sum_{i:I} b_i \, y^{a_i}$. To categorify a function of this kind, we can simply interpret the coefficients and exponents and the variable y as standing for sets rather than mere numbers. In this way, we reinterpret the term y^{a_i} as the representable copresheaf $\mathbf{Set}(a_i, -)$, so that we can substitute in any set X and obtain the exponential X^{a_i} (just as in the classical case). To categorify the sums and products, we can simply use the universal constructions available in the copresheaf category $\mathbf{Set}^{\mathbf{Set}}$: these are still available

---

[10]Such changes can be very general: consider for instance the changes involved in producing sound (*e.g.*, rapid vibration of tissue) or light (*e.g.*, connecting a luminescent circuit, or the molecular interactions involved therein).

in the subcategory **Poly**, since **Poly** is by definition the subcategory of the copresheaf category on sums of representables (and as we have seen, products are equivalently iterated coproducts).

Remark 3.5.1. Limits and colimits in (co)presheaf categories are computed 'pointwise'. Therefore, if F and G are two copresheaves $\mathcal{C} \to \mathbf{Set}$, then their sum $F + G$ is the copresheaf defined by $x \mapsto F(x) + G(x)$ and their product is the copresheaf defined by $x \mapsto F(x) \times G(x)$.

We will adopt the standard notation for polynomial functors of Spivak and Niu [235], so that if p is a polynomial, we will expand it as $\sum_{i:p(1)} y^{p[i]}$. When treating p as encoding the type of a system's interface, we will interpret $p(1)$ as encoding the set of possible configurations (or 'shapes') that the system may adopt, and for each configuration $i : p(1)$, the set $p[i]$ is the set of possible immanent signals ('inputs') that may arrive on the interface in configuration i.

Definition 3.5.2. First, if A be any set, we will denote by y^A its representable copresheaf $y^A := \mathbf{Set}(A, -) : \mathbf{Set} \to \mathbf{Set}$. A *polynomial functor* $p : \mathbf{Set} \to \mathbf{Set}$ is then an indexed coproduct of such representable copresheaves, written $p := \sum_{i:p(1)} y^{p_i}$, where $p(1)$ denotes the indexing set and $p[i]$ the representing set for each i. The category of polynomial functors is the full subcategory $\mathbf{Poly} \hookrightarrow \mathbf{Set}^{\mathbf{Set}}$ of the copresheaf category spanned by coproducts of representables. A morphism of polynomials is thus a natural transformation.

Remark 3.5.3. Note that, given a polynomial functor $p : \mathbf{Set} \to \mathbf{Set}$, the indexing set $p(1)$ is indeed obtained by applying p to the terminal set 1.

We will make much use of the following 'bundle' representation of polynomial functors and their morphisms.

Proposition 3.5.4. Every polynomial functor $\sum_{i:p(1)} y^{p_i}$ corresponds to a bundle (a function) $p : \sum_{i:p(1)} p_i \to p(1)$, where the set $\sum_{i:p(1)} p_i$ is the $p(1)$-indexed coproduct of the representing objects p_i and p is the projection out of the coproduct onto the indexing set $p(1)$.

Every morphism of polynomials $f : p \to q$ corresponds to a pair $(f_1, f^\sharp)$ of a function $f_1 : p(1) \to q(1)$ and a $p(1)$-indexed family of functions $f_i^\sharp : q[f_1(i)] \to p[i]$ making the diagram below commute. We adopt the notation $p[i] := p_i$, and write $f^\sharp$ to denote the coproduct $\sum_i f_i^\sharp$.

$$
\begin{array}{ccccc}
\sum_{i:p(1)} p[i] & \xleftarrow{\;f^\sharp\;} & \sum_{i:p(1)} q[f_1(i)] & \longrightarrow & \sum_{j:q(1)} q[j] \\
{\scriptstyle p}\downarrow & & \downarrow & & \downarrow{\scriptstyle q} \\
p(1) & =\!=\!=\!=\!= & p(1) & \xrightarrow{\;f_1\;} & q(1)
\end{array}
$$

Given $f : p \to q$ and $g : q \to r$, their composite $g \circ f : p \to r$ is as marked in the diagram

$$
\begin{array}{ccccc}
\sum_{i:p(1)} p[i] & \xleftarrow{(gf)^\sharp} & \sum_{i:p(1)} r[g_1 \circ f_1(i)] & \longrightarrow & \sum_{k:r(1)} r[k] \\
p \downarrow & & \downarrow & \lrcorner & \downarrow r \\
p(1) & =\!\!\!=\!\!\!= & p(1) & \xrightarrow{g_1 \circ f_1} & r(1)
\end{array}
$$

where $(gf)^\sharp$ is the coproduct of the $p(1)$-indexed family of composite maps

$$
r[g_1(f_1(i))] \xrightarrow{f^* g^\sharp} q[f_1(i)] \xrightarrow{f^\sharp} p[i] \ .
$$

The identity morphism on a polynomial p is $(\mathsf{id}_{p(1)}, \mathsf{id})$.

Proof. We just need to show that every natural transformation between polynomial functors corresponds to a pair of maps $(f_1, f^\sharp)$ as defined above. The set of natural transformations $\sum_{i:p(1)} y^{p[i]} \Rightarrow \sum_{j:q(1)} y^{q[j]}$ is the hom-set $\mathbf{Set}^{\mathbf{Set}}\left(\sum_{i:p(1)} y^{p[i]}, \sum_{j:q(1)} y^{q[j]}\right)$. Since the contravariant hom functor takes colimits to limits (Remark 2.3.53), this hom-set is isomorphic to $\prod_{i:p(1)} \mathbf{Set}^{\mathbf{Set}}(y^{p[i]}, \sum_{j:q(1)} y^{q[j]})$. By the Yoneda lemma, this is in turn isomorphic to $\prod_{i:p(1)} \sum_{j:q(1)} p[i]^{q[j]}$. And since products distribute over sums, we can rewrite this as $\sum_{f_1:p(1)\to q(1)} \prod_{i:p(1)} p[i]^{q[f_1(i)]}$. The elements of this set are precisely pairs of a function $f_1 : p(1) \to q(1)$ along with a family of functions $q[f_1(i)] \to p[i]$ indexed by $i : p(1)$, such that the diagram above commutes. $\square$

We now recall a handful of useful facts about polynomials and their morphisms, each of which is explained in Spivak and Niu [235] and summarized in Spivak [241].

We will consider the unit polynomial y to represent a 'closed' system, since it has no nontrivial configurations and no possibility of external input. For this reason, morphisms $p \to y$ will represent ways to make an open system closed, and in this context the following fact explains why: such morphisms correspond to a choice of possible input for each p-configuration; that is, they encode "how the environment might respond to p".

Proposition 3.5.5. Polynomial morphisms $p \to y$ correspond to sections $p(1) \to \sum_i p[i]$ of the corresponding function $p : \sum_i p[i] \to p(1)$.

The following embedding of **Set** into **Poly** will be useful in constructing 'hierarchical' dynamical systems.

Proposition 3.5.6. There is an embedding of **Set** into **Poly** given by taking sets X to the linear polynomials $Xy : \mathbf{Poly}$ and functions $f : X \to Y$ to morphisms $(f, \mathsf{id}_X) : Xy \to Yy$.

There are many monoidal structures on **Poly**, but two will be particularly important for us. The first represents the parallel composition of systems.

Proposition 3.5.7. There is a symmetric monoidal structure $(\otimes, y)$ on **Poly** that we call *tensor*, and which is given on objects by $p \otimes q := \sum_{i:p(1)} \sum_{j:q(1)} y^{p[i] \times q[j]}$ and on morphisms $f := (f_1, f^\sharp) : p \to p'$ and $g := (g_1, g^\sharp) : q \to q'$ by $f \otimes g := (f_1 \times g_1, \Sigma(f^\sharp, g^\sharp))$, where $\Sigma(f^\sharp, g^\sharp)$ is the family of functions

$$\Sigma(f^\sharp, g^\sharp)_{i,j} := p'[f_1(i)] \times q'[g_1(j)] \xrightarrow{f_i^\sharp \times g_j^\sharp} p[i] \times q[j]$$

indexed by $(i,j) : p(1) \times q(1)$. This is to say that the 'forwards' component of $f \otimes g$ is the product of the forwards components of f and g, while the 'backwards' component is the *pointwise* product of the respective backwards components.

Proposition 3.5.8. $(\textbf{Poly}, \otimes, y)$ is symmetric monoidal closed, with internal hom denoted $[-, =]$. Explicitly, we have $[p, q] = \sum_{f:p \to q} y^{\sum_{i:p(1)} q[f_1(i)]}$. Given an set A, we have $[Ay, y] \cong y^A$.

The second important monoidal structure is that inherited from the composition of endofunctors. To avoid confusion with other composition operators, we will in this context denote the operation by $\lhd$.

Proposition 3.5.9. The composition of polynomial functors $q \circ p : \mathcal{E} \to \mathcal{E} \to \mathcal{E}$ induces a monoidal structure on $\textbf{Poly}_\mathcal{E}$, which we denote $\lhd$, and call 'composition' or 'substitution'. Its unit is again y.

Comonoids with respect to $\lhd$ play a particularly important rôle in the theory of polynomial functors, and we will make accordingly much use of them.

Proposition 3.5.10 (Ahman and Uustalu [7, §3.2])**.** Comonoids in $(\textbf{Poly}, \lhd, y)$ correspond to small categories. If (c, δ, ϵ) is a comonoid, then the shapes $c(1)$ are the objects of the corresponding category $\mathcal{C}$. For each object $x : c(1)$, $c[i]$ is the set $\sum_{y:c(1)} \mathcal{C}(x, y)$ of morphisms out of x. The counit morphism $\epsilon : c \to y$ is, following Proposition 3.5.5, a section of c, and assigns to each $x : c(1)$ its identity morphism $\text{id}_x : x \to x$. The comultiplication $\delta : c \to c \lhd c$ encodes morphisms' codomains (its forward action) and their composition (its backward action). Finally, the comonoid laws ensure that the category is well defined.

Remark 3.5.11. $\lhd$-comonoid homomorphisms are not, as one might expect, functors; rather, they are 'cofunctors': they act backwards on morphisms. We will not explore the theory of cofunctors any further in this thesis, although we will make frequent use of them later in the context of dynamical systems.

The following $\lhd$-comonoids will play a prominent rôle in our dynamical developments.

Proposition 3.5.12. If $\mathbb{T}$ is a monoid in $(\mathbf{Set}, \times, 1)$, then the comonoid structure on $y^{\mathbb{T}}$ witnesses it as the category $\mathbf{B}\mathbb{T}$.

Proposition 3.5.13. Monomials of the form Sy^S can be equipped with a canonical comonoid structure witnessing the codiscrete groupoid on S: the category with an object for every element s of S and a morphism $s \to t$ for every pair of elements (s, t).

4. The compositional structure of Bayesian inference

This chapter introduces the fundamental concepts and structures needed for the development of statistical games in Chapter 5, and proves the crucial result that Bayesian updating composes according to the 'lens' pattern. To make sense of this statement, we first introduce compositional probability (§4.1), motivating it as a resolution of some imprecision that arises when one works informally with probability and statistics, particularly in the context of 'hierarchical' models. We exhibit categorical probability theory both abstractly (§4.1.2 and §4.1.3) and concretely (using discrete probability in §4.1.1 and 'continuous' probability in §4.1.4). We then move on to construct categories of bidirectional processes in §4.2, by first categorifying our earlier discussion of dependent data using the Grothendieck construction (§4.2.1) and then using this to introduce the lens pattern (§4.2.2).

In §4.3, we present our novel results. First, we introduce the indexed category of "state-dependent channels" in §4.3.1. These formalize the type of Bayesian inversions, and so in §4.3.2 we define the associated notion of *Bayesian lens*, and show in §4.3.3 that Bayesian updating composes according to the lens pattern. We end with a brief discussion of the 'lawfulness' of Bayesian lenses.

Remark 4.0.1. To gain some intuition about the hierarchical compositional structure of Bayesian inference, consider sitting close to the screen at a cinema. Your neural activity encodes a belief about where the characters are on the screen and what they are doing, but your visual field can only capture a part of the image at any one time. These incoming visual signals contain "low-level" information, about the light intensity over the patch of screen you can see, and the first job of the visual system is to infer what this means for what's going on in this patch. Of course, having been following the film so far, your brain encodes a high-level belief about what is going on across the whole screen, and it uses this to predict what to expect in the patch. This intermediate-level belief is then updated using the received visual signals, through a process of (approximate) Bayesian inference. The resulting intermediate-level posterior then supplies the input for a second inference process, updating the prior high-level belief accordingly. Notice that this means that the process of

prediction in such a hierarchical inference system points from inside an agent "towards the world"; and the belief-updating process points the other way, from the world into the agent.

4.1. Compositional probability

In informal literature, Bayes' rule is often written in the following form:

$$P(A|B) = \frac{P(B|A) \cdot P(A)}{P(B)}$$

where $P(A)$ is the probability of the event A, and $P(A|B)$ is the probability of the event A given that the event B occurred; and *vice versa* swapping A and B. Unfortunately, this notation obscures that there is in general no unique assignment of probabilities to events: different observers can hold different beliefs. Moreover, we are usually less interested in the probability of particular events than in the process of assigning probabilities to arbitrarily chosen beliefs; and what should be done if $P(B) = 0$ for some B? The aim in this section is to exhibit a general, precise, and compositional, form of Bayes' rule; we begin, as before, by introducing the intuition.

In the categorical setting, the assignment of probabilities or beliefs to events will formally be the task of a *state* (in the sense of §3.1.1) on the space from which the events are drawn; we should think of states as generalizing distributions or measures. With this notion to hand, we can write $P_\pi(A)$ to denote the probability of A according to the state π.

The formalization of *conditional* probability will be achieved by morphisms that we will call *channels*, meaning that we can write $P_c(B|A)$ to denote the probability of B given A according to the channel c. We can think of the channel c as taking events A as inputs and emitting states $c(A)$ as outputs. This means that we can alternatively write $P_c(B|A) = P_{c(A)}(B)$.

If the input events are drawn from the space X and the output states encode beliefs about Y, then the channel c will be a morphism $X \rightarrowtail Y$. Given a channel $c : X \rightarrowtail Y$ and a channel $d : Y \rightarrowtail Z$, we will understand their composite $d \bullet c : X \rightarrowtail Z$ as marginalizing (averaging) over the possible outcomes in Y. We will see precisely how this works in various settings below.

Notation 4.1.1. In a stochastic context, we will denote channels by the arrow $\rightarrowtail$, and write their composition operator as $\bullet$. We do this to distinguish stochastic channels from deterministic functions, which we will continue to write as $\rightarrow$ with composition $\circ$; in a number of situations, it will be desirable to work with both kinds of morphism and composition.

Given two spaces X and Y of events, we can form beliefs about them jointly, represented by states on the product space denoted $X \otimes Y$. The numerator in Bayes' rule represents such a joint state, by the law of conditional probability or 'product rule':

$$\mathrm{P}_\omega(A, B) = \mathrm{P}_c(B|A) \cdot \mathrm{P}_\pi(A) \tag{4.1}$$

where $\cdot$ is multiplication of probabilities, π is a state on X, and ω denotes the joint state on $X \otimes Y$. By composing c and π to form a state $c \bullet \pi$ on Y, we can write

$$\mathrm{P}_{\omega'}(B, A) = \mathrm{P}_{c_\pi^\dagger}(A|B) \cdot \mathrm{P}_{c \bullet \pi}(B)$$

where $c_\pi^\dagger$ will denote the Bayesian inversion of c with respect to π.

Joint states in classical probability theory are symmetric—and so the tensor $\otimes$ is symmetric—meaning that there is a family of isomorphisms $\mathsf{swap} : X \otimes Y \xrightarrow{\sim} Y \otimes X$, as in §3.1.1, and which will satisfy the symmetric monoidal category axioms (Definition 3.1.4). Consequently, we have $\mathrm{P}_\omega(A, B) = \mathrm{P}_{\omega'}(B, A)$ where $\omega' = \mathsf{swap} \bullet \omega$, and thus

$$\mathrm{P}_c(B|A) \cdot \mathrm{P}_\pi(A) = \mathrm{P}_{c_\pi^\dagger}(A|B) \cdot \mathrm{P}_{c \bullet \pi}(B) \tag{4.2}$$

where both left- and right-hand sides are called *disintegrations* of the joint state ω [60]. From this equality, we can write down the usual form of Bayes' theorem, now with the sources of belief indicated:

$$\mathrm{P}_{c_\pi^\dagger}(A|B) = \frac{\mathrm{P}_c(B|A) \cdot \mathrm{P}_\pi(A)}{\mathrm{P}_{c \bullet \pi}(B)} . \tag{4.3}$$

As long as $\mathrm{P}_{c \bullet \pi}(B) \neq 0$, this equality defines the inverse channel $c_\pi^\dagger$. If the division is undefined, or if we cannot guarantee $\mathrm{P}_{c \bullet \pi}(B) \neq 0$, then $c_\pi^\dagger$ can be any channel satisfying (4.2).

There is therefore generally no unique Bayesian inversion $c^\dagger : Y \rightarrow X$ for a given channel $c : X \rightarrow Y$: rather, we have an inverse $c_\pi^\dagger : Y \rightarrow X$ for each prior state π on X. Moreover, $c_\pi^\dagger$ is not a "posterior distribution" (as written in some literature), but a process which emits a posterior distribution, given an observation in Y. If we denote our category of stochastic channels by $\mathcal{C}$, then, by allowing π to vary, we obtain a map of the form $c_{(\cdot)}^\dagger : \mathcal{P}X \rightarrow \mathcal{C}(Y, X)$, where $\mathcal{P}X$ denotes a space of states on X. Note that here we are not assuming the object $\mathcal{P}X$ to be an object of $\mathcal{C}$ itself (though it often will be), but rather an object in its base of enrichment, so that here we can think of $c_{(\cdot)}^\dagger$ as a kind of externally parameterized channel (in the sense of §3.2.2). Making the type of this 'state-dependent' channel $c_{(\cdot)}^\dagger$ precise is the task of §4.2.1.

Remark 4.1.2. There are two easily confused pieces of terminology here. We will call the channel $c_\pi^\dagger$ the *Bayesian inversion* of the channel c with respect to π. Then, given some $y \in Y$, the state $c_\pi^\dagger(y)$ is a new 'posterior' distribution on X. We will call $c_\pi^\dagger(y)$ the *Bayesian update* of π along c given y.

In the remainder of this section, we instantiate the ideas above in categories of stochastic channels of various levels of generality, beginning with the familiar case of discrete (*i.e.*, finitely supported, or 'categorical') probability.

4.1.1. Discrete probability, algebraically

Interpreting the informal Bayes' rule (4.3) is simplest in the case of discrete or *finitely-supported* probability. Here, every event is a set, generated as the disjoint union of so many atomic (singleton) events, which one can therefore take as the elements of the set. A finitely-suported probability distribution is then simply an assignment of nonzero probabilities to finitely many elements, such that the sum of all the assignments is 1. This condition is a *convexity* condition, and so in this subsection we will introduce discrete compositional probability theory from a geometric perspective, using the algebraic tools of the previous chapter.

Definition 4.1.3. Suppose X is a set. A function $c : X \to [0, 1]$ such that $c(x) > 0$ for only finitely many $x : X$ and $\sum_{x:X} c(x) = 1$ will be called a *discrete* or *finitely-supported distribution* on X. We write $\mathcal{D}X$ to denote the set of discrete distributions on X. A (real-valued) *convex set* is a set X equipped with a function $\varsigma : \mathcal{D}X \to X$ called its *evaluation*.

Convex sets X are sets in which we can form convex combinations of elements. Algebraically, we can model these convex combinations as distributions on X, and the evaluations realize the convex combinations (distributions) as elements again of X: geometrically, the evaluation returns the barycentre of the distribution.

In light of Chapter 3, this situation may seem familiar. Indeed, the assignment $X \mapsto \mathcal{D}X$ is the functor part of a monad on **Set**, whose algebras are convex sets. This monad arises from a free-forgetful adjunction between the category of convex sets (the category of algebras of the monad) and the category **Set**. Later, we will find that the category of finitely-supported conditional probability distributions—the category of discrete stochastic channels—is equivalent to the category of free convex sets and their morphisms: a free convex set on X is equivalently a distribution on X.

Let us first formalize the functor $\mathcal{D}$.

Proposition 4.1.4. The mapping of sets $X \mapsto \mathcal{D}X$ is functorial. Given a function $f : X \to Y$, we obtain a function $\mathcal{D}f : \mathcal{D}X \to \mathcal{D}Y$ mapping $c : \mathcal{D}X$ to the distribution $Df(c) : \mathcal{D}Y$,

$$\mathcal{D}f(c) : Y \to [0,1]$$
$$y \mapsto \sum_{x:f(x)=y} c(x) .$$

Proof. Given $f : X \to Y$ and $g : Y \to Z$, we have

$$\mathcal{D}g(\mathcal{D}f(c)) : Z \to [0,1]$$
$$z \mapsto \sum_{y:g(y)=z} \sum_{x:f(x)=y} c(x)$$
$$= \sum_{x:g\circ f(x)=z} c(x)$$

hence $\mathcal{D}g \circ \mathcal{D}f = \mathcal{D}(g \circ f)$. We also have

$$\mathcal{D}(\mathsf{id}_X)(c) : X \to [0,1]$$
$$x \mapsto \sum_{x':\mathsf{id}_X(x')=x} c(x')$$
$$= c(x)$$

and hence $\mathcal{D}(\mathsf{id}) = \mathsf{id}_{\mathcal{D}}$. $\square$

To obtain the monad structure of $\mathcal{D}$, we will exhibit the free-forgetful adjunction. We start by defining the category of convex sets, and the specific case of a *free* convex set.

Definition 4.1.5. The *category of (real-valued) convex sets* **Conv** has convex sets (X, ς_X) as objects. Its morphisms $(X, \varsigma_X) \to (Y, \varsigma_Y)$ are functions $f : X \to Y$ that preserve the convex structure, in the sense that the following square commutes:

$$
\begin{array}{ccc}
\mathcal{D}X & \xrightarrow{\ \mathcal{D}f\ } & \mathcal{D}Y \\
{\scriptstyle \varsigma_X}\downarrow & & \downarrow{\scriptstyle \varsigma_Y} \\
X & \xrightarrow{\ f\ } & Y
\end{array}
$$

Definition 4.1.6. If X is any set, then the *free convex set* on X is the set $\mathcal{D}X$ equipped with the evaluation $\mu_X : \mathcal{D}\mathcal{D}X \to \mathcal{D}X$ which maps $\alpha : \mathcal{D}\mathcal{D}X$ to the distribution $\mu_X(\alpha) : \mathcal{D}X$,

$$\mu_X(\alpha) : X \to [0,1]$$
$$x \mapsto \sum_{c:\mathcal{D}X} \alpha(c) \cdot c(x) .$$

Notation 4.1.7. To emphasize the algebraic nature of finitely-supported distributions $\pi : \mathcal{D}X$, instead of writing them as functions $x \mapsto \pi(x)$, we can write them as *formal sums* or *formal convex combinations* $\sum_{x:X} \pi(x) \, |x\rangle$, with each element $x : X$ corresponding to a formal "basis vector" $|x\rangle$ with the coefficient $\pi(x)$. If X is a convex set, then the evaluation realizes this formal sum as an actual element ('vector') in X.

We are now in a position to exhibit the adjunction: the reasoning behind the following proposition follows the lines of Example 3.4.19 and Proposition 2.2.25 (on the free-forgetful adjunction between graphs and categories).

Proposition 4.1.8. The mapping of $X : \mathbf{Set}$ to the free convex set $(\mathcal{D}X, \mu_X)$ defines a functor $F : \mathbf{Set} \to \mathbf{Conv}$ which takes functions $f : X \to Y$ to morphisms $Ff : (\mathcal{D}X, \mu_X) \to (\mathcal{D}Y, \mu_Y)$ defined by $\mathcal{D}f : \mathcal{D}X \to \mathcal{D}Y$. This functor F is left adjoint to the forgetful functor $U : \mathbf{Conv} \to \mathbf{Set}$ which acts by $(X, \varsigma_X) \mapsto X$.

Using Proposition 3.4.13, the adjunction gives us a monad.

Corollary 4.1.9. The functor $\mathcal{D} : \mathbf{Set} \to \mathbf{Set}$ is equivalently the functor part of the monad induced by the free-forgetful adjunction on convex sets. It therefore acquires a monad structure (μ, η) where the components of the multiplication μ are the free evaluations $\mu_X : \mathcal{D}\mathcal{D}X \to \mathcal{D}X$, and the unit η has components $\eta_X : X \to \mathcal{D}X$ which return the 'Dirac' distributions, as in

$$\eta_X(x) : X \to [0,1]$$
$$x' \mapsto \begin{cases} 1 & \text{iff } x = x' \\ 0 & \text{otherwise.} \end{cases}$$

And $\mathbf{Conv}$ is the category of algebras for this monad.

Corollary 4.1.10. $\mathbf{Conv} \cong \mathbf{Alg}(\mathcal{D})$.

Using Corollary 4.1.10, we can actually exhibit the relationship between the monad $\mathcal{D}$ and its defining adjunction tautologously: every monad T on a category $\mathcal{C}$ induces an free-forgetful adjunction between its category of algebras $\mathbf{Alg}(T)$ and $\mathcal{C}$ itself, such that the monad generated by this adjunction is again T. This is precisely the situation here.

Proposition 4.1.11. Suppose (T, μ, η) is a monad on the category $\mathcal{C}$. There is a forgetful functor $U : \mathbf{Alg}(T) \to \mathcal{C}$ which has a left adjoint $F : \mathcal{C} \to \mathbf{Alg}(T)$ taking each object $X : \mathcal{C}$ to the *free T-algebra* (TX, μ_X) on X, where $\mu_X : TTX \to TX$ is the component of the monad multiplication μ at X. The unit of the adjunction is the monadic unit η, the counit ϵ is defined by $\epsilon_{(X, \varsigma_X)} := \varsigma_X$, and the monad induced by the adjunction is (T, μ, η).

Proof sketch. The proof that F is left adjoint to U is standard (see Borceux [39, Prop. 4.1.4]), and that the adjunction generates the monad follows almost immediately from Proposition 3.4.13. $\quad\square$

Remark 4.1.12. It must be emphasized that, although every monad arises from such a free-forgetful adjunction, not every adjunction does! (Consider for example the adjunction $\Delta \dashv \lim$ of Proposition 2.3.48: Δ does not assign to each $c : \mathcal{C}$ the "free J-shaped diagram on c", and $\lim$ does not simply forget diagrammatic structure.) Those adjunctions which do arise from monads in this way are called *monadic*.

There is a special name for subcategories of *free* algebras.

Definition 4.1.13. Suppose (T, μ, η) is a monad on $\mathcal{C}$. The subcategory of $\mathbf{Alg}(T)$ on the free T-algebras (TX, μ_X) is called the *Kleisli category* for T, and denoted $\mathcal{K\ell}(T)$.

The following proposition gives us an alternative presentation of $\mathcal{K\ell}(T)$ which, when applied to the monad $\mathcal{D}$, will yield a computationally meaningful category of finitely-supported stochastic channels.

Proposition 4.1.14. The objects of $\mathcal{K\ell}(T)$ are the objects of $\mathcal{C}$. The morphisms $X \nrightarrow Y$ of $\mathcal{K\ell}(T)$ are the morphisms $X \to TY$ of $\mathcal{C}$. Identity morphisms $\mathsf{id}_X : X \nrightarrow X$ are given by the monadic unit $\eta_X : X \to TX$. Composition is defined by *Kleisli extension*: given $g : Y \nrightarrow Z$, we form its Kleisli extension $g^\triangleright : TY \nrightarrow Z$ as the composite $TY \xrightarrow{Tg} TTZ \xrightarrow{\mu_Z} TZ$ in $\mathcal{C}$. Then, given $f : X \nrightarrow Y$, we form the composite $g \bullet f : X \nrightarrow Z$ as $g^\triangleright \circ f \colon X \xrightarrow{f} TY \xrightarrow{Tg} TTZ \xrightarrow{\mu_Z} TZ$.

Proof. Observe that there is a bijection between the objects X of $\mathcal{C}$ and the free T-algebras (TX, μ_X). We therefore only need to establish a bijection between the hom-sets $\mathbf{Alg}(T)\big((TX, \mu_X), (TY, \mu_Y)\big)$ and $\mathcal{K\ell}(T)(X, Y)$, with the latter defined as in the statement of the proposition.

First, we demonstrate that Kleisli extension defines a surjection

$$\mathcal{K\ell}(T)(X, Y) \to \mathbf{Alg}(T)\big((TX, \mu_X), (TY, \mu_Y)\big) \ .$$

Suppose ϕ is any algebra morphism $(TX, \mu_X) \to (TY, \mu_Y)$; we show that it is equal to the Kleisli extension of the Kleisli morphism $X \xrightarrow{\eta_X} TX \xrightarrow{\phi} TY$:

$$
\begin{aligned}
TX \xrightarrow{(\phi \circ \eta_X)^\triangleright} TY &= TX \xrightarrow{T\eta_X} TTX \xrightarrow{T\phi} TTY \xrightarrow{\mu_{TY}} TY \\
&= TX \xrightarrow{T\eta_X} TTX \xrightarrow{\mu_{TX}} TX \xrightarrow{\phi} TY \\
&= TX \xRightarrow{\mathsf{id}_{TX}} TX \xrightarrow{\phi} TY \\
&= TX \xrightarrow{\phi} TY
\end{aligned}
$$

where the first equality holds by definition, the second line by naturality of μ, and the third by the unitality of the monad (T, μ, η). Hence every free algebra morphism is in the image of Kleisli extension, and so Kleisli extension defines a surjection.

Next, we show that this surjection is additionally injective. Suppose f, g are two Kleisli morphisms $X \to TY$ such that their Kleisli extensions are equal

$$TX \xrightarrow{Tf} TTY \xrightarrow{\mu_{TY}} TY = TX \xrightarrow{Tg} TTY \xrightarrow{\mu_{TY}} TY$$

and recall that the identity in $\mathcal{K}\ell(T)$ is η. We therefore have the following equalities:

$$
\begin{aligned}
X \xrightarrow{\eta_X} TX \xrightarrow{Tf} TTY \xrightarrow{\mu_{TY}} TY &= X \xrightarrow{\eta_X} TX \xrightarrow{Tg} TTY \xrightarrow{\mu_{TY}} TY \\
= X \xrightarrow{f} TY \xrightarrow{T\eta_y} TTY \xrightarrow{\mu_{TY}} TY &= X \xrightarrow{g} TY \xrightarrow{T\eta_y} TTY \xrightarrow{\mu_{TY}} TY \\
= X \xrightarrow{f} Y \qquad\qquad\qquad &= X \xrightarrow{g} Y \ .
\end{aligned}
$$

where the equality in the first line holds *ex hypothesi*, the second by naturality, and the third by monadic unitality. Since $f = g$ when their Kleisli extensions are equal, Kleisli extension is injective. Since it is also surjective, we have an isomorphism between $\mathcal{K}\ell(T)(X, Y)$ and $\mathbf{Alg}(T)\big((TX, \mu_X), (TY, \mu_Y)\big)$. Hence $\mathcal{K}\ell(T)$ is the subcategory of $\mathbf{Alg}(T)$ on the free algebras. $\square$

If T is a monad on $\mathcal{C}$, there is a canonical embedding of $\mathcal{C}$ into $\mathcal{K}\ell(T)$. In the case of $\mathcal{K}\ell(\mathcal{D})$, this will yield the subcategory of *deterministic* channels: those which do not add any uncertainty.

Proposition 4.1.15. Suppose T is a monad on $\mathcal{C}$. Then there is an identity-on-objects embedding $\mathcal{C} \hookrightarrow \mathcal{K}\ell(T)$ given on morphisms by mapping $f : X \to Y$ in $\mathcal{C}$ to the Kleisli morphism $X \xrightarrow{\eta_X} TX \xrightarrow{Tf} TY$.

Proof sketch. Functoriality follows from the unitality of η in the monad structure, since Kleisli composition involves post-composing the monad multiplication, and $\mu_T \circ T\eta = \mathsf{id}$. $\square$

4.1.1.1. Stochastic matrices

At this point, let us exhibit $\mathcal{K}\ell(\mathcal{D})$ a little more concretely, by instantiating Proposition 4.1.14. Since a distribution π on the set X is a function $X \to [0, 1]$, and following the "formal sum" intuition, we can alternatively think of π as a vector, whose coefficients are indexed by elements of X (the basis vectors $|x\rangle$). Morphisms $X \rightsquigarrow Y$ in $\mathcal{K}\ell(\mathcal{D})$ are functions $X \to \mathcal{D}Y$, and so we can similarly think of these as stochastic matrices, by the Cartesian closure of $\mathbf{Set}$: a function $X \to \mathcal{D}Y$ is equivalently

a function $X \to [0,1]^Y$, which in turn corresponds to a function $X \times Y \to [0,1]$, which we can read as a (left) stochastic matrix, with only finitely many nonzero coefficients and each of whose columns must sum to 1. We will adopt 'conditional probability' notation for the coefficients of these matrices: given $p : X \nrightarrow Y$, $x \in X$ and $y \in Y$, we write $p(y|x) := p(x)(y) \in [0,1]$ for "the probabilty of y *given* x, according to p".

Composition in $\mathcal{K}\ell(\mathcal{D})$ is then matrix multiplication: given $p : X \to \mathcal{D}Y$ and $q : Y \to \mathcal{D}Z$, we compute their composite $q \bullet p : X \to \mathcal{D}Z$ by 'averaging over' or 'marginalizing out' Y via the Chapman-Kolmogorov equation:

$$q \bullet p : X \to \mathcal{D}Z := x \mapsto \sum_{z:Z} \boxed{\sum_{y:Y} q(z|y) \cdot p(y|x)} \ |z\rangle .$$

Here we have again used the formal sum notation, drawing a box to indicate the coefficients (*i.e.*, the probabilities returned by the conditional distribution $q \bullet p(x)$ for each atomic event z in Z).

Via the monadic unit, identity morphisms $\mathrm{id}_X : X \nrightarrow X$ in $\mathcal{K}\ell(\mathcal{D})$ take points to 'Dirac delta' distributions: $\mathrm{id}_X := x \mapsto 1 \, |x\rangle$. The embedding $\mathbf{Set} \hookrightarrow \mathcal{K}\ell(\mathcal{D})$ makes any function $f : Y \to X$ into a (deterministic) channel $\overline{f} = \eta_X \circ f : Y \to \mathcal{D}X$ by post-composing with η_X.

4.1.1.2. Monoidal structure

We will want to equip $\mathcal{K}\ell(\mathcal{D})$ with a copy-discard category structure, in order to represent joint states (joint distributions) and their marginalization, as well as the copying of information. The first ingredient making a copy-discard category, after the category itself, is a monoidal structure. Once again, in the case of $\mathcal{K}\ell(\mathcal{D})$, this can be obtained abstractly from a more fundamental structure—the categorical product $(\times, 1)$ on $\mathbf{Set}$—as a consequence of $\mathcal{D}$ being a 'monoidal' monad. We will write the induced tensor product on $\mathcal{K}\ell(\mathcal{D})$ as $\otimes$; its monoidal unit remains the object 1.

Definition 4.1.16. A *monoidal monad* is a monad in $\mathbf{MonCat}$. This means that it is a monad (T, μ, η) in $\mathbf{Cat}$ whose functor $T : \mathcal{C} \to \mathcal{C}$ is additionally equipped with a lax monoidal structure (α, ϵ) such that the monad multiplication μ and unit η are monoidal natural transformations accordingly.

With this extra structure, it is not hard to verify that the following proposition makes $\mathcal{K}\ell(T)$ into a well-defined monoidal category.

Proposition 4.1.17. The Kleisli category $\mathcal{K}\ell(T)$ of a monoidal monad $(T, \alpha, \epsilon, \mu, \eta)$ is a monoidal category. The monoidal product is given on objects by the monoidal product $\otimes$ of the base category

$\mathcal{C}$. On Kleisli morphisms $f : X \nrightarrow Y$ and $f' : X' \nrightarrow Y'$, their tensor $f \otimes g$ is given by the following composite in $\mathcal{C}$:

$$X \otimes X' \xrightarrow{f \otimes f'} TX \otimes TX' \xrightarrow{\alpha_{X,X'}} T(X \otimes X')$$

The monoidal unit is the monoidal unit I in $\mathcal{C}$. The associator and unitor of the monoidal category structure are inherited from $\mathcal{C}$ under the embedding $\mathcal{C} \hookrightarrow \mathcal{K}\ell(T)$. When $(\mathcal{C}, \otimes, I)$ is symmetric monoidal, then so is $(\mathcal{K}\ell(T), \otimes, I)$.

In the specific case of $\mathcal{K}\ell(\mathcal{D})$, the tensor product $\otimes$ is given on objects by the product of sets and on stochastic channels $f : X \to \mathcal{D}A$ and $g : Y \to \mathcal{D}B$ as

$$X \times Y \xrightarrow{f \times g} \mathcal{D}A \times \mathcal{D}B \xrightarrow{\alpha_{A,B}} \mathcal{D}(A \times B) .$$

Note that because not all joint states have independent marginals, the monoidal product $\otimes$ is not Cartesian: that is, given an arbitrary $\omega : \mathcal{D}(X \otimes Y)$, we do not necessarily have $\omega = (\rho, \sigma)$ for some $\rho : \mathcal{D}X$ and $\sigma : \mathcal{D}Y$. The laxator takes a pair of distributions (ρ, σ) in $\mathcal{D}X \times \mathcal{D}Y$ to the joint distribution on $X \times Y$ given by $(x, y) \mapsto \rho(x) \cdot \sigma(y)$; ρ and σ are then the (independent) marginals of this joint distribution. (Of course, the joint distribution (ρ, σ) is not the only joint distribution with those marginals: other joint states may have these marginals but also correlations between them, and this is what it means for not all joint states to have independent marginals.)

Since $(\mathbf{Set}, \times, 1)$ is symmetric monoidal, $\mathcal{K}\ell(\mathcal{D})$ is too, with swap isomorphisms $\mathsf{swap}_{X,Y} : X \otimes Y \xrightarrow{\sim} Y \otimes X$ similarly inherited form those of $\times$.

4.1.1.3. Copy-discard structure

The copy-discard structure in $\mathcal{K}\ell(\mathcal{D})$ is inherited from $\mathbf{Set}$ through its embedding: since every object in $\mathcal{K}\ell(\mathcal{D})$ is an object in $\mathbf{Set}$, and every object in $\mathbf{Set}$ is a comonoid (Example 3.4.22), and since functors preserve equalities, these comonoid structures are preserved under the embedding. More explicitly, the discarding channels $\bar{\bar{\top}}_X$ are given by $x \mapsto 1 \ket{*}$, and the copiers $\math%Y_X$ by $x \mapsto 1 \ket{x, x}$. Note that the copiers are not natural in $\mathcal{K}\ell(\mathcal{D})$: in general, $\math%Y \bullet f \neq f \otimes f \bullet \math%Y$, as a result of the possibility of correlations.

Since the projections $\mathsf{proj}_X : X \times Y \to X$ in $\mathbf{Set}$ satisfy $\mathsf{proj}_X = \rho_X \circ (\mathrm{id}_X \times \bar{\bar{\top}}_Y)$ where $\rho_X : X \times 1 \to X$ is component of the right unitor, we can see how discarding and projection give us marginalization, thereby explaining the string diagrams of §3.1.1. Given some joint state $\omega : 1 \nrightarrow X \otimes Y$, its X-marginal $\omega_X : 1 \nrightarrow X$ is given by $\mathsf{proj}_X \bullet \omega$, which in $\mathcal{K}\ell(\mathcal{D})$ is given by the formal sum formula $\sum_{x:X} \boxed{\sum_{y:Y} \omega(x, y)} \ket{x}$, where we have again drawn a box to distinguish the

131

probability assigned to $|x\rangle$, which we note coincides with the classical rule for marginal discrete probability. (The Y-marginal is of course symmetric.)

Remark 4.1.18. A *semicartesian* category is a monoidal category in which the monoidal unit is terminal. In a semicartesian monoidal category, every tensor product $X \otimes Y$ is equipped with a natural family of projections $\mathsf{proj}_X : X \otimes Y \to X$ and $\mathsf{proj}_Y : X \otimes Y \to Y$ given by 'discarding' one of the factors and using the unitor; the existence of such projections is not otherwise implied by a monoidal structure (though of course it does follow when the tensor is the product).

A related notion is that of an *affine* functor, which is one that preserves the terminal object, and of which $\mathcal{D}$ is an example. As a result, and following the discussion above, we can see that $\mathcal{K}\ell(\mathcal{D})$ is an example of a semicartesian category.

Semicartesian copy-discard categories are also known as *Markov categories*, following Fritz [109].

Remark 4.1.19. Since 1 is therefore terminal in $\mathcal{K}\ell(\mathcal{D})$, Proposition 3.4.24 tells us that those channels f that do commute with copying (*i.e.*, for which Ψ is natural; Remark 3.4.26), and which are therefore comonoid morphisms, are precisely the *deterministic* channels: those in the image of the embedding of **Set** (and which therefore emit Dirac delta distributions). As a result, we can think of $\mathbf{Comon}(\mathcal{K}\ell(\mathcal{D}))$ as the subcategory of deterministic channels, and write $\mathbf{Comon}(\mathcal{K}\ell(\mathcal{D})) \cong \mathbf{Set}$. (Intuitively, this follows almost by definition: a deterministic process is one that has no informational side-effects; that is to say, whether we copy a state before performing the process on each copy, or perform the process and then copy the resulting state, or whether we perform the process and then marginalize, or just marginalize, makes no difference to the resulting state.)

4.1.1.4. Bayesian inversion

We can now instantiate Bayesian inversion in $\mathcal{K}\ell(\mathcal{D})$, formalizing Equation (4.3). Given a channel $p : X \to \mathcal{D}Y$ satisfying the condition in Remark 4.1.20 below, its Bayesian inversion is given by the function

$$p^\dagger : \mathcal{D}X \times Y \to \mathcal{D}X := (\pi, y) \mapsto \sum_{x:X} \boxed{\frac{p(y|x) \cdot \pi(x)}{\sum_{x':X} p(y|x') \cdot \pi(x')}} |x\rangle = \sum_{x:X} \boxed{\frac{p(y|x) \cdot \pi(x)}{(p \bullet \pi)(y)}} |x\rangle$$
$$(4.4)$$

so that the Bayesian update of p along π is the conditional distribution defined by

$$p^\dagger_\pi(x|y) = \frac{p(y|x) \cdot \pi(x)}{(p \bullet \pi)(y)} \quad .$$

Note that here we have used the Cartesian closure of **Set**, writing the type of $p^\dagger$ as $\mathcal{D}X \times Y \to \mathcal{D}X$ rather than $\mathcal{D}X \to \mathcal{K}\ell(\mathcal{D})(Y, X)$, where $\mathcal{K}\ell(\mathcal{D})(Y, X) = (\mathcal{D}X)^Y$.

Remark 4.1.20. In the form given above, $p^\dagger$ is only well-defined when the support of $p \bullet \pi$ is the whole of Y, so that, for all y, $(p \bullet \pi)(y) > 0$; otherwise, the division is ill-defined. Henceforth, in the context of Bayesian inversion, we will therefore assume that $p \bullet \pi$ has full support (see Definition 4.1.21).

To avoid this (rather ugly) condition, one can replace it by the assumption that the notion of 'support' is well-defined, and modify the type of $p^\dagger$ accordingly: this is the refinement made by Braithwaite and Hedges [41], and were it not for the presently-uncertain nature of support objects in general, it would now be this author's preferred approach. This leads to writing the type of the inversion $p^\dagger$ as $\sum_{\pi:\mathcal{D}X} \mathsf{supp}(p \bullet \pi) \to \mathcal{D}X$, where $\mathsf{supp}(p \bullet \pi)$ is the subobject of Y on which $p \bullet \pi$ is supported: with this type, $p_\pi^\dagger$ is always a well-defined channel. One can then proceed with the definition of 'dependent' Bayesian lenses accordingly; for the details, we refer the reader to Braithwaite and Hedges [41]. In this thesis, for simplicity of exposition and faithfulness to this author's earlier work, we will proceed under the full-support assumption.

4.1.2. Abstract Bayesian inversion

Beyond the concerns of Remark 4.1.20, in a more general setting it is not always possible to define Bayesian inversion using an equation like Equation (4.4) or Equation (4.3): the expression $p(y|x)$ might not be well-defined, or there might not be a well-defined notion of division. Instead being guided by Equation (4.3) in defining Bayesian inversion, we can use Equation (4.2). Therefore, supposing a channel $c : X \nrightarrow Y$ and a state $\pi : I \nrightarrow X$ in an ambient copy-discard category $\mathcal{C}$, we can ask for the Bayesian inversion $c_\pi^\dagger$ to be any channel satisfying the graphical equality [60, eq. 5]:

$$\tag{4.5}$$

This diagram can be interpreted as follows. Given a prior $\pi : I \nrightarrow X$ and a channel $c : X \nrightarrow Y$, we form the joint distribution $\omega := (\mathsf{id}_X \otimes c) \bullet \mathop{\blacktriangledown}_X \bullet \pi : I \nrightarrow X \otimes Y$ shown on the left hand side: this formalizes the product rule, $\mathrm{P}_\omega(A, B) = \mathrm{P}_c(B|A) \cdot \mathrm{P}_\pi(A)$, and π is the corresponding X-marginal.

As in the concrete case of $\mathcal{K}\ell(\mathcal{D})$, we seek an inverse channel $Y \rightsquigarrow X$ witnessing the 'dual' form of the rule, $\mathrm{P}_\omega(A, B) = \mathrm{P}(A|B) \cdot \mathrm{P}(B)$; this is depicted on the right hand side. By discarding X, we see that $c \bullet \pi : I \rightsquigarrow Y$ is the Y-marginal witnessing $\mathrm{P}(B)$. So any channel $c_\pi^\dagger : Y \rightsquigarrow X$ witnessing $\mathrm{P}(A|B)$ and satisfying the equality above is a Bayesian inverse of c with respect to π.

In light of Remark 4.1.20, we therefore make the following definition.

Definition 4.1.21. We say that a channel $c : X \rightsquigarrow Y$ *admits Bayesian inversion* with respect to $\pi : I \rightsquigarrow X$ if there exists a channel $c_\pi^\dagger : Y \rightsquigarrow X$ satisfying equation (4.5). We say that c admits Bayesian inversion *tout court* if c admits Bayesian inversion with respect to all states $\pi : I \rightsquigarrow X$.

Remark 4.1.22. We need to be careful about the existence of inversions as a consequence of the fact that $c \bullet \pi$ may not always be fully supported on Y (recall Remark 4.1.20). In this thesis we will henceforth assume that $c \bullet \pi$ *is* always fully supported, in order to keep the exposition clear. This is justified in two ways: first, because we can always restrict to a wide subcategory all of whose channels do admit inversion; and second, because we may equally work with dependent Bayesian lenses (as described by Braithwaite and Hedges [41] and noted in Remark 4.1.20).

4.1.3. Density functions

Abstract Bayesian inversion (4.5) generalizes the product rule form of Bayes' theorem (4.2) but in most applications, we are interested in a specific channel witnessing $\mathrm{P}(A|B) = \mathrm{P}(B|A) \cdot \mathrm{P}(A)/\mathrm{P}(B)$. In the typical measure-theoretic setting, this is often written informally as

$$p(x|y) = \frac{p(y|x) \cdot p(x)}{p(y)} = \frac{p(y|x) \cdot p(x)}{\int_{x':X} p(y|x') \cdot p(x') \, \mathrm{d}x'} \tag{4.6}$$

but the formal semantics of such an expression are not trivial: for instance, what is the object $p(y|x)$, and how does it relate to a channel $c : X \rightsquigarrow Y$?

Following Cho and Jacobs [60], we can interpret $p(y|x)$ as a *density function* for a channel, abstractly witnessed by an effect $X \otimes Y \rightsquigarrow I$ in our ambient category $\mathcal{C}$. Consequently, $\mathcal{C}$ cannot be semicartesian—as this would trivialize all density functions—though it must still supply comonoids. We can think of this as expanding the collection of channels in the category to include acausal or 'partial' maps and unnormalized distributions or states.

Example 4.1.23. An example of such a category is $\mathcal{K}\ell(\mathcal{D}_{\leqslant 1})$, whose objects are sets and whose morphisms $X \rightsquigarrow Y$ are functions $X \to \mathcal{D}(Y + 1)$. Then a stochastic map is partial if it sends any probability to the added element $*$, and the subcategory of total (equivalently, causal) maps is $\mathcal{K}\ell(\mathcal{D})$ (see [61] for more details).

A morphism $X \rightarrow 1$ in $\mathcal{K}\ell(\mathcal{D}_{\leqslant 1})$ is therefore a function $X \to \mathcal{D}(1 + 1)$. Now, a distribution π on $1 + 1$ is the same as a number $\bar{\pi}$ in $[0, 1]$: note that $1 + 1$ has two points, and so π assigns $\bar{\pi}$ to one of them and $1 - \bar{\pi}$ to the other. Therefore an effect $X \rightarrow 1$ is equivalently a function $X \to [0, 1]$, which is precisely the type we expect for a density function.

We therefore adopt the following abstract definition.

Definition 4.1.24 (Density functions [60, Def. 8.1]). A channel $c : X \rightarrow Y$ is said to be *represented by an effect* $p : X \otimes Y \rightarrow I$ with respect to $\mu : I \rightarrow Y$ if
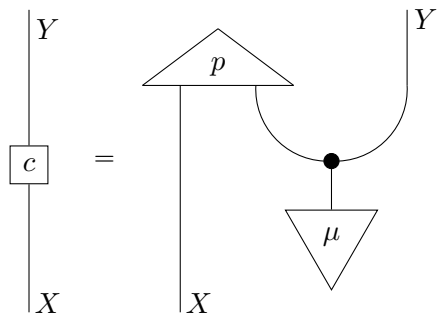
In this case, we call p a *density function* for c.

We will also need the concepts of almost-equality and almost-invertibility.

Definition 4.1.25 (Almost-equality, almost-invertibility [60, Def. 8.2]). Given a state $\pi : I \rightarrow X$, we say that two channels $c : X \rightarrow Y$ and $d : X \rightarrow Y$ are π-*almost-equal*, denoted $c \overset{\pi}{\sim} d$, if
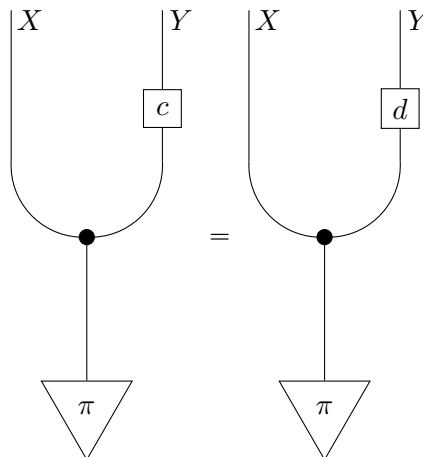
and we say that an effect $p : X \rightarrow I$ is π-*almost-invertible* with π-*almost-inverse* $q : X \rightarrow I$ if
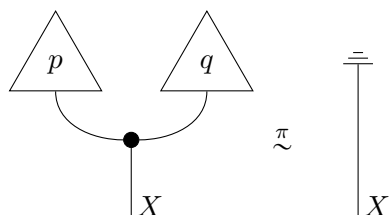
The following basic results about almost-equality will prove helpful.

Proposition 4.1.26 (Composition preserves almost-equality). *If $c \overset{\pi}{\sim} d$, then $f \bullet c \overset{\pi}{\sim} f \bullet d$.*

Proof. Immediate from the definition of almost-equality. □

Proposition 4.1.27 (Almost-inverses are almost-equal). *Suppose $q : X \nrightarrow I$ and $r : X \nrightarrow I$ are both π-almost-inverses for the effect $p : X \nrightarrow I$. Then $q \overset{\pi}{\sim} r$.*

Proof. By assumption, we have
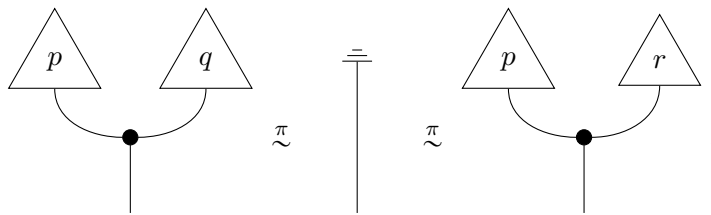
Then, by the definition of almost-equality (Definition 4.1.25):

(4.7)
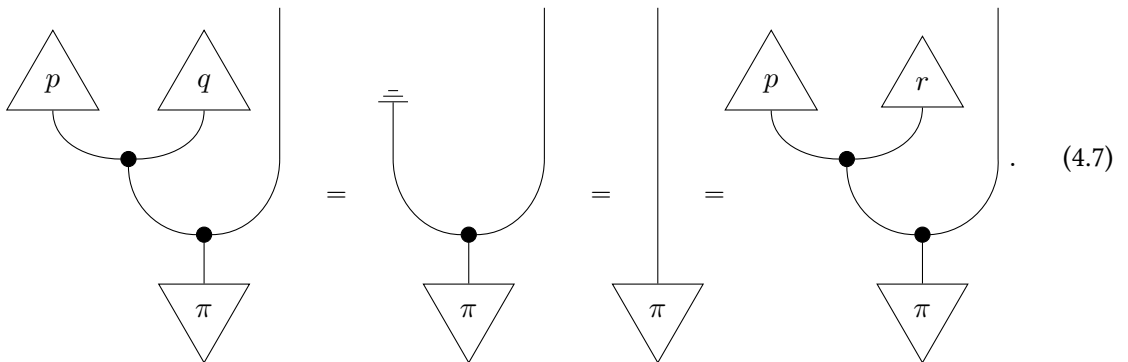
We seek to show that

(4.8)

136
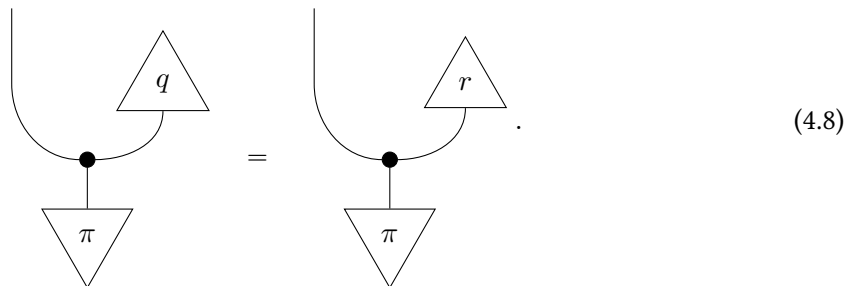
Substituting the right-hand-side of (4.7) for π in the left-hand-side of (4.8), we have that
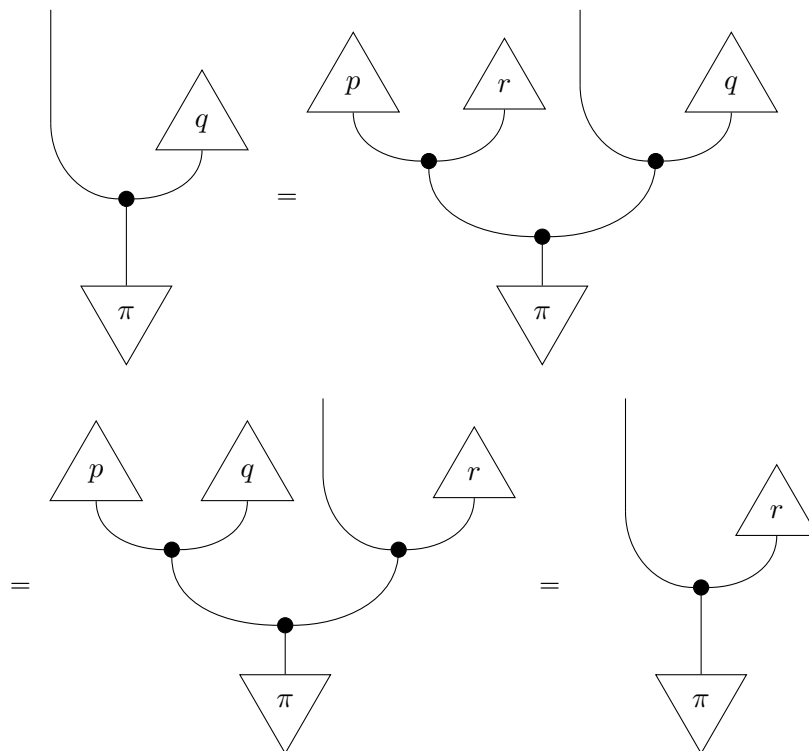
which establishes the result. The second equality follows by the coassociativity of $\curlyvee$ and the third by its counitality. □

With these notions, we can characterise Bayesian inversion via density functions. The result is due to Cho and Jacobs [60], but we include the graphical proof for expository completeness, as an example of string-diagrammatic reasoning.

Proposition 4.1.28 (Bayesian inversion via density functions [60, Thm. 8.3]). Suppose $c : X \rightarrow Y$ is represented by the effect p with respect to μ. The Bayesian inverse $c_\pi^\dagger : Y \rightarrow X$ of c with respect to $\pi : I \rightarrow X$ is given by

137
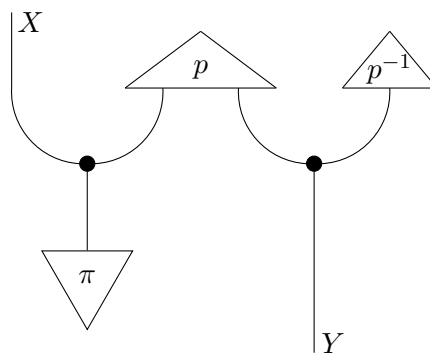
where $p^{-1} : Y \nrightarrow I$ is a μ-almost-inverse for the effect
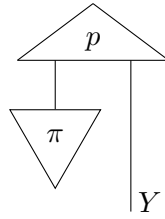
Proof. We seek to establish the relation (4.5) characterizing Bayesian inversion. By substituting the density function representations for c and $c_\pi^\dagger$ into the right-hand-side of (4.5), we have
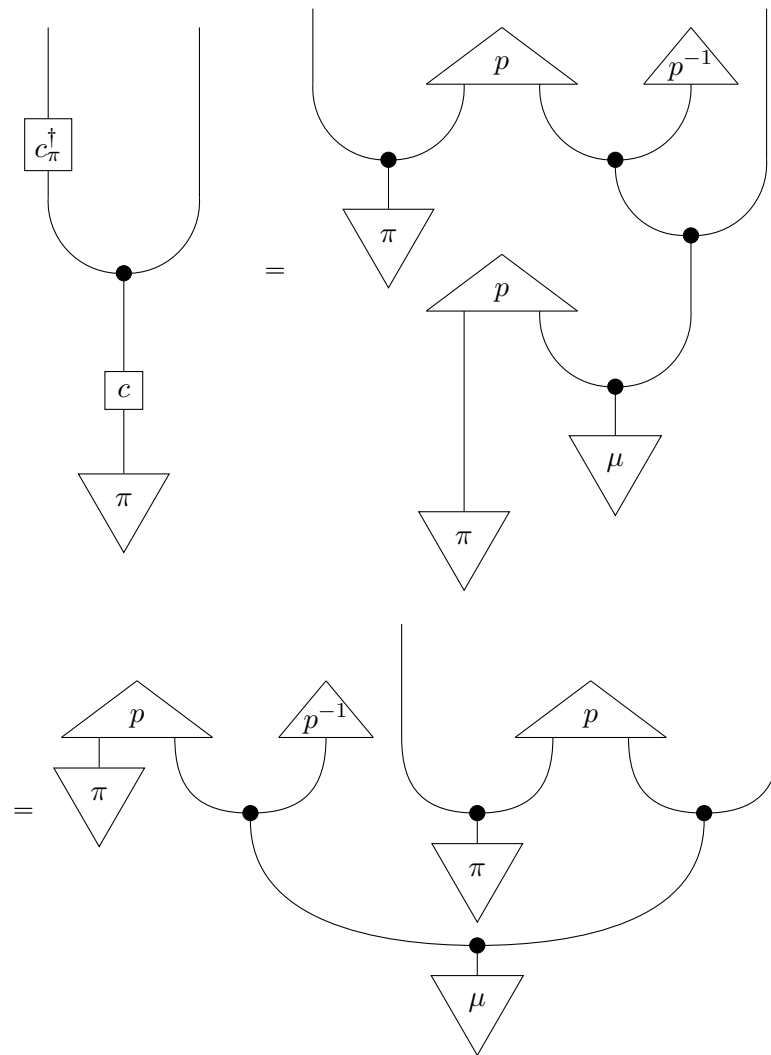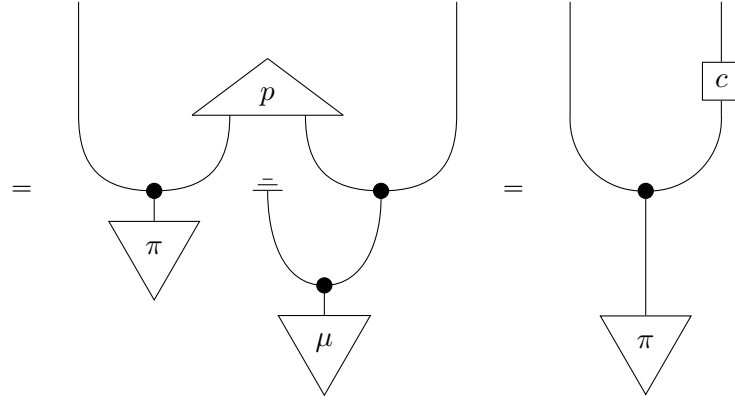
as required. The second equality holds by the coassociativity of $\curlyvee$, the third since p^{-1} is an almost-inverse *ex hypothesi*, and the fourth by the counitality of $(\curlyvee, \bar{\bar{\top}})$ and the density function representation of c. □

The following proposition is an immediate consequence of the definition of almost-equality and of the abstract characterisation of Bayesian inversion (4.5). We omit the proof.

Proposition 4.1.29 (Bayesian inverses are almost-equal). Suppose $\alpha : Y \nrightarrow X$ and $\beta : Y \nrightarrow X$ are both Bayesian inversions of the channel $c : X \nrightarrow Y$ with respect to $\pi : I \nrightarrow X$. Then $\alpha \overset{c \bullet \pi}{\sim} \beta$.

4.1.4. S-finite kernels

To represent channels by concrete density functions, we can work in the category **sfKrn** of measurable spaces and s-finite kernels. We will only sketch the structure of this category, and refer the reader to Cho and Jacobs [60] and Staton [258] for elaboration.

Objects in **sfKrn** are measurable spaces (X, Σ_X); often we will just write X, and leave the σ-algebra Σ_X implicit. Morphisms $(X, \Sigma_X) \nrightarrow (Y, \Sigma_Y)$ are s-finite kernels. A *kernel k* from X to Y is a function $k : X \times \Sigma_Y \to [0, \infty]$ satisfying the following conditions:

- for all $x \in X$, $k(x, -) : \Sigma_Y \to [0, \infty]$ is a measure; and

- for all $B \in \Sigma_Y$, $k(-, B) : X \to [0, \infty]$ is measurable.

A kernel $k : X \times \Sigma_Y \to [0, \infty]$ is *finite* if there exists some $r \in [0, \infty)$ such that, for all $x \in X$, $k(x, Y) \leq r$. And k is *s-finite* if it is the sum of at most countably many finite kernels k_n, $k = \sum_{n:\mathbb{N}} k_n$.

Identity morphisms $\mathsf{id}_X : X \nrightarrow X$ are Dirac kernels $\delta_X : X \times \Sigma_X \to [0, \infty] := x \times A \mapsto 1$ iff $x \in A$ and 0 otherwise. Composition is given by a Chapman-Kolmogorov equation, analogously to

composition in $\mathcal{K}\ell(\mathcal{D})$. Suppose $c : X \rightsquigarrow Y$ and $d : Y \rightsquigarrow Z$. Then

$$d \bullet c : X \times \Sigma_Z \to [0, \infty] := x \times C \mapsto \int_{y:Y} d(C|y)\, c(\mathrm{d}y|x)$$

where we have again used the 'conditional probability' notation $d(C|y) := d \circ (y \times C)$. Reading $d(C|y)$ from left to right, we can think of this notation as akin to reading the string diagrams from top to bottom, *i.e.* from output(s) to input(s).

Monoidal structure on sfKrn There is a monoidal structure on **sfKrn** analogous to that on $\mathcal{K}\ell(\mathcal{D})$. On objects, $X \otimes Y$ is the Cartesian product $X \times Y$ of measurable spaces. On morphisms, $f \otimes g : X \otimes Y \rightsquigarrow A \otimes B$ is given by

$$f \otimes g : (X \times Y) \times \Sigma_{A \times B} := (x \times y) \times E \mapsto \int_{a:A} \int_{b:B} \delta_{A \otimes B}(E|x, y)\, f(\mathrm{d}a|x)\, g(\mathrm{d}b|y)$$

where, as above, $\delta_{A \otimes B}(E|a, b) = 1$ iff $(a, b) \in E$ and 0 otherwise. Note that $(f \otimes g)(E|x, y) = (g \otimes f)(E|y, x)$ for all s-finite kernels (and all E, x and y), by the Fubini-Tonelli theorem for s-finite measures [60, 258], and so $\otimes$ is symmetric on **sfKrn**.

The monoidal unit in **sfKrn** is again $I = 1$, the singleton set. Unlike in $\mathcal{K}\ell(\mathcal{D})$, however, we do have nontrivial effects $p : X \rightsquigarrow I$, given by kernels $p : (X \times \Sigma_1) \cong X \to [0, \infty]$, with which we will represent density functions.

Comonoids in sfKrn Every object in **sfKrn** is a comonoid, analogously to $\mathcal{K}\ell(\mathcal{D})$. Discarding is given by the family of effects $\bar{\bar{\top}}_X : X \to [0, \infty] := x \mapsto 1$, and copying is again Dirac-like: $\bar{\mathbf{Y}}_X : X \times \Sigma_{X \times X} := x \times E \mapsto 1$ iff $(x, x) \in E$ and 0 otherwise. Because we have nontrivial effects, discarding is only natural for causal or 'total' channels: if c satisfies $\bar{\bar{\top}} \bullet c = \bar{\bar{\top}}$, then $c(-|x)$ is a probability measure for all x in the domain[1]. And, once again, copying is natural (that is, $\mathbf{Y} \bullet c = (c \otimes c) \bullet \mathbf{Y}$) if and only if the channel is deterministic.

Channels represented by effects We can interpret the string diagrams of §4.1.3 in **sfKrn**, and we will do so by following the intuition of the conditional probability notation and reading the string diagrams from outputs to inputs. Hence, if $c : X \rightsquigarrow Y$ is represented by the effect $p : X \otimes Y \rightsquigarrow I$ with respect to the measure $\mu : I \rightsquigarrow Y$, then

$$c : X \times \Sigma_Y \to [0, \infty] := x \times B \mapsto \int_{y:B} \mu(\mathrm{d}y)\, p(y|x).$$

---

[1] This means that the subcategory of total maps in **sfKrn** is equivalent to the Kleisli category $\mathcal{K}\ell(\mathcal{G})$ of the *Giry monad* $\mathcal{G}$ taking each measurable space X to the space $\mathcal{G}X$ of measures over X; see Example 4.1.30 for more details.

Note that we also use conditional probability notation for density functions, and so $p(y|x) := p \circ (x \times y)$.

Suppose that $c : X \nrightarrow Y$ is indeed represented by p with respect to μ, and that $d : Y \nrightarrow Z$ is represented by $q : Y \otimes Z \nrightarrow I$ with respect to $\nu : I \nrightarrow Z$. Then in **sfKrn**, $d \bullet c : X \nrightarrow Z$ is given by

$$d \bullet c : X \times \Sigma_Z := x \times C \mapsto \int_{z:C} \nu(\mathrm{d}z) \int_{y:Y} q(z|y) \, \mu(\mathrm{d}y) \, p(y|x)$$

Alternatively, by defining the effect $(p\mu q) : X \otimes Z \nrightarrow I$ as

$$(p\mu q) : X \times Z \to [0, \infty] := x \times z \mapsto \int_{y:Y} q(z|y) \, \mu(\mathrm{d}y) \, p(y|x),$$

we can write $d \bullet c$ as

$$d \bullet c : X \times \Sigma_Z := x \times C \mapsto \int_{z:C} \nu(\mathrm{d}z) \, (p\mu q)(z|x).$$

Bayesian inversion via density functions   Once again writing $\pi : I \nrightarrow X$ for a prior on X, and interpreting the string diagram of Proposition 4.1.28 for $c_\pi^\dagger : Y \nrightarrow X$ in **sfKrn**, we have

$$
\begin{aligned}
c_\pi^\dagger : Y \times \Sigma_X \to [0, \infty] := y \times A \mapsto & \left(\int_{x:A} \pi(\mathrm{d}x) \, p(y|x) \right) p^{-1}(y) \\
& = p^{-1}(y) \int_{x:A} p(y|x) \, \pi(\mathrm{d}x),
\end{aligned}
\tag{4.9}
$$

where $p^{-1} : Y \nrightarrow I$ is a μ-almost-inverse for effect $p \bullet (\pi \otimes \mathrm{id}_Y)$, and is given up to μ-almost-equality by

$$p^{-1} : Y \to [0, \infty] := y \mapsto \left(\int_{x:X} p(y|x) \, \pi(\mathrm{d}x) \right)^{-1}.$$

Note that from this we recover the informal form of Bayes' rule for measurable spaces (4.6). Suppose π is itself represented by a density function p_π with respect to the Lebesgue measure $\mathrm{d}x$. Then

$$c_\pi^\dagger(A|y) = \int_{x:A} \frac{p(y|x) \, p_\pi(x)}{\int_{x':X} p(y|x') \, p_\pi(x') \, \mathrm{d}x'} \, \mathrm{d}x.$$

4.1.5. On probability monads

Later, it will at times be helpful to work in a category of stochastic channels that is the Kleisli category for a monad, without fixing that monad in advance; in this case we will speak of a *probability monad*. Unfortunately, an abstract characterization of probability monads is not presently known to the author, and so we use this term informally. However, when we do so, we have in mind a monoidal monad that maps spaces to spaces of measures or valuations on them, and that maps morphisms to the corresponding pushforwards. In the setting of finitary probability, we have already seen one example, the monad $\mathcal{D}$ explored in §4.1.1. Here we note the existence of others.

Example 4.1.30 (Giry monad [120]). Let **Meas** denote the category of measurable spaces, whose objects are sets equipped with σ-algebras and whose morphisms are measurable functions. The *Giry monad* $\mathcal{G}$: **Meas** $\to$ **Meas** maps each measurable space (X, Σ_X) to the space $\mathcal{G}X$ of probability measures $\alpha : \Sigma_X \to [0, 1]$ over it, equipped with the smallest σ-algebra making the evaluation functions

$$\mathsf{ev}_U : \mathcal{G}X \to [0, 1]$$
$$\alpha \mapsto \alpha(U)$$

measurable for all $U \in \Sigma_X$. Given a measurable function $f : X \to Y$, the function $\mathcal{G}f : \mathcal{G}X \to \mathcal{G}Y$ is defined by pushforwards: that is, for each $\alpha : \mathcal{G}X$, we define

$$\mathcal{G}f(\alpha) : \Sigma_Y \to [0, 1]$$
$$V \mapsto \alpha\big(f^{-1}(V)\big) \,.$$

(We may also write $f_*\alpha$ to denote $\mathcal{G}f(\alpha)$.) The unit of the monad η has components $\eta_X : X \to \mathcal{G}X$ mapping each point x to the corresponding Dirac measure δ_x, which is defined by $\delta_x(U) = 1$ iff $x \in U$ and $\delta_x(U) = 0$ otherwise. Finally, the multiplication μ has components $\mu_X : \mathcal{G}\mathcal{G}X \to \mathcal{G}X$ defined by integration, analogous to the 'evaluation' of $\mathcal{D}$ (Def. 4.1.6): for each $\nu : \mathcal{G}\mathcal{G}X$, define

$$\mu_X(\nu) : \Sigma_X \to [0, 1]$$
$$U \mapsto \int_{\alpha:\mathcal{G}X} \alpha(U) \, \mathrm{d}\nu \,.$$

Note that the subcategory of total morphisms in **sfKrn** is equivalent to $\mathcal{K}\ell(\mathcal{G})$.

The category **Meas** has all finite limits (it has products and equalizers), and this will mean that we will be able in Chapter 6 to define "effectful polynomials" in $\mathcal{K}\ell(\mathcal{G})$, and hence obtain categories of continuous-time continuous-space open Markov processes. However, **Meas** does not have exponentials and is therefore not Cartesian closed, because the evaluation function $\mathsf{ev}_{\mathbb{R},\mathbb{R}} : \mathbf{Meas}(\mathbb{R}, \mathbb{R}) \times \mathbb{R} \to \mathbb{R} : (f, x) \mapsto f(x)$ is not measurable, for any choice of σ-algebra on the function space $\mathbf{Meas}(\mathbb{R}, \mathbb{R})$ [15]. This means that $\mathcal{K}\ell(\mathcal{G})$ cannot be enriched in **Meas**, and so we cannot define Bayesian lenses internally to **Meas**. Circumnavigating this obstruction would complicate our construction of cilia — dynamical systems that control lenses — which are central to our formalization of predictive coding. This is because the output maps of stochastic dynamical systems are deterministic functions: in the case of systems in $\mathcal{K}\ell(\mathcal{G})$, this means they are morphisms in **Meas**; for a general probability monad $\mathcal{P} : \mathcal{E} \to \mathcal{E}$, they are morphisms in $\mathcal{E}$.

For a system to be able to emit a lens, therefore, the hom objects of **BayesLens** must be objects in $\mathcal{E}$, and this in turn requires $\mathcal{Kl}(\mathcal{P})$ to be enriched in $\mathcal{E}$. Fortunately, as the following example notes, a suitable probability monad does exist.

Example 4.1.31 (Quasi-Borel spaces [131]). A *quasi-Borel space* is a set X equipped with a set M_X of 'random variables' on X taking samples from the real line, $M_X \subset X^{\mathbb{R}}$. The set M_X is taken to satisfy three closure properties: (i) M_X contains all constant functions; (ii) M_X is closed under composition with measurable functions, such that if $\rho \in M_X$ and $f : \mathbb{R} \to \mathbb{R}$ is measurable with respect to the standard Borel structure on $\mathbb{R}$, then $\rho \circ f \in M_X$; and (iii) M_X is closed under gluing 'disjoint Borel domains', meaning that if $\mathbb{R}$ is countably partitioned by $\mathbb{R} \cong \sum_{i:\mathbb{N}} S_i$, and if $\{\alpha_i\}_{i:\mathbb{N}} \subset M_X$, then the function $(x \in S_i) \mapsto \alpha_i(x)$ is in M_X. A function $f : X \to Y$ is a morphism of quasi-Borel spaces if for all $\rho \in M_X$, $f \circ \rho \in M_Y$. Quasi-Borel spaces and their morphisms form a category, **QBS**, and this category is Cartesian closed: if X and Y are quasi-Borel spaces, then $\mathbf{QBS}(X, Y)$ can be given a quasi-Borel structure M_{XY} by defining

$$M_{XY} := \{\rho : \mathbb{R} \to \mathbf{QBS}(X, Y) \mid (\rho^\flat : \mathbb{R} \times X \to Y) \in \mathbf{QBS}(\mathbb{R} \times X, Y)\} .$$

A probability measure on a quasi-Borel space X is defined to be a pair of a (standard) probability measure ν on $\mathbb{R}$ and a random variable $\rho \in M_X$. Since two different pairs (ν, ρ) and (μ, τ) may produce equal pushforward measures, $\rho_* \nu = \tau_* \mu$, it makes sense to consider two such QBS measures to be equivalent if their pushforwards are equal. The set $\mathcal{P}X$ of such equivalence classes of QBS measures on X can then be equipped with the structure of a quasi-Borel space, and the assignment $\mathcal{P}$ is made functorial by the pushforwards action. Finally, the functor $\mathcal{P} : \mathbf{QBS} \to \mathbf{QBS}$ can be equipped with the structure of a (monoidal) monad in a manner analogous to the Giry monad: the unit yields Dirac measures, and the multiplication acts by integration.

We end this section by noting that the notions of s-finite measure and s-finite kernel can be reconstructed within **QBS**, so that we may interpret **sfKrn** to be enriched accordingly [270, §11]. Moreover, Vákár and Ong [270] show that the set TX of s-finite measures on X can be given a quasi-Borel structure, and this assignment actually yields a monad $T : \mathbf{QBS} \to \mathbf{QBS}$ (by analogy with the 'continuation' monad). This licenses us to take **sfKrn** to be instead defined as $\mathcal{Kl}(T)$.

For further examples of probability monads, we refer the reader to Jacobs [139].

4.2. Dependent data and bidirectional processes

Two properties of Bayesian inversion are particularly notable. Firstly, given a channel $X \rightarrow Y$, its inversion yields a channel in the opposite direction, $Y \rightarrow X$. Secondly, this inverse channel does not exist in isolation, but rather depends on a supplied 'prior' distribution. In Chapter 7 we will want to assign functorially to stochastic channels dynamical systems that invert them, and to do this requires understanding how inversions compose. The general pattern for the composition of dependent bidirectional processes is called the *lens* pattern, and this section is dedicated to introducing it. The more fundamental aspect is that of dependence, which we began to explore in the context of dependent sums and products in Chapter 2: we therefore begin this section by introducing the *Grothendieck construction*, a 'fibrational' framework for composing dependent processes.

4.2.1. Indexed categories and the Grothendieck construction

At various point above, we have encountered 'dependent' objects and morphisms: indexed and dependent sums (Remark 2.3.10); indexed products (Remark 2.3.20); dependent products (§2.3.4.1); hints at dependent type theory (end of §2.3.4); parameterized morphisms (§3.2.2); circuit algebras (§3.3); and, of course, Bayesian inversions. The Grothendieck construction classifies each of these as examples of a common pattern, allowing us to translate between 'indexed' and 'fibrational' perspectives: from the indexed perspective, we consider functors from an indexing object into a category (think of diagrams); from the fibrational perspective, we consider bundles as projection maps. The correspondence is then, roughly speaking, between "the object indexed by i" and "the subobject that projects to i", which is called the 'fibre' of the bundle over i.

For this reason, categories of bundles are an important part of the story, from which much else is generalized. Recall from Definition 3.2.10 that these categories of bundles are slice categories: the category of bundles over B in $\mathcal{C}$ is the slice $\mathcal{C}/B$, whose objects are pairs (E, p) of an object E and a morphism $p : E \to B$; and whose morphisms $(E, p) \to (E', p')$ are morphisms $\alpha : E \to E'$ of $\mathcal{C}$ such that $p = p' \circ \alpha$. We call this the category of bundles over B as a generalization of the notion of "fibre bundle", from which we inherit the notion of 'fibre'.

Definition 4.2.1. Suppose $\mathcal{C}$ is a category with finite limits. Given a bundle $p : E \to B$ in $\mathcal{C}$, its *fibre over* $b : B$ is the subobject E_b of E such that $p(e) = b$ for all $e : E_b$. The fibre E_b can be

characterized as the following pullback object, where 1 is the terminal object in $\mathcal{C}$:

$$
\begin{array}{ccc}
E_b & \longrightarrow & E \\
\downarrow & \lrcorner & \downarrow p \\
1 & \underset{b}{\longrightarrow} & B
\end{array}
$$

In the case where $\mathcal{C} = \mathbf{Set}$, there is an equivalence between the slice $\mathbf{Set}\,/B$ and a certain presheaf category: the category of B-diagrams in $\mathbf{Set}$, which we can equivalently think of as the category of B-indexed sets.

Definition 4.2.2. Suppose B is a set. The *discrete category* on X is the category whose objects are the elements of B and whose only morphisms are identity morphisms $\mathrm{id}_b : b \to b$ for each element $b : B$. We will denote the discrete category on B simply by B.

Proposition 4.2.3. For each set B, there is an equivalence $\mathbf{Set}\,/B \cong \mathbf{Set}^B$.

Proof. In the direction $\mathbf{Set}\,/B \to \mathbf{Set}^B$, let $p : E \to B$ be a bundle over B. We construct a functor $P : B \to \mathbf{Set}$ by defining $P(b) := E_b$, where E_b is the fibre of p over b; there are no nontrivial morphisms in B, so we are done. Now suppose $f : (E, p) \to (F, q)$ is a morphism of bundles. A natural transformation $\varphi : P \Rightarrow Q$ in $\mathbf{Set}^B$ is just a family of functions $\varphi_b : Pb \to Qb$ indexed by b. Hence, given f, we define φ_b as the restriction of f to E_b for each $b : B$.

In the direction $\mathbf{Set}^B \to \mathbf{Set}\,/B$, let $P : B \to \mathbf{Set}$ be a functor. We define E as the coproduct $\sum_{b:B} P(b)$, and the bundle $p : E \to B$ as the projection $(b, x) \mapsto b$ for every (b, x) in $\sum_{b:B} P(b)$. Now suppose $\varphi : P \Rightarrow Q$ is a natural transformation in $\mathbf{Set}^B$. We define the function $f : (E, p) \to (F, q)$ by the coproduct of the functions φ_b, as $f := \sum_{b:B} \varphi_b$.

These two constructions are easily verified as mutually inverse. $\qquad\square$

If the B in $\mathbf{Set}^B$ is not just a set, but rather a category, then there is a correspondingly categorified notion of the category of bundles.

Definition 4.2.4. Suppose $F : \mathcal{C} \to \mathbf{Set}$ is a copresheaf on $\mathcal{C}$. Its *category of elements* $\mathcal{C}/F$ has for objects pairs (X, x) of an object $X : \mathcal{C}$ and an element $x : FX$. A morphism $(X, x) \to (Y, y)$ is a morphism $f : X \to Y$ in $\mathcal{C}$ such that $Ff(x) = y$, as in the following diagram, where the top triangle commutes in $\mathbf{Set}$:

$$
\begin{array}{ccc}
 & 1 & \\
{}^{x}\swarrow & & \searrow^{y} \\
FX & \underset{Ff}{\longrightarrow} & FY
\end{array}
\quad.
$$

$$
X \xrightarrow{\;f\;} Y
$$

Identities are given by identity morphisms in $\mathcal{C}$, and composition is composition of the underlying morphisms in $\mathcal{C}$. There is an evident forgetful functor $\pi_F : \mathcal{C}/F \to \mathcal{C}$, which acts on objects as $(X, x) \mapsto X$ and on morphisms as $f \mapsto f$.

To validate that the category of elements construction is a good generalization of the slice category, we have the following example.

Example 4.2.5. The category of elements of a representable copresheaf $\mathcal{C}(C, -)$ is equivalent to the slice category $\mathcal{C}/C$, from which we derive the similar notation.

Remark 4.2.6. Another way to look at the morphisms $(X, x) \to (Y, y)$ in $\mathcal{C}/F$ is as *pairs* (f, ι), where f is a morphism $X \to Y$ in $\mathcal{C}$ and ι is an identification $Ff(x) = y$. Then composition in $\mathcal{C}/F$ is not just composition of morphisms in $\mathcal{C}$, but also composition of identifications: given $(f, \iota) : (X, x) \to (Y, y)$ and $(g, \kappa) : (Y, y) \to (Z, z)$, the composite $(g, \kappa) \circ (f, \iota)$ is $(g \circ f, \kappa \circ Fg(\iota))$, where $\kappa \circ Fg(\iota)$ is the composite identification $F(g \circ f)(x) \xeq{Fg(\iota)} Fg(y) \xeq{\kappa} z$. We can think of these identifications as witnesses to the required equalities. This perspective on $\mathcal{C}/F$ is analogous to the process of categorification we considered in Chapter 2, where we added witnesses (fillers) to equations and diagrams.

A better way to validate the category of elements construction is to generalize the Grothendieck correspondence, Proposition 4.2.3, which means we need something to correspond to $\mathbf{Set}^{\mathcal{B}}$: a category of categories of elements. These generalized categories of elements are called "discrete opfibrations", and constitute our first examples of categorified bundles.

Definition 4.2.7. A *discrete opfibration* is a functor $F : \mathcal{E} \to \mathcal{B}$ such that, for every object E in $\mathcal{E}$ and morphism $g : FE \to B$ in $\mathcal{B}$, there exists a unique morphism $h : E \to E'$ in $\mathcal{E}$ such that $Fh = g$ (called the *lift* of g):

$$
\begin{array}{ccc}
E & \dashrightarrow{\ h\ } & E' \\
\Big\vdots & & \\
FE & \xrightarrow{\ g\ } & B
\end{array}
$$

Write $\mathbf{DOpfib}(\mathcal{B})$ to denote the full subcategory of $\mathbf{Cat}/\mathcal{B}$ on those objects which are discrete opfibrations. The subcategory $\mathcal{E}_B$ of $\mathcal{E}$ all of whose objects are mapped by F to B and all of whose morphisms are mapped to id_B is called the *fibre* of F over B.

Example 4.2.8. The forgetful functor $\pi_F : \mathcal{C}/F \to \mathcal{C}$ out of the category of elements of a copresheaf F is a discrete opfibration: for any object (X, x) in $\mathcal{C}/F$ and morphism $g : X \to Y$ in $\mathcal{C}$, there is a unique morphism $g : (X, x) \to (Y, y)$, namely where $y = Ff(x)$.

And thus we obtain a Grothendieck correspondence at the next level of categorification.

Proposition 4.2.9. For any category $\mathcal{B}$, there is an equivalence $\mathbf{DOpfib}(\mathcal{B}) \cong \mathbf{Set}^{\mathcal{B}}$.

Proof sketch. We only sketch the bijection on objects; the correspondence on morphisms subsequently follows quite mechanically.

Given a discrete opfibration $p : \mathcal{E} \to \mathcal{B}$, it is easy to check that each fibre E_b is a discrete category and hence a set. Given a morphism $f : b \to c$ in $\mathcal{B}$, we define a function $\varphi : E_b \to E_c$ by mapping each $e : E_b$ to the codomain of the unique lift h. This defines a functor $\mathcal{B} \to \mathbf{Set}$; functoriality follows from the uniqueness of lifts.

In the inverse direction, given a copresheaf $F : \mathcal{B} \to \mathbf{Set}$, take the forgetful functor $\pi_F : \mathcal{B}/F \to \mathcal{B}$ out of its category of elements, which is a discrete opfibration by the example above. Given a natural transformation $\sigma : F \Rightarrow G$, define a functor $S : \mathcal{B}/F \to \mathcal{B}/G$ on objects as $S(X, x) = (X, \sigma_X(x))$ and on morphisms $f : (X, x) \to (Y, y)$ as $Sf = (X, \sigma_X(x)) \xrightarrow{f} (Y, \sigma_Y(y))$; this is well-defined by the naturality of σ and the definition of f, since $Gf \circ \sigma_X(x) = \sigma_Y \circ Ff(x)$ and $Ff(x) = y$.

The verification that these two constructions are mutually inverse is straightforward. $\qquad\square$

In many cases, the dependent data of interest will have more structure than that of mere sets. For example, in §3.3 we introduced a rate-coded circuit diagrams as an indexing of sets of rate-coded circuits by a category of circuit diagrams; later, we will see that dynamical systems have a canonical notion of morphism, and so our dynamical semantics will take the form of an indexed collection of categories. This requires us to categorify not only the domain of indexing (as we have done above), but also the codomain of values (as we do now). As with monoidal categories—and as in the case of circuit algebras—in this higher-dimensional setting, it becomes necessary to work with weak composition, and the relevant notion of weak functor is the 'pseudofunctor'.

Definition 4.2.10. Suppose $\mathcal{C}$ is a category and $\mathcal{B}$ is a bicategory. A *pseudofunctor* $F : \mathcal{C} \to \mathcal{B}$ is constituted by

1. a function $F_0 : \mathcal{C}_0 \to \mathcal{B}_0$ on objects;

2. for each pair of objects $a, b : \mathcal{C}$, a function $F_{a,b} : \mathcal{C}(a, b) \to \mathcal{B}(F_0 a, F_0 b)_0$ on morphisms;

3. for each object $c : \mathcal{C}$, a 2-isomorphism $F_{\mathsf{id}_c} : \mathsf{id}_{F_0 c} \Rightarrow F_{c,c}(\mathsf{id}_c)$ witnessing *weak unity*, natural in c; and

4. for each composable pair of morphisms $f : a \to b$ and $g : b \to c$ in $\mathcal{C}$, a 2-isomorphism $F_{g,f} : F_{b,c}(g) \diamond F_{a,b}(f) \Rightarrow F_{a,c}(g \circ f)$ witnessing *weak functoriality*, natural in f and g, where we have written composition in $\mathcal{C}$ as $\circ$ and horizontal composition in $\mathcal{B}$ as $\diamond$;

satisfying the following conditions:

(a) coherence with left and right unitality of horizontal composition, so that the respective diagrams of 2-cells commute:

$$
\begin{array}{ccc}
\mathsf{id}_{F_0 b} \diamond F_{a,b}(f) & \xrightarrow{\lambda_{F_{a,b}(f)}} & F_{a,b}(f) \\
\Big\downarrow{\scriptstyle F_{\mathsf{id}_b} \diamond F_{a,b}(f)} & & \Big\| \\
F_{b,b}(\mathsf{id}_b) \diamond F_{a,b}(f) & \xrightarrow{F_{\mathsf{id}_b, f}} & F_{a,b}(\mathsf{id}_b \circ f)
\end{array}
\qquad
\begin{array}{ccc}
F_{a,b}(f) \diamond \mathsf{id}_{F_0 a} & \xrightarrow{\rho_{F_{a,b}(f)}} & F_{a,b}(f) \\
\Big\downarrow{\scriptstyle F_{a,b}(f) \diamond F_{\mathsf{id}_a}} & & \Big\| \\
F_{a,b}(f) \diamond F_{a,a}(\mathsf{id}_a) & \xrightarrow{F_{f,\mathsf{id}_a}} & F_{a,b}(f \circ \mathsf{id}_a)
\end{array}
$$

(b) coherence with associativity of horizontal composition, so that the following diagram of 2-cells commutes:

$$
\begin{array}{ccc}
(F_{c,d}(h) \diamond F_{b,c}(g)) \diamond F_{a,b}(f) & \xrightarrow{\alpha_{F_{c,d}(h), F_{b,c}(g), F_{a,b}(f)}} & F_{c,d}(h) \diamond (F_{b,c}(g) \diamond F_{a,b}(f)) \\
\Big\downarrow{\scriptstyle F_{h,g} \diamond F_{a,b}(f)} & & \Big\downarrow{\scriptstyle F_{c,d}(h) \diamond F_{g,f}} \\
F_{b,d}(h \circ g) \diamond F_{a,b}(f) & & F_{c,d}(h) \diamond F_{a,c}(g \circ f) \\
\Big\downarrow{\scriptstyle F_{h \circ g, f}} & & \Big\downarrow{\scriptstyle F_{h, g \circ f}} \\
F_{a,d}((h \circ g) \circ f) & =\!=\!=\!=\!=\!= & F_{a,d}(h \circ (g \circ f))
\end{array}
\qquad .
$$

Remark 4.2.11. If $\mathcal{C}$ is in fact a nontrivial bicategory, then the definition of pseudofunctor is weakened accordingly: the functions $F_{a,b}$ are replaced by functors between the corresponding hom-categories, and the equalities in the functoriality conditions (a) and (b) are replaced by the relevant unitor or associator isomorphism. We will encounter this more general case in the next chapter, where we introduce the (yet weaker) concept of *lax functor*: see Definition 5.2.8, and the associated footnote 5 for the relationship with the present notion of pseudofunctor.

With pseudofunctors, we gain a notion of indexed category.

Definition 4.2.12. An *indexed category* is a pseudofunctor $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$, for some indexing category $\mathcal{C}$. An *opindexed category* is a pseudofunctor $F : \mathcal{C} \to \mathbf{Cat}$. Given an (op)indexed category F, we call the categories Fc its *fibres*, for each object $c : \mathcal{C}$.

Working with indexed categories rather than indexed sets, the relevant notion of (op)fibration is no longer discrete, as there are now (non-trivial) morphisms to account for. Following the Grothendieck logic, fibrations $p : \mathcal{E} \to \mathcal{B}$ should be in bijective correspondence with indexed categories $F : \mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$. This means that we should be able to turn any indexed category into a fibration by appropriately gluing together its fibres; and conversely, given a fibration p, the assignment $b \mapsto \mathcal{E}_b{}^2$ should define a pseudofunctor $\mathcal{B}^{\mathrm{op}} \to \mathbf{Cat}$. These considerations yield the following definition.

Definition 4.2.13. A *fibration* is a functor $p : \mathcal{E} \to \mathcal{B}$ such that, for every pair of morphisms $f : E' \to E$ and $\phi : E'' \to E$ in $\mathcal{E}$, and for every morphism $g : p(E'') \to p(E')$ such that $p(\phi) = p(f) \circ g$ in $\mathcal{B}$, there exists a unique morphism $h : E'' \to E'$ in $\mathcal{E}$ such that $p(h) = g$ and $\phi = f \circ h$:

The subcategory $\mathcal{E}_B$ of all those objects mapped by p to $B : \mathcal{B}$ and all those morphisms mapped to id_B is called the *fibre* of p over B. An *opfibration* is a functor $p : \mathcal{E} \to \mathcal{B}$ for which $p^{\mathrm{op}} : \mathcal{E}^{\mathrm{op}} \to \mathcal{B}^{\mathrm{op}}$ is a fibration.

Remark 4.2.14. Note that a discrete (op)fibration is a(n) (op)fibration each of whose fibres is a discrete category: this means that in each fibre, there are no non-identity morphisms, so that morphisms f and ϕ in the definition above are trivialized, thereby recovering the form of Definition 4.2.7.

The Grothendieck construction then tells us how to translate from (op)indexed categories to (op)fibrations: in some situations, it will be easier to work with the one, and in others the other. In particular, categories of lenses (and polynomial functors) will be seen to arise as Grothendieck constructions.

Definition 4.2.15. Suppose $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ is a pseudofunctor. Its *(contravariant) Grothendieck construction* is the category $\int F$ defined as follows. The objects of $\int F$ are pairs (X, x) of an object

---

[2]with $\mathcal{E}_b$ being the subcategory of $\mathcal{E}$ sometimes denoted $p^{-1}(b)$ all of whose objects are mapped by p to b, as in the proof of Proposition 4.2.9.

$X : \mathcal{C}$ and an object $x : FX$. A morphism $(X, x) \to (Y, y)$ is a pair (f, φ) of a morphism $f : X \to Y$ in $\mathcal{C}$ and a morphism $\varphi : x \to Ff(y)$ in FX, as in the following diagram, where the upper triangle is interpreted in **Cat** (note the contravariance of Ff):

$$
\begin{array}{ccc}
& 1 & \\
x \swarrow & \underset{\varphi}{\Downarrow} & \searrow y \\
FX \longleftarrow & Ff & FY
\end{array}
$$

$$X \xrightarrow{\quad f \quad} Y$$

We can thus write the hom set $\int F\big((X, x), (Y, y)\big)$ as the dependent sum $\sum_{f : \mathcal{C}(X,Y)} FX\big(x, Ff(y)\big)$. The identity morphism on (X, x) is $(\mathrm{id}_X, \mathrm{id}_x)$, and composition is defined as follows. Given $(f, \varphi) : (X, x) \to (Y, y)$ and $(g, \gamma) : (Y, y) \to (Z, z)$, their composite $(g, \gamma) \circ (f, \varphi)$ is the pair

$$(g \circ f, Ff(\gamma) \circ \varphi) .$$

The following well-known result tells us that the Grothendieck construction yields fibrations.

Proposition 4.2.16 (Johnson and Yau [145, Prop. 10.1.10])**.** Suppose $F : \mathcal{C}^{\mathrm{op}} \to$ **Cat** is an indexed category. Then there is a 'projection' functor $\pi_F : \int F \to \mathcal{C}$ mapping $(X, x) \mapsto X$ and $(f, \varphi) \mapsto f$, and this functor is a fibration.

Remark 4.2.17. Dually, there is a *covariant Grothendieck construction*, for opindexed categories $F : \mathcal{C} \to$ **Cat**. The objects of $\int F$ are again pairs $(X : \mathcal{C}, x : FX)$, but now the morphisms $(X, x) \to (Y, y)$ are pairs (f, φ) with $f : X \to Y$ in $\mathcal{C}$ as before and now $\varphi : Ff(x) \to y$; all that we have done is swapped the direction of the arrow Ff in the diagram in Definition 4.2.15 (compare the identifications in the category of elements of a copresheaf, in Definition 4.2.4). As a result, we can write the hom set $\int F\big((X, x), (Y, y)\big)$ in this case as $\sum_{f : \mathcal{C}(X,Y)} FX\big(Ff(x), y\big)$.

Remark 4.2.18. The Grothendieck construction induces an analogue of Proposition 4.2.9 between the bicategory of pseudofunctors $\mathcal{B}^{\mathrm{op}} \to$ **Cat** and the bicategory of *Grothendieck fibrations* on $\mathcal{B}$ [145, Theorem 10.6.16]. Indeed there are analogues of Propositions 4.2.9 and 4.2.3 in any categorical dimension. Because fibrations are the higher-dimensional analogues of bundles, they have a base category (the codomain) and a 'total' category (the domain), which is a kind of colimit of the fibres (constructed by the Grothendieck construction): strictly speaking, what we have called the Grothendieck construction above is total category of the full fibrational construction; the fibration itself is the corresponding forgetful (projection) functor. For a highly readable exposition of Grothendieck constructions, we refer the reader to Loregian and Riehl [173].

4.2.1.1. The monoidal Grothendieck construction

When $\mathcal{C}$ is a monoidal category with which F is appropriately compatible, then we can 'upgrade' the notions of indexed category and Grothendieck construction accordingly. In this chapter, we will restrict ourselves to locally trivial monoidal indexed categories, those for which the domain $\mathcal{C}$ is only a category; Moeller and Vasilakopoulou [189] work out the structure for bicategorical $\mathcal{C}$. (As noted in Remark 4.2.11, in Chapter 5, we will sketch a notion of monoidal indexed bicategory which amounts to a categorification of the present notion; but that will also in some sense be locally trivial.)

Definition 4.2.19 (After Moeller and Vasilakopoulou [189, §3.2]). Suppose $(\mathcal{C}, \otimes, I)$ is a monoidal category. We say that F is a *monoidal indexed category* when F is a lax monoidal pseudofunctor $(F, \mu, \eta) : (\mathcal{C}^{\mathrm{op}}, \otimes^{\mathrm{op}}, I) \to (\mathbf{Cat}, \times, \mathbf{1})$. This means that the laxator μ is given by a natural family of functors $\mu_{A,B} : FA \times FB \to F(A \otimes B)$ along with, for any morphisms $f : A \to A'$ and $g : B \to B'$ in $\mathcal{C}$, a natural isomorphism $\mu_{f,g} : \mu_{A',B'} \circ (Ff \times Fg) \Rightarrow F(f \otimes g) \circ \mu_{A,B}$. (This makes μ into a pseudonatural transformation in the sense of Definition 5.2.9.) The laxator and the unitor $\eta : \mathbf{1} \to FI$ together satisfy axioms of associativity and unitality that constitute indexed versions of the associators and unitors of a monoidal category (Definition 3.1.4).

Explicitly, this means that there must be three families of natural isomorphisms, indexed by objects $A, B, C : \mathcal{C}$,

1. an *associator* family $\alpha_{A,B,C} : \mu_{A \otimes B, C}(\mu_{A,B}(-, -), -) \Rightarrow \mu_{A, B \otimes C}(-, \mu_{B,C}(-, -))$;

2. a *left unitor* $\lambda_A : \mu_{I,A}(\eta, -) \Rightarrow \mathrm{id}_{FA}$; and

3. a *right unitor* $\rho_A : \mu_{A,I}(-, \eta) \Rightarrow \mathrm{id}_{FA}$

such that the unitors are compatible with the associator, *i.e.* for all $A, B : \mathcal{C}$ the diagram

$$\mu_{A \otimes I, B}(\mu_{A,I}(-, \eta), -) \xRightarrow{\alpha_{A,I,B}(-,\eta,-)} \mu_{A, I \otimes B}(-, \mu_{I,B}(\eta, -))$$

$$\mu_{\mathrm{P}_A, B}(\rho_A, -) \searrow \qquad \swarrow \mu_{A, \Lambda_B}(-, \lambda_B)$$

$$\mu_{A,B}(-, -)$$

commutes (where P and Λ are the right and left associators of the monoidal structure $(\otimes, I)$ on $\mathcal{C}$), and such that the associativity is 'order-independent', *i.e.* for all $A, B, C, D : \mathcal{C}$, the diagram

$$\mu_{A\otimes(B\otimes C),D}(\mu_{A,B\otimes C}(-,\mu_{B,C}(-,-)),-) \xRightarrow{\ \alpha_{A,B\otimes C,D}\ } \mu_{A,(B\otimes C)\otimes D}(-,\mu_{B\otimes C,D}(\mu_{B,C}(-,-),-))$$

$$\Big\uparrow \mu_{A_{A,B,C},D}(\alpha_{A,B,C},-) \qquad\qquad \mu_{A,A_{B,C,D}}(-,\alpha_{B,C,D}) \Big\downarrow$$

$$\mu_{(A\otimes B)\otimes C,D}(\mu_{A\otimes B,C}(\mu_{A,B}(-,-),-),-) \qquad\qquad \mu_{A,B\otimes(C\otimes D)}(-,\mu_{B,C\otimes D}(-,\mu_{C,D}(-,-)))$$

$$\xRightarrow{\ \alpha_{A\otimes B,C,D}\ } \quad \mu_{A\otimes B,C\otimes D}(\mu_{A,B}(-,-),\mu_{C,D}(-,-)) \quad \xRightarrow{\ \alpha_{A,B,C\otimes D}\ }$$

commutes (where A is the associator of the monoidal structure on $\mathcal{C}$).

The following proposition exhibits the monoidal structure carried by the Grothendieck construction when the indexed category is monoidal.

Proposition 4.2.20 (Moeller and Vasilakopoulou [189, §6.1])**.** Suppose $(F,\mu,\eta):(\mathcal{C}^{\,\mathrm{op}},\otimes^{\mathrm{op}},I)\to (\mathbf{Cat},\times,\mathbf{1})$ is a monoidal indexed category. Then the total category of the Grothendieck construction $\int F$ obtains a monoidal structure $(\otimes_\mu, I_\mu)$. On objects, define

$$(C,X)\otimes_\mu(D,Y):=\big(C\otimes D,\mu_{CD}(X,Y)\big)$$

where $\mu_{CD}:FC\times FD\to F(C\otimes D)$ is the component of μ at (C,D). On morphisms $(f,f^\dagger): (C,X)\twoheadrightarrow(C',X')$ and $(g,g^\dagger):(D,Y)\twoheadrightarrow(D',Y')$, define

$$(f,f^\dagger)\otimes_\mu(g,g^\dagger):=\big(f\otimes g,\mu_{CD}(f^\dagger,g^\dagger)\big).$$

The monoidal unit I_μ is defined to be the object $I_\mu:=\big(I,\eta(*)\big)$. Writing $\lambda:I\otimes(-)\Rightarrow(-)$ and $\rho:(-)\otimes I\Rightarrow(-)$ for the left and right unitors of the monoidal structure on $\mathcal{C}$, the left and right unitors in $\int F$ are given by (λ,id) and (ρ,id) respectively. Writing α for the associator of the monoidal structure on $\mathcal{C}$, the associator in $\int F$ is given by (α,id).

Remark 4.2.21. Sometimes, rather than (or in addition to) an indexed category F being lax monoidal as a pseudofunctor (which yields a 'global' monoidal structure), it may in fact be *fibrewise monoidal*, meaning that each fibre FX is itself a monoidal category (yielding 'local' monoidal structures); in this case, the pseudofunctor F can be written with the type $\mathcal{C}^{\,\mathrm{op}}\to\mathbf{MonCat}$. In general, the fibrewise monoidal structures may be independent both of each other and of the lax monoidal structure on F itself, but when $\mathcal{C}$ is in fact *Cartesian* monoidal, the local and global monoidal structures coincide. For more reading on this, we refer the reader to Moeller and Vasilakopoulou [189, §4].

4.2.2. Grothendieck lenses

Lenses formalize bidirectional processes in which the 'backward' process depends on data in the domain of the 'forward' process. The name originates in database theory [34, 99], where the forward process gives a zoomed-in 'view' onto a database record, and the backward process is used to update it. Following an observation of Myers and Spivak [238], lenses of this general shape can be given a concise definition using the Grothendieck construction. In order to obtain the backward directionality of the dependent part, we use the "pointwise opposite" of a pseudofunctor.

Definition 4.2.22. Suppose $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ is a pseudofunctor. We define its *pointwise opposite* $F^p : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ to be the pseudofunctor $c \mapsto Fc^{\mathrm{op}}$ returning the opposite of each category Fc; given $f : c \to c'$, $F^p f : Fc^{\mathrm{op}} \to Fc'^{\mathrm{op}}$ is defined as $(Ff)^{\mathrm{op}} : Fc^{\mathrm{op}} \to Fc'^{\mathrm{op}}$.

Categories of Grothendieck lenses are then obtained via the Grothendieck construction of pointwise opposites of pseudofunctors.

Definition 4.2.23 (Grothendieck lenses [238])**.** We define the category $\mathbf{Lens}_F$ of Grothendieck lenses for a pseudofunctor $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ to be the total category of the Grothendieck construction for the pointwise opposite F^p of F. Explicitly, its objects $(\mathbf{Lens}_F)_0$ are pairs (C, X) of objects C in $\mathcal{C}$ and X in $F(C)$, and its hom-sets $\mathbf{Lens}_F\big((C, X), (C', X')\big)$ are the dependent sums

$$\mathbf{Lens}_F\big((C, X), (C', X')\big) = \sum_{f : \mathcal{C}(C, C')} F(C)\big(F(f)(X'), X\big) \tag{4.10}$$

so that a morphism $(C, X) \nrightarrow (C', X')$ is a pair $(f, f^\dagger)$ of $f : \mathcal{C}(C, C')$ and $f^\dagger : F(C)\big(F(f)(X'), X\big)$. We call such pairs *Grothendieck lenses* for F or *F-lenses*. We say that the morphism f is the *forward* component of the lens, and the morphism $f^\dagger$ the *backward* component.

The identity Grothendieck lens on (C, X) is $\mathrm{id}_{(C,X)} = (\mathrm{id}_C, \mathrm{id}_X)$. Sequential composition is as follows. Given $(f, f^\dagger) : (C, X) \nrightarrow (C', X')$ and $(g, g^\dagger) : (C', X') \nrightarrow (D, Y)$, their composite $(g, g^\dagger) \mathbin{\Phi} (f, f^\dagger)$ is defined to be the lens $\big(g \bullet f, f^\dagger \circ F(f)(g^\dagger)\big) : (C, X) \nrightarrow (D, Y)$.

Notation 4.2.24. In the context of lenses, we will often write the backward map as $f^\dagger$ or $f^\sharp$, with the former particularly used for Bayesian lenses. We will also use $\nrightarrow$ to denote a lens, and $\mathbin{\Phi}$ for lens composition. Above, we additionally used $\bullet$ for composition in the base category and $\circ$ for composition in the fibres.

Since lenses are bidirectional processes and English is read horizontally, when it comes to string diagrams for lenses, we will depict these horizontally, with the forwards direction read from left to right.

Whenever $\mathcal{C}$ is a monoidal category, it gives rise to a canonical category of lenses, in which the forwards morphisms are comonoid morphisms in $\mathcal{C}$ and the backwards morphisms are (internally) parameterized by the domains of the forwards ones. Comonoids and their morphisms are necessary to copy parameters during composition. The resulting 'monoidal' lenses are a natural generalization of the 'Cartesian' lenses used in the database setting, and we will see that Bayesian lenses are similarly constructed using an indexed category of (externally) parameterized morphisms.

Example 4.2.25. Suppose $(\mathcal{C}, \otimes, I)$ is a monoidal category and let $\mathbf{Comon}(\mathcal{C})$ be its subcategory of comonoids and comonoid morphisms. A *monoidal lens* $(X, A) \nrightarrow (Y, B)$ is a pair $(f, f^\sharp)$ of a comonoid morphism $f : X \to Y$ in $\mathbf{Comon}(\mathcal{C})$ and a morphism $f^\sharp : X \otimes B \to A$ in $\mathcal{C}$. Such lenses can be characterized as Grothendieck lenses, following Spivak [238, §3.2].

First, define a pseudofunctor $\mathsf{P} : \mathbf{Comon}(\mathcal{C})^{\mathrm{op}} \to \mathbf{Cat}$ as follows. On objects $X : \mathbf{Comon}(\mathcal{C})$, define $\mathsf{P}X$ as the category with the same objects as $\mathcal{C}$ and with hom-sets given by $\mathsf{P}X(A, B) := \mathcal{C}(X \otimes A, B)$; denote a morphism f from A to B in $\mathsf{P}X$ by $f : A \xrightarrow{X} B$. The identity $\mathsf{id}_A : A \xrightarrow{X} A$ is defined as the projection $\mathsf{proj}_A : X \otimes A \xrightarrow{\bar{\top}_X \otimes \mathsf{id}_A} I \otimes A \xrightarrow{\lambda_A} A$. Given $f : A \xrightarrow{X} B$ and $g : B \xrightarrow{X} C$, their composite $g \circ f : A \xrightarrow{X} C$ is given by the following string diagram in $\mathcal{C}$:

.
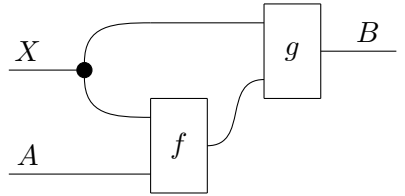
Given $h : X \to Y$ in $\mathbf{Comon}(\mathcal{C})$, the functor $\mathsf{P}h : \mathsf{P}Y \to \mathsf{P}X$ acts by precomposition on morphisms, taking $f : A \xrightarrow{Y} B$ to the morphism $\mathsf{P}h(f) : A \xrightarrow{X} B$ given by

$$X \otimes A \xrightarrow{h \otimes \mathsf{id}_A} Y \otimes A \xrightarrow{f} B \ .$$

(An alternative way to obtain $\mathsf{P}X$ is as the 'coKleisli' category of the comonad $X \otimes (-)$.)

The category of monoidal lenses is then defined to be the category of Grothendieck P-lenses. The objects of $\mathbf{Lens}_\mathsf{P}$ are pairs (X, A) of a comonoid X and an object A in $\mathcal{C}$, and the morphisms are monoidal lenses. Given lenses $(f, f^\sharp) : (X, A) \to (Y, B)$ and $(g, g^\sharp) : (Y, B) \to (Z, C)$, the composite lens has forward component given by $g \circ f : X \to Z$ and backward component given by $f^\sharp \circ \mathsf{P}f(g^\sharp) : C \xrightarrow{X} A$.

We can depict monoidal lenses string-diagrammatically, with the forward and backward components oriented in opposite directions. To exemplify this, note that, because the forwards

components are comonoid morphisms, the following equality holds for all composite monoidal lenses $(g, g^\sharp) \circ (f, f^\sharp)$:
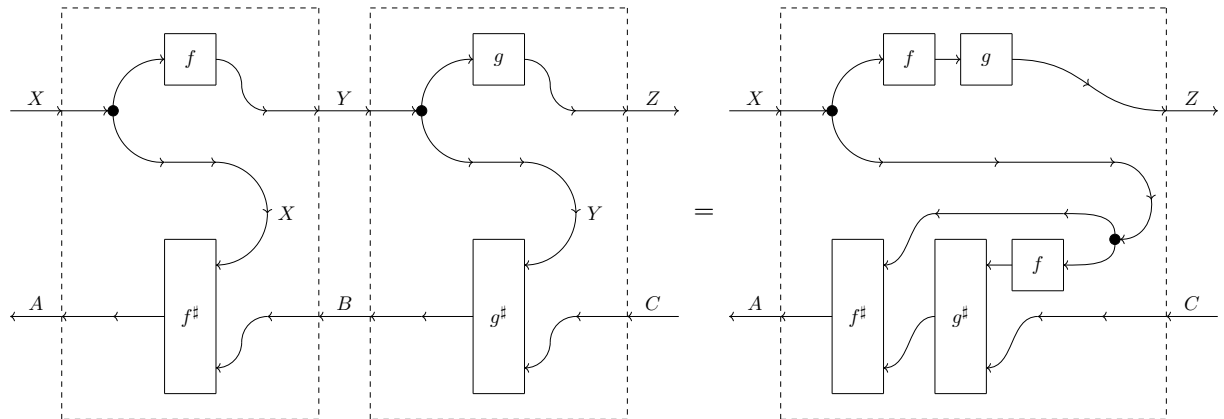
Here, we have decorated the strings with fletches to indicate the direction of information-flow and disambiguate the bidirectionality, and drawn boxes around the pairs that constitute each lens. Note however that the parameterizing input to the backwards component of the first lens is not constrained to be a copy of the input to the forwards component; it is only for compositional convenience that we depict lenses this way.

Definition 4.2.26. When $\mathcal{C}$ is Cartesian monoidal, so that its monoidal structure $(\times, 1)$ is the categorical product, we will call monoidal lenses in $\mathcal{C}$ *Cartesian lenses*.

Remark 4.2.27. The string-diagrammatic depictions of lenses above were not strictly formal, or at least we have not explain how they might be; we have not exhibited a coherence theorem such as 3.1.10. In this case, the diagrams above are depictions in the graphical calculus of Boisseau [35]. An alternative graphical language for a generalization of lenses called *optics*[184, 217] has been described by Román [219].

Monoidal lenses find uses not only in database theory, but in many other situations, too: the general pattern is "interacting systems where information flows bidirectionally". In economics (specifically, compositional game theory), lenses are used to model the pattern of interaction of economic games: the forward maps encode how players act in light of observations, and the backward maps encode how utility is passed "backwards in time" from outcomes, in order to assign credit[119]. In non-probabilistic machine learning, lenses can be used to formalize reverse differentiation and hence the backpropagation of error (another kind of credit assignment): the forwards maps represent differentiable processes (such as neural networks), and the backward maps are the reverse-derivatives used to pass error back (*e.g.*, between neural network layers)[74, 96].

Generalizations of lenses known as optics[184, 217] have also been used both to model economic games with uncertainty ('mixed' strategies)[36] and to model the process of dynamic programming (Bellman iteration) used in the related field of reinforcement learning[128], as well as to model client-server interactions in computing[277].

In systems theory, lenses can be used to formalize various kinds of dynamical system: the forward maps encode their 'outputs' or 'actions', and the backward maps encode how states and inputs give rise to transitions[191]. This latter application will be a particular inspiration to us, and is closely related to Example 4.2.30, which expresses polynomial functors as lenses (thereby explaining Proposition 3.5.4), and for which we need the following canonical family of indexed categories.

Definition 4.2.28. When a category $\mathcal{C}$ has pullbacks, its slice categories $\mathcal{C}/C$ collect into an indexed category $\mathcal{C}/(-) : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ called the *(contravariant[3]) self-indexing* of $\mathcal{C}$, and defined as follows. On objects $C : \mathcal{C}$, the self-indexing unsurprisingly returns the corresponding slice categories $\mathcal{C}/C$.

Given a morphism $f : A \to B$, the functor $\mathcal{C}/f : \mathcal{C}/B \to \mathcal{C}/A$ is defined by pullback. On objects $(E, p) : \mathcal{C}/B$, we define $(\mathcal{C}/f)(E, p) := (f^*E, f^*p)$, where f^*E is the pullback object $A \times_B E$ and f^*p is the associated projection to A. On morphisms $\varphi : (E, p) \to (E', p')$ in $\mathcal{C}/B$, we define $(\mathcal{C}/f)(\varphi)$ as the morphism $f^*\varphi : (f^*E, f^*p) \to (f^*E', f^*p')$ induced by the universal property of the pullback f^*E', as in the commuting diagram

$$
\begin{array}{ccc}
f^*E & \longrightarrow & E \\
\downarrow f^*\varphi & & \downarrow \varphi \\
f^*E' & \longrightarrow & E' \\
\downarrow f^*p' & & \downarrow p' \\
A & \xrightarrow{\ f\ } & B
\end{array}
$$

Remark 4.2.29. The functors $\mathcal{C}/f : \mathcal{C}/B \to \mathcal{C}/A$ are also known as *base-change* functors, as they change the 'base' of the slice category.

Example 4.2.30. The category **Poly** of polynomial functors (§3.5) is equivalent to the category of Grothendieck lenses for the self-indexing of **Set**: that is, $\mathbf{Poly} \cong \mathbf{Lens}_{\mathbf{Set}/(-)}$. To see this, observe that the objects of $\mathbf{Lens}_{\mathbf{Set}/(-)}$ are bundles $p : E \to B$ of sets. If we define the set $p[i]$ to be the fibre E_i of p for each $i : B$, we have an isomorphism $E \cong \sum_{i:B} p[i]$. We can then define

---

a polynomial functor $P := \sum_{i:B} y^{p[i]}$, and then find that $P(1) = B$, which justifies writing the original bundle as $p : \sum_{i:p(1)} p[i] \to p(1)$. We saw in Proposition 3.5.4 how to associate to any polynomial functor P a bundle p, and it is easy to check that applying this construction to the P defined here returns our original bundle p. This shows that the objects of **Poly** are in bijection with the objects of $\mathbf{Lens}_{\mathbf{Set}/(-)}$. What about the morphisms?

A morphism $p \to q$ in $\mathbf{Lens}_{\mathbf{Set}/(-)}$, for $p : X \to A$ and $q : Y \to B$ is a pair of functions $f_1 : A \to B$ and $f^\sharp : f_1^* Y \to X$ such that $f_1^* q = p \circ f^\sharp$, as in the following diagram:

$$
\begin{array}{ccccc}
X & \xleftarrow{\ f^\sharp\ } & f_1^* Y & \longrightarrow & Y \\
{\scriptstyle p}\downarrow & & {\scriptstyle f_1^* q}\downarrow & \lrcorner & \downarrow{\scriptstyle q} \\
A & =\!=\!=\!= & A & \xrightarrow{\ f_1\ } & B
\end{array}
$$

Replacing the bundles p and q by their polynomial representations $p : \sum_{i:p(1)} p[i] \to p(1)$ and $q : \sum_{j:q(1)} q[j] \to q(1)$, we see that the pair $(f_1, f^\sharp)$ is precisely a morphism of polynomials of the form established in Proposition 3.5.4, and that every morphism of polynomials corresponds to such a lens. This establishes an isomorphism of hom-sets, and hence $\mathbf{Poly} \cong \mathbf{Lens}_{\mathbf{Set}/(-)}$.

Lenses are also closely related to wiring diagrams[239, 282] and our linear circuit diagrams (§3.3.2).

Example 4.2.31. Let **FVect** denote the category of finite-dimensional real vector spaces and linear maps between them; write n for the object $\mathbb{R}^n$. **FVect** has a Cartesian monoidal product $(+, 0)$ given by the direct sum of vectors ($n + m = \mathbb{R}^n \oplus \mathbb{R}^m = \mathbb{R}^{n+m}$), and whose unit object is 0. The category of monoidal lenses in $(\mathbf{FVect}, +, 0)$ is the category of linear circuit diagrams (Example 3.3.9).

Cartesian lenses $(X, A) \nrightarrow (Y, B)$ are in some sense 'non-dependent' lenses, because the domain of the backwards map is a simple product $X \times B$, in which the object B does not depend on $x : X$. We can see polynomial functors as a dependent generalization of Cartesian lenses in **Set**.

Proposition 4.2.32. The category of monoidal lenses in $(\mathbf{Set}, \times, 1)$ is equivalently the full subcategory of **Poly** on the monomials Xy^A.

Proof sketch. A morphism of monomials $(f_1, f^\sharp) : Xy^A \to Yy^B$ is a pair of functions $f_1 : X \to Y$ and $f^\sharp : X \times B \to A$; this is a Cartesian lens $(X, A) \to (Y, B)$. There is clearly a bijection of objects $Xy^A \leftrightarrow (X, A)$. $\qquad\square$

In particular, this situation encompasses linear circuit diagrams, which embed into **Poly** accordingly.

Remark 4.2.33. There is a forgetful functor from vector spaces to sets, $U : \mathbf{FVect} \to \mathbf{Set}$. If we write $\mathbf{Lens}(\mathcal{C})$ to denote the category of monoidal lenses in $\mathcal{C}$ (with the relevant monoidal structure left implicit), this forgetful functor induces a 'change of base' $\mathbf{Lens}(U) : \mathbf{Lens}(\mathbf{FVect}) \to \mathbf{Lens}(\mathbf{Set})$, since the Grothendieck construction is functorial by Remark 4.2.18, and hence so is the **Lens** construction. There is therefore a canonical embedding of linear circuit diagrams into **Poly**, $\mathbf{Lens}(\mathbf{FVect}) \xrightarrow{\mathbf{Lens}(U)} \mathbf{Lens}(\mathbf{Set}) \hookrightarrow \mathbf{Poly}$.

Our dynamical semantics for approximate inference (Chapter 7) can, if one squints a little, be therefore seen as a kind of probabilistic generalization of our algebra for rate-coded neural circuits: it will be an algebra for (a stochastic analogue of) the multicategory $\mathcal{O}\mathbf{Poly}$ with semantics in categories of (stochastic) dynamical systems. One can see a morphism of polynomials therefore as a kind of 'dependent' circuit diagram, with the forwards component transporting 'outgoing' information from inside a ('nested') system to its boundary (its external interface), and the backward component transporting 'incoming' information ("immanent signals") from the boundary internally, depending on the configuration of the boundary.

Of course, to give an $\mathcal{O}\mathbf{Poly}$-algebra is to give a lax monoidal functor, which means knowing the relevant monoidal structure. While we saw this in the case of polynomial functors of sets in Proposition 3.5.7, it will be helpful when it comes to generalizing **Poly** to see how this structure is obtained. Moreover, we will want a monoidal structure on Bayesian lenses, in order to define joint approximate inference systems. For these reasons, we now turn to monoidal categories of lenses.

4.2.2.1. Monoidal categories of lenses

The monoidal structures on categories of Grothendieck lenses—at least those of interest here—are a direct corollary of the monoidal Grothendieck construction, Proposition 4.2.20.

Corollary 4.2.34. When $F : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ is equipped with a monoidal indexed category structure (μ, η), its category of lenses $\mathbf{Lens}_F$ becomes a monoidal category $(\mathbf{Lens}_F, \otimes'_\mu, I_\mu)$. On objects $\otimes'_\mu$ is defined as $\otimes_\mu$ in Proposition 4.2.20, as is I_μ. On morphisms $(f, f^\dagger) : (C, X) \twoheadrightarrow (C', X')$ and $(g, g^\dagger) : (D, Y) \twoheadrightarrow (D', Y')$, define

$$(f, f^\dagger) \otimes'_\mu (g, g^\dagger) := \left(f \otimes g, \mu^{\mathrm{op}}_{CD}(f^\dagger, g^\dagger) \right)$$

where $\mu_{CD}^{\mathrm{op}} : F(C)^{\mathrm{op}} \times F(D)^{\mathrm{op}} \to F(C \otimes D)^{\mathrm{op}}$ is the pointwise opposite of μ_{CD}. The associator and unitors are defined as in Proposition 4.2.20.

As an example, this gives us the tensor product on **Poly**, which is inherited by the category of Cartesian lenses in **Set**.

Example 4.2.35. The tensor product structure $(\otimes, y)$ on **Poly** is induced by a monoidal indexed category structure (μ, η) on the self-indexing of $(\mathbf{Set}, \times, 1)$. To define the unitor η, first note that $\mathbf{Set}/1 \cong \mathbf{Set}$, so that η equivalently has the type $\mathbf{1} \to \mathbf{Set}$; we thus make the natural choice for η, the terminal element $* \mapsto 1$. The laxator μ is defined for each $B, C : \mathbf{Set}$ by the functor

$$
\begin{aligned}
\mu_{B,C} \quad : \quad & \mathbf{Set}/B \quad \times \quad \mathbf{Set}/C \quad \to \quad \mathbf{Set}/(B \times C) \\
& \Big(p : \sum_{i:B} p[i] \to B, \quad q : \sum_{j:C} q[j] \to C \Big) \mapsto \sum_{(i,j):B \times C} p[i] \times q[j]
\end{aligned}
$$

the naturality and functoriality of which follow from the functoriality of $\times$. Applying Corollary 4.2.34 to this structure, we obtain precisely the tensor product of polynomials introduced in Proposition 3.5.7.

Corollary 4.2.36. Since the category of Cartesian lenses in **Set** is the monomial subcategory of **Poly**, to which the tensor structure $(\otimes, y)$ restricts, the latter induces a symmetric monoidal structure on the former, the unit of which is the object $(1, 1)$. Given objects (X, A) and (X', A'), their tensor $(X, A) \otimes (X', A')$ is $(X \times X', A \times A')$. Given lenses $(f, f^\sharp) : (X, A) \to (Y, B)$ and $(f', f'^\sharp) : (X', A') \to (Y', B')$, their tensor has forward component $f \times f' : X \times X' \to Y \times Y'$ and backward component

$$
X \times X' \times B \times B' \xrightarrow{\mathrm{id}_X \times \sigma_{X',\mathsf{B}} \times \mathrm{id}_{B'}} X \times B \times X' \times B' \xrightarrow{f^\sharp \times f'^\sharp} A \times A'
$$

where σ is the symmetry of the product $\times$.

We will see that the monoidal structure on Bayesian lenses is defined similarly. First of all, we need to define Bayesian lenses themselves.

4.3. The bidirectional structure of Bayesian updating

In this section, we define a collection of indexed categories, each denoted Stat, whose morphisms can be seen as generalized Bayesian inversions. Following Definition 4.2.23, these induce corresponding categories of lenses which we call *Bayesian lenses*. In §4.3.3, we show abstractly that, for the

subcategories of *exact* Bayesian lenses whose backward channels correspond to 'exact' Bayesian inversions, the Bayesian inversion of a composite of forward channels is given (up to almost-equality) by the lens composite of the corresponding backward channels. This justifies calling these lenses 'Bayesian', and provides the foundation for the study of approximate (non-exact) Bayesian inversion in Chapter 5.

Remark 4.3.1. Bayesian lenses, and the result that "Bayesian updates compose optically", were first introduced by the present author in [248]. Braithwaite and Hedges [41] then elaborated the structure to define *dependent* Bayesian lenses, solving the 'divide-by-zero' issue already indicated in Remark 4.1.22. All three authors then joined forces to produce a paper [42], published at MFCS 2023, which we take to be a canonical summary of the definitions and basic results.

4.3.1. State-dependent channels

As we saw in §4.1, a channel $c : X \to Y$ admitting a Bayesian inversion induces a family of inverse channels $c_\pi^\dagger : Y \to X$, indexed by 'prior' states $\pi : 1 \to X$. Making the state-dependence explicit, in typical cases where c is a probability kernel we obtain a function $c^\dagger : \mathcal{G}X \times Y \to \mathcal{G}X$, under the assumption that $c \bullet \pi$ is fully supported for all $\pi : \mathcal{G}X$ (see Remark 4.1.20 for our justification of this simplifying assumption). In more general situations, and in light of the full-support assumption, we obtain a morphism $c^\dagger : \mathcal{C}(I, X) \to \mathcal{C}(Y, X)$ in the base of enrichment of the monoidal category $(\mathcal{C}, \otimes, I)$ of c, which for simplicity we take to be **Set** (although the construction still succeeds for an arbitrary Cartesian base of enrichment). We call morphisms of this general type *state-dependent channels*, and structure the indexing as an indexed category.

Definition 4.3.2. Let $(\mathcal{C}, \otimes, I)$ be a monoidal category. Define the $\mathcal{C}$-*state-indexed* category $\mathsf{Stat} : \mathcal{C}^{\mathrm{op}} \to \mathbf{Cat}$ as follows.

$$\mathsf{Stat} \ : \ \mathcal{C}^{\mathrm{op}} \ \to \ \mathbf{Cat}$$

$$X \mapsto \mathsf{Stat}(X) := \begin{pmatrix} \mathsf{Stat}(X)_0 & := & \mathcal{C}_0 \\ \mathsf{Stat}(X)(A, B) & := & \mathbf{Set}\big(\mathcal{C}(I, X), \mathcal{C}(A, B)\big) \\ \mathsf{id}_A \ : \ \mathsf{Stat}(X)(A, A) & := & \begin{cases} \mathsf{id}_A : \mathcal{C}(I, X) \to \mathcal{C}(A, A) \\ \rho & \mapsto & \mathsf{id}_A \end{cases} \end{pmatrix}$$

$$(4.11)$$

$$f : \mathcal{C}(Y, X) \mapsto \begin{pmatrix} \mathsf{Stat}(f) \ : & \mathsf{Stat}(X) & \to & \mathsf{Stat}(Y) \\ & \mathsf{Stat}(X)_0 & = & \mathsf{Stat}(Y)_0 \\ & \mathbf{Set}(\mathcal{C}(I, X), \mathcal{C}(A, B)) & \to & \mathbf{Set}\big(\mathcal{C}(I, Y), \mathcal{C}(A, B)\big) \\ & \alpha & \mapsto & \big(\sigma : \mathcal{C}(I, Y)\big) \mapsto \big(\alpha_{f \bullet \sigma} : \mathcal{C}(A, B)\big) \end{pmatrix}$$

Composition in each fibre $\mathsf{Stat}(X)$ is as in $\mathcal{C}$. Explicitly, indicating morphisms $\mathcal{C}(I, X) \to \mathcal{C}(A, B)$ in $\mathsf{Stat}(X)$ by $A \overset{X}{\dashrightarrow} B$, and given $\alpha : A \overset{X}{\dashrightarrow} B$ and $\beta : B \overset{X}{\dashrightarrow} C$, their composite $\beta \circ \alpha : A \overset{X}{\dashrightarrow}$ is defined by $(\beta \circ \alpha)_\rho := \beta_\rho \bullet \alpha_\rho$, where here we indicate composition in $\mathcal{C}$ by $\bullet$ and composition in the fibres $\mathsf{Stat}(X)$ by $\circ$. Given $f : Y \rightarrowtail X$ in $\mathcal{C}$, the induced functor $\mathsf{Stat}(f) : \mathsf{Stat}(X) \to \mathsf{Stat}(Y)$ acts by pre-composition (compare Definition 4.2.28 of the functorial action of the self-indexing); for example:

$$\mathsf{Stat}(f)(\alpha) \; : \; \begin{array}{ccccc} \mathcal{C}(I, Y) & \xrightarrow{\mathcal{C}(I,f)} & \mathcal{C}(I, X) & \xrightarrow{\alpha} & \mathcal{C}(A, B) \\ \sigma & \mapsto & f \bullet \sigma & \mapsto & \alpha_{f \bullet \sigma} \end{array} \; .$$

Remark 4.3.3. If we do not wish to make the full-support assumption, and instead we know that the category $\mathcal{C}$ has a well-defined notion of *support object*[41, 109, 259], then for a given general channel $c : X \rightarrowtail Y$, we can write the type of its Bayesian inversion $c^\dagger$ as $\prod_{\pi : \mathcal{C}(I,X)} \mathcal{C}\big(\mathsf{supp}(c \bullet \pi), Y\big)$. As Braithwaite and Hedges [41] show, this corresponds to a morphism in a certain fibration, and gives rise to a category of dependent Bayesian lenses; see Remark 4.1.20.

Notation 4.3.4. Just as we wrote $X \xrightarrow{M} Y$ for an internally M-parameterized morphism in $\mathcal{C}(M \odot X, Y)$ (see Proposition 3.2.3) and $A \xrightarrow{\Theta} B$ for an externally Θ-parameterized morphism in $\mathcal{E}\big(\Theta, \mathcal{C}(A, B)\big)$ (see Definition 3.2.11), we write $A \overset{X}{\dashrightarrow} B$ for an X-state-dependent morphism in $\mathbf{Set}\big(\mathcal{C}(I, X), \mathcal{C}(A, B)\big)$. Given a state ρ in $\mathcal{C}(I, X)$ and an X-state-dependent morphism $f : A \overset{X}{\dashrightarrow} B$, we write f_ρ for the resulting morphism in $\mathcal{C}(A, B)$.

Remark 4.3.5. The similarities between state-dependent channels and externally parameterized functions are no coincidence: the indexed category Stat is closely related to an indexed category underlying external parameterization in $\mathbf{Set}$, which in previous work, reported by Capucci, Gavranović, and St Clere Smithe [53], we called $\mathbf{Prox}$ (for 'proxies').

When $\mathcal{C}$ is a Kleisli category $\mathcal{K}\ell(T)$, it is of course possible to define a variant of Stat on the other side of the product-exponential adjunction, with state-dependent morphisms $A \overset{X}{\dashrightarrow} B$ having the types $TX \times A \to TB$. This avoids the technical difficulties sketched in the preceding example at the cost of requiring a monad T. However, the exponential form makes for better exegesis, and so we will stick to that.

We will want to place inference systems side-by-side, which means we want a monoidal category structure for Bayesian lenses. Following Corollary 4.2.34, this means Stat needs to be a monoidal indexed category.

Proposition 4.3.6. Stat is a monoidal indexed category, in the sense of Definition 4.2.19. The components $\mu_{XY} : \mathsf{Stat}(X) \times \mathsf{Stat}(Y) \to \mathsf{Stat}(X \otimes Y)$ of the laxator are defined on objects

by $\mu_{XY}(A, A') := A \otimes A'$ and on morphisms $f : A \xrightarrow{X} B$ and $f' : A' \xrightarrow{Y} B'$ as the $X \otimes Y$-state-dependent morphism denoted $f \otimes f'$ and given by the function

$$\mu_{XY}(f, f') : \mathcal{C}(I, X \otimes Y) \to \mathcal{C}(A \otimes A', B \otimes B')$$
$$\omega \mapsto f_{\omega_X} \otimes f'_{\omega_Y}$$

.

Here, ω_X and ω_Y are the X and Y marginals of ω, given by $\omega_X := \mathsf{proj}_X \bullet \omega$ and $\omega_Y := \mathsf{proj}_Y \bullet \omega$. (Note that this makes μ into a strict transformation in the sense of Definition 5.2.9.) The unit $\eta : \mathbf{1} \to \mathsf{Stat}(I)$ of the lax monoidal structure is the functor mapping the unique object $* : \mathbf{1}$ to the unit object $I : \mathsf{Stat}(I)$.

Remark 4.3.7. Note that Stat is also *fibrewise monoidal* in the sense of Remark 4.2.21, as an almost trivial consequence of $\mathcal{C}$ being monoidal. We will not make use of this structure in this chapter, but we will return to it in the construction of statistical games in §5.3.1.

At this point, we can turn to Bayesian lenses themselves.

4.3.2. Bayesian lenses

We define the category of Bayesian lenses in $\mathcal{C}$ to be the category of Grothendieck Stat-lenses.

Definition 4.3.8. The category $\mathbf{BayesLens}_{\mathcal{C}}$ of Bayesian lenses in $\mathcal{C}$ is the category $\mathbf{Lens}_{\mathsf{Stat}}$ of Grothendieck lenses for the functor Stat. A *Bayesian lens* is a morphism in $\mathbf{BayesLens}_{\mathcal{C}}$. Where the category $\mathcal{C}$ is evident from the context, we will just write $\mathbf{BayesLens}$.

Unpacking this definition, we find that the objects of $\mathbf{BayesLens}_{\mathcal{C}}$ are pairs (X, A) of objects of $\mathcal{C}$. Morphisms (that is, Bayesian lenses) $(X, A) \to (Y, B)$ are pairs $(c, c^\dagger)$ of a channel $c : X \to Y$ and a generalized Bayesian inversion $c^\dagger : B \xrightarrow{X} A$; that is, elements of the hom objects

$$\mathbf{BayesLens}_{\mathcal{C}}\big((X, A), (Y, B)\big) := \mathbf{Lens}_{\mathsf{Stat}}\big((X, A), (Y, B)\big)$$
$$\cong \mathcal{C}(X, Y) \times \mathbf{Set}\big(\mathcal{C}(I, X), \mathcal{C}(B, A)\big).$$

The identity Bayesian lens on (X, A) is $(\mathsf{id}_X, \mathsf{id}_A)$, where by abuse of notation $\mathsf{id}_A : \mathcal{C}(I, Y) \to \mathcal{C}(A, A)$ is the constant map id_A defined in Equation (4.11) that takes any state on Y to the identity on A.

The sequential composite $(d, d^\dagger) \Phi (c, c^\dagger)$ of $(c, c^\dagger) : (X, A) \to (Y, B)$ and $(d, d^\dagger) : (Y, B) \to (Z, C)$ is the Bayesian lens $\big((d \bullet c), (c^\dagger \circ c^* d^\dagger)\big) : (X, A) \to (Z, C)$ where $(c^\dagger \circ c^* d^\dagger) : C \xrightarrow{X} A$ takes a state $\pi : I \to X$ to the channel $c_\pi^\dagger \bullet d_{c \bullet \pi}^\dagger : C \to A$.

To emphasize the structural similarity between Bayesian and monoidal lenses, and visualize the channel $c_\pi^\dagger \bullet d_{c\bullet\pi}^\dagger$, note that following Example 4.2.25, we can depict Bayesian lens composition using the graphical calculus of Boisseau [35] as
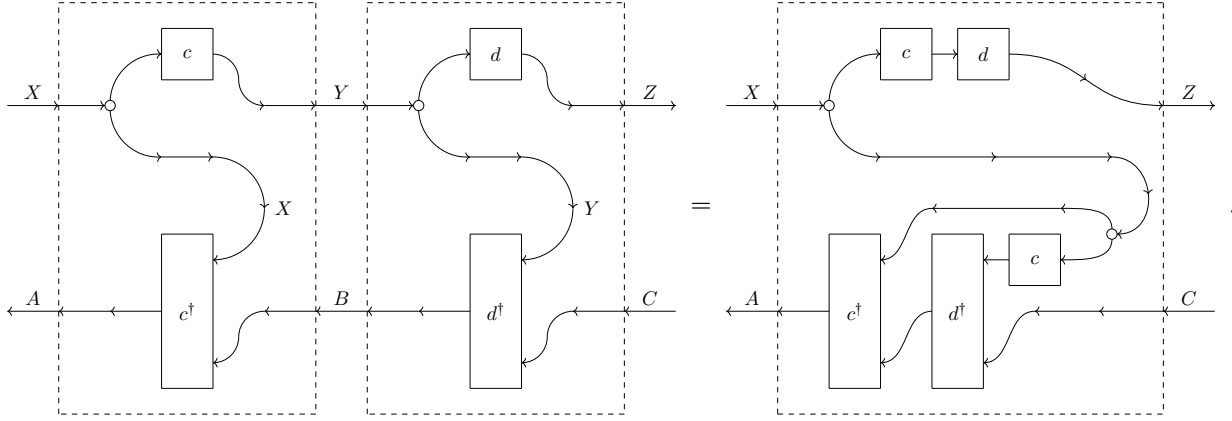
Remark 4.3.9. Strictly speaking, these depictions are diagrams in Boisseau [35]'s calculus of string diagrams for optics, which means that they are not direct depictions of the Bayesian lenses themselves; rather they are depictions of the corresponding optics, which we define and elaborate in [248]. Briefly, these optics are obtained by embedding the categories of forrwards and backwards channels into their corresponding (co)presheaf categories and coupling them together along the 'residual' category $\mathcal{C}$; in the depictions, the string diagrams in the forwards and backwards directions are thus interpreted in these different categories. This explains why we are allowed to 'copy' the channel c in the depiction above, producing the right-hand side by pushing c through the copier as if it were a comonoid morphism: it is because the comonoids in question are $\mathcal{C}(I, X)$ and $\mathcal{C}(I, Y)$, and the function $\mathcal{C}(I, c)$ is indeed a comonoid morphism, even though c is in general not!

Remark 4.3.10. Note that the definition of Stat and hence the definition of $\mathbf{BayesLens}_\mathcal{C}$ do not require $\mathcal{C}$ to be a copy-delete category, even though our motivating categories of stochastic channels are; all that is required for the definition is that $\mathcal{C}$ is monoidal. On the other hand, as we can define Bayesian lenses in any copy-delete category, we can define them in $\mathbf{Set}$, where $\mathbf{Set}(1, X) \cong X$ for every set X: in this case, Bayesian lenses coincide with Cartesian lenses.

Of course, since Stat is a monoidal indexed category, $\mathbf{BayesLens}_\mathcal{C}$ is a monoidal category.

Proposition 4.3.11. $\mathbf{BayesLens}_\mathcal{C}$ is a monoidal category, with structure $\big((\otimes, (I, I)\big)$ inherited from $\mathcal{C}$. On objects, define $(A, A') \otimes (B, B') := (A \otimes A', B \otimes B')$. On morphisms $(f, f^\dagger) : (X, A) \nrightarrow (Y, B)$ and $(g, g^\dagger) : (X', A') \nrightarrow (Y', B')$, define $(f, f^\dagger) \otimes (g, g^\dagger) := (f \otimes g, f^\dagger \otimes g^\dagger)$, where $f^\dagger \otimes g^\dagger : B \otimes B' \xrightarrow{X \otimes X'} A \otimes A'$ acts on states $\omega : I \nrightarrow X \otimes X'$ to return the channel

$f^\dagger_{\omega_X} \otimes g^\dagger_{\omega_{X'}}$, following the definition of the laxator μ in Proposition 4.3.6. The monoidal unit in $\mathbf{BayesLens}_\mathcal{C}$ is the pair (I, I) duplicating the unit in $\mathcal{C}$. When $\mathcal{C}$ is moreover symmetric monoidal, so is $\mathbf{BayesLens}_\mathcal{C}$.

Proof sketch. The main result is immediate from Proposition 4.3.6 and Corollary 4.2.34. When $\otimes$ is symmetric in $\mathcal{C}$, the symmetry lifts to the fibres of Stat and hence to $\mathbf{BayesLens}_\mathcal{C}$. $\qquad\square$

But $\mathbf{BayesLens}_\mathcal{C}$ is not in general a copy-discard category.

Remark 4.3.12. Although $\mathbf{BayesLens}_\mathcal{C}$ is a monoidal category, it does not inherit a copy-discard structure from $\mathcal{C}$, owing to the bidirectionality of its component morphisms. To see this, we can consider morphisms into the monoidal unit (I, I), and find that there is generally no canonical discarding map. For instance, a morphism $(X, A) \rightarrowtail (I, I)$ consists in a pair of a channel $X \rightsquigarrow I$ (which may indeed be a discarding map) and a state-dependent channel $I \overset{X}{\rightsquigarrow} A$, for which there is generally no suitable choice satisfying the comonoid laws. Note, however, that a lens of the type $(X, I) \rightarrowtail (I, B)$ might indeed act by discarding, since we can choose the constant state-dependent channel $B \overset{X}{\rightsquigarrow} I$ on the discarding map $\overline{\overline{\top}} : B \rightsquigarrow I$. By contrast, the Grothendieck category $\int \mathsf{Stat}$ *is* a copy-delete category, as the morphisms $(X, A) \rightarrow (I, I)$ in $\int \mathsf{Stat}$ are pairs $X \rightsquigarrow I$ and $A \overset{X}{\rightsquigarrow} I$, and so for both components we can choose morphisms witnessing the comonoid structure.

4.3.3. Bayesian updates compose optically

In this section we prove the fundamental result that justifies the development of *statistical games* as hierarchical inference systems in Chapter 5: that the Bayesian inversion of a composite channel is given up to almost-equality by the lens composite of the backwards components of the associated 'exact' Bayesian lenses.

Definition 4.3.13. Let $(c, c^\dagger) : (X, X) \rightarrowtail (Y, Y)$ be a Bayesian lens. We say that $(c, c^\dagger)$ is *exact* if c admits Bayesian inversion and, for each $\pi : I \rightsquigarrow X$ such that $c \bullet \pi$ has full support, c and $c^\dagger_\pi$ together satisfy equation (4.5) (p. 133). Bayesian lenses that are not exact are said to be *approximate*.

Theorem 4.3.14. Let $(c, c^\dagger)$ and $(d, d^\dagger)$ be sequentially composable exact Bayesian lenses. Then, for any state π on the domain of c, the contravariant component $c^\dagger \circ c^* d^\dagger$ of the composite lens $(d, d^\dagger) \diamond (c, c^\dagger)$ is the Bayesian inversion of $d \bullet c$. That is to say, *Bayesian updates compose optically*: $(d \bullet c)^\dagger_\pi \overset{d \bullet c \bullet \pi}{\sim} c^\dagger_\pi \bullet d^\dagger_{c \bullet \pi}$.

Proof. For any suitably-typed π, the state-dependent channel $c^\dagger \circ c^* d^\dagger$ returns the channel $c^\dagger_\pi \bullet d^\dagger_{c \bullet \pi}$: $Z \rightarrowtail X$, so to establish the result it suffices to show that
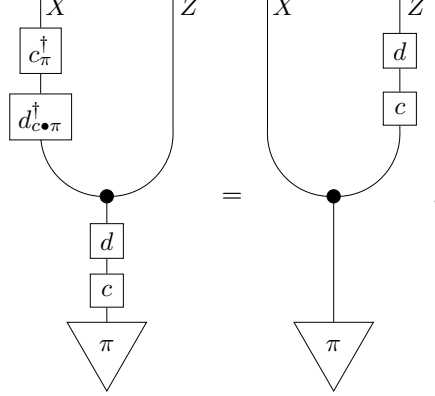
We have
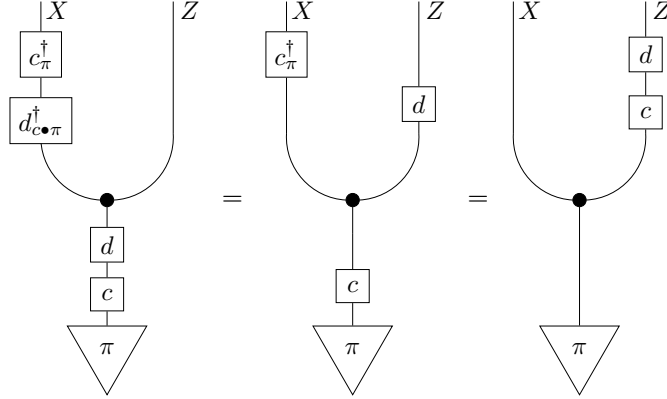
where the first obtains because $d^\dagger_{c \bullet \pi}$ is by assumption a Bayesian inverse of d with respect to $c \bullet \pi$, and the second because $c^\dagger_\pi$ is likewise a Bayesian inverse of c with respect to π. Hence, $c^\dagger_\pi \bullet d^\dagger_{c \bullet \pi}$ and $(d \bullet c)^\dagger_\pi$ are both Bayesian inversions of $d \bullet c$ with respect to π. Since Bayesian inversions are almost-equal (Proposition 4.1.29), we have $c^\dagger_\pi \bullet d^\dagger_{c \bullet \pi} \overset{d \bullet c \bullet \pi}{\sim} (d \bullet c)^\dagger_\pi$, as required. $\square$

This theorem has the following important immediate consequence.

Corollary 4.3.15. Suppose $\mathcal{C}^\dagger$ is a subcategory of $\mathcal{C}$ all of whose channels admit Bayesian inversion, and consider the restriction to $\mathcal{C}^\dagger$ of the fibration $\pi_{\mathsf{Lens}} : \mathbf{BayesLens}_\mathcal{C} \to \mathcal{C}$ of Bayesian lenses, denoted $\pi^\dagger_{\mathsf{Lens}}$. Then there is an almost sure section $\dagger : \mathcal{C}^\dagger \to \mathbf{BayesLens}_\mathcal{C}$ of $\pi^\dagger_{\mathsf{Lens}}$ taking each object X to (X, X) and each channel $c : X \rightarrowtail Y$ to a lens $(c, c^\dagger) : (X, X) \rightarrowtail (Y, Y)$ where $c^\dagger$ is an almost-surely unique Bayesian inversion of c. Hence the composite $\mathcal{C} \overset{\dagger}{\to} \mathbf{BayesLens}_\mathcal{C} \xrightarrow{\pi^\dagger_{\mathsf{Lens}}} \mathcal{C}$ is equal to the identity functor $\mathsf{id}_\mathcal{C}$.

Remark 4.3.16. A morphism $\sigma : B \to E$ is a section of $\pi : E \to B$ when $\pi \circ \sigma = \mathsf{id}_B$. In standard category theory, a section of a fibration π is therefore a functor: but, since Bayesian

inversion is only defined up to almost-equality, the functoriality of the preceding corollary is accordingly weakened. This leads to the notion of *almost sure section*, which we formalize by lifting the relation of almost-equality from $\mathcal{C}$ to $\mathbf{BayesLens}_{\mathcal{C}}$, as follows. Suppose $(c, c^{\sharp})$ and $(d, d^{\sharp})$ are lenses $(X, X) \rightarrow (Y, Y)$. Then we may say that they are equivalent up to almost equality, denoted $(c, c^{\sharp}) \approx (d, d^{\sharp})$, if for all states $\alpha : I \rightarrow X$, we have $c \overset{\alpha}{\sim} d$ and $c^{\sharp}_{\alpha} \overset{c \bullet \alpha}{\sim} d^{\sharp}_{\alpha}$. If additionally we have $c = d$, we write $(c, c^{\sharp}) \simeq (d, d^{\sharp})$ and say that they are strongly equivalent. Note then that the mapping † of the preceding corollary is functorial up to this strong equivalence: $\dagger(d) \phi \dagger(c) \simeq \dagger(d \bullet c)$; this is what we mean by *almost sure section*. We believe this notion (and the implicit more general one of *almost sure functor*) to be new, but do not study it further here.

Remark 4.3.17. In the context of finitely-supported probability (*i.e.*, in $\mathcal{K}\ell(\mathcal{D})$), almost-equality coincides with simple equality over the support, and so Bayesian inversions are then just equal (over the support). This suggests that, in this context, † may be strengthened to a strict functor: but the qualification *over the support* means we must use the machinery of *dependent Bayesian lenses* (Remark 4.1.20); then, † does yield a strict functor.

Remark 4.3.18. Note that the functor † is not monoidal, because inverting the tensor of two channels with respect to a joint distribution is not the same as inverting the two channels independently with respect to the marginals and tensoring them together (unless the joint is already the product of two independent states); that is, $(c \otimes d)^{\dagger}_{\omega} \neq c^{\dagger}_{\omega_1} \otimes d^{\dagger}_{\omega_2}$, where ω_1 and ω_2 are the two marginals of the joint state ω. Technically, this situation obtains because there is no channel $X_1 \otimes X_2 \rightarrow X_1 \otimes X_2$ that performs this marginalization-then-tensoring that could play the part of the laxator of †. (But note that typically a probability monad $\mathcal{P}$ will be 'bimonoidal', with the 'opmonoidal' structure $\mathcal{P}(X_1 \times X_2) \rightarrow \mathcal{P}X_1 \times \mathcal{P}X_2$ witnessing this joint-marginalization operation [110, §4]; the technical hurdle is that this structure typically interacts nicely with the monad structure, since the tensor of two Dirac deltas is again a Dirac delta.)

In §5.4, we will use the machinery of statistical games to measure the error produced by inverting two channels independently, versus inverting them jointly.

Historically, lenses have often been associated with 'lens laws': additional axioms guaranteeing their well-behavedness. These laws originate in the context of database systems, and we now investigate how well they are satisfied by Bayesian lenses, where one might see an inference system as a kind of uncertain database. We will find that Bayesian lenses are not lawful in this traditional sense, because they 'mix' information.

4.3.4. Lawfulness of Bayesian lenses

The study of Cartesian lenses substantially originates in the context of bidirectional transformations of data in the computer science and database community [34, 99], where we can think of the view (or get) function as returning part of a database record, and the update (or put) function as 'putting' a part into a record and returning the updated record. In this setting, axioms known as *lens laws* can be imposed on lenses to ensure that they are 'well-behaved' with respect to database behaviour: for example, that updating a record with some data is idempotent (the 'put-put' law).

We might hope that well-behaved or "very well-behaved" lenses in the database context should roughly correspond to our notion of exact Bayesian lens: with the view that an inference system, formalized by a Bayesian lens, is something like a probabilistic database. However, as we will see, even exact Bayesian lenses are only weakly lawful in the database sense: Bayesian updating mixes information in the prior state (the 'record') with the observation (the 'data'), rather than replacing the prior information outright.

We will concentrate on the three lens laws that have attracted recent study [35, 217]: `GetPut`, `PutGet`, and `PutPut`. A Cartesian lens satisfying the former two is *well-behaved* while a lens satisfying all three is *very well-behaved*, in the terminology of Foster et al. [99]. Informally, `GetPut` says that getting part of a record and putting it straight back returns an unchanged record; `PutGet` says that putting a part into a record and then getting it returns the same part that we started with; and `PutPut` says that putting one part and then putting a second part has the same effect on a record as just putting the second part (that is, update completely overwrites the part in the record). We will express these laws graphically, and consider them each briefly in turn.

Note first that we can lift any channel c in the base category $\mathcal{C}$ into any state-dependent fibre $\mathsf{Stat}(A)$ using the constant (identity-on-objects) functor taking c to the constant-valued state-indexed channel $\rho \mapsto c$ that maps any state ρ to c. We can lift string diagrams in $\mathcal{C}$ into the fibres accordingly.

GetPut

Definition 4.3.19. A lens $(c, c^\dagger)$ is said to satisfy the `GetPut` law if it satisfies the left equality in (4.12) below. Equivalently, because the copier induced by the Cartesian product is natural (*i.e.*, $\curlyvee \circ f = (f \times f) \circ \curlyvee$), for any state π, we say that $(c, c^\dagger)$ satisfies `GetPut` with respect to π if it

satisfies the right equality in (4.12) below.

$$
\begin{array}{ccc}
\begin{array}{c}
\boxed{c^\dagger} \\
\boxed{c} \\
\cup
\end{array}
= &
\implies &
\begin{array}{c}
\boxed{c^\dagger_\pi} \\
\boxed{c} \\
\nabla_\pi
\end{array}
= \;
\nabla_\pi
\end{array}
\tag{4.12}
$$

(Note that here we have written the copying map as $\curlyvee$, since we are assuming an ambient Cartesian monoidal structure; hence for a Bayesian lens we interpret the left diagram above in the image of the Yoneda embedding.)

Proposition 4.3.20. When c is causal, the exact Bayesian lens $(c, c^\dagger)$ satisfies the GetPut law with respect to any state π for which c admits Bayesian inversion.

Proof. Starting from the right-hand-side of (4.12), we have the following chain of equalities

$$
= \qquad = \qquad = \qquad = \qquad
$$

where the first holds by the counitality of $\curlyvee$, the second by the causality of c, the third since c admits Bayesian inversion (4.5) with respect to π, and the fourth again by counitality. $\qquad\square$

Note that by Bayes' law, exact Bayesian lenses only satisfy GetPut with respect to states. This result means that, if we think of c as generating a prediction $c \bullet \pi$ from a prior belief π, then if our observation exactly matches the prediction, updating the prior π according to Bayes' rule results in no change.

PutGet   The `PutGet` law is characterized for a lens (v, u) by the following equality:

$$\text{(diagram)}$$
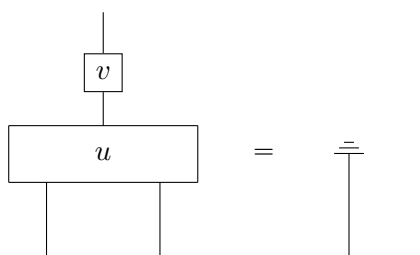
In general, `PutGet` does not hold for exact Bayesian lenses $(c, c^\dagger)$. However, because `GetPut` holds with respect to states π, we do have $c \bullet c_\pi^\dagger \bullet c \bullet \pi = c \bullet \pi$; that is, `PutGet` holds for exact Bayesian lenses $(c, c^\dagger)$ for the prior π and 'input' $c \bullet \pi$.

The reason `PutGet` fails to hold in general is that Bayesian updating mixes information from the prior and the observation, according to the strength of belief. Consequently, updating a belief according to an observed state and then producing a new prediction need not result in the same state as observed; unless, of course, the prediction already matches the observation.

PutPut   Finally, the `PutPut` law for a lens (v, u) is characterized by the following equality:

$$\text{(diagram)}$$
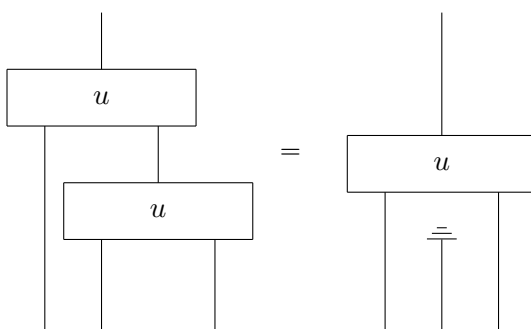
`PutPut` fails to hold for exact Bayesian lenses for the same reason that `PutGet` fails to hold in general: updates mix old and new beliefs, rather than entirely replace the old with the new.

Comment   In the original context of computer databases, there is assumed to be no uncertainty, so a 'belief' is either true or false. Consequently, there can be no 'mixing' of beliefs; and in database applications, such mixing may be highly undesirable. Bayesian lenses, on the other hand, live in a fuzzier world: our present interest in Bayesian lenses originates in their application to describing cognitive and cybernetic processes such as perception and action, and here the ability to mix beliefs according to uncertainty is desirable.

Possibly it would be of interest to give analogous information-theoretic lens laws that characterize exact and approximate Bayesian lenses and their generalizations; and we might then expect the

'Boolean' lens laws to emerge in the extremal case where there is no uncertainty and only Dirac states. We leave such an endeavour for future work: Bayes' law (4.5) is sufficiently concise and productive for our purposes here.

5. Statistical games

In this chapter, we characterize a number of well known systems of approximate inference as *loss models* (defined in §5.3.2): lax sections of 2-fibrations of statistical games, themselves constructed (in §5.3.1) by attaching internally-defined loss functions to Bayesian lenses. Our examples include the relative entropy (§5.3.3.1), which constitutes a *strict* section, and whose chain rule is formalized by the horizontal composition of the 2-fibration. In order to capture this compositional structure, we first introduce the notion of 'copy-composition' (in §5.2.1), alongside corresponding bicategories through which the composition of copy-discard categories factorizes. These latter bicategories are obtained as a variant of the **Copara** construction [54, §2] (dual to the internal parameterization of §3.2.1), and so we additionally introduce coparameterized Bayesian lenses (§5.2.3), proving that coparameterized Bayesian updates compose optically (§5.2.4), as in the non-coparameterized case.

Besides the relative entropy, our other examples of loss models are given by maximum likelihood estimation (§5.3.3.2), the free energy (which gives us in §5.3.3.3 a characterization of *autoencoders*), and the 'Laplace' approximation to the free energy (§5.3.3.4). It is this latter loss model which will, in Chapter 7, finally yield the dynamical semantics for predictive coding.

We begin with a discussion of compositional approximate inference from the 'lens' perspective, focusing on the relative entropy.

5.1. Compositional approximate inference, via the chain rule for relative entropy

In Chapter 4, we observed that the Bayesian inversion of a composite stochastic channel is (almost surely) equal to the 'lens composite' of the inversions of the factors; that is, *Bayesian updates compose optically* ('BUCO', Theorem 4.3.14). Formalizing this statement for a given category $\mathcal{C}$ yields a fibration of *Bayesian lenses* as a Grothendieck construction of the indexed category of state-dependent channels (Definition 4.3.8), and Bayesian inversion almost surely yields a section † of the corresponding fibration (Corollary 4.3.15). This section † picks out a special class of Bayesian lenses, which we call *exact* as they compute 'exact' inversions (Definition 4.3.13), but although

the category $\mathbf{BayesLens}(\mathcal{C})$ has many other morphisms, its construction is not extravagant: by comparison, we think of the non-exact lenses as representing *approximate* inference systems. This is particularly necessary in computational applications, because computing exact inversions is usually intractable, but this creates a new problem: choosing an approximation, and measuring its performance. In this chapter, we formalize this process, by attaching *loss functions* to Bayesian lenses, thus creating another fibration, of *statistical games*. Sections of this latter fibration encode compositionally well-behaved systems of approximation that we call *loss models*.

A classic example of a loss model will be supplied by the relative entropy, which in some sense measures the 'divergence' between distributions: the game here is then to minimize the divergence between the approximate and exact inversions. If π and π' are two distributions on a space X, with corresponding density functions p_π and $p_{\pi'}$ (both with respect to a common measure), then their relative entropy $D(\pi, \pi')$ is the real number given by $\mathbb{E}_{x \sim \pi} [\log p_\pi(x) - \log p_{\pi'}(x)]$[1]. Given a pair of channels $\alpha, \alpha' : A \to B$ (again commensurately associated with densities), we can extend D to a map $D_{\alpha, \alpha'} : A \to \mathbb{R}_+$ in the natural way, writing $a \mapsto D\big(\alpha(a), \alpha'(a)\big)$. We can assign such a map $D_{\alpha, \alpha'}$ to any such parallel pair of channels, and so, following the logic of composition in $\mathcal{C}^2$, we might hope for the following equation to hold for all $a : A$ and composable parallel pairs $\alpha, \alpha' : A \to B$ and $\beta, \beta' : B \to C$:

$$D_{\beta \bullet \alpha, \beta' \bullet \alpha'}(a) = \mathbb{E}_{b \sim \alpha(a)} \big[D_{\beta, \beta'}(b) \big] + D_{\alpha, \alpha'}(a)$$

The right-hand side is known as the *chain rule* for relative entropy, but, unfortunately, the equation does *not* hold in general, because the composites $\beta \bullet \alpha$ and $\beta' \bullet \alpha'$ each involve an extra expectation:

$$D_{\beta \bullet \alpha, \beta' \bullet \alpha'}(a) = \mathbb{E}_{c \sim \beta \bullet \alpha(a)} \big[\log p_{\beta \bullet \alpha(a)}(c) - \log p_{\beta' \bullet \alpha'(a)}(c) \big]$$

$$= \mathbb{E}_{c \sim \beta \bullet \alpha(a)} \Big[\log \mathbb{E}_{b \sim \alpha(a)} \big[p_\beta(c|b) \big] - \log \mathbb{E}_{b \sim \alpha'(a)} \big[p_{\beta'}(c|b) \big] \Big]$$

However, we *can* satisfy an equation of this form by using 'copy-composition': if we write $\curlyvee_B$ to denote the canonical 'copying' comultiplication on B, and define $\beta \bullet^2 \alpha := (\mathrm{id}_B \otimes \beta) \bullet \curlyvee_B \bullet \alpha$, as depicted by the string diagram
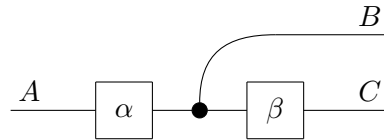
[1] For details about this 'expectation' notation $\mathbb{E}$, see 5.3.19.

[2] In which, following the Chapman-Kolmogorov rule, a composite channel $\beta \bullet \alpha$ can be expressed as the expectation of β under α, *i.e.* $a \mapsto \mathbb{E}_{b \sim \alpha(a)} [\beta(b)]$.

then $D_{\beta \bullet^2 \alpha, \beta' \bullet^2 \alpha'}(a)$ *does* equal the chain-rule form on the right-hand side:

$$
\begin{aligned}
D_{\beta \bullet^2 \alpha, \beta' \bullet^2 \alpha'}(a) &= \mathop{\mathbb{E}}_{b \sim \alpha(a)} \mathop{\mathbb{E}}_{c \sim \beta(b)} \left[\log p_\beta(c|b) p_\alpha(b|a) - \log p_{\beta'}(c|b) p_{\alpha'}(b|a) \right] \\
&= \mathop{\mathbb{E}}_{b \sim \alpha(a)} \left[\mathop{\mathbb{E}}_{c \sim \beta(b)} \left[\log p_\beta(c|b) - \log p_{\beta'}(c|b) \right] + \log p_\alpha(b|a) - \log p_{\alpha'}(b|a) \right] \\
&= \mathop{\mathbb{E}}_{b \sim \alpha(a)} \left[D_{\beta, \beta'}(b) \right] + D_{\alpha, \alpha'}(a)
\end{aligned}
$$

where the second line follows by the linearity of expectation. This result exhibits a general pattern about copy-discard categories (Definition 3.1.3) such as $\mathcal{C}$: composition $\bullet$ can be decomposed into first copying $\curlyvee$, and then discarding $\bar{\top}$. If we don't discard, then we retain the 'intermediate' variables, and this results in a functorial assignment of relative entropies to channels.

The rest of this chapter is dedicated to making use of this observation to construct loss models, including (but not restricted to) the relative entropy. The first complication that we encounter is that copy-composition is not strictly unital, because composing with an identity retains an extra variable. To deal with this, in §5.2, we construct a *bicategory* of copy-composite channels, extending the **Copara** construction, and build coparameterized (copy-composite) Bayesian lenses accordingly; we also prove a corresponding BUCO result. In §5.3, we then construct 2-fibrations of statistical games, defining loss functions internally to any copy-discard category $\mathcal{C}$ that admits "bilinear effects". Because we are dealing with approximate systems, the 2-dimensional structure of the construction is useful: loss models are allowed to be *lax* sections. We then characterize the relative entropy, maximum likelihood estimation, the free energy, and the 'Laplacian' free energy as such loss models.

Unsurprisingly, each of these loss functions are moreover (lax) monoidal, and, assuming $\mathcal{C}$ is symmetric monoidal, each of the constructions mentioned here result in monoidal (2-)fibrations. We explore this monoidal structure in §5.4.

5.2. 'Copy-composite' Bayesian lenses

5.2.1. Copy-composition by coparameterization

In a locally small copy-discard category $\mathcal{C}$ (Definition 3.1.3), every object A is equipped with a canonical comonoid structure $(\curlyvee_A, \bar{\top}_A)$, and so, by the comonoid laws (Definition 3.4.21), it is almost a triviality that the composition functions $\bullet : \mathcal{C}(B, C) \times \mathcal{C}(A, B) \to \mathcal{C}(A, C)$ factorize as

$$
\mathcal{C}(B, C) \times \mathcal{C}(A, B) \xrightarrow{(\mathrm{id}_B \otimes -) \times \mathcal{C}\left(\mathrm{id}_A, \curlyvee_B\right)} \mathcal{C}(B \otimes B, B \otimes C) \times \mathcal{C}(A, B \otimes B) \cdots
$$

$$
\cdots \xrightarrow{\bullet} \mathcal{C}(A, B \otimes C) \xrightarrow{\mathcal{C}(\mathrm{id}_A, \mathrm{proj}_C)} \mathcal{C}(A, C)
$$

where the first factor copies the B output of the first morphism and tensors the second morphism with the identity on B, the second factor composes the latter tensor with the copies, and the third discards the extra copy of B[3]. This is, however, only *almost* trivial, since it witnesses the structure of Chapman-Kolmogorov style composition in categories of stochastic channels such as $\mathcal{K\ell}(\mathcal{D})$, the Kleisli category of the (finitary) distributions monad $\mathcal{D} : \mathbf{Set} \to \mathbf{Set}$ (§4.1.1.1). There, given channels $c : A{\rightarrow}B$ and $d : B{\rightarrow}C$, the composite $d \bullet c$ is formed first by constructing the 'joint' channel, denoted $d \bullet^2 c$ and defined by $(d \bullet^2 c)(b, c|a) := d(c|b)c(b|a)$, and then discarding (marginalizing over) $b : B$, giving

$$(d \bullet c)(c|a) = \sum_{b:B}(d \bullet^2 c)(b, c|a) = \sum_{b:B} d(c|b)c(b|a) \, .$$

Of course, the channel $d \bullet^2 c$ is not a morphism $A{\rightarrow}C$, but rather $A{\rightarrow}B \otimes C$; that is, it is *coparameterized* by B, in a sense formally dual to the notion of parameterization of §3.2.1.

Moreover, as noted above, $\bullet^2$ is not strictly unital: given the composites $\mathsf{id}_B \, \overline{\bullet}^2 f$ and $f \overline{\bullet}^2 \mathsf{id}_A$, we need 2-cells that discard the coparameters introduced by the copying; and, inversely, we need 2-cells $f \mapsto \mathsf{id}_B \, \overline{\bullet}^2 f$ and $f \mapsto f \overline{\bullet}^2 \mathsf{id}_A$ that introduce them. The former are of course given by the discarding structure

(5.1)

(5.2)

while the latter are given by copying:

(5.3)

---

[3] We define $\mathsf{proj}_C := B \otimes C \xrightarrow{\overline{\overline{\top}}_B \otimes \mathsf{id}_C} I \otimes C \xrightarrow{\lambda_C} C$, using the comonoid counit $\overline{\overline{\top}}_B$ and the left unitor λ_C of $\mathcal{C}$'s monoidal structure.

$$(5.4)$$

These putative 2-cells clearly need access to copies of the domain and codomain of f, and hence are not available in the standard **Copara** construction obtained by formally dualizing **Para**. For this reason, we construct a bicategory $\mathbf{Copara}_2(\mathcal{C})$ as a variant of the **Copara** construction, in which a 1-cell $A \to B$ may be any morphism $A \nrightarrow M \otimes B$ in $\mathcal{C}$, and where horizontal composition is precisely copy-composition. We will henceforth drop the cumbersome notation $\bullet^2$, and write simply $\bullet$ for horizontal composition in $\mathbf{Copara}_2(\mathcal{C})$, matching the composition operator of $\mathcal{C}$ itself. (Later, if we need to be explicit about horizontal composition, we will sometimes use the symbol $\diamond$.)

Theorem 5.2.1. Let $(\mathcal{C}, \otimes, I)$ be a copy-discard category. Then there is a bicategory $\mathbf{Copara}_2(\mathcal{C})$ as follows. Its 0-cells are the objects of $\mathcal{C}$. A 1-cell $f : A \xrightarrow[M]{} B$ is a morphism $f : A \to M \otimes B$ in $\mathcal{C}$. A 2-cell $\varphi : f \Rightarrow f'$, with $f : A \xrightarrow[M]{} B$ and $f' : A \xrightarrow[M']{} B$, is a morphism $\varphi : A \otimes M \otimes B \to M'$ of $\mathcal{C}$, satisfying the *change of coparameter* axiom:

Given 2-cells $\varphi : f \Rightarrow f'$ and $\varphi' : f' \Rightarrow f''$, their vertical composite $\varphi' \odot \varphi : f \Rightarrow f''$ is given by the following string diagram:
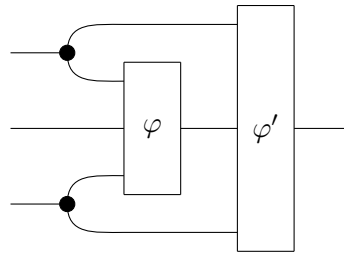
The identity 2-cell $\mathrm{id}_f : f \Rightarrow f$ on $f : A \xrightarrow[M]{} B$ is given by the projection morphism $\mathrm{proj}_M : A \otimes M \otimes B \to M$ obtained by discarding A and B, as in footnote 3.

The horizontal composite $g \circ f : A \xrightarrow[M \otimes B \otimes N]{} C$ of 1-cells $f : A \xrightarrow[M]{} B$ then $g : B \xrightarrow[N]{} C$ is given

by the following string diagram in $\mathcal{C}$:
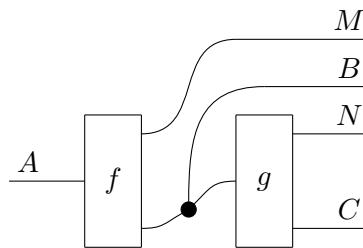
Strictly speaking, we define the coparameter of $g \circ f$ to be $(M \otimes B) \otimes N$. The identity 1-cell id_A on A is given by the inverse of the left unitor of the monoidal structure on $\mathcal{C}$, *i.e.* $\mathrm{id}_A := \lambda_A^{-1} : A \underset{I}{\to} A$, with coparameter thus given by the unit object I.

The horizontal composite $\gamma \circ \varphi : (g \circ f) \Rightarrow (g' \circ f')$ of 2-cells $\varphi : f \Rightarrow f'$ and $\gamma : g \Rightarrow g'$ is given by the string diagram
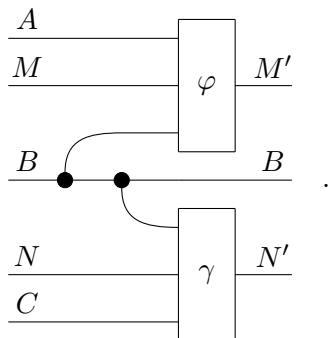
Proof. To show that $\mathbf{Copara}_2(\mathcal{C})$ is a bicategory, we need to establish the unitality and associativity of vertical composition; verify that horizontal composition is well-defined and functorial; establish the weak unitality and associativity of horizontal composition; and confirm that the corresponding unitors and associator satisfy the bicategorical coherence laws. Then, to prove that $\mathbf{Copara}_2(\mathcal{C})$ is moreover monoidal, we need to demonstrate that the tensor as proposed satisfies the data of a monoidal bicategory. However, since the monoidal structure is inherited from that of $\mathcal{C}$, we will elide much of this latter proof, and demonstrate only that the tensor is functorial; the rest follows straightforwardly but tediously.

We begin by confirming that vertical composition $\odot$ is unital and associative. To see that $\odot$ is unital, simply substitute the identity 2-cell (given by projection onto the coparameter) into the string diagram defining $\odot$ and then apply the comonoid counitality law twice (once on the left, once on the right). The associativity of $\odot$ requires that $\varphi'' \odot (\varphi' \odot \varphi) = (\varphi'' \odot \varphi') \odot \varphi$, which

corresponds to the following graphical equation:
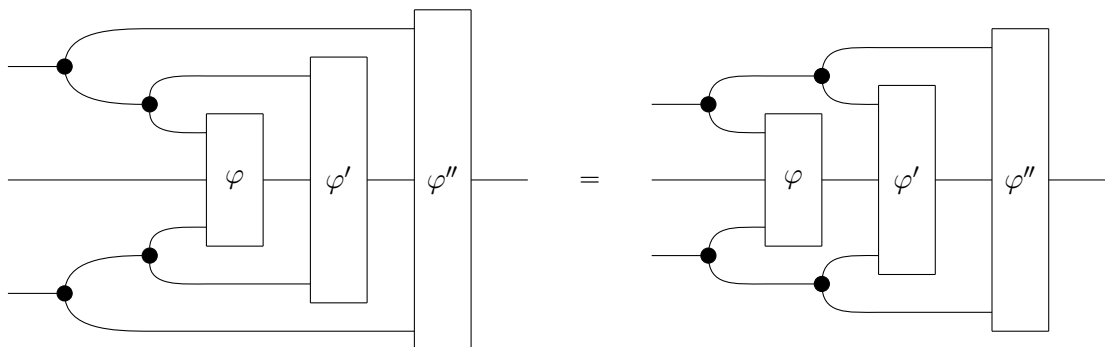
To see that this equation is satisfied, simply apply the comonoid coassociativity law twice (once left, once right).

Next, we check that horizontal composition ∘ is well-defined, which amounts to checking whether the horizontal composite of 2-cells satisfies the change of coparameter axiom. Again, we reason graphically. Given 2-cells φ and γ between composable pairs of 1-cells f, f' and g, g', our task is to verify that
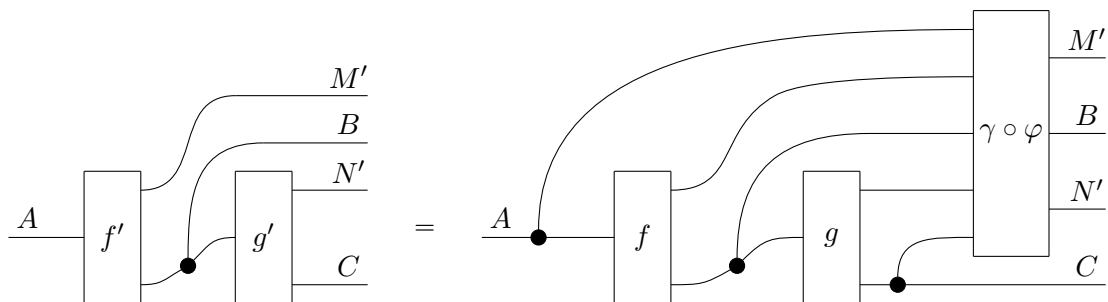
Since φ and γ satisfy change of coparameter *ex hypothesi*, the left hand side is equal to the morphism
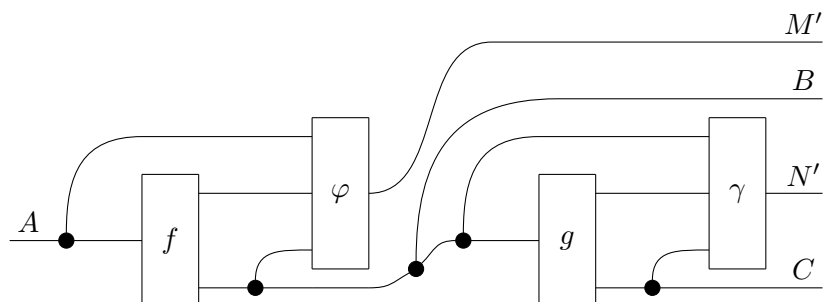
By comonoid coassociativity, this is in turn equal to
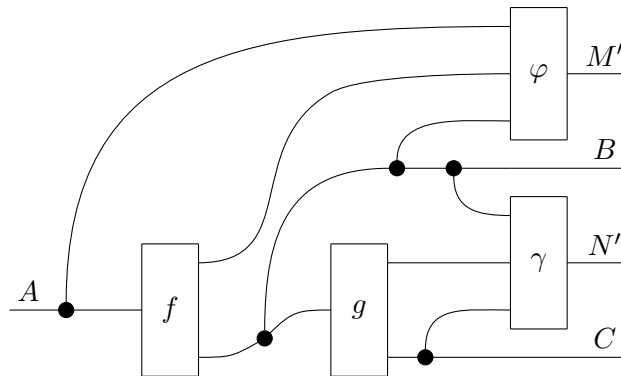
which, by the definition of ∘, is precisely equal to
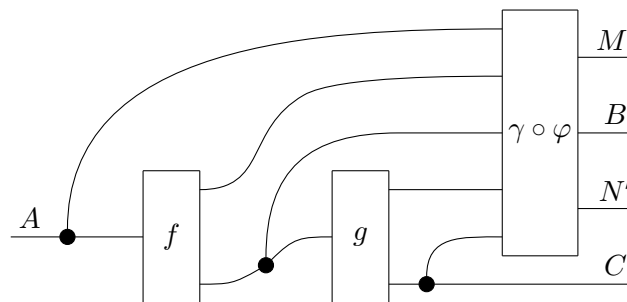
and so this establishes the result.

We now verify that ∘ so defined is functorial on 2-cells, beginning with the preservation of composition. We need to validate the equation $(\gamma' \circ \varphi') \odot (\gamma \circ \varphi) = (\gamma' \odot \gamma) \circ (\varphi' \odot \varphi)$ (for appropriately composable 2-cells). This amounts to checking the following equation, which can be seen to hold by two applications of comonoid coassociativity:
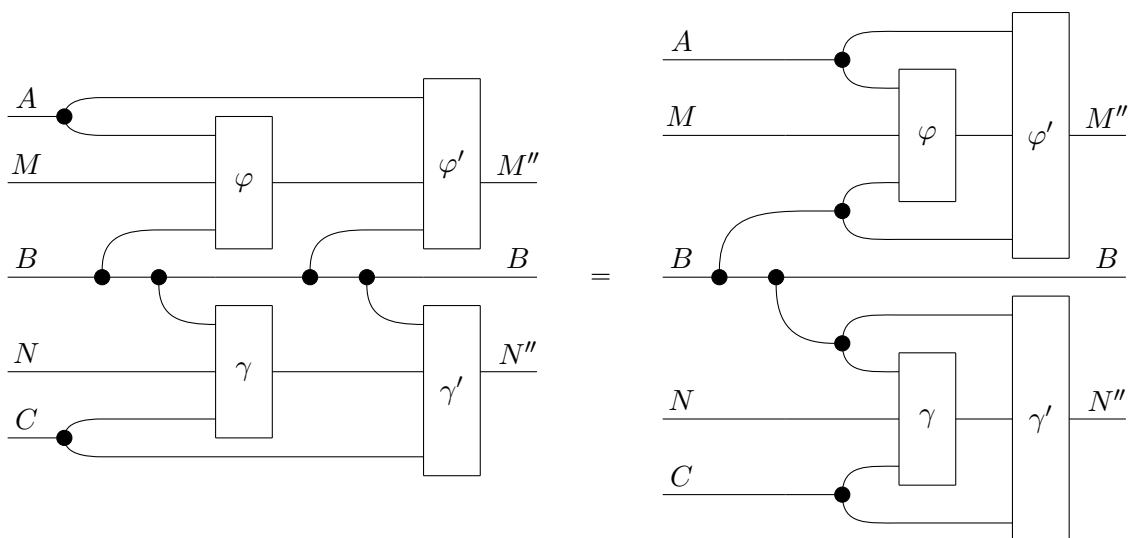
It is easy to verify that ∘ preserves identities, *i.e.* that $\mathrm{id}_g \circ \mathrm{id}_f = \mathrm{id}_{g \circ f}$; just substitute the identity 2-cells into the definition of ∘ on 2-cells, and apply comonoid counitality four times.

Next, we establish that horizontal composition is weakly associative, which requires us to supply isomorphisms $\alpha_{f,g,h} : (h \circ g) \circ f \Rightarrow h \circ (g \circ f)$ natural in composable triples of 1-cells h, g, f. Supposing the three morphisms have the types $f : A \xrightarrow{M} B$, $g : B \xrightarrow{N} C$, and $h : C \xrightarrow{O} D$, we can choose $a_{f,g,h}$ to be the 2-cell represented by the morphism

$$A \otimes \big((M \otimes B) \otimes ((N \otimes C) \otimes O)\big) \otimes D \xrightarrow{\mathsf{proj}} (M \otimes B) \otimes ((N \otimes C) \otimes O) \cdots$$

$$\cdots \xrightarrow{\alpha^{\mathcal{C}}_{(M \otimes B),(N \otimes C),O}} ((M \otimes B) \otimes (N \otimes C)) \otimes O \cdots$$

$$\cdots \xrightarrow{\alpha^{\mathcal{C}}_{(M \otimes B),N,C} \otimes \mathsf{id}_O} (((M \otimes B) \otimes N) \otimes C) \otimes O$$

where the first factor is the projection onto the coparameter and $\alpha^{\mathcal{C}}$ denotes the associator of the monoidal structure $(\otimes, I)$ on $\mathcal{C}$. In the inverse direction, we can choose the component $\alpha^{-1}_{f,g,h} : h \circ (g \circ f) \Rightarrow (h \circ g) \circ f$ to be the 2-cell represented by the morphism

$$A \otimes \big((((M \otimes B) \otimes N) \otimes C) \otimes O\big) \otimes D \xrightarrow{\mathsf{proj}} (((M \otimes B) \otimes N) \otimes C) \otimes O \cdots$$

$$\cdots \xrightarrow{\alpha^{\mathcal{C},-1}_{(M \otimes B),N,C} \otimes \mathsf{id}_O} ((M \otimes B) \otimes (N \otimes C)) \otimes O \cdots$$

$$\cdots \xrightarrow{\alpha^{\mathcal{C},-1}_{(M \otimes B),(N \otimes C),O}} (M \otimes B) \otimes ((N \otimes C) \otimes O)$$

where $\alpha^{\mathcal{C},-1}$ denotes the inverse of the associator on $(\mathcal{C}, \otimes, I)$. That the pair of $\alpha_{f,g,h}$ and $\alpha^{-1}_{f,g,h}$ constitutes an isomorphism in the hom category follows from the counitality of the comonoid structures. That this family of isomorphisms is moreover natural follows from the naturality of the associator on $(\mathcal{C}, \otimes, I)$.

We come to the matter that motivated the construction of $\mathbf{Copara}_2(\mathcal{C})$: the weak unitality of copy-composition, witnessed here by the weak unitality of horizontal composition. We need to exhibit two families of natural isomorphisms: the left unitors with components $\lambda_f : \mathsf{id}_B \circ f \Rightarrow f$, and the right unitors with components $\rho_f : f \circ \mathsf{id}_A \Rightarrow f$, for each morphism $f : A \xrightarrow{M} B$. Each such component will be defined by a projection morphism, and weak unitality will then follow from the counitality of the comonoid structures. More explicitly, λ_f is witnessed by $\mathsf{proj}_M : A \otimes M \otimes B \otimes B \to M$; its inverse λ_f^{-1} is witnessed by $\mathsf{proj}_{M \otimes B} : A \otimes M \otimes B \to M \otimes B$; ρ_f is witnessed by $\mathsf{proj}_M : A \otimes A \otimes M \otimes B \to M$; and its inverse ρ_f^{-1} is witnessed by $\mathsf{proj}_{A \otimes M} : A \otimes M \otimes B \to A \otimes M$. Checking that these definitions give natural isomorphisms is then an exercise in counitality that we leave to the reader.

All that remains of the proof that $\mathbf{Copara}_2(\mathcal{C})$ is indeed a bicategory is to check that the unitors are compatible with the associator (i.e., $(\mathsf{id}_g \circ \lambda_f) \odot \alpha_{g,\mathsf{id}_B,f} = \rho_g \circ \mathsf{id}_f$) and that associativity is order-dependent (i.e., the associator α satisfies the pentagon diagram). The latter follows immediately

from the corresponding fact about the associator $\alpha^{\mathcal{C}}$ on $(\mathcal{C}, \otimes, I)$. To demonstrate the former, it is easier to verify that $(\mathsf{id}_g \circ \lambda_f) \odot \alpha_{g,\mathsf{id}_B,f} \odot (\rho_g^{-1} \circ \mathsf{id}_f) = \mathsf{id}_{g \circ f}$. This amounts to checking that the following string diagram is equally a depiction of the morphism underlying $\mathsf{id}_{g \circ f}$:
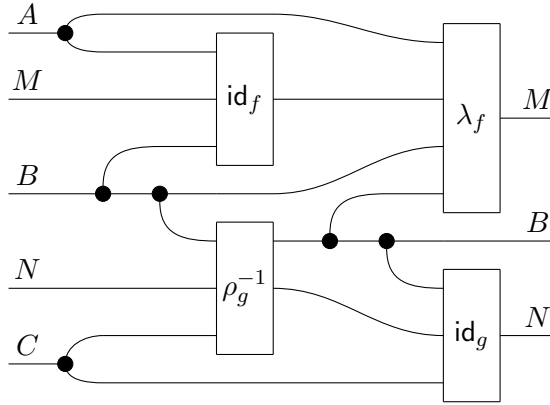
(Note that here we have elided the associator from the depiction. This is allowed by comonoid counitality, and because string diagrams are blind to bracketing.) Substituting the relevant morphisms into the boxes, we see that this diagram is equal to
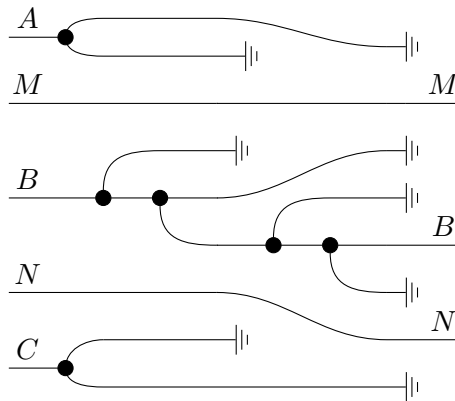
and six applications of counitality give us $\mathsf{id}_{g \circ f}$. This establishes that $\mathbf{Copara}_2(\mathcal{C})$ is a bicategory.

$\square$

Remark 5.2.2. When $\mathcal{C}$ is symmetric monoidal, $\mathbf{Copara}_2(\mathcal{C})$ inherits a monoidal structure, elaborated in Proposition 5.4.1.

Remark 5.2.3. In order to capture the bidirectionality of Bayesian inversion we will need to consider left- and right-handed versions of the $\mathbf{Copara}_2$ construction. These are formally dual, and when $\mathcal{C}$ is symmetric monoidal (as in most examples) they are isomorphic. Nonetheless, it makes formalization easier if we explicitly distinguish $\mathbf{Copara}_2^l(\mathcal{C})$, in which the coparameter is placed on the left of the codomain (as above), from $\mathbf{Copara}_2^r(\mathcal{C})$, in which it is placed on the right. Aside from the swapping of this handedness, the rest of the construction is the same.

We end this section with three easy (and ambidextrous) propositions relating $\mathcal{C}$ and $\mathbf{Copara}_2(\mathcal{C})$.

Proposition 5.2.4. There is an identity-on-objects lax embedding $(-)^{\curlyvee} : \mathcal{C} \hookrightarrow \mathbf{Copara}_2(\mathcal{C})$, mapping $f : X \to Y$ to $f^{\curlyvee} : X \underset{I}{\to} Y$, which (in the left-handed case) has the underlying morphism $X \xrightarrow{f} Y \xrightarrow{\lambda_Y^{-1}} I \otimes Y$ (where λ is the left unitor of the monoidal structure on $\mathcal{C}$). The laxator $\iota(g) \circ \iota(f) \to \iota(g \circ f)$ discards the coparameter obtained from copy-composition.

Remark 5.2.5. We will define the notion of *lax functor* in Definition 5.2.8 below. A lax embedding is a lax functor that is an embedding in the sense of Remark 2.4.5: that is, a lax functor that is faithful on hom categories.

Proposition 5.2.6. There is a 'discarding' functor $(-)^{\curlywedge} : \mathbf{Copara}_2(\mathcal{C}) \to \mathcal{C}$, which takes any coparameterized morphism and discards the coparameter.

Proposition 5.2.7. $(-)^{\curlyvee}$ is a section of $(-)^{\curlywedge}$. That is, $\mathrm{id}_{\mathcal{C}} = \mathcal{C} \xrightarrow{(-)^{\curlyvee}} \mathbf{Copara}_2(\mathcal{C}) \xrightarrow{(-)^{\curlywedge}} \mathcal{C}$; more suggestively, this can be written $(-) = (-)^{\curlyvee\curlywedge}$.

5.2.2. Lax functors, pseudofunctors, their transformations, and indexed bicategories

In order to define bicategories of statistical games, coherently with loss functions like the relative entropy that compose by copy-composition, we first need to define coparameterized (copy-composite) Bayesian lenses. Analogously to non-coparameterized Bayesian lenses, these will be obtained by applying a Grothendieck construction to an indexed bicategory [19, Def. 3.5] of state-dependent channels, Stat_2. Constructing copy-composite Bayesian lenses in this way is the subject of §5.2.3; in this section, we supply the necessary higher-categorical prerequisites.

An indexed category is a homomorphism of bicategories with codomain $\mathbf{Cat}$ and locally trivial domain, and analogously an indexed bicategory will be a homomorphism of *tricategories* with codomain $\mathbf{Bicat}$ (appropriately defined) and locally '2-trivial' domain. In order to stay as close as possible to the matter at hand, we do not give here an explicit definition of 'tricategory' or indeed of 'indexed bicategory', and instead refer the reader to [19, §3]. The definition of Stat_2 will of course provide an example of an indexed bicategory, but in order to state it we will nonetheless need to extend the notion of pseudofunctor from Definition 4.2.10 to the case where the domain is a true bicategory; and we will also need morphisms of pseudofunctors, called *pseudonatural* (or *strong*) *transformations*.

We will begin by defining the notion of *lax functor*, of which pseudofunctors constitute a special case. Just as a lax monoidal functor F is equipped with a natural family of morphisms $F(X) \otimes F(Y) \rightarrow F(X \otimes Y)$ (the laxator; *cf.* Definition 3.1.11), a lax functor is a weak functor equipped with a natural family of 2-cells $F(g) \diamond F(f) \Rightarrow F(g \circ f)$; this *lax functoriality* will be important when we come to study loss models in §5.3.2.

Definition 5.2.8 (Johnson and Yau [145, Def. 4.1.2])**.** Suppose $\mathcal{B}$ and $\mathcal{C}$ are both bicategories. A *lax functor $F : \mathcal{B} \rightarrow \mathcal{C}$* is constituted by

1. a function $F_0 : \mathcal{B}_0 \rightarrow \mathcal{C}_0$ on 0-cells;

2. for each pair of 0-cells $a, b : \mathcal{B}$, a functor $F_{a,b} : \mathcal{B}(a, b) \rightarrow \mathcal{C}(F_0 a, F_0 b)$;

3. for each 0-cell $b : \mathcal{B}$, a natural transformation

$$
\begin{array}{ccc}
\mathbf{1} & \xrightarrow{\ \mathsf{id}_b\ } & \mathcal{B}(b, b) \\
& {\scriptstyle F_1} \nearrow\!\!\!\Rightarrow & \downarrow {\scriptstyle F_{b,b}} \\
\mathsf{id}_{F_0 b} & \searrow & \\
& & \mathcal{C}(F_0 b, F_0 b)
\end{array}
$$

    witnessing *lax unity*, with component 2-cells $F_b : \mathsf{id}_{F_0 b} \Rightarrow F_{b,b}(\mathsf{id}_b)$;

4. for each triple of 0-cells $a, b, c : \mathcal{B}$, a natural transformation

$$
\begin{array}{ccc}
\mathcal{B}(b, c) \times \mathcal{B}(a, b) & \xrightarrow{\quad\quad\circ\quad\quad} & \mathcal{B}(a, c) \\
{\scriptstyle F_{b,c} \times F_{a,b}} \downarrow & {\scriptstyle F_2} \Rightarrow & \downarrow {\scriptstyle F_{a,c}} \\
\mathcal{C}(F_0 b, F_0 c) \times \mathcal{C}(F_0 a, F_0 b) & \xrightarrow{\quad\quad\diamond\quad\quad} & \mathcal{C}(F_0 a, F_0 c)
\end{array}
$$

    witnessing *lax functoriality* and called the *laxator*[4], with component 2-cells

    $$ F_{g,f} : F_{b,c}(g) \diamond F_{a,b}(f) \Rightarrow F_{a,c}(g \circ f) $$

    where $\circ$ and $\diamond$ denote horizontal composition in $\mathcal{B}$ and $\mathcal{C}$ respectively;

satisfying the following conditions:

---

[4]By analogy with the laxator of a lax monoidal functor (Definition 3.1.11). Since *monoidal category* is a special case of *bicategory*, the notion of lax functor (between bicategories) generalizes the notion of lax monoidal functor (between monoidal categories).

(a) coherence with the left and right unitality of horizontal composition, so that for all 1-cells $f : a \to b$ the following diagrams commute:

$$
\begin{array}{ccc}
\mathrm{id}_{F_0 b} \diamond F_{a,b}(f) \xrightarrow{\lambda^{\mathcal{C}}_{F_{a,b}(f)}} F_{a,b}(f) & \qquad & F_{a,b}(f) \diamond \mathrm{id}_{F_0 a} \xrightarrow{\rho^{\mathcal{C}}_{F_{a,b}(f)}} F_{a,b}(f) \\
F_b \diamond F_{a,b}(f) \big\| \qquad \big\uparrow F_{a,b}(\lambda^{\mathcal{B}}_f) & & F_{a,b}(f) \diamond F_a \big\| \qquad \big\uparrow F_{a,b}(\rho^{\mathcal{B}}_f) \\
F_{b,b}(\mathrm{id}_b) \diamond F_{a,b}(f) \xrightarrow{F_{\mathrm{id}_b,f}} F_{a,b}(\mathrm{id}_b \circ f) & & F_{a,b}(f) \diamond F_{a,a}(\mathrm{id}_a) \xrightarrow{F_{f,\mathrm{id}_a}} F_{a,b}(f \circ \mathrm{id}_a)
\end{array}
$$

where $\lambda^{\mathcal{B}}, \lambda^{\mathcal{C}}$ and $\rho^{\mathcal{B}}, \rho^{\mathcal{C}}$ are the left and right unitors for the horizontal composition in $\mathcal{B}$ and $\mathcal{C}$ respectively;

(b) coherence with the associativity of horizontal composition, so that for all 1-cells $f : a \to b$, $g : b \to c$, and $h : c \to d$, the following diagram commutes:

$$
\begin{array}{ccc}
(F_{c,d}(h) \diamond F_{b,c}(g)) \diamond F_{a,b}(f) \xrightarrow{\alpha^{\mathcal{C}}_{F_{c,d}(h),F_{b,c}(g),F_{a,b}(f)}} F_{c,d}(h) \diamond (F_{b,c}(g) \diamond F_{a,b}(f)) \\
F_{h,g} \diamond F_{a,b}(f) \big\downarrow \qquad\qquad\qquad\qquad\qquad\qquad \big\downarrow F_{c,d}(h) \diamond F_{g,f} \\
F_{b,d}(h \circ g) \diamond F_{a,b}(f) \qquad\qquad\qquad\qquad F_{c,d}(h) \diamond F_{a,c}(g \circ f) \\
F_{h\circ g,f} \big\downarrow \qquad\qquad\qquad\qquad\qquad\qquad\qquad \big\downarrow F_{h,g\circ f} \\
F_{a,d}((h \circ g) \circ f) \xrightarrow{F_{a,d}(\alpha^{\mathcal{B}}_{h,g,f})} F_{a,d}(h \circ (g \circ f))
\end{array}
$$

where $\alpha^{\mathcal{B}}$ and $\alpha^{\mathcal{C}}$ are the associators for the horizontal composition in $\mathcal{B}$ and $\mathcal{C}$ respectively.

A *pseudofunctor* is a lax functor F for which F_1 and F_2 are natural isomorphisms[5].

The variable laxness of lax functors is recapitulated in the laxness of their morphisms; again, we begin with the weakest case.

Definition 5.2.9. Suppose F and G are lax functors $\mathcal{B} \to \mathcal{C}$. A *lax transformation* $\alpha : F \to G$ consists of

1. a 1-cell $\alpha_b : Fb \to Gb$ in $\mathcal{C}$ for each 0-cell $b : \mathcal{B}$;

2. a natural transformation $\alpha_{b,c} : \alpha_b{}^* G \Rightarrow \alpha_{c*} F$ (where $\alpha_b{}^*$ denotes pre-composition by α_b, and α_{c*} denotes post-composition by α_c) for each pair b, c of objects in $\mathcal{B}$, with component

---

[5] Compare Definition 4.2.10, treating $\mathcal{C}$ there as a bicategory with discrete hom-categories.

2-cells

$$
\begin{array}{ccc}
Fb & \xrightarrow{\ Ff\ } & Fc \\
{\scriptstyle\alpha_b}\downarrow & \overset{\alpha_f}{\nearrow\!\!\nearrow} & \downarrow{\scriptstyle\alpha_c} \\
Gb & \xrightarrow[\ Gf\ ]{} & Gc
\end{array}
$$

for each 1-cell $f : b \to c$ in $\mathcal{B}$;

satisfying conditions of lax unity and lax naturality whose precise general form the reader may find in Johnson and Yau [145, Def. 4.2.1].

A *strong transformation* (or *pseudonatural transformation*) is a lax transformation for which the component 2-cells constitute natural isomorphisms.

It is notable that, unlike natural transformations, lax transformations do not compose, not even laxly; see Johnson and Yau [145, Motivation 4.6.1]. This means that there is no bicategory of bicategories, lax functors, and lax transformations, analogous to **Cat**. However, *strong* transformations between pseudofunctors *do* compose, weakly, up to isomorphism. These isomorphisms collect into 3-cells known as *modifications*, producing a *tricategory* **Bicat** whose 0-cells are bicategories, 1-cells are pseudofunctors, 2-cells strong transformations, and 3-cells modifications. This tricategory constitutes the codomain of an indexed bicategory.

Remark 5.2.10. There is another notion of composable morphism between *lax* functors, called *icon*, which yields a bicategory **Bicat**$_{\mathrm{ic}}$ of bicategories, lax functors, and icons. Icons are, however, more restrictive than lax transformations, as their 1-cell components must be identities. Nonetheless, this restriction is satisfied by loss models as defined in §5.3.2, and so morphisms of loss models will be icons.

5.2.3. Coparameterized Bayesian lenses

With that categorical background out of the way, we are able to define copy-composite Bayesian lenses, starting with the corresponding indexed bicategory. Let disc denote the functor **Set** $\to$ **Cat** taking sets to discrete categories (*cf.* Definition 4.2.2).

Definition 5.2.11. We define the indexed bicategory $\mathsf{Stat}_2 : \mathbf{Copara}_2^l(\mathcal{C})^{\mathrm{co\,op}} \to \mathbf{Bicat}$ fibrewise as follows.

(i) The 0-cells $\mathsf{Stat}_2(X)_0$ of each fibre $\mathsf{Stat}_2(X)$ are the objects $\mathcal{C}_0$ of $\mathcal{C}$.

(ii) For each pair of 0-cells A, B, the hom-category $\mathsf{Stat}_2(X)(A, B)$ is defined to be the functor category $\mathbf{Cat}\big(\mathsf{disc}\,\mathcal{C}(I, X), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big)$.

(iii) For each 0-cell A, the identity functor $\mathrm{id}_A : \mathbf{1} \to \mathsf{Stat}_2(X)(A, A)$ is the constant functor on the identity on A in $\mathbf{Copara}_2^r(\mathcal{C})$; *i.e.* $\mathrm{disc}\,\mathcal{C}(I, X) \xrightarrow{!} \mathbf{1} \xrightarrow{\mathrm{id}_A} \mathbf{Copara}_2^r(\mathcal{C})(A, A)$.

(iv) For each triple A, B, C of 0-cells, the horizontal composition functor $\circ_{A,B,C}$ is defined by

$$\circ_{A,B,C} : \mathsf{Stat}_2(X)(B, C) \times \mathsf{Stat}_2(X)(A, B) \cdots$$

$$\cdots \xrightarrow{=} \mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, X), \mathbf{Copara}_2^r(\mathcal{C})(B, C)\big) \times \mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, X), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big) \cdots$$

$$\cdots \xrightarrow{\times} \mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, X)^2, \mathbf{Copara}_2^r(\mathcal{C})(B, C) \times \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big) \cdots$$

$$\cdots \xrightarrow{\mathbf{Cat}\left(\mathbf{Y}, \circ\right)} \mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, X), \mathbf{Copara}_2^r(\mathcal{C})(A, C)\big) \cdots$$

$$\cdots \xrightarrow{=} \mathsf{Stat}_2(X)(A, C)$$

where $\mathbf{Cat}\,(\mathbf{Y}, \circ)$ indicates pre-composition with the universal (Cartesian) copying functor in $(\mathbf{Cat}, \times, \mathbf{1})$ and post-composition with the horizontal composition functor of $\mathbf{Copara}_2^r(\mathcal{C})$.

For each pair of 0-cells X, Y in $\mathbf{Copara}^l(\mathcal{C})$, we define the reindexing pseudofunctor $\mathsf{Stat}_{2,X,Y} : \mathbf{Copara}^l(\mathcal{C})(X, Y)^{\mathrm{op}} \to \mathbf{Bicat}\big(\mathsf{Stat}_2(Y), \mathsf{Stat}_2(X)\big)$ as follows.

(a) For each 1-cell f in $\mathbf{Copara}^l(\mathcal{C})(X, Y)$, we obtain a pseudofunctor $\mathsf{Stat}_2(f) : \mathsf{Stat}_2(Y) \to \mathsf{Stat}_2(X)$ which acts as the identity on 0-cells.

(b) For each pair of 0-cells A, B in $\mathsf{Stat}_2(Y)$, the functor $\mathsf{Stat}_2(f)_{A,B}$ is defined as the precomposition functor $\mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, f^{\bar{\dagger}}), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big)$, where $(-)^{\bar{\dagger}}$ is the discarding functor $\mathbf{Copara}_2^l(\mathcal{C}) \to \mathcal{C}$ of Proposition 5.2.6.

(c) For each 2-cell $\varphi : f \Rightarrow f'$ in $\mathbf{Copara}_2^l(\mathcal{C})(X, Y)$, the pseudonatural transformation $\mathsf{Stat}_2(\varphi) : \mathsf{Stat}_2(f') \Rightarrow \mathsf{Stat}_2(f)$ is defined on 0-cells $A : \mathsf{Stat}_2(Y)$ by the discrete natural transformation with components $\mathsf{Stat}_2(\varphi)_A := \mathrm{id}_A$, and on 1-cells $c : \mathsf{Stat}_2(Y)(A, B)$ by the substitution natural transformation with constituent 2-cells $\mathsf{Stat}_2(\varphi)_c : \mathsf{Stat}_2(f)(c) \Rightarrow \mathsf{Stat}_2(f')(c)$ in $\mathsf{Stat}_2(X)$ which acts by replacing $\mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, f^{\bar{\dagger}}), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big)$ by $\mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, f'^{\bar{\dagger}}), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big)$; and which we might alternatively denote by $\mathbf{Cat}\big(\mathrm{disc}\,\mathcal{C}(I, \varphi^{\bar{\dagger}}), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big)$.

Notation 5.2.12. We will write $f : A \xrightarrow[M]{X} B$ to denote a state-dependent coparameterized channel f with coparameter M and state-dependence on X.

Remark 5.2.13. We could give an alternative definition of Stat_2, for which the definition above would give a sub-indexed bicategory, by defining the state-dependence on the whole hom-category

$\mathbf{Copara}_2(\mathcal{C})(I, X)$ rather than just $\mathcal{C}(I, X)$. However, doing this would cause the reindexing pseudofunctors to introduce coparameters (from the now-coparameterized priors), which would contradict the type signature of coparameterized Bayesian inversion: imagine the equation of Definition 5.2.18 below but without the discarding on the right-hand side.

Remark 5.2.14. Similarly, the same definitions would go through upon substituting $\mathbf{Copara}(\mathcal{C})$ for $\mathbf{Copara}_2(\mathcal{C})$; but, as noted above, we need copy-composition for the relative entropy.

As we saw in §4.2.2, lenses in 1-category theory are morphisms in the fibrewise opposite of a fibration. Analogously, our bicategorical Bayesian lenses are obtained as 1-cells in the bicategorical Grothendieck construction of (the pointwise opposite of) the indexed bicategory Stat_2; this yields a 2-fibration. So as not to over-complicate the presentation, we do not present all the details of this construction, and refer the reader instead to Baković [19, §6].

Definition 5.2.15. Fix a copy-discard category $(\mathcal{C}, \otimes, I)$. We define the bicategory of coparameterized Bayesian lenses in $\mathcal{C}$, denoted $\mathbf{BayesLens}_2(\mathcal{C})$ or simply $\mathbf{BayesLens}_2$, to be the bicategorical Grothendieck construction of the pointwise opposite of the corresponding indexed bicategory Stat_2, with the following data:

(i) A 0-cell in $\mathbf{BayesLens}_2$ is a pair (X, A) of an object X in $\mathbf{Copara}_2^l(\mathcal{C})$ and an object A in $\mathsf{Stat}_2(X)$; equivalently, a 0-cell in $\mathbf{BayesLens}_2$ is a pair of objects in $\mathcal{C}$.

(ii) The hom-category $\mathbf{BayesLens}_2\big((X, A), (Y, B)\big)$ is the product category $\mathbf{Copara}_2^l(\mathcal{C})(X, Y) \times \mathsf{Stat}_2(X)(B, A)$.

(iii) The identity on (X, A) is given by the pair $(\mathsf{id}_X, \mathsf{id}_A)$.

(iv) For each triple of 0-cells $(X, A), (Y, B), (Z, C)$, the horizontal composition functor is given

by

$$\mathbf{BayesLens}_2\big((Y,B),(Z,C)\big) \times \mathbf{BayesLens}_2\big((X,A),(Y,B)\big)$$

$$= \quad \mathbf{Copara}_2^l(\mathcal{C})(Y,Z) \times \mathsf{Stat}_2(Y)(C,B) \times \mathbf{Copara}_2^l(\mathcal{C})(X,Y) \times \mathsf{Stat}_2(X)(B,A)$$

$$\xrightarrow{\sim} \sum_{g:\mathbf{Copara}_2^l(\mathcal{C})(Y,Z)} \sum_{f:\mathbf{Copara}_2^l(\mathcal{C})(X,Y)} \mathsf{Stat}_2(Y)(C,B) \times \mathsf{Stat}_2(X)(B,A)$$

$$\xrightarrow{\sum_g \sum_f \mathsf{Stat}_2(f)_{C,B} \times \mathsf{id}} \sum_{g:\mathbf{Copara}_2^l(\mathcal{C})(Y,Z)} \sum_{f:\mathbf{Copara}_2^l(\mathcal{C})(X,Y)} \mathsf{Stat}_2(X)(C,B) \times \mathsf{Stat}_2(X)(B,A)$$

$$\xrightarrow{\sum_{\circ \mathbf{Copara}_2^l(\mathcal{C})} \circ^{\mathsf{Stat}_2(X)}} \sum_{g \circ f:\mathbf{Copara}_2^l(\mathcal{C})(X,Z)} \mathsf{Stat}_2(X)(C,A)$$

$$\xrightarrow{\sim} \quad \mathbf{BayesLens}_2\big((X,A),(Z,C)\big)$$

where the functor in the penultimate line amounts to the product of the horizontal composition functors on $\mathbf{Copara}_2^l(\mathcal{C})$ and $\mathsf{Stat}_2(X)$.

Proposition 5.2.16. There is a projection pseudofunctor $\pi_{\mathsf{Lens}} : \mathbf{BayesLens}_2(\mathcal{C}) \to \mathbf{Copara}_2^l(\mathcal{C})$ mapping each 0-cell (X, A) to X, each 1-cell (f, f') to f, and each 2-cell (φ, φ') to φ. This pseudofunctor is a 2-fibration in the sense of Baković [19, Def. 4.7].

Proof. The claim follows as a consequence of Baković [19, Theorem 6.2]. $\square$

Remark 5.2.17. When $\mathcal{C}$ is symmetric monoidal, Stat_2 acquires the structure of a monoidal indexed bicategory (Definition 5.4.2 and Theorem 5.4.4), and hence $\mathbf{BayesLens}_2$ becomes a monoidal bicategory (Corollary 5.4.5).

5.2.4. Coparameterized Bayesian updates compose optically

So that our generalized Bayesian lenses are worthy of the name, we should also confirm that Bayesian inversions compose according to the lens pattern ('optically') in the coparameterized setting. Such confirmation is the subject of the present section.

Definition 5.2.18. We say that a coparameterized channel $\gamma : A \rightarrowtail M \otimes B$ *admits Bayesian inversion*

if there exists a dually coparameterized channel $\rho_\pi : B \to A \otimes M$ satisfying the graphical equation
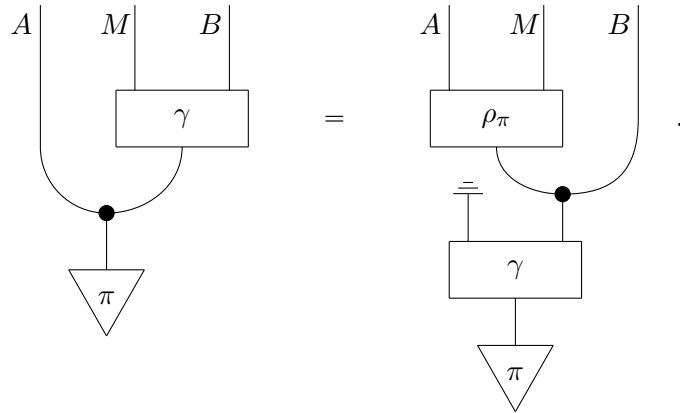
In this case, we say that ρ_π is the *Bayesian inversion of γ with respect to π*.

With this definition, we can supply the desired result that "coparameterized Bayesian updates compose optically".

Theorem 5.2.19. Suppose $(\gamma, \gamma^\dagger) : (A, A) \xrightarrow[M]{} (B, B)$ and $(\delta, \delta^\dagger) : (B, B) \xrightarrow[N]{} (C, C)$ are coparameterized Bayesian lenses in $\mathbf{BayesLens}_2$. Suppose also that $\pi : I \to A$ is a state on A in the underlying category of channels $\mathcal{C}$, such that $\gamma_\pi^\dagger$ is a Bayesian inversion of γ with respect to π, and such that $\delta_{\gamma\pi}^\dagger$ is a Bayesian inversion of δ with respect to $(\gamma\pi)^{\bar{\dagger}}$; where the notation $(-)^{\bar{\dagger}}$ represents discarding coparameters. Then $\gamma_\pi^\dagger \bullet \delta_{\gamma\pi}^\dagger$ is a Bayesian inversion of $\delta \bullet \gamma$ with respect to π. (Here $\bullet$ denotes copy-composition.) Moreover, if $(\delta \bullet \gamma)_\pi^\dagger$ is any Bayesian inversion of $\delta \bullet \gamma$ with respect to π, then $\gamma_\pi^\dagger \bullet \delta_{\gamma\pi}^\dagger$ is $(\delta\gamma\pi)^{\bar{\dagger}}$-almost-surely equal to $(\delta \bullet \gamma)_\pi^\dagger$: that is, $(\delta \bullet \gamma)_\pi^\dagger \overset{(\delta\gamma\pi)^{\bar{\dagger}}}{\sim} \gamma_\pi^\dagger \bullet \delta_{\gamma\pi}^\dagger$.

Proof. We only need to show that $\gamma_\pi^\dagger \bullet \delta_{\gamma\pi}^\dagger$ is a Bayesian inversion of $\delta \bullet \gamma$ with respect to π; the 'moreover' claim follows immediately because Bayesian inversions are almost surely unique (by Proposition 4.1.29). Thus, $\delta \bullet \gamma \bullet \pi$ has the following depiction;

Since $\gamma_\pi^\dagger$ is a Bayesian inversion of γ with respect to π, this is equal to

.

By the coassociativity of copying, this in turn is equal to

.
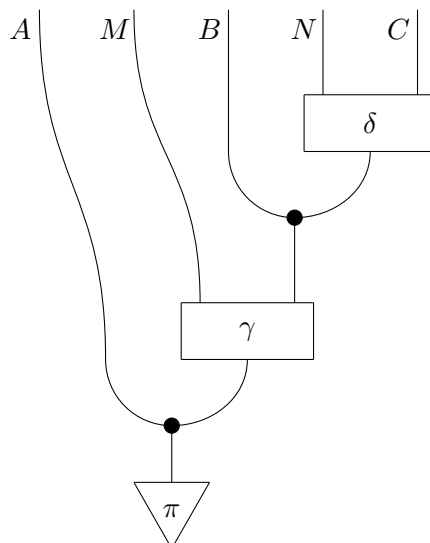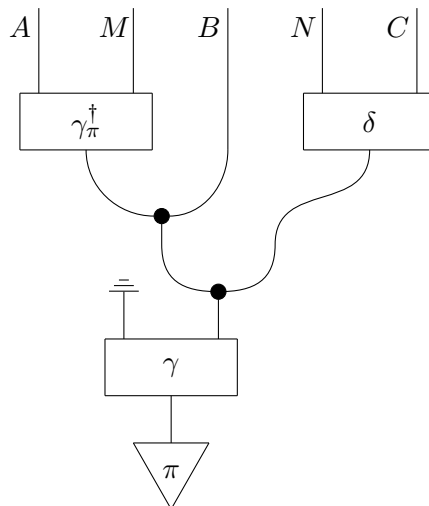
And since $\delta^{\dagger}_{\gamma\pi}$ is a Bayesian inversion of δ with respect to $(\gamma\pi)^{\ddagger}$, this is equal to
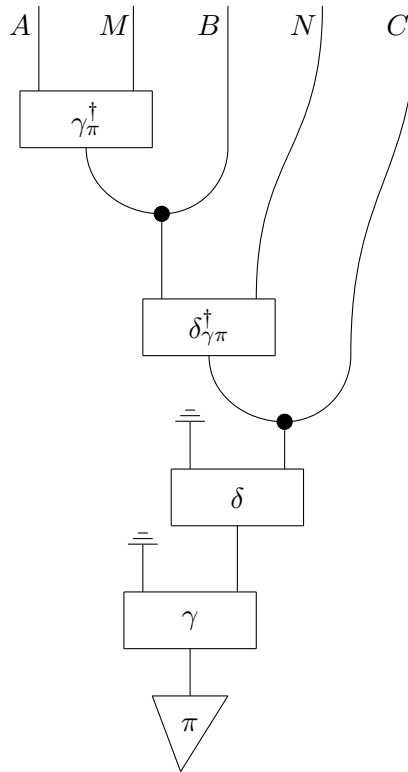
which establishes the result. $\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\square$

In order to satisfy this coparameterized Bayes' rule, a Bayesian lens must be of 'simple' type.

Definition 5.2.20. We say that a coparameterized Bayesian lens (c, c') is *simple* if its domain and codomain are 'diagonal' (duplicate pairs of objects) and if the coparameter of c is equal to the coparameter of c'. In this case, we can write the type of (c, c') as $(X, X)\xrightarrow[M]{\;+\!\!\!\rightarrow\;}(Y, Y)$ or simply $X\xrightarrow[M]{\;+\!\!\!\rightarrow\;}Y$.

Remark 5.2.21. In Remark 5.2.13, we explained that we should restrict the type of state-dependent coparameterized morphisms so that they cohere with the coparameterized Bayes' rule of Definition 5.2.18. The restriction here to simple lenses is by contrast not enforced by the type system, an oversight which (like the failure to restrict to supports noted in Remark 4.1.20) is comparatively inelegant, but which is forced upon us by the Grothendieck construction, which does not have a place for such constraints. We expect that the use of (a bicategorical instance of) *dependent optics* [43, 50, 276] would allow such a constraint to be enforced (alongside support objects), at the cost of requiring yet more high-powered categorical machinery, of which there is probably enough in this thesis. We therefore leave this avenue unexplored for now.

By analogy with Corollary 4.3.15, we have the following important consequence of Theorem 5.2.19.

Corollary 5.2.22. Suppose $\mathbf{Copara}_2^l(\mathcal{C})^\dagger$ is a subbicategory of $\mathbf{Copara}_2^l(\mathcal{C})$ all of whose channels admit Bayesian inversion. Then there is almost surely a pseudofunctor $\dagger : \mathbf{Copara}_2^l(\mathcal{C})^\dagger \to \mathbf{BayesLens}_2$ mapping each 1-cell to its almost-surely unique corresponding exact Bayesian lens. Moreover, $\dagger$ is a section of the 2-fibration $\pi_{\mathsf{Lens}} : \mathbf{BayesLens}_2 \to \mathbf{Copara}_2^l(\mathcal{C})$ induced by the bicategorical Grothendieck construction.

5.3. Statistical games for local approximate inference

5.3.1. Attaching losses to lenses

Statistical games are obtained by attaching to Bayesian lenses *loss functions*, representing 'local' quantifications of the performance of approximate inference systems. Because this performance depends on the system's context (*i.e.*, the prior $\pi : I \nrightarrow X$ and the observed data $b : B$), a loss function at its most concrete will be a function $\mathcal{C}(I, X) \times B \to \mathbb{R}$. To internalize this type in $\mathcal{C}$, we may recall that, when $\mathcal{C}$ is the category $\mathbf{sfKrn}$ of s-finite kernels or the Kleisli category $\mathcal{K}\ell(\mathcal{D}_{\leq 1})$ of the subdistribution monad[6], a density function $p_c : X \times Y \to [0, 1]$ for a channel $c : X \nrightarrow Y$ corresponds to an *effect* (or *costate*) $X \otimes Y \nrightarrow I$. In this way, we can see a loss function as a kind of *state-dependent effect* $B \overset{X}{\nrightarrow} I$ (and not a coparameterized one).

Loss functions will compose by sum, and so we need to ask for the effects in $\mathcal{C}$ to form a monoid. Moreover, we need this monoid to be 'bilinear' with respect to channels, so that Stat-reindexing (*cf.* Definition 4.3.2) preserves sums. These conditions are formalized in the following definition.

Definition 5.3.1. Suppose $(\mathcal{C}, \otimes, I)$ is a copy-discard category. We say that $\mathcal{C}$ *has bilinear effects* if the following conditions are satisfied:

(i) *effect monoid*: there is a natural transformation $+ : \mathcal{C}(-, I) \times \mathcal{C}(=, I) \Rightarrow \mathcal{C}(- \otimes =, I)$ making $\sum_{A:\mathcal{C}} \mathcal{C}(A, I)$ into a commutative monoid with unit $0 : I \nrightarrow I$;

(ii) *bilinearity*: $(g + g') \bullet \curlyvee \bullet f = g \bullet f + g' \bullet f$ for all effects g, g' and morphisms f such that $(g + g') \bullet \curlyvee \bullet f$ exists.

Example 5.3.2. A trivial example of a category with bilinear effects is supplied by any Cartesian category, such as $\mathbf{Set}$, in which there is a unique effect for each object, so the effect monoid

---

[6]Weaken the definition of the distribution monad $\mathcal{D} : \mathbf{Set} \to \mathbf{Set}$ so that distributions may sum to any number in the unit interval.

structure is given only by the product of objects, and bilinearity follows from the terminality of the unit object 1.

Example 5.3.3. We might hope that $\mathcal{K}\ell(\mathcal{D}_{\leqslant 1})$ has bilinear effects, but this is not the case, because the sum of two effects may exceed 1: the effects only form a *partial* monoid[7]. But if M is any monoid in **Set**, then there is a monad $\mathcal{D}_M$ taking each set X to the set $\mathcal{D}_M(X)$ of formal M-linear combinations of elements of X. This is the free M-module on X, just as traditionally $\mathcal{D}X$ is the free convex space on X, and the monad structure is obtained from the adjunction in the same way [137, §2]. An effect $Y \nrightarrow I$ then corresponds to a function $Y \rightarrow M$, and the monoid structure follows from the monoid structure on M; bilinearity follows from the linearity of the (free) module structure:

$$
\begin{aligned}
(g + g') \bullet \, \forall \, \bullet f(x) &= \sum_y \big(g(y) + g'(y)\big) \cdot f(y|x) \\
&= \sum_y g(y) \cdot f(y|x) + g'(y) \cdot f(y|x) \\
&= \sum_y g(y) \cdot f(y|x) + \sum_y g'(y) \cdot f(y|x) \\
&= g \bullet f(x) + g' \bullet f(x)
\end{aligned}
$$

Example 5.3.4. The category **sfKrn** of s-finite kernels [270] has bilinear effects. An effect $Y \nrightarrow I$ is a measurable function $Y \rightarrow [0, \infty]$, and bilinearity follows from the linearity of integration:

$$
\begin{aligned}
(g + g') \bullet \, \forall \, \bullet f(x) &= \int_y \big(g(y) + g'(y)\big) \, f(\mathrm{d}y|x) \\
&= \int_y g(y) \, f(\mathrm{d}y|x) + g'(y) \, f(\mathrm{d}y|x) \\
&= \int_y g(y) \, f(\mathrm{d}y|x) + \int_y g'(y) \, f(\mathrm{d}y|x) \\
&= g \bullet f(x) + g' \bullet f(x)
\end{aligned}
$$

We will typically assume **sfKrn** as our ambient $\mathcal{C}$ for the examples below.

Example 5.3.5. Given a category $\mathcal{C}$ with bilinear effects, we can lift the natural transformation $+$, and hence the bilinear effect structure, to the fibres of $\mathsf{Stat}_\mathcal{C}$, using the universal property of the

---

[7]Indeed, an *effect algebra* is a kind of partial commutative monoid [141, §2], but we do not need the extra complication here.

product of categories:

$$
\begin{aligned}
+_X : \mathsf{Stat}(X)(-,I) &\times \mathsf{Stat}(X)(=,I) = \mathbf{Set}\big(\mathcal{C}(I,X),\mathcal{C}(-,I)\big) \times \mathbf{Set}\big(\mathcal{C}(I,X),\mathcal{C}(=,I)\big) \\
&\xRightarrow{(\cdot,\cdot)} \mathbf{Set}\big(\mathcal{C}(I,X),\mathcal{C}(-,I) \times \mathcal{C}(=,I)\big) \\
&\xRightarrow{\mathbf{Set}\big(\mathcal{C}(I,X),+\big)} \mathbf{Set}\big(\mathcal{C}(I,X),\mathcal{C}(-\otimes=,I)\big) \\
&\xRightarrow{=} \mathsf{Stat}(X)(-\otimes=,I)
\end{aligned}
$$

Here, $(\cdot,\cdot)$ denotes the pairing operation obtained from the universal property. In this way, each $\mathsf{Stat}(X)$ has bilinear effects. Note that this lifting is (strictly) compatible with the reindexing of Stat, so that $+_{(-)}$ defines an indexed natural transformation. This means in particular that *reindexing distributes over sums*: given state-dependent effects $g, g' : B\xrightarrow{Y} I$ and a channel $c : X\rightarrow Y$, we have $(g +_Y g')_c = g_c +_X g'_c$. We will thus generally omit the subscript from the lifted sum operation, and just write $+$.

We are now ready to construct the bicategory of statistical games.

Definition 5.3.6. Suppose $(\mathcal{C},\otimes,I)$ has bilinear effects, and let $\mathbf{BayesLens}_2$ denote the corresponding bicategory of (copy-composite) Bayesian lenses. We will write $\mathbf{SGame}_\mathcal{C}$ to denote the following *bicategory of (copy-composite) statistical games* in $\mathcal{C}$:

- The 0-cells are the 0-cells (X, A) of $\mathbf{BayesLens}_2$;

- the 1-cells, called *statistical games*, $(X, A) \to (Y, B)$ are triples (c, c', L^c) consisting of a 1-cell $(c, c') : (X, A) \rightarrow (Y, B)$ in $\mathbf{BayesLens}_2$ and a *loss* $L^c : B\xrightarrow{X} I$ in $\mathsf{Stat}(X)(B, I)$;

- given 1-cells $(c, c', L^c), (e, e', L^e) : (X, A) \to (Y, B)$, the 2-cells $(c, L^c) \Rightarrow (e, L^e)$ are pairs (α, K^α) of a 2-cell $\alpha : (c, c') \Rightarrow (e, e')$ in $\mathbf{BayesLens}_2$ and a loss $K^\alpha : B\xrightarrow{X} I$ such that $L^c = L^e + K^\alpha$;

- the identity 2-cell on (c, c', L^c) is $(\mathsf{id}_{(c,c')}, 0)$;

- given 2-cells $(\alpha, K^\alpha) : (c, c', L^c) \Rightarrow (d, d', L^d)$ and $(\beta, K^\beta) : (d, d', L^d) \Rightarrow (e, e', L^e)$, their vertical composite is $(\beta \circ \alpha, K^\beta + K^\alpha)$, where $\circ$ here denotes vertical composition in $\mathbf{BayesLens}_2$;

- given 1-cells $(c, c', L^c) : (X, A) \to (Y, B)$ and $(d, d', L^d) : (Y, B) \to (Z, C)$, their horizontal composite is $\big((d, d') \diamond (c, c'), L^d_c + L^c \circ d'_c\big)$; and

– given 2-cells $(\alpha, K^\alpha) : (c, c', L^c) \Rightarrow (e, e', L^e)$ and $(\beta, K^\beta) : (d, d', L^d) \Rightarrow (f, f', L^f)$, their horizontal composite is $(\beta \phi \alpha, K_c^\beta + K^\alpha \circ d_c')$, where ϕ here denotes horizontal composition in $\mathbf{BayesLens}_2$.

Remark 5.3.7. In earlier work (such as versions 1 and 2 of our preprint [250]), we gave a more elaborate but less satisfying definition of "statistical game", as a Bayesian lens equipped with a function from its 'context' to $\mathbb{R}$ (which we also called a loss function). The construction given here shows that the complicated earlier notion of context, which follows the ideas of 'Bayesian open games' [36], is actually unnecessary for the purposes of statistical games. Considering a Bayesian lens in $\mathcal{Kl}(\mathcal{D})$ of type $(X, A) \to (Y, B)$, its 'context' is an element of $\mathcal{D}X \times \mathbf{Set}(\mathcal{D}Y, \mathcal{D}B)$. By comparison, a corresponding loss function of the type given above is equivalently a function with domain $\mathcal{D}X \times B$, and so we have replaced the dependence on 'continuations' in $\mathbf{Set}(\mathcal{D}Y, \mathcal{D}B)$ with a simple dependence on B.

Theorem 5.3.8. Definition 5.3.6 generates a well-defined bicategory.

The proof of this result is that $\mathbf{SGame}_\mathcal{C}$ is obtained via a pair of bicategorical Grothendieck constructions: first to obtain Bayesian lenses; and then to attach the loss functions. The proof depends on the following intermediate result that our effect monoids can be 'upgraded' to monoidal categories; we then use the delooping of this structure to associate (state-dependent) losses to (state-dependent) channels, after discarding the coparameters of the latter.

Lemma 5.3.9. Suppose $(\mathcal{C}, \otimes, I)$ has bilinear effects. Then, for each object B, $\mathcal{C}(B, I)$ has the structure of a symmetric monoidal category. The objects of $\mathcal{C}(B, I)$ are its elements, the effects. If g, g' are two effects, then a morphism $\kappa : g \to g'$ is an effect such that $g = g' + \kappa$; the identity morphism for each effect id_g is then the constant 0 effect. Likewise, the tensor of two effects is their sum, and the corresponding unit is the constant 0. Precomposition by any morphism $c : A \nrightarrow B$ preserves the monoidal category structure, making the presheaf $\mathcal{C}(-, I)$ into a fibrewise-monoidal indexed category $\mathcal{C}^{\mathrm{op}} \to \mathbf{MonCat}$ (*cf.* Remark 4.2.21).

As already indicated, this structure lifts to the fibres of Stat.

Corollary 5.3.10. For each object X in a category with bilinear effects, and for each object B, $\mathsf{Stat}(X)(B, I)$ inherits the symmetric monoidal structure of $\mathcal{C}(B, I)$; note that morphisms of state-dependent effects are likewise state-dependent, and that tensoring (summing) state-dependent effects involves copying the parameterizing state. Moreover, $\mathsf{Stat}(-)(=, I)$ is a fibrewise-monoidal indexed category $\sum_{X:\mathcal{C}^{\mathrm{op}}} \mathsf{Stat}(X)^{\mathrm{op}} \to \mathbf{MonCat}$.

Using this corollary, we can give the abstract proof of Theorem 5.3.8. There are two further observations of note: first, that we can deloop a monoidal category into a bicategory with one object; second, that we can extend $\mathsf{Stat}(-)(=, I)$ to Stat_2 via discarding.

Proof of 5.3.8. Recall from Proposition 3.1.24 that every monoidal category $\mathcal{M}$ can be transformed into a one-object bicategory, its delooping $\mathbf{B}\mathcal{M}$, with the 1-cells and 2-cells being the objects and morphisms of $\mathcal{M}$, vertical composition being composition in $\mathcal{M}$, and horizontal composition being the tensor. This delooping is functorial, giving a 2-functor $\mathbf{B} : \mathbf{MonCat} \to \mathbf{Bicat}$ which, following Corollary 5.3.10, we can compose after $\mathsf{Stat}(-)(=, I)$ (taking its domain as a locally discrete 2-category) to obtain indexed bicategories; we will assume this transformation henceforth.

Next, observe that we can extend the domain of $\mathsf{Stat}(-)(=, I)$ to $\sum_{X:\mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}}} \mathsf{Stat}_2(X)^{\,\mathrm{co\,op}}$ by discarding the coparameters of the (coparameterized) state-dependent channels as well as the coparameter on any reindexing, as in the following diagram of indexed bicategories:

$$\sum_{X:\mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}}} \mathsf{Stat}_2(X)^{\,\mathrm{co\,op}}$$

$$\sum_{\maltese} \dot{=} \qquad \qquad \mathsf{Stat}_2(-)(=,I)$$

$$\sum_{X:\mathcal{C}^{\,\mathrm{op}}} \mathsf{Stat}(X)^{\,\mathrm{op}} \xrightarrow[\mathsf{Stat}(-)(=,I)]{} \mathbf{Bicat}$$

Here, the 2-cell indicates also discarding the coparameters of the 'effects' in $\mathsf{Stat}_2(-)(=, I)$.

If we let L denote the composite functor in the diagram above, we can reason as follows:

$$\cfrac{\cfrac{\cfrac{\cfrac{L : \sum_{X:\mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}}} \mathsf{Stat}_2(X)^{\,\mathrm{co\,op}} \to \mathbf{Bicat}}{\prod_{X:\mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}}} \mathbf{Bicat}^{\mathsf{Stat}_2(X)^{\,\mathrm{co\,op}}}}\ \text{sum/product}}{\prod_{X:\mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}}} \mathbf{2Fib}\,(\mathsf{Stat}_2(X))}\ \textstyle\prod\int}{\mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}} \to \mathbf{Bicat}}\ \text{forget}}{G : \mathbf{Copara}_2^l(\mathcal{C})^{\,\mathrm{co\,op}} \to \mathbf{Bicat}}\ \text{op}$$

where the first step uses the adjointness of (dependent) sums and products; the second applies the bicategorical Grothendieck construction in the codomain; the third forgets the 2-fibrations, to leave only the total bicategory; and the fourth step takes the pointwise opposite. We can thus write the action of G as $G(X) = \left(\int L(X, -)\right)^{\mathrm{op}}$.

Since each bicategory $L(X, B)$ has only a single 0-cell, the 0-cells of each $G(X)$ are equivalently just the objects of $\mathcal{C}$, and the hom-categories $G(X)(A, B)$ are equivalent to the product categories $\mathsf{Stat}_2(X)(B, A) \times \mathsf{Stat}(X)(B, I)$. That is to say, a 1-cell $A \to B$ in $G(X)$ is a pair of a state-dependent channel $B \xrightarrow{X} A$ along with a correspondingly state-dependent effect on its domain B. We therefore seem to approach the notion of statistical game, but in fact we are already there:

$\mathbf{SGame}_{\mathcal{C}}$ is simply $\int G$, by the bicategorical Grothendieck construction. To see this is only a matter of further unfolding the definition. $\qquad\square$

Remark 5.3.11. There are two notable details that the abstractness of the preceding proof obscures. Firstly, the horizontal composition of effects in $\mathbf{SGame}_{\mathcal{C}}$ is strict. To see this, let $(c, \iota) : A \to B$ and $(d, \kappa) : B \to C$ and $(e, \lambda) : C \to D$ be 1-cells in $G(X)$, and for concision write the horizontal composite of effects by concatenation, so that $\kappa\iota = \kappa + \iota \circ d^{\dagger}$ (by the Grothendieck construction). Then strict associativity demands that $\lambda(\kappa\iota) = (\lambda\kappa)\iota$. This obtains as follows:

$$\begin{aligned}
\lambda(\kappa\iota) &= \lambda + (\kappa\iota) \circ e^{\dagger} && \text{by Grothendieck}\\
&= \lambda + (\kappa + \iota \circ d^{\dagger}) \circ e^{\dagger} && \text{by Grothendieck}\\
&= \lambda + (\kappa \circ e^{\dagger} + \iota \circ d^{\dagger} \circ e^{\dagger}) && \text{by bilinearity}\\
&= \lambda + (\kappa \circ e^{\dagger} + \iota \circ (d \circ e)^{\dagger}) && \text{by functoriality}\\
&= \lambda + (\kappa \circ e^{\dagger} + \iota \circ (e \circ^{\mathrm{op}} d)^{\dagger}) && \text{by ``pointwise opposite''}\\
&= (\lambda + \kappa \circ e^{\dagger}) + \iota \circ (e \circ^{\mathrm{op}} d)^{\dagger} && \text{by monoid associativity}\\
&= (\lambda\kappa)\iota && \text{by Grothendieck}
\end{aligned}$$

Since the identity effect is the constant 0, it is easy to see that horizontal composition is strictly unital on effects:

$$0\kappa = 0 + \kappa \circ \mathsf{id} = \kappa = \kappa + 0 \circ d^{\dagger} = \kappa 0$$

Secondly, note that the well-definedness of horizontal composition in $\mathbf{SGame}_{\mathcal{C}}$ depends furthermore on the distributivity of reindexing over sums (*cf.* Example 5.3.5). Suppose we have 1-cells and 2-cells in $\mathbf{SGame}_{\mathcal{C}}$ as in the following diagram:

Then, writing ϕ for horizontal composition in $\mathbf{SGame}_{\mathcal{C}}$ and $\circ$ for composition in Stat_2 (and leaving the discarding of coparameters implicit):

$$\begin{aligned}
L^{d'} \phi L^{c'} &= (L^d + K^{\beta}) \phi (L^c + K^{\alpha}) && \textit{ex hypothesi}\\
&= (L^d + K^{\beta})_c + (L^c + K^{\alpha}) \circ \overline{d}_c && \text{by Grothendieck}\\
&= L^d_c + K^{\beta}_c + (L^c \circ \overline{d}_c) + (K^{\alpha} \circ \overline{d}_c) && \text{by distributivity and bilinearity}\\
&= L^d_c + (L^c \circ \overline{d}_c) + K^{\beta}_c + (K^{\alpha} \circ \overline{d}_c) && \text{by commutativity of the effect monoid}\\
&= L^d \phi L^c + K^{\beta} \phi K^{\alpha} && \text{by Grothendieck}
\end{aligned}$$

Remark 5.3.12. Of course, we don't strictly need to use $\mathbf{BayesLens}_2$ in the preceding; the structure equally makes sense if we work only with 'marginalized' lenses in $\mathbf{BayesLens}$. In this case, although $\mathbf{BayesLens}$ is a 1-category, one still obtains 2-cells between statistical games, because it remains possible to consider their differences.

5.3.2. Inference systems and loss models

In the context of approximate inference, one often does not have a single statistical model to evaluate, but a whole family of them. In particularly nice situations, this family is actually a subcategory $\mathcal{D}$ of $\mathcal{C}$, with the family of statistical models being all those that can be composed in $\mathcal{D}$. The problem of approximate inference can then be formalized as follows. Since both $\mathbf{BayesLens}_2$ and $\mathbf{SGame}_{\mathcal{C}}$ were obtained by bicategorical Grothendieck constructions, we have a pair of 2-fibrations $\mathbf{SGame}_{\mathcal{C}} \xrightarrow{\pi_{\mathsf{Loss}}} \mathbf{BayesLens}_2 \xrightarrow{\pi_{\mathsf{Lens}}} \mathbf{Copara}_2^l(\mathcal{C})$. Each of π_{Loss}, π_{Lens}, and the discarding functor $(-)^{\updownarrow}$ can be restricted to the subcategory $\mathcal{D}$. The inclusion $(-)^{\curlyvee} : \mathcal{D} \hookrightarrow \mathbf{Copara}_2^l(\mathcal{D})$ restricts to a section of this restriction of $(-)^{\updownarrow}$; the assignment of inversions to channels in $\mathcal{D}$ then corresponds to a 2-section of the 2-fibration π_{Lens} (restricted to $\mathcal{D}$); and the subsequent assignment of losses is a further 2-section of π_{Loss}. This situation is depicted in the following diagram of bicategories:

$$
\begin{array}{ccc}
\mathbf{SGame}_{\mathcal{D}} & \lhook\joinrel\longrightarrow & \mathbf{SGame}_{\mathcal{C}} \\
{\scriptstyle\nearrow} \Big\downarrow {\scriptstyle \pi_{\mathsf{Loss}}|_{\mathcal{D}}} & & \Big\downarrow {\scriptstyle \pi_{\mathsf{Loss}}} \\
\mathbf{BayesLens}_2|_{\mathcal{D}} & \lhook\joinrel\longrightarrow & \mathbf{BayesLens}_2 \\
{\scriptstyle\nearrow} \Big\downarrow {\scriptstyle \pi_{\mathsf{Lens}}|_{\mathcal{D}}} & & \Big\downarrow {\scriptstyle \pi_{\mathsf{Lens}}} \\
\mathbf{Copara}_2^l(\mathcal{D}) & \lhook\joinrel\longrightarrow & \mathbf{Copara}_2^l(\mathcal{C}) \\
{\scriptstyle\curvearrowright} \Big\downarrow {\scriptstyle \overline{\overline{\top}}|_{\mathcal{D}}} & & \Big\downarrow {\scriptstyle \overline{\overline{\top}}} \\
\mathcal{D} & \lhook\joinrel\longrightarrow & \mathcal{C}
\end{array}
$$

This motivates the following definitions of *inference system* and *loss model*, although, for the sake of our examples, we will explicitly allow the loss-assignment to be *lax*: if L is a loss model and c and d are composable lenses, then rather than an equality or natural isomorphism $L(d) \diamond L(c) \cong L(d \varoslash c)$, we will only require a natural transformation $L(d) \diamond L(c) \Rightarrow L(d \varoslash c)$.

Before defining loss models and inference systems, it helps to recall the concept of *essential image*: a generalization of the notion of *image* from functions to functors.

Definition 5.3.13 ([196]). Suppose $F : \mathcal{C} \to \mathcal{D}$ is an n-functor (a possibly weak homomorphism of weak n-categories). The *image* of F is the smallest sub-n-category of $\mathcal{D}$ that contains $F(\alpha)$ for all k-cells α in $\mathcal{C}$, along with any $(k + 1)$-cells relating images of composites and composites of images, for all $0 \leqslant k \leqslant n$. We say that a sub-n-category $\mathcal{D}$ is *replete* if, for any k-cells α in $\mathcal{D}$

and β in $\mathcal{C}$ (with $0 \leqslant k < n$) such that $f : \alpha \Rightarrow \beta$ is a $(k+1)$-isomorphism in $\mathcal{C}$, then f is also a $(k+1)$-isomorphism in $\mathcal{D}$. The *essential image* of F, denoted $\mathsf{im}(F)$, is then the smallest replete sub-n-category of $\mathcal{D}$ containing the image of F.

With these concepts in mind, we state our definitions.

Definition 5.3.14. Suppose $(\mathcal{C}, \otimes, I)$ is a copy-delete category. An *inference system* in $\mathcal{C}$ is a pair $(\mathcal{D}, \ell)$ of a subcategory $\mathcal{D} \hookrightarrow \mathcal{C}$ along with a section $\ell : \mathcal{D}^{\curlyvee} \to \mathbf{BayesLens}_2|_{\mathcal{D}}$ of $\pi_{\mathsf{Lens}}|_{\mathcal{D}}$, where $\mathcal{D}^{\curlyvee}$ is the essential image of the canonical lax inclusion $(-)^{\curlyvee} : \mathcal{D} \hookrightarrow \mathbf{Copara}_2^l(\mathcal{D})$.

Definition 5.3.15. Suppose $(\mathcal{C}, \otimes, I)$ has bilinear effects and $\mathcal{B}$ is a subbicategory of $\mathbf{BayesLens}_2$. A *loss model* for $\mathcal{B}$ is a lax section L of the restriction $\pi_{\mathsf{Loss}}|_{\mathcal{B}}$ of π_{Loss} to $\mathcal{B}$. We say that L is a *strict* loss model if it is in fact a strict 2-functor, and a *strong* loss model if it is in fact a pseudofunctor.

Remark 5.3.16. We may often be interested in loss models for which $\mathcal{B}$ is in fact the essential image of an inference system, but we do not stipulate this requirement in the definition as it is not necessary for the following development.

In order for two loss models F and G to be comparable, they must both be sections of the same fibration of statistical games. One consequence of this is that both F and G must map each 0-cell (X, A) in the bicategory of lenses to the same 0-cell in the bicategory of games, which (by the definition of the bicategory of games) must again be (X, A). In such circumstances, the relevant type of morphism of lax functors is the *icon*, whose definition we now review.

Definition 5.3.17 (Johnson and Yau [145, Def. 4.6.2]). Suppose F and G are lax functors $\mathcal{B} \to \mathcal{C}$ such that, for all $b : \mathcal{B}$, $Fb = Gb$. An *icon* (or *identity component oplax natural transformation*) $\alpha : F \to G$ consists of a family of natural transformations

$$
\mathcal{B}(a, b) \quad
\begin{array}{c}
\overset{F_{a,b}}{\overbrace{\hspace{3cm}}} \\[-0.3em]
\alpha_{a,b} \Big\Downarrow \qquad \mathcal{C}(Fa, Fb) = \mathcal{C}(Ga, Gb) \\[-0.3em]
\underset{G_{a,b}}{\underbrace{\hspace{3cm}}}
\end{array}
$$

for each pair a, b of 0-cells in $\mathcal{B}$, satisfying coherence conditions corresponding to unity and oplax naturality, and whose component 2-cells we write as $\alpha_f : Ff \Rightarrow Gf$ for each 1-cell f in $\mathcal{B}$.

Lax functors $\mathcal{B} \to \mathcal{C}$ and icons between them constitute the objects and morphisms of a category, $\mathbf{Bicat}_{\mathrm{ic}}(\mathcal{B}, \mathcal{C})$, which we can use to construct categories of loss models. Moreover, owing to the monoidality of $+$, this category will be moreover monoidal: a property that we will use to define

the free energy loss model below. (Note that this monoidal structure, on the *category* of loss models, is distinct from the monoidal structure that we will attach to loss models themselves in §5.4.)

Proposition 5.3.18. Loss models for $\mathcal{B}$ constitute the objects of a symmetric monoidal category $\big(\mathsf{Loss}(\mathcal{B}), +, 0\big)$. The morphisms of $\mathsf{Loss}(\mathcal{B})$ are icons between the corresponding lax functors, and they compose accordingly. The monoidal structure is given by sums of losses.

Proof sketch. From Johnson and Yau [145, Theorem 4.6.13], we know that icons compose, forming the morphisms of a category. Next, note that for any two loss models F and G and any k-cell α (for any $k \in \{0, 1, 2\}$), $F(\alpha)$ and $G(\alpha)$ must only differ on the loss component, and so we can sum the losses; this gives the monoidal product. The monoidal unit is necessarily the constant 0 loss. Finally, observe that the structure is symmetric becauase effect monoids are commutative (by Definition 5.3.1). □

5.3.3. Examples

Each of our examples involves taking expectations of log-densities, and so to make sense of them it first helps to understand what we mean by "taking expectations".

Notation 5.3.19 (Expectations). Written as a function, a density p on X has the type $X \to \mathbb{R}_+$; written as an effect, the type is $X \nrightarrow I$. Given a measure or distribution π on X (equivalently, a state $\pi : I \nrightarrow X$), we can compute the expectation of p under π as the composite $p \bullet \pi$. We write the resulting quantity as $\mathbb{E}_\pi[p]$, or more explicitly as $\mathbb{E}_{x \sim \pi}\big[p(x)\big]$. We can think of this expectation as representing the 'validity' (or truth value) of the 'predicate' p given the state π [142].

5.3.3.1. Relative entropy and Bayesian inference

For our first example, we return to the subject with which we opened this paper: the compositional structure of the relative entropy. We begin by giving a precise definition.

Definition 5.3.20. Suppose α, β are both measures on X, with α absolutely continuous with respect to β. Then the *relative entropy* or *Kullback-Leibler divergence* from α to β is the quantity $D_{KL}(\alpha, \beta) := \mathbb{E}_\alpha\left[\log \frac{\alpha}{\beta}\right]$, where $\frac{\alpha}{\beta}$ is the Radon-Nikodym derivative of α with respect to β.

Remark 5.3.21. When α and β admit density functions p_α and p_β with respect to the same base measure $\mathrm{d}x$, then $D_{KL}(\alpha, \beta)$ can equally be computed as $\mathbb{E}_{x \sim \alpha}\big[\log p_\alpha(x) - \log p_\beta(x)\big]$. It it this form that we will adopt henceforth.

Proposition 5.3.22. Let $\mathcal{B}$ be a subbicategory of simple lenses in $\mathbf{BayesLens}_2$, all of whose channels admit density functions with respect to a common measure and whose forward channels admit Bayesian inversion (and whose forward and backward coparameters coincide), and with only structural 2-cells. Then the relative entropy defines a strict loss model $\mathsf{KL} : \mathcal{B} \to \mathbf{SGame}$. Given a lens $(c, c') : (X, X) \nrightarrow (Y, Y)$, KL assigns the loss function $\mathsf{KL}(c, c') : Y \overset{X}{\nrightarrow} I$ defined, for $\pi : I \nrightarrow X$ and $y : Y$, by the relative entropy $\mathsf{KL}(c, c')_\pi(y) := D_{KL}\big(c'_\pi(y), c^\dagger_\pi(y)\big)$, where $c^\dagger$ is the exact inversion of c.

Proof. Being a section of $\pi_{\mathsf{Loss}}|_\mathcal{B}$, KL leaves lenses unchanged, only acting to attach loss functions. It therefore suffices to check that this assignment of losses is strictly functorial. Writing $\bullet$ for composition in $\mathcal{C}$, $\circ$ for horizontal composition in Stat_2, ϕ in $\mathbf{BayesLens}_2$, and $\diamond$ for horizontal composition of losses in $\mathbf{SGame}$, we have the following chain of equalities:

$$
\begin{aligned}
\mathsf{KL}\big((d, d') \phi (c, c')\big)_\pi(z) &= \underset{(x,m,y,n) \sim (c' \circ d'_c)_\pi(z)}{\mathbb{E}} \Big[\log p_{(c' \circ d'_c)_\pi}(x, m, y, n | z) \\
&\qquad\qquad\qquad - \log p_{(c^\dagger \circ d^\dagger_c)_\pi}(x, m, y, n | z) \Big] \\
&= \underset{(y,n) \sim d'_{c \bullet \pi}(z)}{\mathbb{E}} \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \Big[\log p_{c'_\pi}(x, m | y)\, p_{d'_{c \bullet \pi}}(y, n | z) \\
&\qquad\qquad\qquad - \log p_{c^\dagger_\pi}(x, m | y)\, p_{d^\dagger_{c \bullet \pi}}(y, n | z) \Big] \\
&= \underset{(y,n) \sim d'_{c \bullet \pi}(z)}{\mathbb{E}} \Big[\log p_{d'_{c \bullet \pi}}(y, n | z) - \log p_{d^\dagger_{c \bullet \pi}}(y, n | z) \\
&\qquad + \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \Big[\log p_{c'_\pi}(x, m | y) - \log p_{c^\dagger_\pi}(x, m | y) \Big] \Big] \\
&= D_{KL}\big(d'_{c \bullet \pi}(z), d^\dagger_{\bullet \pi}(z)\big) + \underset{(y,n) \sim d'_{c \bullet \pi}(z)}{\mathbb{E}} \Big[D_{KL}\big(c'_\pi(y), c^\dagger_\pi(y)\big) \Big] \\
&= \mathsf{KL}(d, d')_{c \bullet \pi}(z) + \big(\mathsf{KL}(c, c') \circ d'_c\big)_\pi(z) \\
&= \big(\mathsf{KL}(d, d') \diamond \mathsf{KL}(c, c')\big)_\pi(z)
\end{aligned}
$$

The first line obtains by definition of KL and ϕ; the second by definition of $\circ$; the third by the $\log$ adjunction ($\log ab = \log a + \log b$) and by linearity of $\mathbb{E}$; the fourth by definition of D_{KL}; the fifth by definition of KL and of $\circ$; and the sixth by definition of $\diamond$.

This establishes that $\mathsf{KL}\big((d, d') \phi (c, c')\big) = \mathsf{KL}(d, d') \diamond \mathsf{KL}(c, c')$ and hence that KL is strictly functorial on 1-cells. Since we have assumed that the only 2-cells are the structural 2-cells (*e.g.*, the horizontal unitors), which do not result in any difference between the losses assigned to the corresponding 1-cells, the only loss 2-cell available to be assigned is the 0 loss; which assignment is easily seen to be vertically functorial. Hence KL is a strict 2-functor, and moreover a section of $\pi_{\mathsf{Loss}}|_\mathcal{B}$ as required. $\qquad\square$

Successfully playing a relative entropy game entails minimizing the divergence from the approximate to the exact posterior. This divergence is minimized when the two coincide, and so KL represents a form of approximate Bayesian inference.

Remark 5.3.23. We opened the chapter by observing that the relative entropy satisfies a chain rule defined not on Bayesian lenses, but simply on pairs of channels: to formalize this simpler case, we do not need the full machinery of statistical games (which is useful when we have bidirectional inference systems); but we do need some of it.

If c and c' are parallel channels $X \rightarrowtail Y$, then $D_{KL}(c(-), c'(-))$ defines an effect $X \rightarrowtail I$. This means we can use the statistical games idea to equip parallel (copy-composite) channels in $\mathcal{C}$ with such non-state-dependent loss functions; and the relative entropy will again form a strict section of the resulting Grothendieck fibration.

Therefore, let $\mathcal{B}$ be the bicategory whose 0-cells are the objects of $\mathcal{C}$, but whose 1-cells and 2-cells are parallel pairs of 1-cells and 2-cells in $\mathbf{Copara}_2(\mathcal{C})$; equivalently, the subbicategory of $\mathbf{Copara}_2(\mathcal{C})^\mathbf{2}$ which is diagonal on 0-cells.

Next, let K denote the indexed bicategory $\mathcal{B}^{\mathrm{co\,op}} \to \mathbf{Bicat}$ obtained as the composite

$$\mathcal{B}^{\mathrm{co\,op}} \xrightarrow{\mathsf{proj}_1} \mathbf{Copara}_2(\mathcal{C})^{\mathrm{co\,op}} \xrightarrow{\overline{\overline{\top}}^{\mathrm{co\,op}}} \mathcal{C}^{\mathrm{op}} \xrightarrow{\mathcal{C}(-,I)} \mathbf{MonCat} \xrightarrow{\mathbf{B}} \mathbf{Bicat}$$

where proj_1 indicates the projection of the 1st factor of the parallel pairs of 1-cells and 2-cells. Applying the Grothendieck construction to K yields a 2-fibration $\int K \xrightarrow{\pi_K} \mathcal{B}$. The 0-cells of $\int K$ are the objects of $\mathcal{C}$. The 1-cells $X \to Y$ are triples (c, c', L) where c and c' are parallel coparameterized channels $X \rightarrowtail Y$ and L is an effect (loss function) $X \rightarrowtail I$. Given composable 1-cells $(c, c', L) : X \to Y$ and $(d, d', M) : Y \to Z$, their horizontal composite is defined on the parallel channels as copy-composition, and on the loss functions as $M \bullet c^{\overline{\top}} + L$ (where $\bullet$ here is composition in $\mathcal{C}$). 2-cells are pairs of 2-cells in $\mathbf{Copara}_2(\mathcal{C})$ and differences of losses.

Finally, the relative entropy D_{KL} defines a strict section of π_K, mapping the parallel pair (c, c') to $(c, c', D_{KL}(c, c'))$. Its chain rule is thus formalized by the horizontal composition in $\int K$. $\qquad\square$

5.3.3.2. Maximum likelihood estimation

A statistical system may be more interested in predicting observations than updating beliefs. This is captured by the process of *maximum* or *marginal likelihood estimation*.

Definition 5.3.24. Let $(c, c') : (X, X) \rightarrowtail (Y, Y)$ be a simple lens whose forward channel c admits a density function p_c. Then its *log marginal likelihood* is the loss function given by the marginal log evidence $\mathsf{MLE}(c, c')_\pi(y) := -\log p_{c^{\overline{\top}} \bullet \pi}(y)$.

Proposition 5.3.25. Let $\mathcal{B}$ be a subbicategory of lenses in $\mathbf{BayesLens}_2$ all of which admit density functions with respect to a common measure, and with only structural 2-cells. Then the assignment $(c, c') \mapsto \mathsf{MLE}(c, c')$ defines a lax loss model $\mathsf{MLE} : \mathcal{B} \to \mathbf{SGame}$.

Proof. We adopt the notational conventions of the proof of Proposition 5.3.22. Observe that

$$\mathsf{MLE}\big((d, d') \mathbin{\phi} (c, c')\big)_\pi(z) = -\log p_{d^\dagger \bullet c^\dagger \bullet \pi}(z) = \mathsf{MLE}(d, d')_{c \bullet \pi}(z) \,.$$

By definition, we have

$$\big(\mathsf{MLE}(d, d') \diamond \mathsf{MLE}(c, c')\big)_\pi(z) = \mathsf{MLE}(d, d')_{c \bullet \pi}(z) + \big(\mathsf{MLE}(c, c') \circ d'_c\big)_\pi(z)$$

and hence by substitution

$$\big(\mathsf{MLE}(d, d') \diamond \mathsf{MLE}(c, c')\big)_\pi(z) = \mathsf{MLE}\big((d, d') \mathbin{\phi} (c, c')\big)_\pi(z) + \big(\mathsf{MLE}(c, c') \circ d'_c\big)_\pi(z) \,.$$

Therefore, $\mathsf{MLE}(c, c') \circ d'_c$ constitutes a 2-cell from $\mathsf{MLE}(d, d') \diamond \mathsf{MLE}(c, c')$ to $\mathsf{MLE}\big((d, d') \mathbin{\phi} (c, c')\big)$, and hence MLE is a lax functor. It is evidently moreover a section of $\pi_{\mathsf{Loss}}|_{\mathcal{B}}$, and, like KL, acts trivially on the (purely structural) 2-cells. $\qquad\square$

Successfully playing a maximum likelihood game involves maximizing the log-likelihood that the system assigns to its observations $y : Y$. This process amounts to choosing a channel c that assigns high likelihood to likely observations, and thus encodes a valid model of the data distribution.

5.3.3.3. Autoencoders via the free energy

Many adaptive systems neither just infer nor just predict: they do both, building a model of their observations that they also invert to update their beliefs. In machine learning, such systems are known as *autoencoders*, as they 'encode' (infer) and 'decode' (predict), 'autoassociatively' [161]. In a Bayesian context, they are known as *variational autoencoders* [154], and their loss function is the *free energy* [80].

Definition 5.3.26. The *free energy* loss model is the sum of the relative entropy and the likelihood loss models: $\mathsf{FE} := \mathsf{KL} + \mathsf{MLE}$. Given a simple lens $(c, c') : (X, X) \nrightarrow (Y, Y)$ admitting Bayesian inversion and with densities, FE assigns the loss function

$$\begin{aligned}
\mathsf{FE}(c, c')_\pi(y) &= (\mathsf{KL} + \mathsf{MLE})(c, c')_\pi(y) \\
&= D_{KL}\big(c'_\pi(y), c^\dagger_\pi(y)\big) - \log p_{c^\dagger \bullet \pi}(y)
\end{aligned}$$

Note that this means that optimizing the free energy is not guaranteed to optimize either KL or MLE individually, although by definition FE is an upper bound on them both (and hence often known in machine learning by the alternative name, the *evidence upper bound*, thinking of MLE as encoding a measure of 'evidence').

Remark 5.3.27. Beyond its autoencoding impetus, another important property of the free energy is its improved computational tractability compared to either the relative entropy or the likelihood loss. This property is a consequence of the following fact: although obtained as the sum of terms which both depend on an expensive marginalization[8], the free energy itself does not. This can be seen by expanding the definitions of the relative entropy and of $c_\pi^\dagger$ and rearranging terms:

$$
\begin{aligned}
\mathsf{FE}(c, c')_\pi(y) &= D_{KL}\big(c'_\pi(y), c_\pi^\dagger(y)\big) - \log p_{c^{\bar{\ast}} \bullet \pi}(y) \\
&= \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \big[\log p_{c'_\pi}(x, m|y) - \log p_{c_\pi^\dagger}(x, m|y)\big] - \log p_{c^{\bar{\ast}} \bullet \pi}(y) \\
&= \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \big[\log p_{c'_\pi}(x, m|y) - \log p_{c_\pi^\dagger}(x, m|y) - \log p_{c^{\bar{\ast}} \bullet \pi}(y)\big] \\
&= \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \Big[\log p_{c'_\pi}(x, m|y) - \log \frac{p_c(m, y|x) p_\pi(x)}{p_{c^{\bar{\ast}} \bullet \pi}(y)} - \log p_{c^{\bar{\ast}} \bullet \pi}(y)\Big] \\
&= \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \big[\log p_{c'_\pi}(x, m|y) - \log p_c(m, y|x) - \log p_\pi(x)\big] \qquad (5.5) \\
&= D_{KL}\big(c'_\pi(y), \pi \otimes \mathbf{1}\big) - \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \big[\log p_c(m, y|x)\big]
\end{aligned}
$$

Here, $\mathbf{1}$ denotes the measure with density 1 everywhere. Note that when the coparameter is trivial, $\mathsf{FE}(c, c')_\pi(y)$ reduces to

$$
D_{KL}\big(c'_\pi(y), \pi\big) - \underset{x \sim c'_\pi(y)}{\mathbb{E}} \big[\log p_c(y|x)\big] .
$$

Remark 5.3.28. The name *free energy* is due to an analogy with the Helmholtz free energy in thermodynamics, as we can write it as the difference between an (expected) energy and an entropy term:

$$
\begin{aligned}
\mathsf{FE}(c, c')_\pi(y) &= \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \big[-\log p_c(m, y|x) - \log p_\pi(x)\big] - S_{X \otimes M}\big[c'_\pi(y)\big] \\
&= \underset{(x,m) \sim c'_\pi(y)}{\mathbb{E}} \big[E_{(c,\pi)}(x, m, y)\big] - S_{X \otimes M}\big[c'_\pi(y)\big] = U - TS
\end{aligned}
$$

where we call $E_{(c,\pi)} : X \otimes M \otimes Y \xrightarrow{X} I$ the *energy*, and where $S_{X \otimes M} : I \xrightarrow{X \otimes M} I$ is the Shannon entropy. The last equality makes the thermodynamic analogy: U here is the *internal energy* of the system; $T = 1$ is the *temperature*; and S is again the entropy.

---

[8] Evaluating the pushforward $c^{\bar{\ast}} \bullet \pi$ involves marginalizing over the intermediate variable; and evaluating $c_\pi^\dagger(y)$ also involves evaluating $c^{\bar{\ast}} \bullet \pi$.

5.3.3.4. The Laplace approximation

Although optimizing the free energy does not necessitate access to exact inversions, it does still entail computing an expectation under the approximate inversion (*cf.* equation (5.5) of Remark 5.3.27 above), which may remain non-trivial. When one is interested in optimizing a model by gradient descent, this becomes particularly pressing, as one needs to form an estimate of the gradient of this expectation with respect to the parameters (which is not in general equal to the expectation of the gradient of the energy). In machine learning, optimizing variational autoencoders typically involves a "reparameterization trick" [155, §2.5] to circumvent this difficulty, but in the context of neuroscientific modelling (where one is concerned with biological plausibility), this option is not generally available.

An alternative strategy is to make simplifying assumptions, enabling the desired computations without totally sacrificing biological realism. In the context of predictive coding, a typical such assumption is that all measures are Gaussian [21, 33, 48, 104, 216]. This is motivated not only by hypotheses about the nature of biological noise (related to the Central Limit Theorem), but also by expediency, as a Gaussian distribution is determined by just two sufficient statistics: its mean and variance. If one first restricts to lenses with Gaussian channels, and then to lenses whose inversion morphisms are constrained to emit 'tightly-peaked' Gaussians (*i.e.*, with small variances), then one can eliminate the expectation from the expected energy, and simply evaluate the energy at the posterior mean.

The conceptual justification for this approximation is due to Laplace [163, p.367], who observed that, given a function of the form $f(x) = e^{n\,h(x)}$ with h having a maximum at x_0, the only non-negligible contributions to its integral as $n \to \infty$ are those near to x_0[9]. Consequently, the function h can be approximated by the quadratic form obtained from its its 2nd-order Taylor expansion about x_0, so that, in the one-dimensional (univariate) case,

$$\int f(x)\,\mathrm{d}x \approx e^{n\,h(x_0)} \int e^{-\frac{1}{2\sigma^2}(x-x_0)^2}\,\mathrm{d}x$$

for $\sigma = \left(n\,\partial_x^2 h(x_0)\right)^{-1/2}$. Notably, the integrand on the right-hand side is a Gaussian function: it has the form of the density of a normal distribution.

In the present context, we are generally interested in expectations of the form $\mathbb{E}_{x\sim\pi}\big[g(x)\big]$, which correspond to integrals $\int g(x)\,e^{\log p_\pi(x)}\,\mathrm{d}x$. It is possible to extend the foregoing reasoning to this case: supposing that $\log p_\pi(x) \propto n\,h(x)$ for some function h with a maximum at x_0, then as

---

[9]A demonstration of this can be found on Wikipedia at `https://en.wikipedia.org/w/index.php?title=Laplace%27s_method&oldid=1154930495`.

$n \to \infty$, we can approximate both g and h by their 2nd-order expansions, thereby approximating π by a Gaussian and g by a quadratic form.

This method of approximating integrals is known as *Laplace's method*, and it has been widely applied in statistics[10] [22, 101, 151, 280] [176, Chp. 27], in some circumstances, even yielding exact posteriors [267, §10.2]. For further exposition (and more rigour) in the finite-dimensional case, we refer the reader to Bruijn [46, Ch. 4] and Olver [200, §3.7]; for the general case in Banach spaces, the reader may consult Piterbarg and Fatalov [211]. And for an analysis of the specific case of approximating Bayesian posteriors (beyond the exact case), with consideration of the approximation errors, one may refer to Kass, Tierney, and Kadane [150] or the technical report accompanying Tierney and Kadane [263].

This latter situation is of course closely related to the matter at hand. Here, rather than approximating the posterior by a Gaussian, we *assume* it to have Gaussian form.

Remark 5.3.29. We say that a channel $c : X \nrightarrow Y$ is *Gaussian* if $c(x)$ is a Gaussian measure for every x in its domain. We denote the mean and variance of $c(x)$ by $\mu_c(x)$ and $\Sigma_c(x)$ respectively, and write its (log) density function as

$$\log p_c(y|x) = \frac{1}{2} \left\langle \epsilon_c(y, x), \Sigma_c(x)^{-1} \epsilon_c(y, x) \right\rangle - \log \sqrt{(2\pi)^n \det \Sigma_c(x)}$$

having also defined the 'error' function $\epsilon_c : Y \times X \to Y$ by $\epsilon_c(y, x) = y - \mu_c(x)$. In §7.1, we give a full definition of a category of (nonlinear) Gaussian channels.

We will still be concerned with approximating expectations $\mathbb{E}_{x \sim d(y)}\big[g(x)\big]$ by the quadratic expansion of g, and so to license Laplace's method we need an analogue of the condition $n \to \infty$. This will be supplied by the further assumption that $\Sigma_d(y)$ has small eigenvalues: that is, we work in the limit $\mathrm{tr}\,(\Sigma_d(y)) \to 0$. With these two assumptions, we can write

$$\mathbb{E}_{x \sim d(y)}\big[g(x)\big] \propto \int_{x:X} g(x) \exp \left\langle \epsilon_d(x, y), \Sigma_d(y)^{-1} \epsilon_d(x, y) \right\rangle \mathrm{d}x$$

and observe that as $\mathrm{tr}\,(\Sigma_d(y)) \to 0$, we must have $\mathrm{tr}\left(\Sigma_d(y)^{-1}\right) \to \infty$. Thus, by Laplace's reasoning, the contributions to the integral are only appreciably non-zero near the mean $\mu_d(y)$. This licenses the approximation of g by its quadratic expansion around $\mu_d(y)$, and leads to the following approximation of the free energy, known in the predictive coding literature as the *Laplace approximation* [101]. (Consistent with the other examples in this chapter, we consider the coparameterized case.)

---

[10]In statistics, making Gaussian assumptions about Bayesian posteriors, or equivalently using second-order approximations to log posteriors, is also known as *variational Laplace* [101].

Definition 5.3.30. A *Cartesian space* is an object X that is isomorphic to $\mathbb{R}^n$ for some $n : \mathbb{N}$.

Proposition 5.3.31 (Laplacian free energy). Suppose $(\gamma, \rho) : (X, X) \nrightarrow (Y, Y)$ is a Bayesian lens with Gaussian channels between finite-dimensional Cartesian spaces, for which, for all $y : Y$ and Gaussian priors $\pi : I \nrightarrow X$, the eigenvalues of $\Sigma_{\rho_\pi}(y)$ are small. Then the free energy $\mathsf{FE}(\gamma, \rho)_\pi(y)$ can be approximated by the *Laplacian free energy*

$$\mathsf{FE}(\gamma, \rho)_\pi(y) \approx \mathsf{LFE}(\gamma, \rho)_\pi(y) \tag{5.6}$$

$$:= E_{(\gamma, \pi)}\big(\mu_{\rho_\pi}(y), y\big) - S_{X \otimes M}\big[\rho_\pi(y)\big] \tag{5.7}$$

$$= -\log p_\gamma(\mu_{\rho_\pi}(y), y) - \log p_\pi(\mu_{\rho_\pi}(y)|_X) - S_{X \otimes M}\big[\rho_\pi(y)\big]$$

where we have written the argument of the density p_γ in 'function' style; where $(-)_X$ denotes the projection onto X; and where $S_{X \otimes M}[\rho_\pi(y)] = \mathbb{E}_{(x,m) \sim \rho_\pi(y)}[-\log p_{\rho_\pi}(x, m|y)]$ is the Shannon entropy of $\rho_\pi(y)$. The approximation is valid when Σ_{ρ_π} satisfies

$$\Sigma_{\rho_\pi}(y) = \left(\partial^2_{(x,m)} E_{(\gamma, \pi)}\right)(\mu_{\rho_\pi}(y), y)^{-1} . \tag{5.8}$$

We call $E_{(\gamma, \pi)}$ the *Laplacian energy*.

Proof. Recall that we can write the free energy $\mathsf{FE}(\gamma, \rho)_\pi(y)$ as the difference between expected energy and entropy:

$$\mathsf{FE}(\gamma, \rho)_\pi(y) = \mathop{\mathbb{E}}_{(x,m) \sim \rho_\pi(y)}\big[-\log p_\gamma(m, y|x) - \log p_\pi(x)\big] - S_{X \otimes M}\big[\rho_\pi(y)\big]$$

$$= \mathop{\mathbb{E}}_{(x,m) \sim \rho_\pi(y)}\big[E_{(\gamma, \pi)}(x, m, y)\big] - S_X\big[\rho_\pi(y)\big]$$

Next, since the eigenvalues of $\Sigma_{\rho_\pi}(y)$ are small for all $y : Y$, we can approximate the expected energy by its second-order Taylor expansion around the mean $\mu_{\rho_\pi}(y)$, following Laplace:

$$\mathsf{FE}(\gamma, \rho)_\pi(y) \approx \mathop{\mathbb{E}}_{(x,m) \sim \rho_\pi(y)}\Bigg[E_{(\gamma, \pi)}(\mu_{\rho_\pi}(y), y) + \Big\langle \epsilon_{\rho_\pi}(x, m, y),\ \big(\partial_{(x,m)} E_{(\gamma, \pi)}\big)(\mu_{\rho_\pi}(y), y)\Big\rangle$$

$$+ \frac{1}{2}\Big\langle \epsilon_{\rho_\pi}(x, m, y),\ \big(\partial^2_{(x,m)} E_{(\gamma, \pi)}\big)(\mu_{\rho_\pi}(y), y) \cdot \epsilon_{\rho_\pi}(x, m, y)\Big\rangle\Bigg]$$

$$- S_{X \otimes M}\big[\rho_\pi(y)\big]$$

$$\overset{(a)}{=} E_{(\gamma, \pi)}(\mu_{\rho_\pi}(y), y) + \Big\langle \mathop{\mathbb{E}}_{(x,m) \sim \rho_\pi(y)}\big[\epsilon_{\rho_\pi}(x, m, y)\big],\ \big(\partial_{(x,m)} E_{(\gamma, \pi)}\big)(\mu_{\rho_\pi}(y), y)\Big\rangle$$

$$+ \frac{1}{2}\operatorname{tr}\Big[\big(\partial^2_{(x,m)} E_{(\gamma, \pi)}\big)(\mu_{\rho_\pi}(y), y)\ \Sigma_{\rho_\pi}(y)\Big] - S_{X \otimes M}\big[\rho_\pi(y)\big]$$

$$\overset{(b)}{=} E_{(\gamma, \pi)}(\mu_{\rho_\pi}(y), y) + \frac{1}{2}\operatorname{tr}\Big[\big(\partial^2_{(x,m)} E_{(\gamma, \pi)}\big)(\mu_{\rho_\pi}(y), y)\ \Sigma_{\rho_\pi}(y)\Big] - S_{X \otimes M}\big[\rho_\pi(y)\big]$$

where $\left(\partial^2_{(x,m)} E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y)$ is the Hessian of $E_{(\gamma,\pi)}$ with respect to (x, m) evaluated at $(\mu_{\rho_\pi}(y), y)$. The equality marked (a) holds first by the linearity of expectations and second because

$$
\mathbb{E}_{(x,m)\sim\rho_\pi(y)} \left[\left\langle \epsilon_{\rho_\pi}(x, m, y), \left(\partial^2_{(x,m)} E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y) \cdot \epsilon_{\rho_\pi}(x, m, y) \right\rangle \right]
$$

$$
= \mathbb{E}_{(x,m)\sim\rho_\pi(y)} \left[\operatorname{tr} \left[\left(\partial^2_{(x,m)} E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y) \; \epsilon_{\rho_\pi}(x, m, y) \, \epsilon_{\rho_\pi}(x, m, y)^T \right] \right]
$$

$$
= \operatorname{tr} \left[\left(\partial^2_{(x,m)} E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y) \; \mathbb{E}_{(x,m)\sim\rho_\pi(y)} \left[\epsilon_{\rho_\pi}(x, m, y) \, \epsilon_{\rho_\pi}(x, m, y)^T \right] \right]
$$

$$
= \operatorname{tr} \left[\left(\partial^2_{(x,m)} E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y) \; \Sigma_{\rho_\pi}(y) \right] \tag{5.9}
$$

where the first equality obtains because the trace of an outer product equals an inner product; the second by linearity of the trace; and the third by the definition of the covariance $\Sigma_{\rho_\pi}(y)$. The equality marked (b) above then holds because $\mathbb{E}_{(x,m)\sim\rho_\pi(y)}\left[\epsilon_{\rho_\pi}(x, m, y)\right] = 0$.

Next, note that the entropy of a Gaussian measure depends only on its covariance,

$$
S_{X\otimes M}\big[\rho_\pi(y)\big] = \frac{1}{2} \log \det \left(2\pi \, e \, \Sigma_{\rho_\pi}(y)\right) ,
$$

and that the energy $E_{(\gamma,\pi)}(\mu_{\rho_\pi}(y), y)$ does not depend on $\Sigma_{\rho_\pi}(y)$. We can therefore write down directly the covariance $\Sigma^*_{\rho_\pi}(y)$ minimizing $\mathsf{FE}(\gamma, \rho)_\pi(y)$ as a function of y. We have

$$
\partial_{\Sigma_{\rho_\pi}} \mathsf{FE}(\gamma, \rho)_\pi(y) \stackrel{(b)}{\approx} \frac{1}{2} \left(\partial^2_{(x,m)} E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y) + \frac{1}{2} \Sigma_{\rho_\pi}^{-1}
$$

by equation (b) above. Setting $\partial_{\Sigma_{\rho_\pi}} \mathsf{FE}(\gamma, \rho)_\pi(y) = 0$, we find the optimum as expressed by equation (5.8):

$$
\Sigma^*_{\rho_\pi}(y) = \left(\partial^2_x E_{(\gamma,\pi)}\right)(\mu_{\rho_\pi}(y), y)^{-1} .
$$

Finally, by substituting $\Sigma^*_{\rho_\pi}(y)$ in equation (5.9), we obtain the desired expression, equation (5.6):

$$
\mathsf{FE}(\gamma, \rho)_\pi(y) \approx E_{(\gamma,\pi)}(\mu_{\rho_\pi}(y), y) - S_{X\otimes M}\big[\rho_\pi(y)\big] =: \mathsf{LFE}(\gamma, \rho)_\pi(y) .
$$

$\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad$ □

Remark 5.3.32. The usual form of the Laplace model in the literature omits the coparameters. It is of course easy to recover the non-coparameterized form by taking $M = 1$.

As well as being an approximation to a particular statistical game, the Laplacian free energy defines a lax loss model.

Proposition 5.3.33. Let $\mathcal{B}$ be a subbicategory of $\mathbf{BayesLens}_2$ of Gaussian lenses between Cartesian spaces whose backward channels have small variance, and with only structural 2-cells[11]. Then LFE defines a lax loss model $\mathcal{B} \to \mathbf{SGame}$.

Proof. Again we follow the notational conventions of the proof of Proposition 5.3.22. Additionally, if ω is a state on a tensor product such as $X \otimes Y$, we will write ω_X and ω_Y to denote its X and Y marginals. We will continue to write $c^{\tilde{\ast}}$ to denote the result of discarding the coparameters of a coparameterized channel c.

Observe that, by repeated application of the linearity of $\mathbb{E}$, the log adjunction, and the definitions of $\bullet$ and $\circ$,

$$
\big(\mathsf{LFE}(d,d') \diamond \mathsf{LFE}(c,c')\big)_\pi(z)
$$
$$
= \mathsf{LFE}(d,d')_{c \bullet \pi}(z) + \big(\mathsf{LFE}(c,c') \circ d'_c\big)_\pi(y)
$$
$$
= \mathsf{LFE}(d,d')_{c \bullet \pi}(z) + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\big[\mathsf{LFE}(c,c')_\pi(y)\big]
$$
$$
= -\log p_d\big(\mu_{d'_{c \bullet \pi}}(z), z\big) - \log p_{c^{\tilde{\ast}} \bullet \pi}\big(\mu_{d'_{c \bullet \pi}}(z)_Y\big)
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\bigg[\log p_{d'_{c \bullet \pi}}(y,n|z) - \log p_c\big(\mu_{c'_\pi}(y), y\big) - \log p_\pi\big(\mu_{c'_\pi}(y)_X\big)
$$
$$
\quad\quad\quad\quad + \mathop{\mathbb{E}}_{(x,m) \sim c'_\pi(y)}\big[\log p_{c'_\pi}(x,m|y)\big]\bigg]
$$
$$
= -\log p_d\big(\mu_{d'_{c \bullet \pi}}(z), z\big) - \log p_{c^{\tilde{\ast}} \bullet \pi}\big(\mu_{d'_{c \bullet \pi}}(z)_Y\big)
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\big[-\log p_c\big(\mu_{c'_\pi}(y), y\big) - \log p_\pi\big(\mu_{c'_\pi}(y)_X\big)\big]
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\bigg[\log p_{d'_{c \bullet \pi}}(y,n|z) + \mathop{\mathbb{E}}_{(x,m) \sim c'_\pi(y)}\big[\log p_{c'_\pi}(x,m|y)\big]\bigg]
$$
$$
= -\log p_d\big(\mu_{d'_{c \bullet \pi}}(z), z\big) - \log p_{c^{\tilde{\ast}} \bullet \pi}\big(\mu_{d'_{c \bullet \pi}}(z)_Y\big)
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\big[-\log p_c\big(\mu_{c'_\pi}(y), y\big) - \log p_\pi\big(\mu_{c'_\pi}(y)_X\big)\big]
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)} \mathop{\mathbb{E}}_{(x,m) \sim c'_\pi(y)}\big[\log p_{d'_{c \bullet \pi}}(y,n|z) + \log p_{c'_\pi}(x,m|y)\big]
$$
$$
= -\log p_d\big(\mu_{d'_{c \bullet \pi}}(z), z\big) - \log p_{c^{\tilde{\ast}} \bullet \pi}\big(\mu_{d'_{c \bullet \pi}}(z)_Y\big)
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\big[-\log p_c\big(\mu_{c'_\pi}(y), y\big) - \log p_\pi\big(\mu_{c'_\pi}(y)_X\big)\big]
$$
$$
\quad + \mathop{\mathbb{E}}_{(x,m,y,n) \sim (c' \circ d'_c)_\pi(z)}\big[-\log p_{(c' \circ d'_c)_\pi}(x,m,y,n|z)\big]
$$
$$
= -\log p_d\big(\mu_{d'_{c \bullet \pi}}(z), z\big) - \log p_{c^{\tilde{\ast}} \bullet \pi}\big(\mu_{d'_{c \bullet \pi}}(z)_Y\big)
$$
$$
\quad + \mathop{\mathbb{E}}_{(y,n) \sim d'_{c \bullet \pi}(z)}\big[-\log p_c\big(\mu_{c'_\pi}(y), y\big) - \log p_\pi\big(\mu_{c'_\pi}(y)_X\big)\big] - S_{XMYN}\big[(c' \circ d'_c)_\pi(z)\big]
$$

---

[11]An example of $\mathcal{B}$ here is obtained by restricting $\mathbf{BayesLens}_2$ to the category $\mathbf{FdGauss}$ of Definition 7.1.9, and by excluding all but the structural 2-cells

$$= -\log p_d\big(\mu_{d'_c\bullet\pi}(z), z\big) - \log p_{c^{\bar{\ast}}\bullet\pi}\big(\mu_{d'_c\bullet\pi}(z)_Y\big)$$

$$+ \underset{(y,n)\sim d'_c\bullet\pi(z)}{\mathbb{E}}\big[E_{(c,\pi)}\big(\mu_{c'_\pi}(y), y\big)\big] - S_{XMYN}\big[(c' \circ d'_c)_\pi(z)\big]$$

$$= E_{(d,c\bullet\pi)}(\mu_{d'_c\bullet\pi}(z), z) + \underset{(y,n)\sim d'_c\bullet\pi(z)}{\mathbb{E}}\big[E_{(c,\pi)}\big(\mu_{c'_\pi}(y), y\big)\big] - S_{XMYN}\big[(c' \circ d'_c)_\pi(z)\big]$$

where $XMYN$ is shorthand for $X \otimes M \otimes Y \otimes N$.

Now, writing $E^\mu_{(c,\pi)}(y) := E_{(c,\pi)}\big(\mu_{c'_\pi}(y), y\big)$, by the Laplace assumption, we have

$$\underset{(y,n)\sim d'_c\bullet\pi(z)}{\mathbb{E}}\big[E^\mu_{(c,\pi)}(y)\big] \approx E^\mu_{(c,\pi)}(\mu_{d'_c\bullet\pi}(z)_Y) + \frac{1}{2}\operatorname{tr}\Big[\big(\partial_y^2 E^\mu_{(c,\pi)}\big)\big(\mu_{d'_c\bullet\pi}(z)_Y\big) \Sigma_{d'_c\bullet\pi}(z)_{YY}\Big]$$

and so we can write

$$\big(\mathsf{LFE}(d, d') \diamond \mathsf{LFE}(c, c')\big)_\pi(z)$$

$$\approx E_{(d,c\bullet\pi)}(\mu_{d'_c\bullet\pi}(z), z) + E^\mu_{(c,\pi)}(\mu_{d'_c\bullet\pi}(z)_Y) - S_{XMYN}\big[(c' \circ d'_c)_\pi(z)\big]$$

$$+ \frac{1}{2}\operatorname{tr}\Big[\big(\partial_y^2 E^\mu_{(c,\pi)}\big)\big(\mu_{d'_c\bullet\pi}(z)_Y\big) \Sigma_{d'_c\bullet\pi}(z)_{YY}\Big]$$

$$= -\log p_d\big(\mu_{d'_c\bullet\pi}(z), z\big) - \log p_c\big(\mu_{c'_\pi}(\mu_{d'_c\bullet\pi}(z)_Y), \mu_{d'_c\bullet\pi}(z)_Y\big) - \log p_\pi\big(\mu_{c'_\pi}(\mu_{d'_c\bullet\pi}(z)_Y)_X\big)$$

$$- S_{XMYN}\big[(c' \circ d'_c)_\pi(z)\big] - \log p_{c^{\bar{\ast}}\bullet\pi}\big(\mu_{d'_c\bullet\pi}(z)_Y\big)$$

$$+ \frac{1}{2}\operatorname{tr}\Big[\big(\partial_y^2 E^\mu_{(c,\pi)}\big)\big(\mu_{d'_c\bullet\pi}(z)_Y\big) \Sigma_{d'_c\bullet\pi}(z)_{YY}\Big]$$

$$= E_{(d\bullet c,\pi)}\big(\mu_{(c'\circ d'_c)_\pi}(z), z\big) - S_{XMYN}\big[(c' \circ d'_c)_\pi(z)\big]$$

$$- \log p_{c^{\bar{\ast}}\bullet\pi}\big(\mu_{d'_c\bullet\pi}(z)_Y\big) + \frac{1}{2}\operatorname{tr}\Big[\big(\partial_y^2 E^\mu_{(c,\pi)}\big)\big(\mu_{d'_c\bullet\pi}(z)_Y\big) \Sigma_{d'_c\bullet\pi}(z)_{YY}\Big]$$

$$= \mathsf{LFE}\big((d, d') \diamond (c, c')\big)_\pi(z) - \log p_{c^{\bar{\ast}}\bullet\pi}\big(\mu_{d'_c\bullet\pi}(z)_Y\big)$$

$$+ \frac{1}{2}\operatorname{tr}\Big[\big(\partial_y^2 E^\mu_{(c,\pi)}\big)\big(\mu_{d'_c\bullet\pi}(z)_Y\big) \Sigma_{d'_c\bullet\pi}(z)_{YY}\Big] .$$

Therefore, if we define a loss function κ by

$$\kappa_\pi(z) := \frac{1}{2}\operatorname{tr}\Big[\big(\partial_y^2 E^\mu_{(c,\pi)}\big)\big(\mu_{d'_c\bullet\pi}(z)_Y\big) \Sigma_{d'_c\bullet\pi}(z)_{YY}\Big] - \log p_{c^{\bar{\ast}}\bullet\pi}\big(\mu_{d'_c\bullet\pi}(z)_Y\big)$$

then κ constitutes a 2-cell $\mathsf{LFE}(d, d') \diamond \mathsf{LFE}(c, c') \Rightarrow \mathsf{LFE}\big((d, d') \diamond (c, c')\big)$, as required. $\qquad\square$

Effectively, this proposition says that, under the stated conditions, the free energy and the Laplacian free energy coincide. Consequently, successfully playing a Laplacian free energy game has the same autoencoding effect as playing a free energy game in the sense of §5.3.3.3.

Remark 5.3.34. We formalized the idea of a Gaussian having *small* or *tightly-peaked* variance as meaning its covariance matrix Σ has small eigenvalues. We do not specify precisely what 'small' means here: only, it must be enough to license the use of Laplace's method. Of course, as the

eigenvalues approach 0, the Gaussian approaches a Dirac delta distribution. In this case, one may truncate the approximating expansion at first order and just work with the means — in fact, the inversions become deterministic — and indeed, this is the choice made in some of the predictive coding literature [33].

5.4. Monoidal statistical games

In Remark 4.3.18, we noted that the canonical section $\dagger$ taking a channel c to the lens equipped with its exact inversion $c^\dagger$ is not monoidal, because inverting the tensor of two channels with respect to a joint state is in general not the same as inverting the two channels independently with respect to the marginals, owing to the possibility of correlations. At the same time, we know from Proposition 4.3.11 that the category $\mathbf{BayesLens}_{\mathcal{C}}$ of non-coparameterized Bayesian lenses in $\mathcal{C}$ is nonetheless a monoidal category (and it is moreover symmetric monoidal when $\mathcal{C}$ is); and we saw in Corollary 5.3.10 that Stat, and hence $\mathbf{BayesLens}_{\mathcal{C}}$, are additionally fibrewise monoidal. In this section, we establish analogous results for copy-composite Bayesian lenses, and statistical games and loss models in turn, as well as demonstrating that each of our loss models is accordingly monoidal. This monoidal structure on loss models can then be used to measure the error obtained by inverting channels independently with respect to the marginals of a joint prior.

Because statistical games are defined over copy-composite channels, our starting point must be to establish a monoidal structure on $\mathbf{Copara}_2(\mathcal{C})$.

Proposition 5.4.1. If the copy-discard category $\mathcal{C}$ is symmetric monoidal, then $\mathbf{Copara}_2(\mathcal{C})$ inherits a monoidal structure $(\otimes, I)$, with the same unit object I as in $\mathcal{C}$. On 1-cells $f : A \xrightarrow[M]{} B$ and $f' : A' \xrightarrow[M']{} B'$, the tensor $f \otimes f' : A \otimes A' \xrightarrow[M \otimes M']{} B \otimes B'$ is defined by
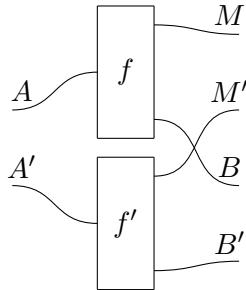
On 2-cells $\varphi : f \Rightarrow g$ and $\varphi' : f' \Rightarrow g'$, the tensor $\varphi \otimes \varphi' : (f \otimes f') \Rightarrow (g \otimes g')$ is given by the

string diagram
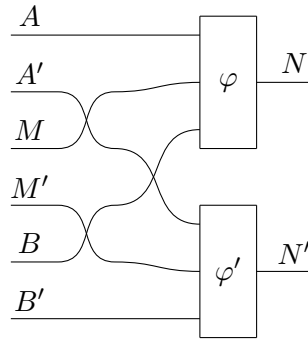
Proof. To establish that $(\mathbf{Copara}_2(\mathcal{C}), \otimes, I)$ is a monoidal bicategory, we need to show that $\otimes$ is a pseudofunctor $\mathbf{Copara}_2(\mathcal{C}) \times \mathbf{Copara}_2(\mathcal{C}) \to \mathbf{Copara}_2(\mathcal{C})$ and that I induces a pseudofunctor $\mathbf{1} \to \mathbf{Copara}_2(\mathcal{C})$, such that the pair of pseudofunctors satisfies the relevant coherence data. We will omit the coherence data, and only sketch that the pseudofunctor $\otimes$ is well defined, leaving a full proof for later work. (In the sequel here, we will not make very much use of this tensor.)

First, we confirm that $\otimes$ is locally functorial, meaning that our definition gives a functor on each pair of hom categories. We begin by noting that $\otimes$ is well-defined on 2-cells, that $\varphi \otimes \varphi'$ satisfies that change of coparameter axiom for $f \otimes f'$; this is immediate from instantiating the axiom's string diagram. Next, we note that $\otimes$ preserves identity 2-cells; again, this is immediate upon substituting identities into the defining diagram. We therefore turn to the preservation of composites, which requires that $(\gamma \odot \varphi) \otimes (\gamma' \odot \varphi') = (\gamma \otimes \gamma') \odot (\varphi \otimes \varphi')$, and which translates to the following graphical equation:
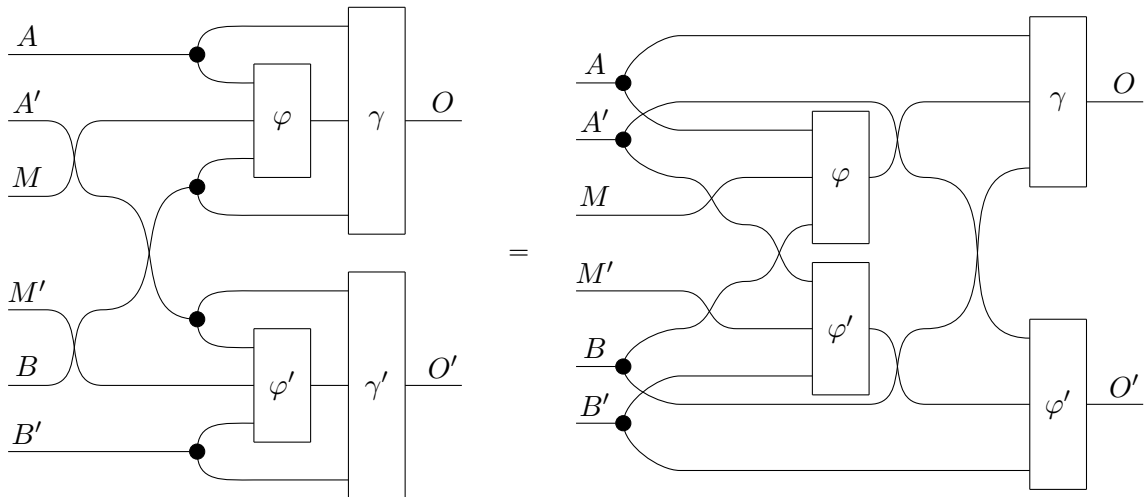
It is easy to see that this equation is satisfied: use the naturality of the symmetry of $(\mathcal{C}, \otimes, I)$. This establishes that $\otimes$ is locally functorial.

Next, we confirm that $\otimes$ is horizontally (pseudo) functorial. First, we note that $\mathsf{id}_f \otimes \mathsf{id}_{f'} = \mathsf{id}_{f \otimes f'}$ by the naturality of the symmetry of $(\mathcal{C}, \otimes, I)$. Second, we exhibit a multiplication natural

isomorphism, witnessing pseudofunctoriality, with components $\mu_{g,g',f,f'} : (g \otimes g') \circ (f \otimes f') \Rightarrow$
$(g \circ f) \otimes (g' \circ f')$ for all composable pairs of 1-cells g, f and g', f'. Let these 1-cells be such that
$(g \otimes g') \circ (f \otimes f')$ has the underlying depiction
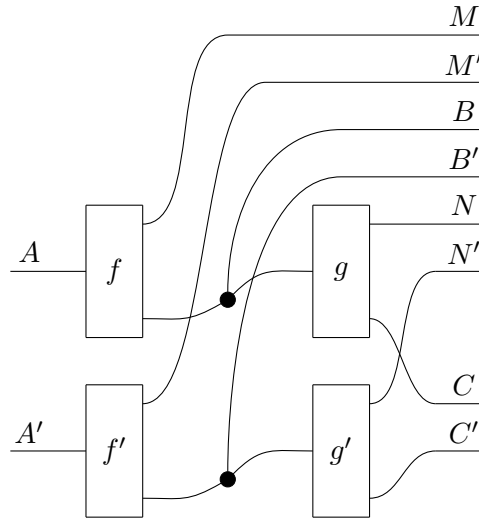
and so $(g \circ f) \otimes (g' \circ f')$ has the depiction

.
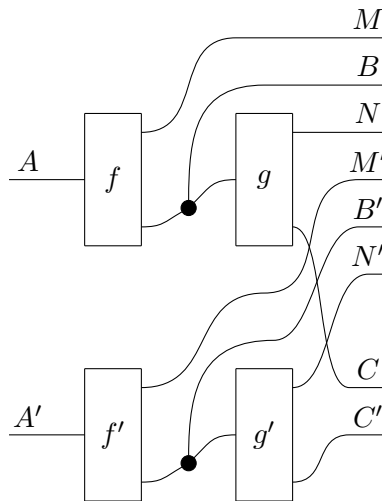
It is then easy to see that defining $\mu_{g,g',f,f'}$ and its inverse $\mu^{-1}_{g,g',f,f'}$ as the 2-cells with the following

respective underlying depictions gives us the desired isomorphism:

$$
\begin{array}{ccc}
\begin{array}{cc}
A & \\
A' & \\
M & M \\
M' & B \\
B & N \\
B' & M' \\
N & B' \\
N' & N' \\
C & \\
C' &
\end{array}
& \text{and} &
\begin{array}{cc}
A & \\
A' & \\
M & M \\
B & M' \\
N & B \\
M' & B' \\
B' & N \\
N' & N' \\
C & \\
C' &
\end{array}
& .
\end{array}
$$

The naturality of this definition is a consequence of the naturality of the symmetry of $(\mathcal{C}, \otimes, I)$.

That this tensor satisfies the monoidal bicategory axioms — of associativity, unitality, and coherence — follows from the fact that the monoidal structure $(\otimes, I)$ satisfies correspondingly decategorified versions of these axioms; we leave the details to subsequent exposition. □

Following the monoidal Grothendieck recipe, establishing that $\mathbf{BayesLens}_2$ is monoidal entails establishing that Stat_2 is a monoidal indexed bicategory. But first we must define the latter concept, by categorifying Definition 4.2.19.

Definition 5.4.2. Suppose $(\mathcal{B}, \otimes, I)$ is a monoidal bicategory. We will say that $F : \mathcal{B}^{\mathrm{co\,op}} \to \mathbf{Bicat}$ is a *monoidal indexed bicategory* when it is equipped with the structure of a weak monoid object in the 3-category of indexed bicategories, indexed pseudofunctors, indexed pseudonatural transformations, and indexed modifications.

More explicitly, we will take F to be a monoidal indexed bicategory when it is equipped with

(i) an indexed pseudofunctor $\mu : F(-) \times F(=) \to F(-\otimes =)$ called the *multiplication*, *i.e.*,

    (a) a family of pseudofunctors $\mu_{X,Y} : FX \times FY \to F(X \otimes Y)$, along with

    (b) for any 1-cells $f : X \to X'$ and $g : Y \to Y'$ in $\mathcal{B}$, a pseudonatural isomorphism

$$\mu_{f,g} : \mu_{X',Y'} \circ (Ff \times Fg) \Rightarrow F(f \otimes g) \circ \mu_{X,Y};$$

(ii) a pseudofunctor $\eta : \mathbf{1} \to FI$ called the *unit*;

as well as three indexed pseudonatural isomorphisms — an associator, a left unitor, and a right unitor — which satisfy weak analogues of the coherence conditions for a monoidal indexed category [189, §3.2], up to invertible indexed modifications.

Remark 5.4.3. Because it is not our main purpose, and because the coherence data for higher-dimensional structures rapidly becomes cumbersome, the preceding definition only suggests the form of this coherence data. Unfortunately, we are not presently aware of a full explicit definition in the literature of the concept of monoidal indexed bicategory.

Using this notion, we can establish that Stat_2 is monoidal.

Theorem 5.4.4. Stat_2 is a monoidal indexed bicategory, in the explicit sense of Definition 5.4.2.

Proof sketch. We only check the explicit requirements of the preceding definition, and expect that the higher coherence data is satisfied by the fact that each of our high-dimensional structures is obtained from a well-behaved lower-dimensional one using canonical categorical machinery.

In this way, the multiplication μ is given first by the family of pseudofunctors $\mu_{X,Y} : \mathsf{Stat}_2(X) \times \mathsf{Stat}_2(Y) \to \mathsf{Stat}_2(X \otimes Y)$ which are defined on objects simply by tensor

$$\mu_{X,Y}(A, B) = A \otimes B$$

since the objects do not vary between the fibres of Stat_2, and on hom categories by the functors

$\mathsf{Stat}_2(X)(A, B) \times \mathsf{Stat}_2(Y)(A', B')$

$= \mathbf{Cat}\big(\mathsf{disc}\,\mathcal{C}(I, X), \mathbf{Copara}_2^r(\mathcal{C})(A, B)\big) \times \mathbf{Cat}\big(\mathsf{disc}\,\mathcal{C}(I, Y), \mathbf{Copara}_2^r(\mathcal{C})(A', B')\big)$

$\cong \mathbf{Cat}\big(\mathsf{disc}\,\mathcal{C}(I, X) \times \mathsf{disc}\,\mathcal{C}(I, Y), \mathbf{Copara}_2^r(\mathcal{C})(A, B) \times \mathbf{Copara}_2^r(\mathcal{C})(A', B')\big)$

$\xrightarrow{\mathbf{Cat}(\mathsf{disc}\,\mathcal{C}(I,\mathsf{proj}_X) \times \mathsf{disc}\,\mathcal{C}(I,\mathsf{proj}_Y),\otimes)} \mathbf{Cat}\big(\mathsf{disc}\,\mathcal{C}(I, X \otimes Y)^2, \mathbf{Copara}_2^r(\mathcal{C})(A \otimes A', B \otimes B')\big)$

$\xrightarrow{\mathbf{Cat}(\mathbf{Y},\mathsf{id})} \mathbf{Cat}(\mathsf{disc}\,\mathcal{C}(I, X \otimes Y), \mathbf{Copara}_2^r(C)(A \otimes A', B \otimes B'))$

$= \mathsf{Stat}_2(X \otimes Y)(A \otimes A', B \otimes B') .$

where $\mathbf{Cat}\,(\mathbf{Y}, \mathsf{id})$ indicates pre-composition with the universal (Cartesian) copying functor. For all $f : X \to X'$ and $g : Y \to Y'$ in $\mathbf{Copara}_2^l(\mathcal{C})$, the pseudonatural isomorphisms

$$\mu_{f,g} : \mu_{X',Y'} \circ \big(\mathsf{Stat}_2(f) \times \mathsf{Stat}_2(g)\big) \Rightarrow \mathsf{Stat}_2(f \otimes g) \circ \mu_{X,Y}$$

are obtained from the universal property of the product $\times$ of categories. The unit $\eta : \mathbf{1} \to \mathsf{Stat}_2(I)$ is the pseudofunctor mapping the unique object of $\mathbf{1}$ to the monoidal unit I. Associativity and unitality of this monoidal structure follow from the functoriality of the construction, given the monoidal structures on $\mathcal{C}$ and $\mathbf{Cat}$. $\qquad\square$

Just as the monoidal Grothendieck construction induces a monoidal structure on categories of lenses for monoidal pseudofunctors [189], we obtain a monoidal structure on the bicategory of copy-composite bayesian lenses.

Corollary 5.4.5. The bicategory of copy-composite Bayesian lenses $\mathbf{BayesLens}_2$ is a monoidal bicategory. The monoidal unit is the object (I, I). The tensor $\otimes$ is given on 0-cells by $(X, A) \otimes (X', A') := (X \otimes X', A \otimes A')$, and on hom-categories by

$$
\begin{aligned}
&\mathbf{BayesLens}_2\big((X, A), (Y, B)\big) \times \mathbf{BayesLens}_2\big((X, A), (Y, B)\big) \\
&= \mathbf{Copara}_2^l(\mathcal{C})(X, Y) \times \mathsf{Stat}_2(X)(B, A) \times \mathbf{Copara}_2^l(\mathcal{C})(X', Y') \times \mathsf{Stat}_2(X')(B', A') \\
&\xrightarrow{\sim} \mathbf{Copara}_2^l(\mathcal{C})(X, Y) \times \mathbf{Copara}_2^l(\mathcal{C})(X', Y') \times \mathsf{Stat}_2(X)(B, A) \times \mathsf{Stat}_2(X')(B', A') \\
&\xrightarrow{\otimes \times \mu_{X,X'}^{\mathrm{op}}} \mathbf{Copara}_2^l(\mathcal{C})(X \otimes X', Y \otimes Y') \times \mathsf{Stat}_2(X \otimes X')(B \otimes B', A \otimes A') \\
&= \mathbf{BayesLens}_2\big((X, A) \otimes (X', A'), (Y, B) \otimes (Y', B')\big) \ .
\end{aligned}
$$

And similarly, we obtain a monoidal structure on statistical games.

Proposition 5.4.6. The bicategory of copy-composite statistical games $\mathbf{SGame}$ is a monoidal bicategory. The monoidal unit is the object (I, I). The tensor $\otimes$ is given on 0-cells as for the tensor of Bayesian lenses, and on hom-categories by

$$
\begin{aligned}
&\mathbf{SGame}\big((X, A), (Y, B)\big) \times \mathbf{SGame}\big((X', A'), (Y', B')\big) \\
&= \mathbf{BayesLens}_2\big((X, A), (Y, B)\big) \times \mathsf{Stat}(X)(B, I) \\
&\qquad \times \mathbf{BayesLens}_2\big((X', A'), (Y', B')\big) \times \mathsf{Stat}(X')(B', I) \\
&\xrightarrow{\sim} \mathbf{BayesLens}_2\big((X, A), (Y, B)\big) \times \mathbf{BayesLens}_2\big((X', A'), (Y', B')\big) \\
&\qquad \times \mathsf{Stat}(X)(B, I) \times \mathsf{Stat}(X')(B', I) \\
&\xrightarrow{\otimes \times \mu_{X,X'}} \mathbf{BayesLens}_2\big((X, A) \otimes (X', A'), (Y, B) \otimes (Y', B')\big) \times \mathsf{Stat}(X \otimes X')(B \otimes B', I \otimes I) \\
&\xrightarrow{\sim} \mathbf{SGame}\big((X, A) \otimes (X', A'), (Y, B) \otimes (Y', B')\big)
\end{aligned}
$$

where here μ indicates the multiplication of the monoidal structure on Stat (*cf.* Proposition 4.3.6).

Having obtained a monoidal structure on statistical games, we are in a position to ask for monoidal structures on inference systems and loss models:

Definition 5.4.7. A *monoidal inference system* is an inference system $(\mathcal{D}, \ell)$ for which ℓ is a lax monoidal pseudofunctor. A *monoidal loss model* is a loss model L which is a lax monoidal lax functor.

To make sense of this definition, we need a notion of lax monoidal structure appropriate for strong (pseudo-) and lax functors: a lax-functor generalization of the notion of lax monoidal functor[12] from Definition 3.1.11. Just as a lax monoidal structure on a functor is given by equipping the functor with natural transformations, a lax monoidal structure on a lax functor is given by equipping it with pseudonatural transformations. The general structure is given by Moeller and Vasilakopoulou [189, §2.2] for the case of pseudofunctors; the lax case is similar.

In the following remark, we instantiate this structure for loss models.

Remark 5.4.8. A loss model $L : \mathcal{B} \to \mathbf{SGame}$ is lax monoidal when it is equipped with strong transformations

$$
\begin{array}{ccc}
\mathcal{B} \times \mathcal{B} \xrightarrow{L \times L} \mathbf{SGame} \times \mathbf{SGame} & & 1 \xrightarrow{(I,I)} \\
\otimes_{\mathcal{B}} \downarrow \quad \xrightarrow{\lambda} \quad \downarrow \otimes_{\mathbf{G}} & \text{and} & (I,I) \downarrow \quad \xrightarrow{\lambda_0} \quad \downarrow \\
\mathcal{B} \xrightarrow{L} \mathbf{SGame} & & \mathcal{B} \xrightarrow{L} \mathbf{SGame}
\end{array}
$$

where $\otimes_{\mathcal{B}}$ and $\otimes_{\mathbf{G}}$ denote the monoidal products on $\mathcal{B} \hookrightarrow \mathbf{BayesLens}_2$ and $\mathbf{SGame}$ respectively, and when λ and λ_0 are themselves equipped with invertible modifications satisfying coherence axioms, as in Moeller and Vasilakopoulou [189, §2.2].

Note that, because L must be a (lax) section of the 2-fibration $\pi_{\mathsf{Loss}}|_{\mathcal{B}} : \mathbf{SGame}|_{\mathcal{B}} \to \mathcal{B}$, the unitor λ_0 is forced to be trivial, picking out the identity on the monoidal unit (I, I). Likewise, the laxator $\lambda : L(-) \otimes L(=) \Rightarrow L(- \otimes =)$ must have 1-cell components which are identities:

$$
L(X, A) \otimes L(X', A) = (X, A) \otimes (X', A') = (X \otimes X', A \otimes A') = L\big((X, A) \otimes L(X', A)\big)
$$

The interesting structure is therefore entirely in the 2-cells. We follow the convention of [145, Def. 4.2.1] that a strong transformation is a lax transformation with invertible 2-cell components. Supposing that $(c, c') : (X, A) \nrightarrow (Y, B)$ and $(d, d') : (X', A') \nrightarrow (Y', B')$ are 1-cells in $\mathcal{B}$, the corresponding 2-cell component of λ has the form $\lambda_{c,d} : L\big((c, c') \otimes (d, d')\big) \Rightarrow L(c, c') \otimes L(d, d')$, hence filling the following square in $\mathbf{SGame}$:

$$
\begin{array}{ccc}
(X, A) \otimes (X', A') & \xrightarrow{L(c,c') \otimes L(d,d')} & (Y, B) \otimes (Y', B') \\
\| & \xRightarrow{\lambda_{c,d}} & \| \\
(X, A) \otimes (X', A') & \xrightarrow{L((c,c') \otimes (d,d'))} & (Y, B) \otimes (Y', B')
\end{array}
$$

Intuitively, these 2-cells witness the failure of the tensor $L(c, c') \otimes L(d, d')$ of the parts to account for correlations that may be evident to the "whole system" $L\big((c, c') \otimes (d, d')\big)$.

---

[12]Note that, although lax functors themselves generalize lax monoidal functors (as bicategories generalize monoidal categories), *lax monoidal lax functors* are different again, adding another dimension (as monoidal functors add a dimension to functors): a lax monoidal lax functor is equivalently a homomorphism of one-object tricategories.

Just as there is a notion of monoidal natural transformation accompanying the notion of monoidal functor (recall Definition 3.1.13), there is a notion of monoidal icon between lax monoidal lax functors[13], from which we obtain a symmetric monoidal category of monoidal loss models.

Proposition 5.4.9. Monoidal loss models and monoidal icons form a subcategory $\mathsf{MonLoss}(\mathcal{B})$ of $\mathsf{Loss}(\mathcal{B})$, and the symmetric monoidal structure $(+, 0)$ on the latter restricts to the former.

5.4.1. Examples

In this section, we present the monoidal structure on the loss models considered above. Because loss models L are (lax) sections, following Remark 5.4.8, this monoidal structure is given in each case by a lax natural family of 2-cells $\lambda_{c,d} : L\big((c, c') \otimes (d, d')\big) \Rightarrow L(c, c') \otimes L(d, d')$, for each pair of lenses $(c, c') : (X, A) \nrightarrow (Y, B)$ and $(d, d') : (X', A') \nrightarrow (Y', B')$. Such a 2-cell $\lambda_{c,d}$ is itself given by a loss function of type $B \otimes B' \xrightarrow{X \otimes X'} I$ satisfying the equation $L\big((c, c') \otimes (d, d')\big) = L(c, c') \otimes L(d, d') + \lambda_{c,d}$; we can think of it as measuring the difference between the joint game $L\big((c, c') \otimes (d, d')\big)$ and the "mean field" games $L(c, c')$ and $L(d, d')$ taken together.

Following Johnson and Yau [145, Eq. 4.2.3], lax naturality requires that λ satisfy the following equation of 2-cells, where K denotes the laxator (with respect to horizontal composition $\diamond$) with components $K(e, c) : Le \diamond Lc \Rightarrow L(e \phi c)$:

[13]The notion of monoidal icon can be obtained by weakening the notion of monoidal pseudonatural transformation given by Moeller and Vasilakopoulou [189, §2.2].

Since vertical composition in **SGame** is given on losses by $+$, we can write this equation as

$$\lambda(e \circ c, f \circ d) + K(e \otimes f, c \otimes d)$$
$$= \lambda(e, f) \diamond \lambda(c, d) + K(e, c) \otimes K(f, d)$$
$$= \lambda(e, f)_{c \otimes d} + \lambda(c, d) \circ (e' \otimes f')_{c \otimes d} + K(e, c) \otimes K(f, d) \ . \tag{5.10}$$

In each of the examples below, therefore, we establish the definition of the laxator λ and check that it satisfies equation 5.10.

We will often use the notation $(-)_X$ to denote projection onto a factor X of a monoidal product.

5.4.1.1. Relative entropy

Proposition 5.4.10. The loss model KL of Proposition 5.3.22 is lax monoidal. Supposing that $(c, c') : (X, X) \nrightarrow (Y, Y)$ and $(d, d') : (X', X') \nrightarrow (Y', Y')$ are lenses in $\mathcal{B}$, the corresponding component $\lambda^{\mathsf{KL}}(c, d)$ of the laxator is given, for $\omega : I \nrightarrow X \otimes X'$ and $(y, y') : Y \otimes Y'$, by

$$\lambda^{\mathsf{KL}}(c, d)_\omega(y, y') := \mathop{\mathbb{E}}_{\substack{(x, x', m, m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y, y')}} \left[\log \frac{p_{\omega_X \otimes \omega_{X'}}(x, x')}{p_\omega(x, x')} \right] + \log \frac{p_{(c \otimes d)^{\dagger} \bullet \omega}(y, y')}{p_{(c \otimes d)^{\dagger} \bullet (\omega_X \otimes \omega_{X'})}(y, y')} \ .$$

(Note that the first term has the form of a "posterior mutual information" and the second a log-likelihood ratio.)

Proof. We have

$$\big(\mathsf{KL}(c) \otimes \mathsf{KL}(d)\big)_\omega(y, y')$$
$$= \mathop{\mathbb{E}}_{(x, m) \sim c'_{\omega_X}(y)} \left[\log p_{c'_{\omega_X}}(x, m|y) - \log p_{c^{\dagger}_{\omega_X}}(x, m|y) \right]$$
$$\quad + \mathop{\mathbb{E}}_{(x', m') \sim d'_{\omega_{X'}}(y')} \left[\log p_{d'_{\omega_{X'}}}(x', m'|y') - \log p_{d^{\dagger}_{\omega_{X'}}}(x', m'|y') \right]$$
$$= \mathop{\mathbb{E}}_{\substack{(x, x', m, m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y, y')}} \left[\log p_{c'_{\omega_X} \otimes d'_{\omega_{X'}}}(x, x', m, m'|y, y') - \log p_{c^{\dagger}_{\omega_X} \otimes d^{\dagger}_{\omega_{X'}}}(x, x', m, m'|y, y') \right]$$

and

$$\big(\mathsf{KL}(c \otimes d)_\omega(y, y')$$
$$= \mathop{\mathbb{E}}_{\substack{(x, x', m, m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y, y')}} \left[\log p_{c'_{\omega_X} \otimes d'_{\omega_{X'}}}(x, x', m, m'|y, y') - \log p_{(c \otimes d)^{\dagger}_\omega}(x, x', m, m'|y, y') \right] \ .$$

Using Bayes' rule, we can rewrite the exact inversions in these expressions, obtaining

$$
\begin{aligned}
&\big(\mathsf{KL}(c) \otimes \mathsf{KL}(d)\big)_\omega(y, y') \\
&= \underset{\substack{(x,x',m,m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y,y')}}{\mathbb{E}} \Big[\log p_{c'_{\omega_X} \otimes d'_{\omega_{X'}}}(x, x', m, m'|y, y') - \log p_c(y, m|x) - \log p_d(y', m'|x') \\
&\qquad\qquad - \log p_{\omega_X}(x) - \log p_{\omega_{X'}}(x') + \log p_{c^{\bar{\dagger}} \bullet \omega_X}(y) + \log p_{d^{\bar{\dagger}} \bullet \omega_{X'}}(y') \Big]
\end{aligned}
$$

and

$$
\begin{aligned}
&\big(\mathsf{KL}(c \otimes d)_\omega(y, y') \\
&= \underset{\substack{(x,x',m,m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y,y')}}{\mathbb{E}} \Big[\log p_{c'_{\omega_X} \otimes d'_{\omega_{X'}}}(x, x', m, m'|y, y') - \log p_c(y, m|x) - \log p_d(y', m'|x') \\
&\qquad\qquad - \log p_\omega(x, x') + \log p_{(c \otimes d)^{\bar{\dagger}} \bullet \omega}(y, y') \Big] .
\end{aligned}
$$

We define $\lambda^{\mathsf{KL}}(c, d)_\omega(y, y')$ as the difference from $\big(\mathsf{KL}(c \otimes d)_\omega(y, y')$ to $\big(\mathsf{KL}(c) \otimes \mathsf{KL}(d)\big)_\omega(y, y')$, and so, with a little rearranging, we obtain the expression above:

$$
\begin{aligned}
\lambda^{\mathsf{KL}}(c, d)_\omega(y, y') &:= \big(\mathsf{KL}(c \otimes d)_\omega(y, y') - \big(\mathsf{KL}(c) \otimes \mathsf{KL}(d)\big)_\omega(y, y') \\
&= \underset{\substack{(x,x',m,m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y,y')}}{\mathbb{E}} \left[\log \frac{p_{\omega_X \otimes \omega_{X'}}(x, x')}{p_\omega(x, x')} \right] + \log \frac{p_{(c \otimes d)^{\bar{\dagger}} \bullet \omega}(y, y')}{p_{(c \otimes d)^{\bar{\dagger}} \bullet (\omega_X \otimes \omega_{X'})}(y, y')} .
\end{aligned}
$$

Next, we need to validate lax naturality. Since KL is strict on losses, we need only check that

$$
\lambda^{\mathsf{KL}}(e \mathbin{\phi} c, f \mathbin{\phi} d) = \lambda^{\mathsf{KL}}(e, f)_{c \otimes d} + \lambda^{\mathsf{KL}}(c, d) \circ (e' \otimes f')_{c \otimes d} .
$$

By definition, we have

$$
\begin{aligned}
&\big(\lambda^{\mathsf{KL}}(e, f)_{c \otimes d}\big)_\omega(z, z') \\
&= \underset{\substack{(y,y',n,n') \sim \\ (e'_c \otimes f'_d)_\omega(z,z')}}{\mathbb{E}} \left[\log \frac{p_{(c \otimes d)^{\bar{\dagger}} \bullet (\omega_X \otimes \omega_{X'})}(y, y')}{p_{(c \otimes d)^{\bar{\dagger}} \bullet \omega}(y, y')} \right] + \log \frac{p_{(e \otimes f)^{\bar{\dagger}} \bullet (c \otimes d)^{\bar{\dagger}} \bullet \omega}(z, z')}{p_{(e \otimes f)^{\bar{\dagger}} \bullet (c \otimes d)^{\bar{\dagger}} \bullet (\omega_X \otimes \omega_{X'})}(z, z')}
\end{aligned}
$$

and

$$
\begin{aligned}
&\big(\lambda^{\mathsf{KL}}(c, d) \circ (e' \otimes f')_{c \otimes d}\big)_\omega(z, z') \\
&= \underset{\substack{(y,y',n,n') \sim \\ (e'_c \otimes f'_d)_\omega(z,z')}}{\mathbb{E}} \left[\underset{\substack{(x,x',m,m') \sim \\ (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y,y')}}{\mathbb{E}} \left[\log \frac{p_{\omega_X \otimes \omega_{X'}}(x, x')}{p_\omega(x, x')} \right] + \log \frac{p_{(c \otimes d)^{\bar{\dagger}} \bullet \omega}(y, y')}{p_{(c \otimes d)^{\bar{\dagger}} \bullet (\omega_X \otimes \omega_{X'})}(y, y')} \right] .
\end{aligned}
$$

And so we also have

$$
\lambda^{\mathsf{KL}}(e \mathbin{\phi} c, f \mathbin{\phi} d)_\omega(z, z')
$$

$$
= \underset{\substack{(x,x',m,m') \sim \\ \left((c' \circ e'_c) \otimes (d' \circ f'_d)\right)_\omega(z,z')}}{\mathbb{E}} \left[\log \frac{p_{\omega_X \otimes \omega_{X'}}(x, x')}{p_\omega(x, x')} \right] + \log \frac{p_{(e \otimes f)^{\dagger} \bullet (c \otimes d)^{\dagger} \bullet \omega}(z, z')}{p_{(e \otimes f)^{\dagger} \bullet (c \otimes d)^{\dagger} \bullet (\omega_X \otimes \omega_{X'})}(z, z')}
$$

$$
= \left(\lambda^{\mathsf{KL}}(c, d) \circ (e' \otimes f')_{c \otimes d} \right)_\omega(z, z') + \left(\lambda^{\mathsf{KL}}(e, f)_{c \otimes d} \right)_\omega(z, z')
$$

thereby establishing the lax naturality of λ^{KL}, by the commutativity of $+$. $\qquad\square$

Remark 5.4.11. Although KL is lax monoidal, its laxness arises from the state-dependence of the inversions, and we saw in the opening of this chapter, and then more formally in Remark 5.3.23, that in its simplest form the relative entropy does not depend on the inversions; in some sense, the statistical game structure is extraneous.

In Remark 5.3.23, we saw that D_{KL} defines a strict section of a 2-fibration $\int K \xrightarrow{\pi_K} \mathcal{B}$, attaching relative entropies to parallel pairs of channels and capturing their chain rule compositionally. Since this section does not involve any inversions, we may thus wonder whether it is more than lax monoidal: and indeed it is! D_{KL} is in fact a *strong* monoidal section which is moreover strict monoidal on the losses themselves. The laxator simply maps $(c, c', D_{KL}(c, c'))$ and $(d, d', D_{KL}(d, d'))$ to $(c \otimes d, c' \otimes d', D_{KL}(c, c') + D_{KL}(d, d'))$; and indeed it is easy to verify that $D_{KL}(c \otimes d, c' \otimes d') = D_{KL}(c, c') + D_{KL}(d, d')$.

5.4.1.2. Maximum likelihood estimation

Proposition 5.4.12. The loss model MLE of Proposition 5.3.25 is lax monoidal. Supposing that $(c, c') : (X, X) \nrightarrow (Y, Y)$ and $(d, d') : (X', X') \nrightarrow (Y', Y')$ are lenses in $\mathcal{B}$, the corresponding component $\lambda^{\mathsf{MLE}}(c, d)$ of the laxator is given, for $\omega : I \nrightarrow X \otimes X'$ and $(y, y') : Y \otimes Y'$, by

$$
\lambda^{\mathsf{MLE}}(c, d)_\omega(y, y') := \log \frac{p_{(c \otimes d)^{\dagger} \bullet (\omega_X \otimes \omega_{X'})}(y, y')}{p_{(c \otimes d)^{\dagger} \bullet \omega}(y, y')} .
$$

Proof. To obtain the definition of $\lambda^{\mathsf{MLE}}(c, d)$, we consider the difference from $\mathsf{MLE}(c \otimes d)$ to $\mathsf{MLE}(c) \otimes \mathsf{MLE}(d)$:

$$
\lambda^{\mathsf{MLE}}(c, d)_\omega(y, y') := \mathsf{MLE}(c \otimes d)_\omega(y, y') - \left(\mathsf{MLE}(c) \otimes \mathsf{MLE}(d) \right)_\omega(y, y')
$$

$$
= -\log p_{(c \otimes d)^{\dagger} \bullet \omega}(y, y') + \log p_{c^{\dagger} \bullet \omega_X}(y) - \log p_{d^{\dagger} \bullet \omega_{X'}}(y')
$$

$$
= \log \frac{p_{(c \otimes d)^{\dagger} \bullet (\omega_X \otimes \omega_{X'})}(y, y')}{p_{(c \otimes d)^{\dagger} \bullet \omega}(y, y')} .
$$

To demonstrate lax naturality, recall that MLE is a lax section, so we need to consider the corresponding $\diamond$-laxator. From Proposition 5.3.25, the laxator $K^{\mathsf{MLE}}(e, c) : \mathsf{MLE}(e) \diamond \mathsf{MLE}(c) \Rightarrow \mathsf{MLE}(e \mathbin{\phi} c)$ is given by $K^{\mathsf{MLE}}(e, c) := \mathsf{MLE}(c) \circ e'_c$. Next, observe that

$$
\begin{aligned}
\lambda^{\mathsf{MLE}}(e \mathbin{\phi} c, f \mathbin{\phi} d)_\omega(z, z') &= \log \frac{p_{\big((e \bullet c)^{\dagger} \otimes (f \bullet d)^{\dagger}\big) \bullet (\omega_X \otimes \omega_{X'})}(z, z')}{p_{\big((e \bullet c)^{\dagger} \otimes (f \bullet d)^{\dagger}\big) \bullet \omega}(z, z')} \\
&= \log \frac{p_{(e \otimes f)^{\dagger} \bullet (c \otimes d)^{\dagger} \bullet (\omega_X \otimes \omega_{X'})}(z, z')}{p_{(e \otimes f)^{\dagger} \bullet (c \otimes d)^{\dagger} \bullet \omega}(z, z')} \\
&= \lambda^{\mathsf{MLE}}(e, f)_{(c \otimes d) \bullet \omega}(z, z') \, .
\end{aligned}
$$

Consequently, we need to verify the equation

$$
\mathsf{MLE}(c \otimes d) \circ (e \otimes f')_{c \otimes d} = \lambda^{\mathsf{MLE}}(c, d) \circ (e' \otimes f')_{c \otimes d} + \big(\mathsf{MLE}(c) \otimes \mathsf{MLE}(d)\big) \circ (e' \otimes f')_{c \otimes d}
$$

which, by bilinearity of effects, is equivalent to verifying

$$
\mathsf{MLE}(c \otimes d) = \lambda^{\mathsf{MLE}}(c, d) + \mathsf{MLE}(c) \otimes \mathsf{MLE}(d) \, .
$$

But, since $+$ is commutative, this is satisfied by the definition of $\lambda^{\mathsf{MLE}}(c, d)$ as a 2-cell of type $\mathsf{MLE}(c \otimes d) \Rightarrow \mathsf{MLE}(c) \otimes \mathsf{MLE}(d)$. $\qquad\square$

5.4.1.3. Free energy

Since KL and MLE are both lax monoidal, it follows that so is FE.

Corollary 5.4.13. The loss model FE of Definition 5.3.26 is lax monoidal. Supposing that $(c, c') : (X, X) \rightarrowtail (Y, Y)$ and $(d, d') : (X', X') \rightarrowtail (Y', Y')$ are lenses in $\mathcal{B}$, the corresponding component $\lambda^{\mathsf{FE}}(c, d)$ of the laxator is given, for $\omega : I \rightarrowtail X \otimes X'$ and $(y, y') : Y \otimes Y'$, by

$$
\lambda^{\mathsf{FE}}(c, d)_\omega(y, y') := \mathop{\mathbb{E}}_{(x, x') \sim (c'_{\omega_X} \otimes d'_{\omega_{X'}})(y, y')} \left[\log \frac{p_{\omega_X \otimes \omega_{X'}}(x, x')}{p_\omega(x, x')} \right] \, .
$$

Proof. FE is defined as KL $+$ MLE, and hence λ^{FE} is obtained as $\lambda^{\mathsf{KL}} + \lambda^{\mathsf{MLE}}$. Since $+$ is functorial, it preserves lax naturality, and so λ^{FE} is also lax natural. λ^{FE} is thus a strong transformation $\mathsf{FE}(-) \otimes \mathsf{FE}(=) \Rightarrow \mathsf{FE}(- \otimes =)$, and hence FE is lax monoidal by Remark 5.4.8. $\qquad\square$

5.4.1.4. Laplacian free energy

In order to demonstrate that the lax monoidal structure on FE is not destroyed by the Laplace approximation, we prove explicitly that LFE is also lax monoidal.

Proposition 5.4.14. The loss model LFE of Propositions 5.3.31 and 5.3.33 is lax monoidal. Supposing that $(c, c') : (X, X) \nrightarrow (Y, Y)$ and $(d, d') : (X', X') \nrightarrow (Y', Y')$ are lenses in $\mathcal{B}$, the corresponding component $\lambda^{\mathsf{LFE}}(c, d)$ of the laxator is given, for $\omega : I \nrightarrow X \otimes X'$ and $(y, y') : Y \otimes Y'$, by

$$\lambda^{\mathsf{LFE}}(c, d)_\omega(y, y') := \log \frac{p_{\omega_X \otimes \omega_{X'}}(\mu_{(c \otimes d)'_\omega}(y, y')_{XX'})}{p_\omega(\mu_{(c \otimes d)'_\omega}(y, y')_{XX'})}$$

where $\mu_{(c \otimes d)'_\omega}(y, y')_{XX'}$ is the $(X \otimes X')$-mean of the Gaussian distribution $(c'_{\omega_X} \otimes d'_{\omega_{X'}})(y, y')$.

Proof. We have

$$\mathsf{LFE}(c \otimes d)_\omega(y, y')$$

$$= -\log p_{c \otimes d}(\mu_{(c' \otimes d')_\omega}(y, y'), y, y') - \log p_\omega(\mu_{(c' \otimes d')_\omega}(y, y')_{XX'})$$

$$\qquad - S_{XX'MM'}\left[(c' \otimes d')_\omega(y, y')\right]$$

$$= -\log p_c(\mu_{c'_{\omega_X}}(y), y) - \log p_d(\mu_{d'_{\omega_{X'}}}(y'), y') - p_\omega(\mu_{(c' \otimes d')_\omega}(y, y')_{XX'})$$

$$\qquad - S_{XM}\left[c'_{\omega_X}(y)\right] - S_{X'M'}\left[d'_{\omega_{X'}}(y')\right]$$

and

$$\left(\mathsf{LFE}(c) \otimes \mathsf{LFE}(d)\right)_\omega(y, y')$$

$$= \mathsf{LFE}(c)_{\omega_X}(y) + \mathsf{LFE}(d)_{\omega_{X'}}(y')$$

$$= -\log p_c(\mu_{c'_{\omega_X}}(y), y) - p_{\omega_X}(\mu_{c_{\omega_X}}(y)_X) - S_{XM}\left[c'_{\omega_X}(y)\right]$$

$$\qquad - \log p_d(\mu_{d'_{\omega_{X'}}}(y'), y') - p_{\omega_{X'}}(\mu_{d'_{\omega_{X'}}}(y')_{X'}) - S_{X'M'}\left[d'_{\omega_{X'}}(y')\right]$$

so that

$$\lambda^{\mathsf{LFE}}(c, d)_\omega(y, y') = \mathsf{LFE}(c \otimes d)_\omega(y, y') - \left(\mathsf{LFE}(c) \otimes \mathsf{LFE}(d)\right)_\omega(y, y')$$

$$= \log \frac{p_{\omega_X \otimes \omega_{X'}}(\mu_{(c \otimes d)'_\omega}(y, y')_{XX'})}{p_\omega(\mu_{(c \otimes d)'_\omega}(y, y')_{XX'})}$$

as given above.

We need to verify lax naturality, which means checking the equation

$$\lambda^{\mathsf{LFE}}(e \diamond c, f \diamond d) + \kappa(e \otimes f, c \otimes d) = \lambda^{\mathsf{LFE}}(e, f)_{c \otimes d} + \lambda^{\mathsf{LFE}}(c, d) \circ (e' \otimes f')_{c \otimes d} + \kappa(e, c) \otimes \kappa(f, d)$$

where κ is the $\diamond$-laxator with components $\kappa(e, c) : \mathsf{LFE}(e) \diamond \mathsf{LFE}(c) \Rightarrow \mathsf{LFE}(e \diamond c)$ given by

$$\kappa(e, c)_\pi(z) = \frac{1}{2} \operatorname{tr}\left[\left(\partial_y^2 E^\mu_{(c, \pi)}\right)\left(\mu_{e'_{c \bullet \pi}}(z)_Y\right) \Sigma_{e'_{c \bullet \pi}}(z)_{YY}\right] - \log p_{c^{\bar{\dagger}} \bullet \pi}\left(\mu_{e'_{c \bullet \pi}}(z)_Y\right).$$

(see Proposition 5.3.33). We have

$$
\begin{aligned}
\lambda^{\mathsf{LFE}}(e \Phi c, f \Phi d) &= \log \frac{p_{\omega_X \otimes \omega_{X'}}(\mu_{(c \otimes d)'_{\omega}}(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'})_{XX'})}{p_{\omega}(\mu_{(c \otimes d)'_{\omega}}(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'})_{XX'})} \\
&= \lambda^{\mathsf{LFE}}(c, d)_{\omega}(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'})_{XX'})
\end{aligned}
$$

and, by the Laplace approximation,

$$
\begin{aligned}
&\big(\lambda^{\mathsf{LFE}}(c, d) \circ (e' \otimes f')_{c \otimes d}\big)_{\omega}(z, z') \\
&= \underset{\substack{(y, y', n, n') \sim \\ (e'_c \otimes f'_d)_{\omega}(z, z')}}{\mathbb{E}} \left[\lambda^{\mathsf{LFE}}(c, d)_{\omega}(y, y') \right] \\
&\approx \lambda^{\mathsf{LFE}}(c, d)_{\omega}(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'}) \\
&\quad + \frac{1}{2} \operatorname{tr}\left[\left(\partial^2_{(y, y')} \lambda^{\mathsf{LFE}}(c, d)_{\omega} \right) \left(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'} \right) \Sigma_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{(YY')(YY')} \right] .
\end{aligned}
$$

We also have

$$
\begin{aligned}
&\big(\kappa(e, c) \otimes \kappa(f, d)\big)_{\omega}(z, z') \\
&= \kappa(e, c)_{\omega_X}(z) + \kappa(f, d)_{\omega_{X'}}(z') \\
&= \frac{1}{2} \operatorname{tr}\left[\left(\partial^2_y E^{\mu}_{(c, \omega_X)} \right) \left(\mu_{e'_c \bullet \omega_X}(z)_Y \right) \Sigma_{e'_c \bullet \omega_X}(z)_{YY} \right] - \log p_{c^{\bar{\top}} \bullet \omega_X}\big(\mu_{e'_c \bullet \omega_X}(z)_Y \big) \\
&\quad + \frac{1}{2} \operatorname{tr}\left[\left(\partial^2_{y'} E^{\mu}_{(d, \omega_{X'})} \right) \left(\mu_{f'_d \bullet \omega_{X'}}(z')_{Y'} \right) \Sigma_{f'_d \bullet \omega_{X'}}(z')_{Y'Y'} \right] - \log p_{d^{\bar{\top}} \bullet \omega_{X'}}\big(\mu_{f'_d \bullet \omega_{X'}}(z')_{Y'} \big) \\
&= \frac{1}{2} \operatorname{tr}\Big[\left(\partial^2_{(y, y')} E^{\mu}_{(c \otimes d, \omega_X \otimes \omega_{X'})} \right) \left(\mu_{(e \otimes f)'_{(c \otimes d)\bullet(\omega_X \otimes \omega_{X'})}}(z, z')_{YY'} \right) \\
&\qquad\qquad \Sigma_{(e \otimes f)'_{(c \otimes d)\bullet(\omega_X \otimes \omega_{X'})}}(z, z')_{(YY')(YY')} \Big] \\
&\quad - \log p_{(c \otimes d)^{\bar{\top}} \bullet (\omega_X \otimes \omega_{X'})}\big(\mu_{(e \otimes f)'_{(c \otimes d)\bullet(\omega_X \otimes \omega_{X'})}}(z, z')_{YY'} \big) .
\end{aligned}
$$

The left-hand side of the lax naturality equation is therefore given by

$$
\begin{aligned}
&\big(\lambda^{\mathsf{LFE}}(e \Phi c, f \Phi d) + \kappa(e \otimes f, c \otimes d)\big)_{\omega}(z, z') \\
&= \lambda^{\mathsf{LFE}}(c, d)_{\omega}(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'}) \\
&\quad + \frac{1}{2} \operatorname{tr}\left[\left(\partial^2_{(y, y')} E^{\mu}_{(c \otimes d, \omega)} \right) \left(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'} \right) \Sigma_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{(YY')(YY')} \right] \\
&\quad - \log p_{(c \otimes d)\bullet\omega}\big(\mu_{(e \otimes f)'_{(c \otimes d)\bullet\omega}}(z, z')_{YY'} \big)
\end{aligned}
$$

while the right-hand side is given by

$$
\left(\lambda^{\mathsf{LFE}}(e,f)_{c\otimes d} + \lambda^{\mathsf{LFE}}(c,d) \circ (e'\otimes f')_{c\otimes d} + \kappa(e,c)\otimes\kappa(f,d) \right)_\omega (z,z')
$$

$$
= \log \frac{p_{(c\otimes d)^{\bar\top}\bullet(\omega_X\otimes\omega_{X'})}\big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\big)}{p_{(c\otimes d)^{\bar\top}\bullet\omega}\big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\big)}
$$

$$
+ \lambda^{\mathsf{LFE}}(c,d)_\omega\big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\big)
$$

$$
+ \frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}\lambda^{\mathsf{LFE}}(c,d)_\omega\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\Big)\,\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{(YY')(YY')}\Big]
$$

$$
+ \frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}E^\mu_{(c\otimes d,\omega_X\otimes\omega_{X'})}\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{YY'}\Big)
$$

$$
\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{(YY')(YY')}\Big]
$$

$$
- \log p_{(c\otimes d)^{\bar\top}\bullet(\omega_X\otimes\omega_{X'})}\big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{YY'}\big)
$$

$$
= - \log p_{(c\otimes d)^{\bar\top}\bullet\omega}\big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\big) + \lambda^{\mathsf{LFE}}(c,d)_\omega\big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\big)
$$

$$
+ \frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}\lambda^{\mathsf{LFE}}(c,d)_\omega\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\Big)\,\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{(YY')(YY')}\Big]
$$

$$
+ \frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}E^\mu_{(c\otimes d,\omega_X\otimes\omega_{X'})}\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{YY'}\Big)
$$

$$
\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{(YY')(YY')}\Big].
$$

The difference from the left- to the right-hand side is thus

$$
\frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}E^\mu_{(c\otimes d,\omega)}\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\Big)\,\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{(YY')(YY')}\Big]
$$

$$
- \frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}E^\mu_{(c\otimes d,\omega_X\otimes\omega_{X'})}\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{YY'}\Big)
$$

$$
\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}(z,z')_{(YY')(YY')}\Big]
$$

$$
- \frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}\lambda^{\mathsf{LFE}}(c,d)_\omega\Big)\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\Big)\,\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{(YY')(YY')}\Big].
$$

Now, by definition $\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet\omega}} = \Sigma_{(e\otimes f)'_{(c\otimes d)\bullet(\omega_X\otimes\omega_{X'})}}$, and so by the linearity of the trace and of derivation, this difference simplifies to

$$
\frac{1}{2}\operatorname{tr}\Big[\Big(\partial^2_{(y,y')}\Big(E^\mu_{(c\otimes d,\omega)} - E^\mu_{(c\otimes d,\omega_X\otimes\omega_{X'})} - \lambda^{\mathsf{LFE}}(c,d)_\omega\Big)\Big)
$$

$$
\Big(\mu_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{YY'}\Big)\,\Sigma_{(e\otimes f)'_{(c\otimes d)\bullet\omega}}(z,z')_{(YY')(YY')}\Big].
$$

Recall from the proof of Proposition 5.3.33 that $E^\mu_{(c,\pi)}(y) := E_{(c,\pi)}\big(\mu_{c'_\pi}(y), y\big)$, and hence

$$
\begin{aligned}
&\big(E^\mu_{(c\otimes d,\omega)} - E^\mu_{(c\otimes d,\omega_X\otimes\omega_{X'})}\big)(y,y') \\
&= \big(E_{(c\otimes d,\omega)} - E_{(c\otimes d,\omega_X\otimes\omega_{X'})}\big)\big(\mu_{(c\otimes d)'_\omega}(y,y'), y, y'\big) \\
&= -\log p_\omega(\mu_{(c\otimes d)'_\omega}(y,y')_{XX'}) + \log p_{\omega_X\otimes\omega_{X'}}(\mu_{(c\otimes d)'_\omega}(y,y')_{XX'}) \\
&= \log \frac{p_{\omega_X\otimes\omega_{X'}}(\mu_{(c\otimes d)'_\omega}(y,y')_{XX'})}{p_\omega(\mu_{(c\otimes d)'_\omega}(y,y')_{XX'})} \\
&= \lambda^{\mathsf{LFE}}(c,d)_\omega(y,y')
\end{aligned}
$$

so that $E^\mu_{(c\otimes d,\omega)} - E^\mu_{(c\otimes d,\omega_X\otimes\omega_{X'})} - \lambda^{\mathsf{LFE}}(c,d)_\omega = 0$. This establishes that λ^{LFE} is lax natural. $\qquad\square$

5.5. Discussion

Having established the basic structure of statistical games and a handful of examples, there is much more to be done, and so in this section we discuss a number of seemingly fruitful avenues of future research.

An important such avenue is the link between this structure and the similar structure of diegetic open (economic) games [51], a recent reformulation of compositional game theory [119], which can also be understood as a constituting a fibration over lenses. Accordingly, the close connection between game theory and reinforcement learning [25, 128] suggests that algorithms for approximate inference (such as expectation-maximization) and reinforcement learning (such as dynamic programming) are more than superficially similar. More broadly, we expect all three of active inference, game theory, and reinforcement learning to fit into the general programme of categorical systems theory [192] (with cybernetic extensions [54, 251]), and we expect that reframing these disciplines in this way will elucidate their relationships. In Chapter 7, we supply functorial dynamical semantics for approximate inference — a form of approximate inference algorithm — but we leave the expression of this in systems-theoretic terms to future work. Likewise, we leave to the future the study of the performance and convergence of algorithms built upon these compositional foundations[14].

Another avenue for further investigation concerns mathematical neatness. First, we seek an abstract characterization of copy-composition and $\mathbf{Copara}_2$: Owen Lynch has suggested to us

---

[14]It is not clear that the fixed points of jointly optimizing the factors of a composite statistical game are the same as those of the optimization of the composite. If one is only concerned with optimizing the inversions, then the lens-like composition rule tells us that we may proceed by backward induction, first optimizing the factor nearest the codomain, and then optimizing each remaining factor in turn back towards the domain. But the problem is harder if we also wish to optimize the forward channels, as the inversion nearest the codomain still depends on the forward channel nearest the domain.

that the computation by compilers of "static single-assignment form" (SSA) [152] by compilers may have a similar structure, and so we expect an abstract characterization to capture both SSA and our examples; we also hope that a more abstract approach will alleviate some of the higher-categorical complexity resulting from the weakness of copy-composition. Second, the explicit constraint defining simple coparameterized Bayesian lenses is inelegant; as indicated in Remark 5.2.21, we expect that using dependent optics [43, 50, 276] may help to encode this constraint in the type signature, at the cost of higher-powered mathematical machinery.

Finally, we seek further examples of loss models, and more abstract (and hopefully universal) characterizations of those we already have; for example, it is known that the Shannon entropy has a topological origin [40] via a "nonlinear derivation" [169], and we expect that we can follow this connection further. In following this path, we expect to make use of the duality between algebra and geometry [180, 194] (and their intersection in (quantitative) categorical logic [55, 140]), for as we have already noted, loss functions have a natural algebraic structure. We consider such investigations part of the nascent field of categorical information geometry.

6. Open dynamical systems, coalgebraically

In Chapter 3, we saw how to compose neural circuits together using an algebraic approach to connectomics. These neural circuits are dynamical systems, formalized as sets of ordinary differential equations. However, simply specifying these sets obscures the general compositional structure of dynamical systems themselves, the revelation of which supports a subtler intertwining of syntax and semantics, form and function—or, as it happens, algebra and coalgebra. In this chapter we begin by introducing categorical language for describing general dynamical systems 'behaviourally'. These systems will be 'closed' (non-interacting), and so we then explain how the language of coalgebra, and specifically polynomial coalgebras, can be used to open them up.

However, traditional coalgebraic methods are restricted to discrete-time dynamical systems, whereas we are also interested in the continuous-time systems that are commonly used in science, such as our earlier neural circuits. This motivates the development of a class of generalized polynomial coalgebras that model open systems governed by a general time monoid, and which therefore encompass systems of dynamically interacting ordinary differential equations. In order to account for stochastic dynamics, we generalize the situation still further, by redefining the category of polynomial functors so that it can be instantiated in a nondeterministic setting. This will show us how to define open Markov processes coalgebraically, and we also demonstrate related categories of open random dynamical systems.

Finally, we use the polynomial setting to package these systems into monoidal bicategories of 'hierarchical' cybernetic systems, of which some are usefully generated differentially. In the next chapter, these bicategories will provide the setting in which we cast the dynamical semantics of approximate inference.

Remark 6.0.1. The story told in this chapter is of a form similar to that of categorical systems theory [191, 192], in which systems on interfaces collect into (doubly) indexed (double) categories. That story tells a general tale, but here we are interested in a specific case: coalgebraic systems with polynomial interfaces whose time evolution is governed by an arbitrary monoid and which may have non-determinism or side effects governed by a monad. Such systems appear to sit at a

sweet spot of scientific utility; in particular, the next chapter will use them to formalize models of predictive coding. In future work, we intend to connect the two stories, expressing our generalized polynomial coalgebras in the double-categorical framework.

6.1. Categorical background on dynamics and coalgebra

In this section, we introduce the background material needed for our development of open dynamical systems as polynomial coalgebras.

6.1.1. Dynamical systems and Markov chains

We begin by recalling a 'behavioural' approach to dynamical systems popularized by Lawvere and Schnauel [164] (who give a pedagogical account). These systems are 'closed' in the sense that they do not require environmental interaction for their evolution. Later, when we consider open systems, their 'closures' (induced by interaction with an environment) will constitute dynamical systems of this form.

The evolution of dynamics is measured by time, and we will take time to be represented by an arbitrary monoid $(\mathbb{T}, +, 0)$. This allows us to consider time-evolution that is not necessarily reversible, such as governed by $\mathbb{N}$ or $\mathbb{R}_+$, as well as reversible evolution that is properly governed by groups such as $\mathbb{Z}$ or $\mathbb{R}$. With this in mind, we give a classic definition of dynamical system, as a $\mathbb{T}$-action.

Remark 6.1.1. We will work in an abstract category $\mathcal{E}$ whose objects are considered to be "state spaces"; its morphisms will determine the nature of the dynamical evolution. Therefore, for deterministic systems, we can take $\mathcal{E}$ simply to be **Set**, or alternatively some other Cartesian category or category of comonoid homomorphisms. For stochastic systems, we may take $\mathcal{E}$ to be a copy-discard category such as $\mathcal{K}\ell(\mathcal{D})$ or **sfKrn**, or some other category whose morphisms are considered to be stochastic maps. For differential systems, we will require $\mathcal{E}$ to be equipped with a tangent bundle endofunctor T; more on this in §6.3.2.

Definition 6.1.2. Let $(\mathbb{T}, +, 0)$ be a monoid, representing time. Let $X : \mathcal{E}$ be some space, called the *state space*. Then a *closed dynamical system ϑ with state space X and time* $\mathbb{T}$ is an action of $\mathbb{T}$ on X. When $\mathbb{T}$ is also an object of $\mathcal{E}$, then this amounts to a morphism $\vartheta : \mathbb{T} \times X \to X$ (or equivalently, a time-indexed family of X-endomorphisms, $\vartheta(t) : X \to X$), such that $\vartheta(0) = \mathrm{id}_X$ and $\vartheta(s + t) = \vartheta(s) \circ \vartheta(t)$. In this dynamical context, we will refer to the action axioms as the *flow conditions*, as they ensure that the dynamics can 'flow'.

Note that, in discrete time, this definition implies that a dynamical system is governed by a single *transition map.*

Proposition 6.1.3. In discrete time $\mathbb{T} = \mathbb{N}$, any dynamical system ϑ is entirely determined by its action at $1 : \mathbb{T}$. That is, letting the state space be X, we have $\vartheta(t) = \vartheta(1)^{\circ t}$ where $\vartheta(1)^{\circ t}$ means "compose $\vartheta(1) : X \to X$ with itself t times".

Proof. The proof is by induction on $t : \mathbb{T}$. We must have $\vartheta(0) = \mathrm{id}_X$ and $\vartheta(t + s) = \vartheta(t) \circ \vartheta(s)$. So for any t, we must have $\vartheta(t + 1) = \vartheta(t) \circ \vartheta(1)$. The result follows immediately; note for example that $\vartheta(2) = \vartheta(1 + 1) = \vartheta(1) \circ \vartheta(1)$. $\qquad\square$

An ordinary differential equation $\dot{x} = f(x)$ defines a vector field $x \mapsto (x, f(x))$ on its state space X, and its solutions $x(t)$ for $t : \mathbb{R}$ define in turn a closed dynamical system, as the following example sketches.

Example 6.1.4. Let T denote a tangent bundle functor $\mathcal{E} \to \mathcal{E}$ on the ambient category of spaces $\mathcal{E}$. Suppose $X : U \to \mathsf{T}U$ is a vector field on U, with a corresponding solution (integral curve) $\chi_x : \mathbb{R} \to U$ for all $x : U$; that is, $\chi'(t) = X(\chi_x(t))$ and $\chi_x(0) = x$. Then letting the point x vary, we obtain a map $\chi : \mathbb{R} \times U \to U$. This χ is a closed dynamical system with state space U and time $\mathbb{R}$.

So far, we have abstained from using much categorical language. But these closed dynamical systems have a simple categorical representation.

Proposition 6.1.5. Closed dynamical systems with state spaces in $\mathcal{E}$ and time $\mathbb{T}$ are the objects of the functor category $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathcal{E})$, where $\mathbf{B}\mathbb{T}$ is the delooping of the monoid $\mathbb{T}$. (Recall delooping from Prop. 3.1.24.) Morphisms of dynamical systems are therefore natural transformations.

Proof. The category $\mathbf{B}\mathbb{T}$ has a single object $*$ and morphisms $t : * \to *$ for each point $t : \mathbb{T}$; the identity is the monoidal unit $0 : \mathbb{T}$ and composition is given by $+$. A functor $\vartheta : \mathbf{B}\mathbb{T} \to \mathcal{E}$ therefore picks out an object $\vartheta(*) : \mathcal{E}$, and, for each $t : \mathbb{T}$, a morphism $\vartheta(t) : \vartheta(*) \to \vartheta(*)$, such that the functoriality condition is satisfied. Functoriality requires that identities map to identities and composition is preserved, so we require that $\vartheta(0) = \mathrm{id}_{\vartheta(*)}$ and that $\vartheta(s + t) = \vartheta(s) \circ \vartheta(t)$. Hence the data for a functor $\vartheta : \mathbf{B}\mathbb{T} \to \mathcal{E}$ amount to the data for a closed dynamical system in $\mathcal{E}$ with time $\mathbb{T}$, and the functoriality condition amounts precisely to the flow condition. A morphism of closed dynamical systems $f : \vartheta \to \psi$ is a map on the state spaces $f : \vartheta(*) \to \psi(*)$ that commutes

with the flow, meaning that f satisfies $f \circ \vartheta(t) = \psi(t) \circ f$ for all times $t : \mathbb{T}$; this is precisely the definition of a natural transformation $f : \vartheta \to \psi$ between the corresponding functors. $\qquad\square$

By changing the state space category $\mathcal{E}$, this simple framework can represent different kinds of dynamics. For example, by choosing $\mathcal{E}$ to be a category of stochastic channels, such as $\mathcal{K}\ell(\mathcal{D})$ or **sfKrn**, we obtain categories of closed Markov processes.

Example 6.1.6 (Closed Markov chains and Markov processes). A closed *Markov chain* is given by a stochastic transition map $X \rightarrowtail X$, typically interpreted as a Kleisli morphism $X \to \mathcal{P}X$ for some probability monad $\mathcal{P} : \mathcal{E} \to \mathcal{E}$ (*cf.* §4.1.5 on probability monads). Following the discussion above, a closed Markov chain is therefore an object in $\mathbf{Cat}\big(\mathbf{B}\mathbb{N}, \mathcal{K}\ell(\mathcal{P})\big)$. With more general time $\mathbb{T}$, one obtains closed *Markov processes*: objects in $\mathbf{Cat}\big(\mathbf{B}\mathbb{T}, \mathcal{K}\ell(\mathcal{P})\big)$. More explicitly, a closed Markov process is a time-indexed family of Markov kernels; that is, a morphism $\vartheta : \mathbb{T} \times X \to \mathcal{P}X$ such that, for all times $s, t : \mathbb{T}$, $\vartheta_{s+t} = \vartheta_s \bullet \vartheta_t$ as a morphism in $\mathcal{K}\ell(\mathcal{P})$. Note that composition $\bullet$ in $\mathcal{K}\ell(\mathcal{P})$ is typically given by the Chapman-Kolmogorov equation, so this means that

$$\vartheta_{s+t}(y|x) = \int_{x':X} \vartheta_s(y|x')\, \vartheta_t(\mathrm{d}x'|x)\,.$$

6.1.2. Coalgebra

We saw above that a closed discrete-time deterministic dynamical system is a function $X \to X$, and that a closed discrete-time Markov chain is a function $X \to \mathcal{P}X$. This suggests a general pattern for discrete-time dynamical systems, as morphisms $X \to FX$ for some endofunctor F: such a morphism is called a *coalgebra* for the endofunctor F.

Definition 6.1.7. Let $F : \mathcal{E} \to \mathcal{E}$ be an endofunctor. A *coalgebra for F*, or *F-coalgebra*, is a pair (X, c) of an object $X : \mathcal{E}$ and a morphism $c : X \to FX$.

A morphism of F-coalgebras or *coalgebra morphism* $(X, c) \to (X', c')$ is a morphism $f : X \to X'$ that commutes with the coalgebra structures; *i.e.*, that makes the following diagram commute:

$$
\begin{array}{ccc}
X & \xrightarrow{\ f\ } & X' \\
{\scriptstyle c}\big\downarrow & & \big\downarrow{\scriptstyle c'} \\
FX & \xrightarrow[Ff]{} & FX'
\end{array}
$$

F-coalgebras and their morphisms constitute a category, denoted $\mathbf{Coalg}(F)$. The identity morphism on (X, c) is simply the identity morphism $\mathrm{id}_X : X \to X$.

Remark 6.1.8. In §3.4.1, we briefly discussed the notion of *coalgebra for a comonad*, which is a coalgebra in the sense of the preceding definition that additionally satisfies axioms dual to those defining algebras for a monad (Definition 3.4.16). In our dynamical applications, the endofunctors not in general be comonads, and so it does not make sense to demand such axioms.

Remark 6.1.9. At the same time, the duality of algebra and coalgebra underlies the subtle powers of the field of coalgebraic logic, in which the algebraic structure of logical syntax is used to define constraints on or propositions about the behaviours of dynamical systems[62, 72, 138, 162, 207]. These tools are particularly useful in setting of formal verification, where it is desirable to prove that systems behave according to a specification (for instance, for safety reasons).

With the notion of F-coalgebra to hand, we immediately obtain categories of closed discrete-time deterministic systems and Markov chains:

Example 6.1.10. The category of closed discrete-time deterministic dynamical systems in $\mathcal{E}$ is the category $\mathbf{Coalg}(\mathrm{id})$ of coalgebras for the identity endofunctor $\mathrm{id}_\mathcal{E} : \mathcal{E} \to \mathcal{E}$.

Example 6.1.11. Let $\mathcal{P} : \mathcal{E} \to \mathcal{E}$ be a probability monad on $\mathcal{E}$. The category of Markov chains is the category $\mathbf{Coalg}(\mathcal{P})$ of $\mathcal{P}$-coalgebras.

Of course, polynomial functors are endofunctors $\mathbf{Set} \to \mathbf{Set}$, so they come with a notion of coalgebra, and we may ask how such objects behave.

Example 6.1.12. Suppose $p : \mathbf{Set} \to \mathbf{Set}$ is a polynomial functor. A coalgebra for p is a function $c : X \to pX$ for some set X. By Definition 3.5.2, we can write p as $\sum_{i:p(1)} y^{p[i]}$, and hence the p-coalgebra c has the form $c : X \to \sum_{i:p(1)} X^{p[i]}$. Such a function corresponds to a choice, for each $x : X$, of an element of $p(1)$ which we denote $c^o(x)$ and an associated function $c_x^u : p[c_1(x)] \to X$. We can therefore write c equivalently as a pair (c^o, c^u) where c^u is the coproduct $\sum_x c_x^u : \sum_x p[c^o(x)] \to X$. We think of p as defining the interface of the dynamical system represented by c, with $p(1)$ encoding the set of possible 'outputs' or 'configurations' of the system, each $p[i]$ the set of possible 'inputs' for the system when it is in configuration $i : p(1)$, and X as the dynamical state space. The coalgebra c can then be understood as an *open* discrete-time dynamical system: the map c^u takes a state $x : X$ and a corresponding input in $p[c^o(x)]$ and returns the next state; and the map c^o takes a state $x : X$ and returns the system's corresponding output or configuration $c^o(x)$.

A pair of functions $c^o : X \to p(1)$ and $c^u : \sum_x p[c^o(x)] \to X$ is precisely a morphism $c : Xy^X \to p$ of polynomials, and so we have established a mapping from p-coalgebras (X, c) to morphisms $Xy^X \to p$. In fact, we have a stronger result.

Proposition 6.1.13. There is an isomorphism of hom-sets $\mathbf{Poly}(Ay^B, p) \cong \mathbf{Set}(A, pB)$ natural in A, B, p, and hence adjunctions $(-)y^B \dashv (-) \circ B : \mathbf{Poly} \to \mathbf{Set}$ and $Ay^{(-)} \dashv p \circ (-) : \mathbf{Poly}^{\mathrm{op}} \to \mathbf{Set}$.

Proof sketch. In Example 6.1.12, we established a mapping $\mathbf{Set}(A, pB) \to \mathbf{Poly}(Ay^B, p)$ for the case where $A = B$; the general case is analogous. The inverse mapping follows directly from Proposition 3.5.4. Naturality in A and B follows from naturality of pre-composition; naturality in p follows from naturality of post-composition. $\qquad\square$

Polynomial coalgebras therefore constitute a type of open discrete-time dynamical systems. But what if we want open *continuous-time* dynamical systems: do these fit into the coalgebra formalism? In a different direction, what if we want *open Markov chains*? In discrete time, we should be able to consider coalgebras for composite endofunctors $p\mathcal{P}$, but what if we want to do this in general time?

Let us turn now to answering these questions.

6.2. Open dynamical systems on polynomial interfaces

In this section, we begin by incorporating dynamical systems in general time into the coalgebraic framework, before generalizing the notion of polynomial functor to incorporate 'side-effects' such as randomness. The resulting framework will allow us to define types of system of interest, such as open Markov processes, quite generally using coalgebraic methods, and in the subsequent sections we will make much use of the newly available compositionality.

6.2.1. Deterministic systems in general time

In this section, let us suppose for simplicity that the ambient category $\mathcal{E}$ is **Set**. We will begin by stating our general definition, before explaining the structures and intuitions that justify it.

Definition 6.2.1. A *deterministic open dynamical system* with interface $p : \mathbf{Poly}$, state space $S : \mathbf{Set}$ and time $\mathbb{T} : \mathbf{Set}$ is a morphism $\beta : Sy^S \to [\mathbb{T}y, p]$ of polynomials, such that, for any section $\sigma : p \to y$ of p, the induced morphism

$$Sy^S \xrightarrow{\beta} [\mathbb{T}y, p] \xrightarrow{[\mathbb{T}y, \sigma]} [\mathbb{T}y, y] \xrightarrow{\sim} y^{\mathbb{T}}$$

is a $\triangleleft$-comonoid homomorphism.

To see how such a morphism β is like an 'open' version of the closed dynamical systems of §6.1.1, note that by the tensor-hom adjunction, β can equivalently be written with the type $\mathbb{T}y \otimes Sy^S \to p$. In turn, such a morphism corresponds to a pair (β^o, β^u), where β^o is the component 'on configurations' with the type $\mathbb{T} \times S \to p(1)$, and β^u is the component 'on inputs' with the type $\sum_{t:\mathbb{T}} \sum_{s:S} p[\beta^o(t,s)] \to S$. We will call the map β^o the *output map*, as it chooses an output configuration for each state and moment in time; and we will call the map β^u the *update map*, as it takes a state $s : S$, a quantity of time $t : \mathbb{T}$, and an input in $p[\beta^o(t,s)]$, and returns a new state. We might imagine the new state as being given by evolving the system from s for time t, and the input as supplied while the system is in the configuration corresponding to (s,t).

It is, however, not sufficient to consider merely such pairs $\beta = (\beta^o, \beta^u)$ to be our open dynamical systems, for we need them to be like 'open' monoid actions: evolving for time t then for time s must be equivalent to evolving for time $t + s$, given the same inputs. It is fairly easy to prove the following proposition, whose proof we defer until after establishing the categories $\mathbf{Coalg}^\mathbb{T}(p)$, when we prove it in an alternate form as Proposition 6.2.4.

Proposition 6.2.2. Comonoid homomorphisms $Sy^S \to y^\mathbb{T}$ correspond bijectively with closed dynamical systems with state space S, in the sense given by functors $\mathbf{B}\mathbb{T} \to \mathbf{Set}$.

This establishes that seeking such a comonoid homomorphism will give us the monoid action property that we seek, and so it remains to show that a composite comonoid homomorphism of the form $[\mathbb{T}y, \sigma] \circ \beta$ is a closed dynamical system with the "right inputs". Unwinding this composite, we find that the condition that it be a comonoid homomorphism corresponds to the requirement that, for any $t : \mathbb{T}$, the *closure* $\beta^\sigma : \mathbb{T} \times S \to S$ of β by σ given by

$$\beta^\sigma(t) := S \xrightarrow{\beta^o(t)^*\sigma} \sum_{s:S} p[\beta^o(t,s)] \xrightarrow{\beta^u} S$$

constitutes a closed dynamical system on S. The idea here is that σ gives the 'context' in which we can make an open system closed, thereby formalizing the "given the same inputs" requirement above.

With this conceptual framework in mind, we are in a position to render open dynamical systems on p with time $\mathbb{T}$ into a category, which we will denote by $\mathbf{Coalg}^\mathbb{T}(p)$. Its objects will be pairs (S, β) with S and β an open dynamical on p with state space S; we will often write these pairs equivalently as triples (S, β^o, β^u), making explicit the output and update maps. Morphisms will be maps of state spaces that commute with the dynamics:

Proposition 6.2.3. Open dynamical systems over p with time $\mathbb{T}$ form a category, denoted $\mathbf{Coalg}^{\mathbb{T}}(p)$. Its morphisms are defined as follows. Let $\vartheta := (X, \vartheta^o, \vartheta^u)$ and $\psi := (Y, \psi^o, \psi^u)$ be two dynamical systems over p. A morphism $f : \vartheta \to \psi$ consists in a morphism $f : X \to Y$ such that, for any time $t : \mathbb{T}$ and section $\sigma : p(1) \to \sum\limits_{i:p(1)} p[i]$ of p, the following naturality squares commute:

$$
\begin{array}{ccc}
X & \xrightarrow{\vartheta^o(t)^*\sigma} \sum\limits_{x:X} p[\vartheta^o(t,x)] \xrightarrow{\vartheta^u(t)} & X \\
\downarrow{\scriptstyle f} & & \downarrow{\scriptstyle f} \\
Y & \xrightarrow[\psi^o(t)^*\sigma]{} \sum\limits_{y:Y} p[\psi^o(t,y)] \xrightarrow[\psi^u(t)]{} & Y
\end{array}
$$

The identity morphism id_ϑ on the dynamical system ϑ is given by the identity morphism id_X on its state space X. Composition of morphisms of dynamical systems is given by composition of the morphisms of the state spaces.

Proof. We need to check unitality and associativity of composition. This amounts to checking that the composite naturality squares commute. But this follows immediately, since the composite of two commutative diagrams along a common edge is again a commutative diagram. $\square$

We can alternatively state Proposition 6.2.2 as follows, noting that the polynomial y represents the trivial interface, exposing no configuration to any environment nor receiving any signals from it:

Proposition 6.2.4. $\mathbf{Coalg}^{\mathbb{T}}_{\mathrm{id}}(y)$ is equivalent to the classical category $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathbf{Set})$ of closed dynamical systems in $\mathbf{Set}$ with time $\mathbb{T}$.

Proof. The trivial interface y corresponds to the trivial bundle $\mathrm{id}_1 : 1 \to 1$. Therefore, a dynamical system over y consists of a choice of state space S along with a trivial output map $\vartheta^o = \bar{\top} : \mathbb{T} \times S \to 1$ and a time-indexed update map $\vartheta^u : \mathbb{T} \times S \to S$. This therefore has the form of a classical closed dynamical system, so it remains to check the monoid action. There is only one section of id_1, which is again id_1. Pulling this back along the unique map $\vartheta^o(t) : S \to 1$ gives $\vartheta^o(t)^* \mathrm{id}_1 = \mathrm{id}_S$. Therefore the requirement that, given any section σ of y, the maps $\vartheta^u \circ \vartheta^o(t)^*\sigma$ form an action means in turn that so does $\vartheta^u : \mathbb{T} \times S \to S$. Since the pullback of the unique section id_1 along the trivial output map $\vartheta^o(t) = \bar{\top} : S \to 1$ of any dynamical system in $\mathbf{Coalg}^{\mathbb{T}}_{\mathrm{id}}(y)$ is the identity of the corresponding state space id_S, a morphism $f : (\vartheta(*), \vartheta^u, \bar{\top}) \to (\psi(*), \psi^u, \bar{\top})$ in $\mathbf{Coalg}^{\mathbb{T}}_{\mathrm{id}}(y)$ amounts precisely to a map $f : \vartheta(*) \to \psi(*)$ on the state spaces in $\mathbf{Set}$ such that the

naturality condition $f \circ \vartheta^u(t) = \psi^u(t) \circ f$ of Proposition 6.1.5 is satisfied, and every morphism in $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathbf{Set})$ corresponds to a morphism in $\mathbf{Coalg}_{\mathrm{id}}^{\mathbb{T}}(y)$ in this way. $\qquad\square$

Now that we know that our concept of open dynamical system subsumes closed systems, let us consider some more examples.

Example 6.2.5. Consider a dynamical system $(S, \vartheta^o, \vartheta^u)$ with outputs but no inputs. Such a system has a 'linear' interface $p := Oy$ for some O; alternatively, we can write its interface p as the 'bundle' $\mathrm{id}_O : O \to O$. A section of this bundle must again be id_O, and so $\vartheta^o(t)^* \, \mathrm{id}_O = \mathrm{id}_S$. Once again, the update maps collect into to a closed dynamical system in $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathbf{Set})$; just now we have outputs $\vartheta^o : \mathbb{T} \times S \to p(1) = O$ exposed to the environment.

Proposition 6.2.6. When time is discrete, as with $\mathbb{T} = \mathbb{N}$, any open dynamical system $(X, \vartheta^o, \vartheta^u)$ over p is entirely determined by its components at $1 : \mathbb{T}$. That is, we have $\vartheta^o(t) = \vartheta^o(1) : X \to p(1)$ and $\vartheta^u(t) = \vartheta^u(1) : \sum_{x:X} p[\vartheta^o(x)] \to X$. A discrete-time open dynamical system is therefore a triple $(X, \vartheta^o, \vartheta^u)$, where the two maps have types $\vartheta^o : X \to p(1)$ and $\vartheta^u : \sum_{x:X} p[\vartheta^o(x)] \to X$.

Proof. Suppose σ is a section of p. We require each closure ϑ^σ to satisfy the flow conditions, that $\vartheta^\sigma(0) = \mathrm{id}_X$ and $\vartheta^\sigma(t+s) = \vartheta^\sigma(t) \circ \vartheta^\sigma(s)$. In particular, we must have $\vartheta^\sigma(t+1) = \vartheta^\sigma(t) \circ \vartheta^\sigma(1)$. By induction, this means that we must have $\vartheta^\sigma(t) = \vartheta^\sigma(1)^{\circ t}$ (compare Proposition 6.1.3). Therefore we must in general have $\vartheta^o(t) = \vartheta^o(1)$ and $\vartheta^u(t) = \vartheta^u(1)$. $\qquad\square$

Remark 6.2.7. Note that the preceding proposition means that the objects of $\mathbf{Coalg}^{\mathbb{N}}(p)$ are the objects of the traditional category $\mathbf{Coalg}(p)$ of p-coalgebras. In fact, we have more than this: $\mathbf{Coalg}^{\mathbb{N}}(p) \cong \mathbf{Coalg}(p)$; *cf.* Example 6.1.12 and Proposition 6.1.13.

Example 6.2.8. We can express 'open' vector fields in this framework. Suppose therefore that X is a differentiable manifold (and write X equally for its underlying set of points), and let $\dot{x} = f(x, a)$ and $b = g(x)$, with $f : X \times A \to \mathsf{T}X$ and $g : X \to B$. Then, as for the 'closed' vector fields of Example 6.1.4, this induces an open dynamical system $(X, \int f, g) : \mathbf{Coalg}^{\mathbb{R}}(By^A)$, where $\int f : \mathbb{R} \times X \times A \to X$ returns the (X, A)-indexed solutions of f.

Example 6.2.9. The preceding example is easily extended to the case of a general polynomial interface. Suppose similarly that $\dot{x} = f(x, a_x)$ and $b = g(x)$, now with $f : \sum_{x:X} p[g(x)] \to \mathsf{T}X$ and $g : X \to p(1)$. Then we obtain an open dynamical system $(X, \int f, g) : \mathbf{Coalg}^{\mathbb{R}}(p)$, where now $\int f : \mathbb{R} \times \sum_{x:X} p[g(x)] \to X$ is the 'update' and $g : X \to p(1)$ the 'output' map.

By letting the polynomial p vary, it is quite straightforward to extend $\mathbf{Coalg}^{\mathbb{T}}(p)$ to an opindexed category $\mathbf{Coalg}^{\mathbb{T}}$.

Proposition 6.2.10. $\mathbf{Coalg}^{\mathbb{T}}$ extends to an opindexed category $\mathbf{Coalg}^{\mathbb{T}} : \mathbf{Poly} \to \mathbf{Cat}$. On objects (polynomials), it returns the categories above. On morphisms of polynomials, we simply post-compose: given $\varphi : p \to q$ and $\beta : Sy^S \to [\mathbb{T}y, p]$, obtain $Sy^S \to [\mathbb{T}y, p] \to [\mathbb{T}y, q]$ in the obvious way.

When we introduced $\mathbf{Poly}$ in §3.5, it was as a "syntax for interacting adaptive systems", and we know that we can understand $\mathbf{Poly}$ multicategorically, as it has a monoidal structure $(\otimes, y)$ allowing us to place systems' interfaces side-by-side (and which therefore gives us a multicategory, $\mathcal{O}\mathbf{Poly}$ by Proposition 3.3.4). We motivated our development of coalgebraic dynamical systems as a compositional extension of the sets of ordinary differential equations that we used to formalize rate-coded neural circuits (Definition 3.3.10), and we have seen that linear circuit diagrams embed into $\mathbf{Poly}$ (Remark 4.2.33).

One may wonder, therefore, whether the opindexed categories $\mathbf{Coalg}^{\mathbb{T}}$ might supply the general "semantics for interacting adaptive systems" that we seek: more precisely, is $\mathbf{Coalg}^{\mathbb{T}}$ a $\mathbf{Poly}$-algebra? This question can be answered affirmatively, as $\mathbf{Coalg}^{\mathbb{T}}$ is lax monoidal: more precisely, it is a strong monoidal opindexed category.

Proposition 6.2.11. $\mathbf{Coalg}^{\mathbb{T}}$ is a monoidal opindexed category $(\mathbf{Poly}, \otimes, y) \to (\mathbf{Cat}, \times, 1)$.

Proof. We need to define a natural family of functors $\mu_{p,q} : \mathbf{Coalg}^{\mathbb{T}}(p) \times \mathbf{Coalg}^{\mathbb{T}}(q) \to \mathbf{Coalg}^{\mathbb{T}}(p \otimes q)$ constituting the laxator, and a unit $\eta : \mathbf{1} \to \mathbf{Coalg}^{\mathbb{T}}(y)$, along with associators α and left and right unitors λ and ρ satisfying the pseudomonoid axioms of Definition 4.2.19.

The unit $\eta : \mathbf{1} \to \mathbf{Coalg}^{\mathbb{T}}(y)$ is given by the trivial system $(1, !, !)$ with the trivial state space and the trivial interface: the output map is the unique map $1 \to 1$ (the identity); likewise, the update map is the unique map $1 \times 1 \to 1$. Note that $1 \times 1 \cong 1$.

The laxator $\mu_{p,q}$ is given on objects $(X, \vartheta) : \mathbf{Coalg}^{\mathbb{T}}(p)$ and $(Y, \varphi) : \mathbf{Coalg}^{\mathbb{T}}(q)$ by the $\mu_{p,q}(\vartheta, \varphi) := \big(XY, (\vartheta\varphi)\big)$ where the state space $XY = X \times Y$ and $(\vartheta\varphi)$ is the system given by the right adjunct of

$$XYy^{XY} \otimes \mathbb{T}y \xrightarrow{\sim \otimes \blacktriangledown y} Xy^X \otimes Yy^Y \otimes \mathbb{T}y \otimes \mathbb{T}y \xrightarrow{Xy^X \otimes \mathsf{swap} \otimes \mathbb{T}y} Xy^X \otimes \mathbb{T}y \otimes Yy^Y \otimes \mathbb{T}y \xrightarrow{\vartheta^\flat \otimes \varphi^\flat} p \otimes q$$

under the tensor-hom adjunction in $\mathbf{Poly}$, where $\vartheta^\flat$ and $\varphi^\flat$ are the corresponding left adjuncts of ϑ and φ, and where $\sim$ is the isomorphism $XYy^{XY} \xrightarrow{\sim} Xy^X \otimes Yy^Y$. On morphisms $f : (X, \vartheta) \to$

(X', ϑ') and $g : (Y, \varphi) \to (Y', \varphi)$, $\mu_{p,q}$ acts as $\mu_{p,q}(f, g) := f \times g$; functoriality hence follows from that of $\times$.

Next, we need to define μ on morphisms $\zeta : p \to p'$ and $\xi : q \to q'$ of polynomials, giving natural isomorphisms $\mu_{\zeta,\xi} : \mu_{p',q'} \circ \left(\mathbf{Coalg}^{\mathbb{T}}(\zeta) \times \mathbf{Coalg}^{\mathbb{T}}(\xi)\right) \Rightarrow \mathbf{Coalg}^{\mathbb{T}}(\zeta \otimes \xi) \circ \mu_{p,q}$. But it is easy to see in fact that $\mu_{p',q'} \circ \left(\mathbf{Coalg}^{\mathbb{T}}(\zeta) \times \mathbf{Coalg}^{\mathbb{T}}(\xi)\right) = \mathbf{Coalg}^{\mathbb{T}}(\zeta \otimes \xi) \circ \mu_{p,q}$, as both sides act by post-composing $\zeta \otimes \xi$.

The associator is defined componentwise on objects as

$$\alpha_{p,q,r} : \left((XY)Zy^{(XY)Z} \xrightarrow{(\vartheta \otimes \varphi) \otimes \psi} [(p \otimes q) \otimes r, \mathbb{T}y]\right) \mapsto \left(X(YZ)y^{X(YZ)} \xrightarrow{\vartheta \otimes (\varphi \otimes \psi)} [p \otimes (q \otimes r), \mathbb{T}y]\right)$$

and on morphisms as $\alpha_{p,q,r} : (f \times g) \times h \mapsto f \times (g \times h)$, implicitly using the associators of $\otimes$ on **Poly** and $\times$ on **Set**.

Likewise, the left unitor is defined by

$$\lambda_p : \left(1Xy^{1X} \xrightarrow{\mu_{y,p}(\eta, \vartheta)} [y \otimes p, \mathbb{T}y]\right) \mapsto \left(Xy^X \xrightarrow{\vartheta} [p, \mathbb{T}y]\right)$$

implicitly using the left unitors of $\otimes$ on **Poly** and $\times$ on **Set**; and the right unitor is defined dually, using the corresponding right unitors on **Poly** and **Set**.

That the associators and unitors satisfy the indexed monoidal category axioms follows from the satisfaction of the monoidal category axioms by $(\mathbf{Poly}, \otimes, y)$ and $(\mathbf{Set}, \times, 1)$. (But it is easy, though laborious, to verify this manually.) $\square$

Remark 6.2.12. We emphasize that the functor $\mathbf{Coalg}^{\mathbb{T}}$ is *lax* monoidal—the laxators are not equivalences—since not all systems over the parallel interface $p \otimes q$ factor into a system over p alongside a system over q.

With this indexed monoidal structure, we can show that, as we might hope from a general semantics for interacting dynamical systems, **Coalg** subsumes our earlier linear circuit algebra of rate-coded neural circuits.

Proposition 6.2.13. There is an inclusion ι of monoidal indexed categories as in the diagram

$$
\begin{array}{ccc}
\left(\mathbf{LinCirc}, +, (0,0)\right) & \xrightarrow{\quad R \quad} & (\mathbf{Set}, \times, 1) \\
\big\downarrow & \Big\Vert \iota & \big\downarrow \\
(\mathbf{Poly}, \otimes, y) & \xrightarrow{\mathbf{Coalg}^{\mathbb{R}}} & (\mathbf{Cat}, \times, \mathbf{1})
\end{array}
$$

where R is the algebra from Proposition 3.3.12.

Proof sketch. ι is defined by a family of functors $\iota_{(n_o, n_i)} : R(n_o, n_i) \to \mathbf{Coalg}^{\mathbb{R}}(\mathbb{R}^{n_o} y^{\mathbb{R}^{n_i}})$, where each set $R(n_o, n_i)$ is treated as the corresponding discrete category; this means that $\iota_{(n_o, n_i)}$ is trivially functorial, and needs only be defined on objects (rate-coded neural circuits). Each such circuit $(\lambda, \alpha, \beta, \gamma, W)$ defines an 'open' ordinary differential equation by Definition 3.3.10 with inputs $i : \mathbb{R}^{n_i}$. $\iota_{(n_o, n_i)}$ is then defined by taking this ordinary differential equation to a corresponding open dynamical system following Example 6.2.8, where the output space is the same as the state space $\mathbb{R}^{n_o}$ and the output map is $\mathrm{id}_{\mathbb{R}^{n_o}}$.

We then need to check that this definition of ι is natural, meaning that the following diagram commutes for each linear circuit diagram $(A, B) : (n_o, n_i) \to (m_o, m_i)$, where $\mathbf{Coalg}^{\mathbb{R}}(A, B)$ is defined by treating (A, B) as a lens and hence a morphism of monomials of the type indicated.

$$
\begin{array}{ccc}
R(n_o, n_i) & \xrightarrow{\ \ R(A,B)\ \ } & R(m_o, m_i) \\
\Big\downarrow{\scriptstyle \iota_{(n_o, n_i)}} & & \Big\downarrow{\scriptstyle \iota_{(m_o, m_i)}} \\
\mathbf{Coalg}^{\mathbb{R}}(\mathbb{R}^{n_o} y^{\mathbb{R}^{n_i}}) & \xrightarrow{\ \mathbf{Coalg}^{\mathbb{R}}(A,B)\ } & \mathbf{Coalg}^{\mathbb{R}}(\mathbb{R}^{m_o} y^{\mathbb{R}^{m_i}})
\end{array}
$$

To see that this diagram commutes, observe that we can write a rate-coded neural circuit κ as a morphism $\mathbb{R}^{n_o} y^{\mathsf{T}\mathbb{R}^{n_o}} \to \mathbb{R}^{n_o} y^{\mathbb{R}^{n_i}}$ of polynomials, where T is the tangent bundle functor; and observe that the action of $R(A, B)$ is to post-compose the lens (A, B) after κ, as in $\mathbb{R}^{n_o} y^{\mathsf{T}\mathbb{R}^{n_o}} \xrightarrow{\kappa} \mathbb{R}^{n_o} y^{\mathbb{R}^{n_i}} \xrightarrow{(A,B)} \mathbb{R}^{m_o} y^{\mathbb{R}^{m_i}}$. Now, $\iota_{(n_o, n_i)}$ acts by taking κ to a system $\mathbb{R}^{n_o} y^{\mathbb{R}^{n_o}} \to [\mathbb{T}y, \mathbb{R}^{n_o} y^{\mathbb{R}^{n_i}}]$, and $\mathbf{Coalg}^{\mathbb{R}}(A, B)$ post-composes $[\mathbb{T}y, (A, B)]$, so we obtain the system

$$
\mathbb{R}^{n_o} y^{\mathbb{R}^{n_o}} \xrightarrow{\ \iota_{(n_o, n_i)}(\kappa)\ } [\mathbb{T}y, \mathbb{R}^{n_o} y^{\mathbb{R}^{n_i}}] \xrightarrow{\ [\mathbb{T}y, (A,B)]\ } [\mathbb{T}y, \mathbb{R}^{m_o} y^{\mathbb{R}^{m_i}}] \ .
$$

This is precisely the system obtained by applying $\iota_{(m_o, m_i)}$ to $(A, B) \circ \kappa$, and hence ι is natural.

Finally, it is easy to check that ι is a monoidal natural transformation (Definition 3.1.13), which by Moeller and Vasilakopoulou [189, Proposition 3.6] entails that ι is a morphism of monoidal indexed categories. That it is additionally an inclusion follows from the evident fact that each functor $\iota_{(n_o, n_i)}$ is an embedding. $\qquad\square$

At some point during the preceding exposition, the reader may have wondered in what sense these open dynamical systems are coalgebras. To answer this, recall from Proposition 6.1.13 that a polynomial morphism $Sy^S \to q$ is equivalently a function $S \to qS$ and hence by Example 6.1.12 a q-coalgebra. Then, by setting $q = [\mathbb{T}y, p]$, we see the connection immediately: the objects of $\mathbf{Coalg}^{\mathbb{T}}(p)$ are $[\mathbb{T}y, p]$-coalgebras that satisfy the $\lhd$-comonoid condition, and the morphisms of $\mathbf{Coalg}^{\mathbb{T}}(p)$ are coalgebra morphisms.

In the following subsection, we generalize the constructions above to allow for non-deterministic ('effectful') feedback, using a generalization of the category **Poly**.

6.2.2. Polynomials with 'effectful' feedback, and open Markov processes

The category **Poly** of polynomial functors **Set** $\to$ **Set** can be considered as a category of 'deterministic' polynomial interaction; notably, morphisms of such polynomials, which we take to encode the coupling of systems' interfaces, do not explicitly incorporate any kind of randomness or uncertainty. Even if the universe is deterministic, however, the finiteness of systems and their general inability to perceive the totality of their environments make it a convenient modelling choice to suppose that systems' interactions may be uncertain; this will be useful not only in allowing for stochastic interactions between systems, but also to define stochastic dynamical systems 'internally' to a category of polynomials.

To reach the desired generalization, we begin by recalling that **Poly** is equivalent to the category of Grothendieck lenses for the self-indexing of **Set** (Example 4.2.30). We define our categories of generalized polynomials from this perspective, by considering Kleisli categories indexed by their "deterministic subcategories". This allows us to define categories of Grothendieck lenses which behave like **Poly** when restricted to the deterministic case, but also admit uncertain inputs. In order to apply the Grothendieck construction, we begin by defining an indexed category.

Definition 6.2.14. Suppose $\mathcal{E}$ is a category with all limits, and suppose $M : \mathcal{E} \to \mathcal{E}$ is a monad on $\mathcal{E}$. Define the indexed category $\mathcal{E}_M/- : \mathcal{E}^{\mathrm{op}} \to \mathbf{Cat}$ as follows. On objects $B : \mathcal{E}$, we define $\mathcal{E}_M/B$ to be the full subcategory of $\mathcal{K}\ell(M)/B$ on those objects $\iota p : E \twoheadrightarrow B$ which correspond to maps $E \xrightarrow{p} B \xrightarrow{\eta_B} MB$ in the image of ι. Now suppose $f : C \to B$ is a map in $\mathcal{E}$. We define $\mathcal{E}_M/f : \mathcal{E}_M/B \to \mathcal{E}_M/C$ as follows. The functor $\mathcal{E}_M/f$ takes objects $\iota p : E \twoheadrightarrow B$ to $\iota(f^*p) : f^*E \twoheadrightarrow C$ where f^*p is the pullback of p along f in $\mathcal{E}$, included into $\mathcal{K}\ell(M)$ by ι.

To define the action of $\mathcal{E}_M/f$ on morphisms $\alpha : (E, \iota p : E \twoheadrightarrow B) \to (F, \iota q : F \twoheadrightarrow B)$, note that since we must have $\iota q \bullet \alpha = \iota p$, we can alternatively write α as the B-dependent sum $\sum_{b:B} \alpha_b : \sum_{b:B} p[b] \to \sum_{b:B} Mq[b]$. Then we can define $(\mathcal{E}_M/f)(\alpha)$ accordingly as $(\mathcal{E}_M/f)(\alpha) := \sum_{c:C} \alpha_{f(c)} : \sum_{c:C} p[f(c)] \to \sum_{c:C} Mq[f(c)]$.

Definition 6.2.15. We define $\mathbf{Poly}_M$ to be the category of Grothendieck lenses for $\mathcal{E}_M/-$. That is, $\mathbf{Poly}_M := \int \mathcal{E}_M/-^{\mathrm{op}}$, where the opposite is again taken pointwise.

Example 6.2.16. When $\mathcal{E} = \mathbf{Set}$ and $M = \mathrm{id}_{\mathbf{Set}}$, Definition 6.2.14 recovers our earlier definition of **Poly**.

Example 6.2.17. When M is a monad on **Set**, we find that the objects of $\mathbf{Poly}_M$ are the same polynomial functors as constitute the objects of **Poly**. The morphisms $f : p \to q$ are pairs $(f_1, f^\sharp)$, where $f_1 : B \to C$ is a function in **Set** and $f^\sharp$ is a family of morphisms $q[f_1(x)] \rightarrowtail p[x]$ in $\mathcal{K}\ell(M)$, making the following diagram commute:

$$
\begin{array}{ccccc}
\sum_{x:B} Mp[x] & \xleftarrow{\;\;f^\sharp\;\;} & \sum_{b:B} q[f_1(x)] & \longrightarrow & \sum_{y:C} q[y] \\
{\scriptstyle \eta_B * p}\downarrow & & \downarrow & & \downarrow{\scriptstyle q} \\
B & =\!=\!=\!=\!= & B & \xrightarrow{\;\;f_1\;\;} & C
\end{array}
$$

Remark 6.2.18. Consequently, we can think of $\mathbf{Poly}_M$ as a dependent version of the category of M-monadic lenses, in the sense of Clarke et al. [65, §3.1.3].

Remark 6.2.19. Any monad (M, μ, η) on **Set** induces a comonad $(\bar{M}, \delta, \epsilon)$ on the category **Poly** of polynomial functors **Set** $\to$ **Set**, and $\mathbf{Poly}_M$ can be recovered as the coKleisli category of this comonad, $\mathbf{Poly}_M \cong \mathrm{co}\mathcal{K}\ell(\bar{M})$. We heard of this idea from David Spivak.

On objects (polynomial functors), $\bar{M} : \mathbf{Poly} \to \mathbf{Poly}$ acts to map $p : \sum_{i:p(1)} p[i] \to p(1)$ to $\bar{M}p : \sum_{i:p(1)} Mp[i] \to p(1)$. Given a morphism of polynomials $\varphi : p \to q$, $\bar{M}$ returns the morphism $\bar{M}(\varphi)$ whose forward component is again φ_1 and whose backward component is defined by $\bar{M}(\varphi)_i^\sharp := M(\varphi_i^\sharp)$ for each i in $p(1)$.

The counit $\epsilon : \bar{M} \Rightarrow \mathrm{id}_{\mathbf{Poly}}$ is defined for each p as the morphism $\epsilon_p : \bar{M}p \to p$ whose forward component is $\mathrm{id}_{p(1)}$ and whose backward component is given for each $i : p(1)$ by the unit $\eta_{p[i]} : p[i] \to Mp[i]$ of the monad M. Similarly, the comultiplication $\delta : \bar{M}\bar{M} \Rightarrow \bar{M}$ is defined for each p as the morphism $\delta_p : \bar{M}\bar{M}p \to \bar{M}p$ whose forward component is again the identity and whose backward components are given by the multiplication of the monad μ, *i.e.* $(\delta_p^\sharp)_i := \mu_{p[i]}$.

Finally, the coKleisli category $\mathrm{co}\mathcal{K}\ell(\bar{M})$ has the same objects as **Poly**. A morphism $p \to q$ in $\mathrm{co}\mathcal{K}\ell(\bar{M})$ is a morphism $\bar{M}p \to q$ in **Poly**. Composition in $\mathrm{co}\mathcal{K}\ell(\bar{M})$ is the usual composition in the forward direction and Kleisli composition in the backward direction.

Remark 6.2.20. Since $\mathcal{E}$ is assumed to have all limits, it must have a product structure $(\times, 1)$. When M is additionally a monoidal monad (Definition 4.1.16), then $\mathbf{Poly}_M$ acquires a tensor akin to that defined for **Poly** in Proposition 3.5.7, and which we also denote by $(\otimes, I)$: the definition only differs by substituting the structure $(\otimes, I)$ on $\mathcal{K}\ell(M)$ for the product $(\times, 1)$ on **Set**. This monoidal structure follows as before from the monoidal Grothendieck construction: $\mathcal{E}_M/-$ is lax monoidal, with laxator taking $p : \mathcal{E}_M/B$ and $q : \mathcal{E}_M/C$ to $p \otimes q : \mathcal{E}_M/(B \otimes C)$.

On the other hand, for $\mathbf{Poly}_M$ also to have an internal hom $[q, r]$ requires each fibre of $\mathcal{E}_M/-$ to be closed with respect to the monoidal structure. In cases of particular interest, $\mathcal{E}$ will be

locally Cartesian closed, and restricting $\mathcal{E}_M/-$ to the self-indexing $\mathcal{E}/-$ gives fibres which are thus Cartesian closed. In these cases, we can think of the broader fibres of $\mathcal{E}_M/-$, and thus $\mathbf{Poly}_M$ itself, as being 'deterministically' closed. This means, for the stochastic example $\mathbf{Poly}_{\mathcal{P}}$ for $\mathcal{P}$ a probability monad, we get an internal hom satisfying the adjunction $\mathbf{Poly}_{\mathcal{P}}(p \otimes q, r) \cong \mathbf{Poly}_{\mathcal{P}}(p, [q, r])$ only when the backwards components of morphisms $p \otimes q \to r$ are 'uncorrelated' between p and q.

Remark 6.2.21. For $\mathbf{Poly}_M$ to behave faithfully like the category $\mathbf{Poly}$ of polynomial functors of sets and their morphisms, we should want the substitution functors $\mathcal{E}_M/f : \mathcal{E}_M/C \to \mathcal{E}_M/B$ to have left and right adjoints (corresponding respectively to dependent sum and product). Although we do not spell it out here, it is quite straightforward to exhibit concretely the left adjoints. On the other hand, writing f^* as shorthand for $\mathcal{E}_M/f$, we can see that a right adjoint only obtains in restricted circumstances. Denote the putative right adjoint by $\Pi_f : \mathcal{E}_M/B \to \mathcal{E}_M/C$, and for $\iota p : E \twoheadrightarrow B$ suppose that $(\Pi_f E)[y]$ is given by the set of 'partial sections' $\sigma : f^{-1}\{y\} \to ME$ of p over $f^{-1}\{y\}$ as in the commutative diagram:

$$
\begin{array}{ccc}
 & f^{-1}\{y\} \longrightarrow & \{y\} \\
\sigma \nearrow & \downarrow \quad \lrcorner & \downarrow \\
ME \xrightarrow{\ \eta_B{}^*p\ } & B \xrightarrow{\ \ f\ \ } & C
\end{array}
$$

Then we would need to exhibit a natural isomorphism $\mathcal{E}_M/B(f^*D, E) \cong \mathcal{E}_M/C(D, \Pi_f E)$. But this will only obtain when the 'backwards' components $h_y^\sharp : D[y] \to M(\Pi_f E)[y]$ are in the image of ι—otherwise, it is not generally possible to pull $f^{-1}\{y\}$ out of M.

Despite these restrictions, we do have enough structure at hand to instantiate $\mathbf{Coalg}^{\mathbb{T}}$ in $\mathbf{Poly}_M$. The only piece remaining is the composition product $\lhd$, but for our purposes it suffices to define its action on objects, which is identical to its action on objects in $\mathbf{Poly}$[1], and then to consider $\lhd$-comonoids in $\mathbf{Poly}_M$. The comonoid laws force the structure maps to be deterministic (*i.e.*, in the image of ι), and so $\lhd$-comonoids in $\mathbf{Poly}_M$ are just $\lhd$-comonoids in $\mathbf{Poly}_{\mathrm{id}_{\mathbf{Set}}}$.

Finally, we note that, even if the internal hom $[-, -]$ is not available in general, we can define morphisms $\beta : Sy^S \to [\mathbb{T}y, p]$: these again just correspond to morphisms $\mathbb{T}y \otimes Sy^S \to p$, and the condition that the backwards maps be uncorrelated between $\mathbb{T}y$ and p is incontrovertibly satisfied because $\mathbb{T}y$ has a trivial exponent. Unwinding such a β according to the definition of $\mathbf{Poly}_M$ indeed gives precisely a pair (β^o, β^u) of the requisite types; and a comonoid homomorphism $Sy^S \to y^{\mathbb{T}}$ in $\mathbf{Poly}_M$ is precisely a functor $\mathbf{B}\mathbb{T} \to \mathcal{K}\ell(M)$, thereby establishing equivalence between the objects of $\mathbf{Coalg}^{\mathbb{T}}(p)$ established in $\mathbf{Poly}_M$ and the objects of $\mathbf{Coalg}_{\mathcal{C}}^{\mathbb{T}}(p)$.

---

[1]We leave the full exposition of $\lhd$ in $\mathbf{Poly}_M$ to future work.

Henceforth, therefore, we will write $\mathbf{Coalg}_M^{\mathbb{T}}$ to denote the instantiation of $\mathbf{Coalg}^{\mathbb{T}}$ in $\mathbf{Poly}_M$. We will call the objects of $\mathbf{Coalg}_M^{\mathbb{T}}(p)$ pM-*coalgebras with time* $\mathbb{T}$, and to get a sense of how, in the case where M is a probability monad, they provide a notion of open Markov process, we can read off the definition a little more explictly.

Proposition 6.2.22. A pM-*coalgebra with time* $\mathbb{T}$ consists of a triple $\vartheta := (S, \vartheta^o, \vartheta^u)$ of a state space $S : \mathcal{E}$ and two morphisms $\vartheta^o : \mathbb{T} \times S \to p(1)$ and $\vartheta^u : \sum_{t:\mathbb{T}} \sum_{s:\mathbb{S}} p[\vartheta^o(t,s)] \to MS$, such that, for any section $\sigma : p(1) \to \sum_{i:p(1)} p[i]$ of p in $\mathcal{E}$, the maps $\vartheta^\sigma : \mathbb{T} \times S \to MS$ given by

$$\sum_{t:\mathbb{T}} S \xrightarrow{\vartheta^o(t)^*\sigma} \sum_{t:\mathbb{T}} \sum_{s:S} p[\vartheta^o(t,s)] \xrightarrow{\vartheta^u} MS$$

constitute an object in the functor category $\mathbf{Cat}\big(\mathbf{B}\mathbb{T}, \mathcal{K}\ell(M)\big)$, where $\mathbf{B}\mathbb{T}$ is the delooping of $\mathbb{T}$ and $\mathcal{K}\ell(M)$ is the Kleisli category of M. (Once more, we call the closed system ϑ^σ, induced by a section σ of p, the *closure* of ϑ by σ.)

Following Example 6.1.6 and the intuition of Example 6.2.9, we can see how this produces an open version of a Markov process.

Since stochastic dynamical systems are often alternatively presented as random dynamical systems, we now briefly consider how these can be incorporated into the coalgebraic framework.

6.2.3. Open random dynamical systems

In the analysis of stochastic systems, it is often fruitful to consider two perspectives: on one side, one considers explicitly the evolution of the distribution of the states of the system, by following (for instance) a Markov process, or Fokker-Planck equation. On the other side, one considers the system as if it were a deterministic system, perturbed by noisy inputs, giving rise to the frameworks of stochastic differential equations and associated *random dynamical systems*.

Whereas a (closed) Markov process is typically given by the action of a time monoid on an object in a Kleisli category of a probability monad, a (closed) random dynamical system is given by a bundle of closed dynamical systems, where the base system is equipped with a probability measure which it preserves: the idea being that a random dynamical system can be thought of as a 'random' choice of dynamical system on the total space at each moment in time, with the base measure-preserving system being the source of the randomness [13].

This idea corresponds in non-dynamical settings to the notion of *randomness pushback* [109, Def. 11.19], by which a stochastic map $f : A \to \mathcal{P}B$ can be presented as a deterministic map $f^\flat : \Omega \times A \to B$ where (Ω, ω) is a measure space such that, for any $a : A$, pushing ω forward

through $f^\flat(\text{-}, a)$ gives the state $f(a)$; that is, ω induces a random choice of map $f^\flat(\omega, \text{-}) : A \to B$. Similarly, under nice conditions, random dynamical systems and Markov processes do coincide, although they have different suitability in applications.

In this section, we sketch how the generalized-coalgebraic structures developed above extend also to random dynamical systems. We begin by defining the concept of measure-preserving dynamical system, which itself requires the notion of measure space (in order that measure can be preserved); we define the corresponding category abstractly, using a notion of slice category dual to that of Definition 3.2.10.

Definition 6.2.23. Suppose X is an object of a category $\mathcal{E}$. The slice of $\mathcal{E}$ *under* X, denoted $X/\mathcal{E}$, is the category whose objects (A, i) are morphisms $X \xrightarrow{i} A$ out of X in $\mathcal{E}$, and whose morphisms $f : (A, i) \to (B, j)$ are the evident triangles:

$$
\begin{array}{ccc}
 & X & \\
{\scriptstyle i}\swarrow & & \searrow{\scriptstyle j} \\
A & \xrightarrow{\quad f \quad} & B
\end{array}
$$

There is a projection functor $F : X/\mathcal{E} \to \mathcal{E}$ mapping each object (A, i) to A and each morphism $f : (A, i) \to (B, j)$ to $f : A \to B$.

We can use this notion to define a notion of 'pointed' category.

Definition 6.2.24. Let $(\mathcal{C}, \otimes, I)$ be a monoidal category, $\mathcal{D}$ be a subcategory $\mathcal{D} \hookrightarrow \mathcal{C}$, and let F denote the projection $I/\mathcal{C} \to \mathcal{C}$. We define the category $\mathcal{D}_*$ to be the pullback category over the diagram $\mathcal{D} \hookrightarrow \mathcal{C} \xleftarrow{F} I/\mathcal{C}$.

The category $\mathcal{D}_*$ has objects 'pointed' by corresponding states in $\mathcal{C}$, and its morphisms are those that preserve these states. The category of measure spaces is obtained as an example accordingly.

Example 6.2.25. Consider the deterministic subcategory $\mathbf{Meas} \hookrightarrow \mathbf{sfKrn}$. The pointed category $\mathbf{Meas}_*$ obtained from Definition 6.2.24 is the category whose objects are *measure spaces* (M, μ) with μ a measure $1 \nrightarrow M$, and whose morphisms $f : (M, \mu) \to (N, \nu)$ are measure-preserving maps; *i.e.*, measurable functions $f : M \to N$ such that $\nu = f \bullet \mu$ in $\mathbf{sfKrn}$. Likewise, if $\mathcal{P}$ is a probability monad on $\mathcal{E}$, then we have $\mathcal{E} \hookrightarrow \mathcal{K\ell}(\mathcal{P})$ and hence can understand $\mathcal{E}_*$ as a category of abstract measure spaces.

Proposition 6.2.26. There is a projection functor $U : \mathcal{E}_* \to \mathcal{E}$ taking measure spaces (B, β) to the underlying spaces B and their morphisms $f : (A, \alpha) \to (B, \beta)$ to the underlying maps

$f : A \to \mathcal{P}B$. We will write B to refer to the space in $\mathcal{E}$ underlying a measure space (B, β), in the image of U.

Proof. The functor is obtained as the projection induced by the universal property of the pullback.
□

Definition 6.2.27. Let (B, β) be a measure space in $\mathcal{E} \hookrightarrow \mathcal{K}\ell(\mathcal{P})$. A closed *metric* or *measure-preserving* dynamical system (ϑ, β) on (B, β) with time $\mathbb{T}$ is a closed dynamical system ϑ with state space $B : \mathcal{E}$ such that, for all $t : \mathbb{T}$, $\mathcal{P}\vartheta(t) \circ \beta = \beta$; that is, each $\vartheta(t)$ is a (B, β)-endomorphism in $1/\mathcal{K}\ell(\mathcal{P})$.

Proposition 6.2.28. Closed measure-preserving dynamical systems in $\mathcal{E}$ with time $\mathbb{T}$ form the objects of a category $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathcal{E}_*)$ whose morphisms $f : (\vartheta, \alpha) \to (\psi, \beta)$ are maps $f : \vartheta(*) \to \psi(*)$ in $\mathcal{E}$ between the state spaces that preserve both flow and measure, as in the following commutative diagram, which also indicates their composition:

Proof. The identity morphism on a closed measure-preserving dynamical system is the identity map on its state space. It is easy to check that composition as in the diagram above is thus both associative and unital with respect to these identities.
□

As we indicated in the introduction to this section, closed random dynamical systems are bundles of deterministic systems over metric systems.

Definition 6.2.29. Let (ϑ, β) be a closed measure-preserving dynamical system. A closed *random dynamical system* over (ϑ, β) is an object of the slice category $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathcal{E})/\vartheta$; it is therefore a bundle of the corresponding functors.

Example 6.2.30. The solutions $X(t, \omega; x_0) : \mathbb{R}_+ \times \Omega \times M \to M$ to a stochastic differential equation $\mathrm{d}X_t = f(t, X_t)\mathrm{d}t + \sigma(t, X_t)\mathrm{d}W_t$, where $W : \mathbb{R}_+ \times \Omega \to M$ is a Wiener process in M,

define a random dynamical system $\mathbb{R}_+ \times \Omega \times M \to M : (t, \omega, x) \mapsto X(t, \omega; x_0)$ over the Wiener base flow $\theta : \mathbb{R}_+ \times \Omega \to \Omega : (t, \omega) \mapsto W(s + t, \omega) - W(t, \omega)$ for any $s : \mathbb{R}_+$.

We can use the same trick, of opening up closed systems along a polynomial interface, to define a notion of open random dynamical system — although at this point we do not have an elegant concise definition.

Definition 6.2.31. Let (θ, β) be a closed measure-preserving dynamical system in $\mathcal{E}$ with time $\mathbb{T}$, and let $p : \mathbf{Poly}_{\mathsf{id}_{\mathcal{E}}}$ be a polynomial in $\mathcal{E}$. Write $\Omega := \theta(*)$ for the state space of θ, and let $\pi : S \to \Omega$ be an object (bundle) in $\mathcal{E}/\Omega$. An *open random dynamical system over (θ, β) on the interface p with state space $\pi : S \to \Omega$ and time $\mathbb{T}$* consists in a pair of morphisms $\vartheta^o : \mathbb{T} \times S \to p(1)$ and $\vartheta^u : \sum_{t:\mathbb{T}} \sum_{s:S} p[\vartheta^o(t,s)] \to S$, such that, for any section $\sigma : p(1) \to \sum_{i:p(1)} p[i]$ of p, the maps $\vartheta^\sigma : \mathbb{T} \times S \to S$ defined as

$$\sum_{t:\mathbb{T}} S \xrightarrow{\vartheta^o(-)^* \sigma} \sum_{t:\mathbb{T}} \sum_{s:S} p[\vartheta^o(-, s)] \xrightarrow{\vartheta^u} S$$

form a closed random dynamical system in $\mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathcal{E})/\theta$, in the sense that, for all $t : \mathbb{T}$ and sections σ, the following diagram commutes:

$$
\begin{array}{ccccc}
S & \xrightarrow{\vartheta^o(t)^* \sigma} & \sum_{s:S} p[\vartheta^o(t, s)] & \xrightarrow{\vartheta^u(t)} & S \\
\downarrow{\scriptstyle \pi} & & & & \downarrow{\scriptstyle \pi} \\
\Omega & & \xrightarrow{\qquad\qquad \theta(t) \qquad\qquad} & & \Omega
\end{array}
$$

Proposition 6.2.32. Let (θ, β) be a closed measure-preserving dynamical system in $\mathcal{E}$ with time $\mathbb{T}$, and let $p : \mathbf{Poly}_{\mathsf{id}_{\mathcal{E}}}$ be a polynomial in $\mathcal{E}$. Open random dynamical systems over (θ, β) on the interface p form the objects of a category $\mathbf{RDyn}^{\mathbb{T}}(p, \theta)$. Writing $\vartheta := (\pi_X, \vartheta^o, \vartheta^u)$ and $\psi := (\pi_Y, \psi^o, \psi^u)$, a morphism $f : \vartheta \to \psi$ is a map $f : X \to Y$ in $\mathcal{E}$ making the following diagram commute for all times $t : \mathbb{T}$ and sections σ of p:

245

Identities are given by the identity maps on state-spaces. Composition is given by pasting of diagrams.

Proposition 6.2.33. The categories $\mathbf{RDyn}^{\mathbb{T}}(p, \theta)$ collect into a doubly-indexed category of the form $\mathbf{RDyn}^{\mathbb{T}} : \mathbf{Poly}_{\mathsf{id}_{\mathcal{E}}} \times \mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathcal{E}_*) \to \mathbf{Cat}$. By the universal property of the product $\times$ in $\mathbf{Cat}$, it suffices to define the actions of $\mathbf{RDyn}^{\mathbb{T}}$ separately on morphisms of polynomials and on morphisms of closed measure-preserving systems.

Suppose therefore that $\varphi : p \to q$ is a morphism of polynomials. Then, for each measure-preserving system $(\theta, \beta) : \mathbf{Cat}(\mathbf{B}\mathbb{T}, \mathcal{E}_*)$, we define the functor $\mathbf{RDyn}^{\mathbb{T}}(\varphi, \theta) : \mathbf{RDyn}^{\mathbb{T}}(p, \theta) \to \mathbf{RDyn}^{\mathbb{T}}(q, \theta)$ as follows. Let $\vartheta := (\pi_X : X \to \Omega, \vartheta^o, \vartheta^u) : \mathbf{RDyn}^{\mathbb{T}}(p, \theta)$ be an object (open random dynamical system) in $\mathbf{RDyn}^{\mathbb{T}}(p, \theta)$. Then $\mathbf{RDyn}^{\mathbb{T}}(\varphi, \theta)(\vartheta)$ is defined as the triple $(\pi_X, \varphi_1 \circ \vartheta^o, \vartheta^u \circ \varphi^{o*}\varphi^{\sharp}) : \mathbf{RDyn}^{\mathbb{T}}(q, \theta)$, where the two maps are explicitly the following composites:

$$\mathbb{T} \times X \xrightarrow{\vartheta^o} p(1) \xrightarrow{\varphi_1} q(1), \qquad \sum_{t:\mathbb{T}} \sum_{x:X} q[\varphi_1 \circ \vartheta^o(t, x)] \xrightarrow{\vartheta^{o*}\varphi^{\sharp}} \sum_{t:\mathbb{T}} \sum_{x:X} p[\vartheta^o(t, x)] \xrightarrow{\vartheta^u} X \,.$$

On morphisms $f : (\pi_X : X \to \Omega, \vartheta^o, \vartheta^u) \to (\pi_Y : Y \to \Omega, \psi^o, \psi^u)$, the image $\mathbf{RDyn}^{\mathbb{T}}(\varphi, \theta)(f) : \mathbf{RDyn}^{\mathbb{T}}(\varphi, \theta)(\pi_X, \vartheta^o, \vartheta^u) \to \mathbf{RDyn}^{\mathbb{T}}(\varphi, \theta)(\pi_Y, \psi^o, \psi^u)$ is given by the same underlying map $f : X \to Y$ of state spaces.

Next, suppose that $\phi : (\theta, \beta) \to (\theta', \beta')$ is a morphism of closed measure-preserving dynamical systems, and let $\Omega' := \theta'(*)$ be the state space of the system θ'. By Proposition 6.2.28, the morphism ϕ corresponds to a map $\phi : \Omega \to \Omega'$ on the state spaces that preserves both flow and measure. Therefore, for each polynomial $p : \mathbf{Poly}_{\mathsf{id}_{\mathcal{E}}}$, we define the functor $\mathbf{RDyn}^{\mathbb{T}}(p, \phi) : \mathbf{RDyn}^{\mathbb{T}}(p, \theta) \to \mathbf{RDyn}^{\mathbb{T}}(p, \theta')$ by post-composition. That is, suppose given open random dynamical systems and morphisms over (p, θ) as in the diagram of Proposition 6.2.32. Then $\mathbf{RDyn}^{\mathbb{T}}(p, \phi)$ returns the following diagram:

That is, $\mathbf{RDyn}^{\mathbb{T}}(p, \phi)(\vartheta) := (\phi \circ \pi_X, \vartheta^o, \vartheta^u)$ and $\mathbf{RDyn}^{\mathbb{T}}(p, \phi)(f)$ is given by the same underlying map $f : X \to Y$ on state spaces.

6.3. Cilia: monoidal bicategories of cybernetic systems

Whereas it is the morphisms (1-cells) of process-theoretic categories—such as categories of lenses, or the categories of statistical games to be defined in Chapter 7—that represent open systems, it is the objects (0-cells) of the opindexed categories $\mathbf{Coalg}_M^{\mathbb{T}}$[2] that play this rôle; in fact, the objects of $\mathbf{Coalg}_M^{\mathbb{T}}$ each represent both an open system and its (polynomial) interface. In order to supply dynamical semantics for statistical games—functors from categories of statistical games to categories of dynamical systems—we need to cleave the dynamical systems from their interfaces, making the interfaces into 0-cells and systems into 1-cells between them, thereby letting the systems' types and composition match those of the games. Doing this is the job of this section, which we first perform in the case of the general categories $\mathbf{Coalg}_M^{\mathbb{T}}$, followed by the specific case of systems generated differentially, as in the vector-field Examples 6.2.8 and 6.2.9.

6.3.1. Hierarchical bidirectional dynamical systems

To construct "hierarchical bidirectional systems", we will associate to each pair of objects (A, S) and (B, T) of a category of (for our purposes, Bayesian) lenses a polynomial $[\![Ay^S, By^T]\!]$ whose configurations correspond to lenses and whose inputs correspond to the lenses' inputs. The categories $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}([\![Ay^S, By^T]\!])$ will then form the hom-categories of bicategories of hierarchical inference systems called *cilia*[3], and it is in these bicategories that we will find our dynamical semantics.

Throughout this subsection, we will fix a category $\mathcal{C}$ of stochastic channels, defined by $\mathcal{C} := \mathcal{K}\ell(\mathcal{P})$ as the Kleisli category of a probability monad $\mathcal{P} : \mathcal{E} \to \mathcal{E}$, which we will also take to define a category $\mathbf{Poly}_{\mathcal{P}}$ of polynomials with stochastic feedback. We will assume $\mathcal{P}$ to be a monoidal monad, and we will write the monoidal structure on $\mathcal{C}$ as $(\otimes, I)$. Finally, we will assume that $\mathcal{C}$ is enriched in its underlying category of spaces $\mathcal{E}$.

Definition 6.3.1. Let $\mathbf{BayesLens}$ be the category of Bayesian lenses in $\mathcal{C}$. Then for any pair of objects (A, S) and (B, T) in $\mathbf{BayesLens}$, we define a polynomial $[\![Ay^S, By^T]\!]$ in $\mathbf{Poly}_{\mathcal{P}}$ by

$$[\![Ay^S, By^T]\!] := \sum_{l : \mathbf{BayesLens}\big((A,S),(B,T)\big)} y^{\mathcal{C}(I,A) \times T}.$$

[2]or, more precisely, their corresponding opfibrations $\int \mathbf{Coalg}_M^{\mathbb{T}}$

[3]'Cilia', because they "control optics", like the ciliary muscles of the eye.

Remark 6.3.2. We can think of $[\![Ay^S, By^T]\!]$ as an 'external hom' polynomial for **BayesLens**, playing a rôle analogous to the internal hom $[p, q]$ in $\mathbf{Poly}_\mathcal{P}$. Its 'bifunctorial' structure—with domain and codomain parts—is what enables cleaving systems from their interfaces, which are given by these parts. The definition, and the following construction of the monoidal bicategory, are inspired by the operad **Org** introduced by Spivak [240].

Remark 6.3.3. Note that $[\![Ay^S, By^T]\!]$ is strictly speaking a monomial, since it can be written in the form Iy^J for $I = \mathbf{BayesLens}\big((A, S), (B, T)\big)$ and $J = \mathcal{C}(I, A) \times T$. However, we have written it in polynomial form with the view to extending it in future work to dependent lenses and dependent optics [43, 276] and these generalized external homs will in fact be true polynomials.

Proposition 6.3.4. Definition 6.3.1 defines a functor $\mathbf{BayesLens}^{\mathrm{op}} \times \mathbf{BayesLens} \to \mathbf{Poly}_\mathcal{P}$. Suppose $c := (c_1, c^\sharp) : (Z, R) \nrightarrow (A, S)$ and $d := (d_1, d^\sharp) : (B, T) \nrightarrow (C, U)$ are Bayesian lenses. We obtain a morphism of polynomials $[\![c, d]\!] : [\![Ay^S, By^T]\!] \to [\![Zy^R, Cy^U]\!]$ as follows. Since the configurations of $[\![Ay^S, By^T]\!]$ are lenses $(A, S) \nrightarrow (B, T)$, the forwards map acts by pre- and post-composition:

$$[\![c, d]\!]_1 := d \mathbin{\varphi} (-) \mathbin{\varphi} c : \mathbf{BayesLens}\big((A, S), (B, T)\big) \to \mathbf{BayesLens}\big((Z, R), (C, U)\big)$$

$$l \mapsto d \mathbin{\varphi} l \mathbin{\varphi} c$$

For each such l, the backwards map $[\![c, d]\!]_l^\sharp$ has type $\mathcal{C}(I, Z) \otimes U \to \mathcal{C}(I, A) \otimes T$ in $\mathcal{C}$, and is obtained by analogy with the backwards composition rule for Bayesian lenses. We define

$$[\![c, d]\!]_l^\sharp := \mathcal{C}(I, Z) \otimes U \xrightarrow{c_{1*} \otimes U} \mathcal{C}(I, A) \otimes U \xrightarrow{\mathbf{\curlyvee} \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(I, A) \otimes U \cdots$$

$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes l_{1*} \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(I, B) \otimes U \xrightarrow{\mathcal{C}(I,A) \otimes d^\sharp \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(U, T) \otimes U \cdots$$

$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes \mathsf{ev}_{U,T}} \mathcal{C}(I, A) \otimes T$$

where l_1 is the forwards part of the lens $l : (A, S) \nrightarrow (B, T)$, and $c_{1*} := \mathcal{C}(I, c_1)$ and $l_{1*} := \mathcal{C}(I, l_1)$ are the push-forwards along c_1 and l_1, and $\mathsf{ev}_{U,T}$ is the evaluation map induced by the enrichment of $\mathcal{C}$ in $\mathcal{E}$.

Less abstractly, with $\mathcal{C} = \mathcal{Kl}(\mathcal{P})$, we can write $[\![c, d]\!]_l^\sharp$ as the following map in $\mathcal{E}$, depicted as a

string diagram:

$$\llbracket c, d \rrbracket_l^\sharp \quad = \quad$$
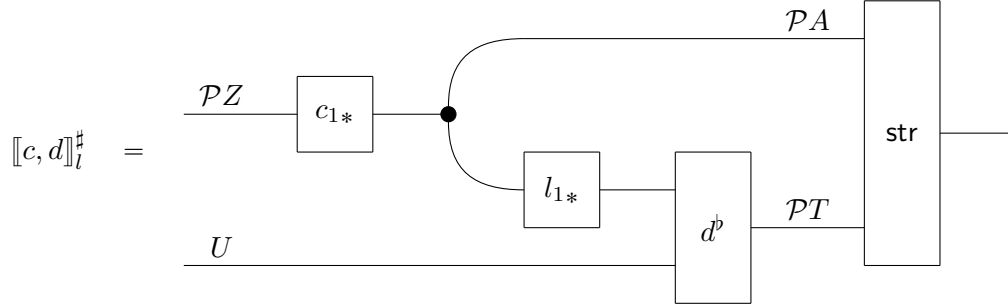
Here, we have assumed that $\mathcal{K}\ell(\mathcal{P})(I, A) = \mathcal{P}A$, and define $d^\flat : \mathcal{P}B \times U \to \mathcal{P}T$ to be the image of $d^\sharp : \mathcal{P}B \to \mathcal{K}\ell(\mathcal{P})(U, T)$ under the Cartesian closure of $\mathcal{E}$, and $\mathsf{str} : \mathcal{P}A \times \mathcal{P}T \to \mathcal{P}\big(\mathcal{P}A \times T\big)$ the (left) strength of the monoidal monad $\mathcal{P}$.

The morphism $\llbracket c, d \rrbracket_l$ acts to 'wrap' the lens l by pre-composing with c and post-composing with d. The backwards component $\llbracket c, d \rrbracket_l^\sharp$ therefore acts to take the inputs of the resulting composite $d \mathbin{\phi} l \mathbin{\phi} c$ to appropriate inputs for l; that is, it maps a pair (π, u) to $(c_1 \bullet \pi, d_{l_1 \bullet c_1 \bullet \pi}^\sharp(u))$.

Proof. We need to check that the mappings defined above respect identities and composition. It is easy to see that the definition preserves identities: in the forwards direction, this follows from the unitality of composition in **BayesLens**; in the backwards direction, because pushing forwards along the identity is again the identity, and because the backwards component of the identity Bayesian lens is the constant state-dependent morphism on the identity in $\mathcal{C}$.

To check that the mapping preserves composition, we consider the contravariant and covariant parts separately. Suppose $b := (b_1, b^\sharp) : (Y, Q) \rightarrowtail (Z, R)$ and $e := (e_1, e^\sharp) : (C, U) \rightarrowtail (D, V)$ are Bayesian lenses. We consider the contravariant case first: we check that $\llbracket c \mathbin{\phi} b, By^T \rrbracket = \llbracket b, By^T \rrbracket \circ \llbracket c, By^T \rrbracket$. The forwards direction holds by pre-composition of lenses. In the backwards direction, we note from the definition that only the forwards channel c_1 plays a rôle in $\llbracket c, By^T \rrbracket_l^\sharp$, and that rôle is again pre-composition. We therefore only need to check that $(c_1 \bullet b_1)_* = c_{1*} \circ b_{1*}$, and this follows immediately from the functoriality of $\mathcal{C}(I, -)$.

We now consider the covariant case, that $\llbracket Ay^S, e \mathbin{\phi} d \rrbracket = \llbracket Ay^S, e \rrbracket \circ \llbracket Ay^S, d \rrbracket$. Once again, the forwards direction holds by composition of lenses. For simplicity of exposition, we consider the backwards direction (with $\mathcal{C} = \mathcal{K}\ell(\mathcal{P})$) and reason graphically. In this case, the backwards map on

the right-hand side is given, for a lens $l : (A, S) \nrightarrow (B, T)$ by the following string diagram:
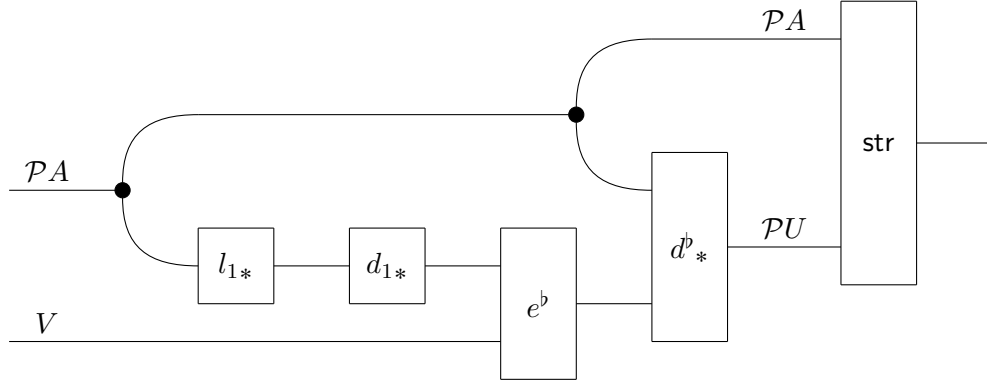
It is easy to verify that the composition of backwards channels here is precisely the backwards channel given by $e \mathbin{\phi} d$—see Theorem 4.3.14—which establishes the result. $\qquad\square$

Remark 6.3.5. Above, we claimed that a monoidal monad $\mathcal{P} : \mathcal{E} \to \mathcal{E}$ on a symmetric monoidal category $(\mathcal{E}, \times, 1)$ is equipped with a (left) strength $\mathsf{str}_{X,Y} : X \times \mathcal{P}Y \to \mathcal{P}(X \times Y)$, in the sense of Definition 3.2.4. This can be obtained from the unit η and the laxator α of the monad as follows:

$$\mathsf{str}_{X,Y} : X \times \mathcal{P}Y \xrightarrow{\eta_X \times \mathsf{id}_{\mathcal{P}Y}} \mathcal{P}X \times \mathcal{P}Y \xrightarrow{\alpha_{X,Y}} \mathcal{P}(X \times Y)$$

Using the monad laws, a strength obtained in this way can be shown to satisfy the following axioms (that the strength commutes with the monad structure), and so one may say that $\mathcal{P}$ is a *strong monad*:

$$
\begin{array}{ccc}
 & A \times B & \\
{\scriptstyle \mathsf{id}_A \times \eta_B} \swarrow & & \searrow {\scriptstyle \eta_{A \times B}} \\
A \times \mathcal{P}B & \xrightarrow[\mathsf{str}_{A,B}]{} & \mathcal{P}(A \times B)
\end{array}
$$

$$
\begin{array}{ccccc}
A \times \mathcal{P}\mathcal{P}B & \xrightarrow{\mathsf{str}_{A,\mathcal{P}B}} & \mathcal{P}(A \times \mathcal{P}B) & \xrightarrow{\mathcal{P}(\mathsf{str}_{A,B})} & \mathcal{P}\mathcal{P}(A \times B) \\
{\scriptstyle A \times \mu_B} \downarrow & & & & \downarrow {\scriptstyle \mu_{A \times B}} \\
A \times \mathcal{P}B & & \xrightarrow{\mathsf{str}_{A,B}} & & \mathcal{P}(A \times B)
\end{array}
$$

Now that we have an 'external hom', we might expect also to have a corresponding 'external composition', represented by a family of morphisms of polynomials; we establish such a family now, and it will be important in our bicategorical construction.

Definition 6.3.6. We define an 'external composition' natural transformation c, with components

$$\llbracket Ay^S, By^T \rrbracket \otimes \llbracket By^T, Cy^U \rrbracket \to \llbracket Ay^S, Cy^U \rrbracket$$

given in the forwards direction by composition of Bayesian lenses. In the backwards direction, for each pair of lenses $c : (A, S) \nrightarrow (B, T)$ and $d : (B, T) \nrightarrow (C, U)$, we need a map

$$\mathsf{c}^{\sharp}_{c,d} : \mathcal{C}(I, A) \otimes U \to \mathcal{C}(I, A) \otimes T \otimes \mathcal{C}(I, B) \otimes U)$$

which we define as follows:

$$\mathsf{c}^{\sharp}_{c,d} := \mathcal{C}(I, A) \otimes U \xrightarrow{\curlyeqprec \otimes \curlyeqprec} \mathcal{C}(I, A) \otimes \mathcal{C}(I, A) \otimes U \otimes U \cdots$$
$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes c_{1*} \otimes U \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(I, B) \otimes U \otimes U \cdots$$
$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes \curlyeqprec \otimes \mathcal{C}(I,B) \otimes U \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(I, B) \otimes \mathcal{C}(I, B) \otimes U \otimes U$$
$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes \mathcal{C}(I,B) \otimes d^{\sharp} \otimes U \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(I, B) \otimes \mathcal{C}(U, T) \otimes Y \otimes U$$
$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes \mathcal{C}(I,B) \mathsf{ev}_{U,T} \otimes U} \mathcal{C}(I, A) \otimes \mathcal{C}(I, B) \otimes T \otimes U$$
$$\cdots \xrightarrow{\mathcal{C}(I,A) \otimes \mathsf{swap} \otimes U} \mathcal{C}(I, A) \otimes T \otimes \mathcal{C}(I, B) \otimes U$$

where c_{1*} and $\mathsf{ev}_{U,T}$ are as in 6.3.4.

With $\mathcal{C} = \mathcal{K}\ell(\mathcal{P})$, we can equivalently (and more legibly) define $\mathsf{c}^{\sharp}_{c,d}$ by the following string diagram:
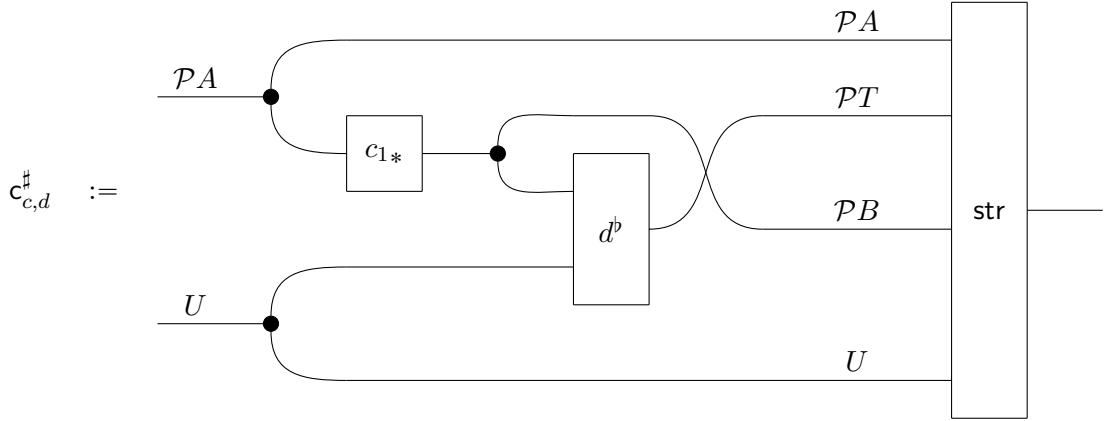
where $d^{\flat}$ and str are also as in Proposition 6.3.4.

We can therefore understand $\mathsf{c}^{\sharp}_{c,d}$ as mapping forward and backward inputs for the composite lens $d \mathbin{\phi} c$ to appropriate inputs for the constituent lenses c and d; that is, $\mathsf{c}^{\sharp}_{c,d}$ maps (π, u) to $(\pi, d^{\sharp}_{c_1 \bullet \pi}(u), c_1 \bullet \pi, u)$. The resulting inputs to the lens c are therefore $(\pi, d^{\sharp}_{c_1 \bullet \pi}(u))$, and those to d are $(c_1 \bullet \pi, u)$. (This is precisely as the law of lens composition stipulates: the forwards input to d is obtained by pushing forwards through d; and the backwards input to c is obtained from the backwards component of d.)

251

We leave to the reader the detailed proof that this definition produces a well-defined natural transformation, noting only that the argument is analogous to that of Proposition 6.3.4: one observes that, in the forwards direction, the definition is simply composition of Bayesian lenses (which is immediately natural); in the backwards direction, one observes that the definition again mirrors that of the backwards composition of Bayesian lenses.

Next, we establish the structure needed to make our bicategory monoidal.

Definition 6.3.7. We define a distributive law d of $[\![-, =]\!]$ over $\otimes$, a natural transformation with components

$$[\![Ay^S, By^T]\!] \otimes [\![A'y^{S'}, B'y^{T'}]\!] \to [\![Ay^S \otimes A'y^{S'}, By^T \otimes B'y^{T'}]\!],$$

noting that $Ay^S \otimes A'y^{S'} = (A \otimes A')y^{(S \otimes S')}$ and $By^T \otimes B'y^{T'} = (B \otimes B')y^{(T \otimes T')}$. The forwards component is given simply by taking the tensor of the corresponding Bayesian lenses, using the monoidal product (also denoted $\otimes$) in **BayesLens**. Backwards, for each pair of lenses $c : (A, S) \nrightarrow (B, T)$ and $c' : (A', S') \nrightarrow (B', T')$, we need a map

$$\mathsf{d}^{\sharp}_{c,c'} : \mathcal{C}(I, A \otimes A') \otimes T \otimes T' \to \mathcal{C}(I, A) \times T \times \mathcal{C}(I, A') \times T'$$

for which we choose

$$\mathcal{C}(I, A \otimes A') \otimes T \otimes T' \xrightarrow{\curlyvee \otimes T \otimes T'} \mathcal{C}(I, A \otimes A') \otimes \mathcal{C}(I, A \otimes A') \otimes T \otimes T' \cdots$$

$$\cdots \xrightarrow{\mathcal{C}(I, \mathsf{proj}_A) \otimes \mathcal{C}(I, \mathsf{proj}_{A'}) \otimes T \otimes T'} \mathcal{C}(I, A) \otimes \mathcal{C}(I, A') \otimes T \otimes T' \cdots$$

$$\cdots \xrightarrow{\mathcal{C}(I, A) \otimes \mathsf{swap} \otimes T'} \mathcal{C}(I, A) \otimes T \otimes \mathcal{C}(I, A') \otimes T'$$

where swap is the symmetry of the tensor $\otimes$ in $\mathcal{C}$. Note that $\mathsf{d}^{\sharp}_{c,c'}$ so defined does not in fact depend on either c or c'.

We now have everything we need to construct a monoidal bicategory $\mathbf{Cilia}^{\mathbb{T}}_{\mathcal{P}}$ of dynamical hierarchical inference systems in $\mathcal{C}$, following the intuition outlined at the beginning of this section. We call systems over such external hom polynomials *cilia*, as they "control optics", akin to the ciliary muscles of the eye. In future work, we will study the general structure of these categories and their relationship to categorical systems theory [191, 192] and related work in categorical cybernetics [51].

Definition 6.3.8. Let $\mathbf{Cilia}^{\mathbb{T}}_{\mathcal{P}}$ denote the monoidal bicategory whose 0-cells are objects (A, S) in **BayesLens**, and whose hom-categories $\mathbf{Cilia}^{\mathbb{T}}_{\mathcal{P}}\big((A, S), (B, T)\big)$ are given by

$\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}(\llbracket Ay^S, By^T \rrbracket)$. The identity 1-cell $\mathrm{id}_{(A,S)} : (A,S) \to (A,S)$ on (A,S) is given by the system with trivial state space 1, trivial update map, and output map that constantly emits the identity Bayesian lens $(A,S) \rightsquigarrow (A,S)$. The composition of a system $(A,S) \to (B,T)$ then a system $(B,T) \to (C,U)$ is defined by the functor

$$\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A,S),(B,T)\big) \times \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((B,T),(C,U)\big)$$
$$= \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket Ay^S, By^T \rrbracket\big) \times \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket By^T, Cy^U \rrbracket\big)$$
$$\xrightarrow{\lambda} \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket Ay^S, By^T \rrbracket \otimes \llbracket By^T, Cy^U \rrbracket\big)$$
$$\xrightarrow{\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}(\mathsf{c})} \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket Ay^S, Cy^U \rrbracket\big) = \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A,S),(C,U)\big)$$

where λ is the laxator and c is the external composition morphism of Definition 6.3.6.

The monoidal structure $(\otimes, y)$ on $\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}$ derives from the structures on $\mathbf{Poly}_{\mathcal{P}}$ and $\mathbf{BayesLens}$, justifying our overloaded notation. On 0-cells, $(A,S) \otimes (A',S') := (A \otimes A', S \otimes S')$. On 1-cells $(A,S) \to (B,T)$ and $(A',S') \to (B',T')$, the tensor is given by

$$\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A,S),(B,T)\big) \times \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A',S'),(B',T')\big)$$
$$= \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket Ay^S, By^T \rrbracket\big) \times \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket A'y^{S'}, B'y^{T'} \rrbracket\big)$$
$$\xrightarrow{\lambda} \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket Ay^S, By^T \rrbracket \otimes \llbracket A'y^{S'}, B'y^{T'} \rrbracket\big)$$
$$\xrightarrow{\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}(\mathsf{d})} \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}\big(\llbracket Ay^S \otimes A'y^{S'}, By^T \otimes B'y^{T'} \rrbracket\big)$$
$$= \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A,S) \otimes (A',S'), (B,T) \otimes (B',T')\big)$$

where d is the distributive law of Definition 6.3.7. The same functors

$$\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A,S),(B,T)\big) \times \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A',S'),(B',T')\big) \to \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}\big((A,S) \otimes (A',S'), (B,T) \otimes (B',T')\big)$$

induce the tensor of 2-cells; concretely, this is given on morphisms of dynamical systems by taking the product of the corresponding morphisms between state spaces.

We do not give here a proof that this makes $\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}$ into a well-defined monoidal bicategory; briefly, the result follows from the facts that the external composition c and the tensor $\otimes$ are appropriately associative and unital, that $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}$ is lax monoidal, that $\llbracket -, = \rrbracket$ is functorial in both positions, and that $\llbracket -, = \rrbracket$ distributes naturally over $\otimes$.

Before we move on, it will be useful to spell out concretely the elements of a 'cilium' (a 1-cell) $(A,S) \to (B,T)$ in $\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}$.

Proposition 6.3.9. Suppose $\mathcal{P}$ is a monad on a Cartesian closed category $\mathcal{E}$. Then a 1-cell $\vartheta :$ $(A, S) \to (B, T)$ in $\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}$ is given by a tuple $\vartheta := (X, \vartheta_1^o, \vartheta_2^o, \vartheta^u)$ of

- a choice of state space X,

- a forwards output map $\vartheta_1^o : \mathbb{T} \times X \times A \to \mathcal{P}B$ in $\mathcal{E}$,

- a backwards output map $\vartheta_2^o : \mathbb{T} \times X \times \mathcal{P}A \times T \to \mathcal{P}S$ in $\mathcal{E}$, and

- an update map $\vartheta^u : \mathbb{T} \times X \times \mathcal{P}A \times T \to \mathcal{P}X$ in $\mathcal{E}$,

satisfying the 'flow' condition of Proposition 6.2.22.

Proof. The result follows immediately upon unpacking the definitions, using the Cartesian closure of $\mathcal{E}$. □

6.3.2. Differential systems

Approximate inference doctrines describe how systems play statistical games, and are particularly of interest when one asks how systems' performance may improve during such game-playing. One prominent method of performance improvement involves descending the gradient of a statistical game's loss function, and we will see below that this method is adopted by both the Laplace and the Hebb-Laplace doctrines. The appearance of gradient descent prompts questions about the connections between such statistical systems and other 'cybernetic' systems such as deep learners or players of economic games, both of which may also involve gradient descent [51, 74]; indeed, it has been proposed [54] that parameterized gradient descent should form the basis of a compositional account of cybernetic systems in general[4].

In order to incorporate gradient descent explicitly into our own compositional framework, we follow the recipes above to define here first a category of differential systems opindexed by polynomial interfaces and then a monoidal bicategory of differential hierarchical inference systems. We then show how we can obtain dynamical from differential systems by integration, and sketch how this induces a "change of base" from dynamical to differential hierarchical inference systems.

---

[4]Our own view on cybernetics is somewhat more general, since not all systems that may be seen as cybernetic are explicitly structured as gradient-descenders, and nor even is explicit differential structure always apparent. In earlier work, we suggested that statistical inference was perhaps more inherent to cybernetics [251], although today we believe that a better, though more informal, definition of cybernetic system is perhaps "an intentionally-controlled open dynamical system". (Slightly more formally, we can understand this as "an open dynamical system clad in a controller", with the possible 'cladding' collected into a fibration over systems of each given type.) Nonetheless, we acknowledge that this notion of "intentional control" may generally be reducible to a stationary action principle, again indicating the importance of differential structure. We leave the statement and proof of this general principle to future work.

Differential systems require differential structure, but we are here still concerned with statistical systems whose time evolution is stochastic. This means that a differential system will be given by a *stochastic vector field*: a stochastic section of the tangent bundle over the system's state space. However, as we have seen, the state spaces of stochastic systems are naturally found in a category of measurable spaces, but such a categorical setting does not generally supply differential structure too, and without this we cannot define tangent bundles. This poses our first hurdle.

We will not here entirely vault this hurdle, for the interplay of randomness and smoothness is subtle and untangling it is not our purpose in this thesis. However, we can overcome it in a manner which is satisfactory for our present needs, by noting that all our state spaces of later interest will be Euclidean, meaning that we can equip them with their standard Borel measurable structure. In future work, we hope to generalize this situation, possibly using the notion of *relative monad* [8].

Definition 6.3.10. Let **Euc** denote the category whose objects are finite-dimensional Euclidean spaces $\mathbb{R}^n$ and whose morphisms are smooth maps between them.

Euclidean spaces are trivially manifolds: the tangent space over each point $x \in \mathbb{R}^n$ is again $\mathbb{R}^n$. Hence, if X is a Euclidean space, then the tangent bundle $\mathsf{T}X \to X$ is simply the projection $X \times X \to X$ mapping (x, v) to x. As in general differential geometry, T yields a functor **Euc** $\to$ **Euc**.

Proposition 6.3.11. The tangent bundle functor $\mathsf{T} : \mathbf{Euc} \to \mathbf{Euc}$ maps each Euclidean space $\mathbb{R}^n$ to $\mathbb{R}^n \times \mathbb{R}^n$ and each smooth map $f : \mathbb{R}^m \to \mathbb{R}^n$ to its differential $df : \mathbb{R}^m \times \mathbb{R}^m \to \mathbb{R}^n \times \mathbb{R}^n$, which in turn maps (x, v) to $\big(f(x), \partial_x f(v)\big)$, where $\partial_x f$ denotes the (total) derivative of f at x, which can be represented by its $n \times m$ Jacobian matrix.

Remark 6.3.12. Differentials compose by pushforward, which yields the chain rule of differential calculus. Earlier we have seen that chain rules indicate the presence of a fibration, and indeed this is also the case here: T is properly a functor into the fibration of vector bundles over the category of spaces; composing this functor with the projection out of the fibration yields the endofunctor we have sketched in the preceding proposition.

Ordinary differential equations define vector fields, which are (deterministic) sections of the tangent bundle over a space; these are deterministic closed differential systems. We are interested in *open* differential systems that may have effectful (*e.g.* stochastic) evolution: for openness, we will use the trick of §6.2; for stochasticity, we will need *stochastic sections*, which means transporting the tangent bundles into a category of stochastic maps and considering their sections there.

Proposition 6.3.13. There is a functor $J : \mathbf{Euc} \to \mathbf{Meas}$ that takes each Euclidean space and exhibits it as a measurable space equipped with its standard Borel σ-algebra, and which takes each smooth map and exhibits it as a measurable function. This functor preserves products.

Proposition 6.3.14 (Heunen et al. [131, §III.B])**.** There is a functor $R : \mathbf{Meas} \to \mathbf{QBS}$ which is full and faithful when restricted to the subcategory $\mathbf{Borel} \hookrightarrow \mathbf{Meas}$ of standard Borel spaces.

Using these functors, we can transport a tangent bundle $\pi_X : \mathsf{T}X \to X$ in $\mathbf{Euc}$ to $\mathbf{QBS}$, as $RJ\pi_X$. Then, if we let $\mathcal{P} : \mathbf{QBS} \to \mathbf{QBS}$ denote the probability monad on quasi-Borel spaces introduced in Example 4.1.31, we can take the sections of $RJ\pi_X$ in $\mathcal{K}\ell(\mathcal{P})$ to be the stochastic vector fields over the space X. Moreover, since $\mathbf{QBS}$ is finitely complete and Cartesian closed, it is sufficiently structured that we may instantiate the category $\mathbf{Poly}_{\mathcal{P}}$ of polynomials with $\mathcal{P}$-effectful feedback.

Using these two ideas, we may define our desired categories of stochastic differential systems. Recall that morphisms $Ay^B \to p$ in $\mathbf{Poly}_{\mathcal{P}}$ correspond to morphisms $A {\rightarrow\!\!\!\rightarrow} pB$ in $\mathcal{K}\ell(\mathcal{P})$.

Notation 6.3.15. In this section, let us write $(\widetilde{-})$ to denote the functor $RJ : \mathbf{Euc} \to \mathbf{QBS}$.

Definition 6.3.16. For each $p : \mathbf{Poly}_{\mathcal{P}}$, define a category $\mathbf{DiffSys}(p)$ as follows. Its objects are pairs (M, m) of a Euclidean space $M : \mathbf{Euc}$ and a morphism $m : \widetilde{M}y^{\widetilde{\mathsf{T}M}} \to p$ of polynomials in $\mathbf{Poly}_{\mathcal{P}}$, such that for any section $\sigma : p \to y$ of p, the composite morphism $\sigma \circ m : \widetilde{M}y^{\widetilde{\mathsf{T}M}} \to y$ corresponds to a stochastic section $m^\sigma : \widetilde{M} {\rightarrow\!\!\!\rightarrow} \widetilde{\mathsf{T}M}$ of the tangent bundle $\mathsf{T}M \to M$ under RJ. A morphism $\alpha : (M, m) \to (M', m')$ in $\mathbf{DiffSys}(p)$ is a smooth map $\alpha : M \to M'$ in $\mathbf{Euc}$ such that the following diagram commutes:

$$
\begin{array}{ccc}
\widetilde{M} & \xrightarrow{\ m\ } & p\widetilde{\mathsf{T}M} \\
{\scriptstyle \tilde{\alpha}}\downarrow & & \downarrow{\scriptstyle p\widetilde{\mathsf{T}\alpha}} \\
\widetilde{M'} & \xrightarrow[\ m'\ ]{} & p\widetilde{\mathsf{T}M'}
\end{array}
$$

We obtain a monoidal opindexed category from this data in much the same way as we did for $\mathbf{Coalg}^{\mathbb{T}}$.

Proposition 6.3.17. $\mathbf{DiffSys}$ defines an opindexed category $\mathbf{Poly}_{\mathcal{P}} \to \mathbf{Cat}$. Given a morphism $\varphi : p \to q$ of polynomials, $\mathbf{DiffSys}(\varphi) : \mathbf{DiffSys}(p) \to \mathbf{DiffSys}(q)$ acts on objects by postcomposition and trivially on morphisms.

Proposition 6.3.18. The functor $\mathbf{DiffSys}$ is lax monoidal $(\mathbf{Poly}_{\mathcal{P}}, \otimes, y) \to (\mathbf{Cat}, \times, \mathbf{1})$.

Proof sketch. Note that T is strong monoidal, with $\mathsf{T}(\mathbb{R}^0) \cong \mathbb{R}^0$ and $\mathsf{T}(M) \times \mathsf{T}(N) \cong \mathsf{T}(M \times N)$, that RJ preserves products, and that $RJ(\mathbb{R}^0) = 1$. The unitor $\mathbf{1} \to \mathbf{DiffSys}(y)$ is given by the isomorphism $\widetilde{\mathbb{R}^0} y^{\widetilde{\mathsf{T}\mathbb{R}^0}} \cong 1y^1 \cong y$ induced by the strong monoidal structure of T. The laxator $\lambda_{p,q} : \mathbf{DiffSys}(p) \times \mathbf{DiffSys}(q) \to \mathbf{DiffSys}(p \otimes q)$ is similarly determined: given objects $m : \widetilde{M} y^{\widetilde{\mathsf{T}M}} \to p$ and $n : \widetilde{N} y^{\widetilde{\mathsf{T}N}} \to q$, take their tensor $m \otimes n : (\widetilde{M} \otimes \widetilde{N}) y^{\widetilde{\mathsf{T}M} \otimes \widetilde{\mathsf{T}N}} \to p \otimes q$ and precompose with the induced morphism $(\widetilde{M \times N}) y^{\mathsf{T}(\widetilde{M \times N})} \to (\widetilde{M} \otimes \widetilde{N}) y^{\widetilde{\mathsf{T}M} \otimes \widetilde{\mathsf{T}M}}$; proceed similarly on morphisms of differential systems. The satisfaction of the unitality and associativity laws follows from the monoidality of T. $\qquad\square$

We now define a monoidal bicategory $\mathbf{DiffCilia}$ of differential hierarchical inference systems, following the definition of $\mathbf{Cilia}$ above.

Definition 6.3.19. Let $\mathbf{DiffCilia}$ denote the monoidal bicategory whose 0-cells are the objects (A, S) of $\mathbf{BayesLens}_{\mathcal{K}\ell(\mathcal{P})}$ and whose hom-categories $\mathbf{DiffCilia}\big((A, S), (B, T)\big)$ are given by $\mathbf{DiffSys}\big([\![Ay^S, By^T]\!]\big)$. The identity 1-cell $\mathrm{id}_{(A,S)} : (A, S) \to (A, S)$ on (A, S) is given by the differential system $y \to [\![Ay^S, By^T]\!]$ with state space $\mathbb{R}^0$, trivial backwards component, and forwards component that picks the identity Bayesian lens on (A, S). The composition of differential systems $(A, S) \to (B, T)$ then $(B, T) \to (C, U)$ is defined by the functor

$$
\begin{aligned}
&\mathbf{DiffCilia}\big((A, S), (B, T)\big) \times \mathbf{DiffCilia}\big((B, T), (C, U)\big) \\
&= \mathbf{DiffSys}\big([\![Ay^S, By^T]\!]\big) \times \mathbf{DiffSys}\big([\![By^T, Cy^U]\!]\big) \\
&\xrightarrow{\lambda} \mathbf{DiffSys}\big([\![Ay^S, By^T]\!] \otimes [\![By^T, Cy^U]\!]\big) \\
&\xrightarrow{\mathbf{DiffSys}(\mathsf{c})} \mathbf{DiffSys}\big([\![Ay^S, Cy^U]\!]\big) = \mathbf{DiffCilia}\big((A, S), (C, U)\big)
\end{aligned}
$$

where λ is the laxator of Proposition 6.3.18 and c is the external composition morphism of Definition 6.3.6.

The monoidal structure $(\otimes, y)$ on $\mathbf{DiffCilia}$ is similarly defined following that of $\mathbf{Cilia}_{\mathcal{P}}^{\mathbb{T}}$. On 0-cells, $(A, S) \otimes (A', S') := (A \otimes A', S \otimes S')$. On 1-cells $(A, S) \to (B, T)$ and $(A', S') \to (B', T')$ (and their 2-cells), the tensor is given by the functors

$$
\begin{aligned}
&\mathbf{DiffCilia}\big((A, S), (B, T)\big) \times \mathbf{DiffCilia}\big((A', S'), (B', T')\big) \\
&= \mathbf{DiffSys}\big([\![Ay^S, By^T]\!]\big) \times \mathbf{DiffSys}\big([\![A'y^{S'}, B'y^{T'}]\!]\big) \\
&\xrightarrow{\lambda} \mathbf{DiffSys}\big([\![Ay^S, By^T]\!] \otimes [\![A'y^{S'}, B'y^{T'}]\!]\big) \\
&\xrightarrow{\mathbf{DiffSys}(\mathsf{d})} \mathbf{DiffSys}_{\mathcal{P}}\big([\![Ay^S \otimes A'y^{S'}, By^T \otimes B'y^{T'}]\!]\big) \\
&= \mathbf{DiffCilia}\big((A, S) \otimes (A', S'), (B, T) \otimes (B', T')\big)
\end{aligned}
$$

where d is the distributive law of Definition 6.3.7.

Following Proposition 6.3.9, we have the following characterization of a differential hierarchical inference system $(A, S) \to (B, T)$ in $\mathcal{K}\ell(\mathcal{P})$.

Proposition 6.3.20. A 1-cell $\delta : (A, S) \to (B, T)$ in **DiffCilia** is given by a tuple $\delta := (X, \delta_1^o, \delta_2^o, \delta^\sharp)$ of

- a choice of state space X : **Euc**;

- a forwards output map $\delta_1^o : \widetilde{X} \times A \to \mathcal{P}B$,

- a backwards output map $\delta_2^o : \widetilde{X} \times \mathcal{P}A \times T \to \mathcal{P}S$,

- a stochastic vector field $\delta^\sharp : \widetilde{X} \times \mathcal{P}A \times T \to \mathcal{P}\widetilde{\mathsf{T}X}$.

At least for deterministic differential systems, we can obtain continuous-time dynamical systems from differential systems by integration. We may then discretize these flows to give discrete-time dynamical systems.

Proposition 6.3.21. For the purposes of this proposition, let $\mathcal{P}$ be the identity monad on a finitely complete category $\mathcal{E}$ of manifolds, let $(\widetilde{-})$ be the corresponding inclusion **Euc** $\hookrightarrow \mathcal{E}$, and let **DiffSys** be instantiated accordingly. Then integration induces an indexed functor Flow : **DiffSys** $\to$ **Coalg**$_{\mathcal{P}}^{\mathbb{R}}$.

Proof. Suppose (M, m) is an object in **DiffSys**(p). The morphism $m : \widetilde{M}y^{\widetilde{\mathsf{T}M}} \to p$ consists of functions $m_1 : \widetilde{M} \to p(1)$ and $m^\sharp : \sum_{x:\widetilde{M}} p[m_1(x)] \to \widetilde{\mathsf{T}M}$. Since, for any section $\sigma : p \to y$, the induced map $m^\sigma : \widetilde{M} \to \widetilde{\mathsf{T}M}$ is a vector field on a compact manifold, it generates a unique global flow Flow$(p)(m)^\sigma : \mathbb{R} \times \widetilde{M} \to \widetilde{M}$ [167, Thm.s 12.9, 12.12], which factors as

$$\sum_{t:\mathbb{R}} \widetilde{M} \xrightarrow{m_1^* \sigma} \sum_{t:\mathbb{R}} \sum_{x:\widetilde{M}} p[m_1(x)] \xrightarrow{\mathsf{Flow}(p)(m)^u} \widetilde{M} \, .$$

We therefore define the system Flow$(p)(m)$ to have state space $\widetilde{M}$, output map m_1 (for all $t : \mathbb{R}$), and update map Flow$(p)(m)^u$. Since Flow$(p)(m)^\sigma$ is a flow for any section σ, it immediately satisfies the monoid action condition. On morphisms $\alpha : m \to m'$, we define Flow$(p)(\alpha)$ by the same underlying map on state spaces; this is again well-defined by the condition that α is compatible with the tangent structure. Given a morphism $\varphi : p \to q$ of polynomials, both the reindexing **DiffSys**(φ) and **Coalg**$_{\mathcal{P}}^{\mathbb{R}}(\varphi)$ act by postcomposition, and so it is easy to see that **Coalg**$_{\mathcal{P}}^{\mathbb{R}}(\varphi) \circ \mathsf{Flow}(p) \cong \mathsf{Flow}(q) \circ \mathbf{DiffSys}_{\mathcal{P}}(\varphi)$ naturally. $\square$

Remark 6.3.22. The question of integration of *stochastic* systems is more vexed and we will not treat it in this thesis.

Not only may we integrate a differential system to obtain a continuous-time dynamical system, we can also variously discretize the continuous-time system to obtain a discrete-time one.

Proposition 6.3.23. Any map $f : \mathbb{T}' \to \mathbb{T}$ of monoids induces an indexed functor (a natural transformation) $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}} \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}'}$.

Proof. We first consider the induced functor $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}(p) \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}'}(p)$, which we denote by Δ_f^p. Note that we have a morphism $[fy, p] : [\mathbb{T}y, p] \to [\mathbb{T}'y, p]$ of polynomials by substitution (precomposition). A system β in $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}$ is a morphism $Sy^S \to [\mathbb{T}y, p]$ for some S, and so we define $\Delta_f^p(\beta)$ to be $[f, p] \circ \beta : Sy^S \to [\mathbb{T}y, p] \to [\mathbb{T}'y, p]$. To see that this satisfies the monoid action axiom, consider that the closure $\Delta_f^p(\beta)^\sigma$ for any section $\sigma : p \to y$ is given by

$$\sum_{t:\mathbb{T}'} S \xrightarrow{\beta^o(f(t))^*\sigma} \sum_{t:\mathbb{T}'}\sum_{s:S} p[\beta^o(f(t), s)] \xrightarrow{\beta^u} S$$

which is an object in the functor category $\mathbf{Cat}\big(\mathbf{B}\mathbb{T}', \mathcal{K}\ell(\mathcal{P})\big)$ since f is a monoid homomorphism. On morphisms of systems, the functor Δ_f^p acts trivially.

To see that Δ_f collects into an indexed functor, consider that it is defined on each polynomial p by the contravariant action $[f, p]$ of the internal hom $[-, =]$, and that the reindexing $\mathbf{Coalg}^{\mathbb{T}}(\varphi)$ for any morphism φ of polynomials is similarly defined by the covariant action $[\mathbb{T}y, \varphi]$. By the bifunctoriality of $[-, =]$, we have $[\mathbb{T}'y, \varphi] \circ [fy, p] = [fy, \varphi] = [fy, q] \circ [\mathbb{T}y, \varphi]$, and so $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}'}(\varphi) \circ \Delta_f^p = \Delta_f^q \circ \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{T}}$. $\qquad\square$

Corollary 6.3.24. For each $k : \mathbb{R}$, the canonical inclusion $\iota_k : \mathbb{N} \hookrightarrow \mathbb{R} : i \mapsto ki$ induces a corresponding 'discretization' indexed functor $\mathsf{Disc}_k := \Delta_\iota : \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{R}} \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{N}}$.

Remark 6.3.25. From Proposition 6.3.21 and Corollary 6.3.24 we obtain a family of composite indexed functors $\mathbf{DiffSys} \xrightarrow{\mathsf{Flow}} \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{R}} \xrightarrow{\mathsf{Disc}_k} \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{N}}$ taking each differential system to a discrete-time dynamical system in $\mathcal{C}$. Below, we will define approximate inference doctrines in discrete time that arise from processes of (stochastic) gradient descent, and which therefore factor through differential systems, but the form in which these are given—and in which these are found in the informal literature (*e.g.*, Bogacz [33])—is not obtained via the composite $\mathsf{Disc}_k \circ \mathsf{Flow}$ for any k, even though there is a free parameter k that plays the same rôle (intuitively, a 'learning rate'). Instead, one typically adopts the following scheme, sometimes known as *Euler integration* or the *Euler method*.

Euler integration induces a family of indexed functors $\mathsf{Euler}_k : \mathbf{DiffSys} \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{N}}$, for $k : \mathbb{R}$, which we illustrate for a single system $(\mathbb{R}^n, m)$ over a fixed polynomial p, with $m : \mathbb{R}^n y^{\mathbb{R}^n \times \mathbb{R}^n} \to p$. This system is determined by a pair of morphisms $m_1 : \mathbb{R}^n \to p(1)$ and $m^\sharp : \sum_{x:\mathbb{R}^n} p[m_1(x)] \rightarrowtail \mathbb{R}^n \times \mathbb{R}^n$, and we can write the action of $m^\sharp$ as $(x, y) \mapsto (x, v_x(y))$.

Using these, we define a discrete-time dynamical system β over p with state space $\mathbb{R}^n$. This β is given by an output map β^o, which we define to be equal to m_1, $\beta^o := m_1$, and an update map $\beta^u : \sum_{x:\mathbb{R}^n} p[\beta^o(x)] \rightarrowtail \mathbb{R}^n$, which we define by $(x, y) \mapsto x + k \, v_x(y)$. Together, these define a system in $\mathbf{Coalg}_{\mathcal{P}}^{\mathbb{N}}(p)$, and the collection of these systems β produces an indexed functor by the definition $\mathsf{Euler}_k(p)(m) := \beta$.

By contrast, the discrete-time system obtained via $\mathsf{Disc}_k \circ \mathsf{Flow}$ involves integrating a continuous-time system for k units of real time for each unit of discrete time: although this in general produces a more accurate simulation of the trajectories implied by the vector field, it is computationally more arduous; to trade off simulation accuracy against computational feasibility, one may choose a more sophisticated discretization scheme than that sketched above, or at least choose a "sufficiently small" timescale k.

Finally, we can use the foregoing ideas to translate differential hierarchical inference systems to dynamical hierarchical inference systems.

Corollary 6.3.26. The indexed functors $\mathsf{Disc}_k : \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{R}} \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{N}}$, $\mathsf{Flow} : \mathbf{DiffSys} \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{R}}$, and $\mathsf{Euler}_k : \mathbf{DiffSys} \to \mathbf{Coalg}_{\mathcal{P}}^{\mathbb{N}}$ induce functors (respectively) $\mathbf{HDisc}_k : \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{R}} \to \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{N}}$, $\mathbf{HFlow} : \mathbf{DiffCilia} \to \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{R}}$ and $\mathbf{HEuler}_k : \mathbf{DiffCilia} \to \mathbf{Cilia}_{\mathcal{P}}^{\mathbb{N}}$ by change of base of enrichment.

7. Approximate inference doctrines for predictive coding

The construction of the predictive coding models that underlie the theory of the Bayesian brain involves mapping a ('generative') statistical model, representing how the modeller believes the brain to understand the world, to a dynamical system which plays the rôle of the neural circuits which are hypothesized to instantiate that model. This dynamical system is then simulated and the resulting trajectories studied: for instance, to compare with experimental neural or psychological data, or to judge against a synthetic benchmark.

Typically, both the generative model and the resulting dynamical systems are 'modular', and the mapping from the former to the latter preserves this structure: that is to say, predictive coding forms an example of functorial semantics, of which we saw a rudimentary example in §3.3, when we considered an algebra of rate-coded neural circuits. This chapter makes this functoriality explicit, which we hope to have a useful scientific consequence: it often seems to be the case that researchers manually derive complicated dynamical systems from their statistical models [21, 48, 76, 108, 148, 264, 265] [205, Chapter 5], but once functoriality is established, this manual labour is unnecessary; the functor represents a machine with which the process may be automated.

We call such functors *approximate inference doctrines*. In defining them, we bring together the statistical games of Chapter 5 (which supply the 'syntax' of generative models) and the open dynamical systems of Chapter 6 (which supply the 'semantics'), and we explain precisely how these doctrines may factorize through the various components we have seen: the stochastic channels, the inference systems, the loss models, the differential systems, and the cilia. This is the work of §7.3, which also establishes the functoriality of predictive coding under the (Laplacian) free energy principle. Before we get there, we construct some final pieces of technical machinery, aspects of which we have seen before: stochastic channels with Gaussian noise, to model functions of the form $f(x) + \omega$ with ω Gaussian-distributed (§7.1); and externally parameterized Bayesian lenses, so that our constructions have the freedom to learn (§7.2).

7.1. Channels with Gaussian noise

Our motivating examples from the predictive coding literature in computational neuroscience are defined over a subcategory of channels between Cartesian spaces with additive Gaussian noise [33, 48, 101]; typically one writes $x \mapsto f(x) + \omega$, with $f : X \to Y$ a deterministic map and ω sampled from a Gaussian distribution over Y. This choice is made, as we saw in §5.3.3.4, because it permits some simplifying assumptions, and the resulting dynamical systems resemble known neural circuits.

In this section, we develop some categorical language in which we can express such Gaussian channels, expanding on the informal definition given in Remark 5.3.29. We do so by thinking of $x \mapsto f(x) + \omega$ as a map parameterized by a noise source, and so to construct a category of such channels, we can use the **Para** construction, following Proposition 3.2.3. Because the noise comes from the parameter, we need a category whose objects are spaces equipped with measures. For this, we can use the 'pointing' construction introduced in §6.2.3; as we saw in Example 6.2.25, this gives us a category of measure spaces. The next step is to spell out an actegory structure that induces the parameterization we seek.

Proposition 7.1.1. Suppose $(\mathcal{C}, \otimes, I)$ is a monoidal category, and suppose $\mathcal{D} \hookrightarrow \mathcal{C}$ is a subcategory to which the monoidal structure restricts. Then there is a $\mathcal{D}_*$-actegory structure $\mathcal{D}_* \to \mathbf{Cat}(\mathcal{D}, \mathcal{D})$ on $\mathcal{D}$ as follows. For each $(M, \mu) : \mathcal{D}_*$, define $(M, \mu) * (-) : \mathcal{D} \to \mathcal{D}$ by $(M, \mu) * X := M \otimes (-)$. For each morphism $f : (M, \mu) \to (N, \nu)$ in $\mathcal{D}_*$, define $f * (-) := f \otimes (-)$.

Proof sketch. The action on morphisms is well-defined because each morphism $f : (M, \mu) \to (N, \nu)$ in $\mathcal{D}_*$ projects to a map $f : M \to N$ in $\mathcal{D}$; it is clearly functorial. The unitor and associator of the actegory structure are inherited from the monoidal structure, and they satisfy the actegory axioms for the same reason. $\square$

Remark 7.1.2. Note that the construction of $*$ is easily extended to an action on the whole of $\mathcal{C}$. We will however be concerned only with the action of $\mathcal{D}_*$ on $\mathcal{D}$.

When we instantiate $*$ in the context of $\mathbf{Meas} \hookrightarrow \mathbf{sfKrn}$, the resulting **Para** bicategory $\mathbf{Para}(*)$ can be thought of as a bicategory of maps each of which is equipped with an independent noise source; the composition of maps takes the product of the noise sources, and 2-cells are noise-source reparameterizations.

In this case, the actegory structure $*$ is moreover symmetric monoidal, and the 1-categorical truncation $\mathbf{Para}(*)_1$ (*cf.* Proposition 3.2.8) is a copy-delete category, as we now sketch.

Proposition 7.1.3. Suppose $(\mathcal{C}, \otimes, I)$ is a symmetric monoidal copy-discard category, and let the symmetry and copy-discard structure restrict to $\mathcal{D} \hookrightarrow \mathcal{C}$. Then $\mathbf{Para}(*)_1$ is also a symmetric monoidal copy-delete category.

Proof sketch. The monoidal structure is defined following Proposition 3.2.5. We need to define a right costrength ρ with components $(N, \nu) * (X \otimes Y) \xrightarrow{\sim} X \otimes ((N, \nu) * Y)$. Since $*$ is defined by forgetting the pointing and taking the monoidal product, the costrength is given by the associator and symmetry in $\mathcal{D}$:

$$(N, \nu)*(X\otimes Y) = N\otimes(X\otimes Y) \xrightarrow{\sim} N\otimes(Y\otimes X) \xrightarrow{\sim} (N\otimes Y)\otimes X \xrightarrow{\sim} X\otimes(N\otimes Y) = X\otimes((N,\nu)*Y)$$

As the composite of natural isomorphisms, this definition gives again a natural isomorphism; the rest of the monoidal structure follows from that of the monoidal product on $\mathcal{C}$.

We now need to define a symmetry natural isomorphism $\beta_{X,Y} : X \otimes Y \xrightarrow{\sim} Y \otimes X$ in $\mathbf{Para}(*)$. This is given by the symmetry of the monoidal product in $\mathcal{D}$, under the embedding of $\mathcal{D}$ in $\mathbf{Para}(*)$ that takes every map to its parameterization by the monoidal unit.

The rest of the copy-delete structure is inherited similarly from $\mathcal{C}$ via $\mathcal{D}$. $\qquad\square$

When $\mathcal{C}$ is a category of Markov kernels, we will typically think of the morphisms of $\mathbf{Para}(*)_1$ as kernels whose uncertainty arises from a noisy parameter. To formalize this we can push forward the noise to obtain again a morphism in $\mathcal{C}$. This yields a functor $\mathsf{Push} : \mathbf{Para}(*)_1 \to \mathcal{C}$.

Proposition 7.1.4. There is a strict monoidal functor $\mathsf{Push} : \mathbf{Para}(*)_1 \to \mathcal{C}$. Given a morphism in $\mathbf{Para}(*)_1$ represented by $f : X \xrightarrow{(\Omega,\mu)} Y$, let $\mathsf{Push}(f)$ be the composite $f \bullet (\mu \otimes \mathsf{id}_X) : X \to Y$ in $\mathcal{C}$.

Proof sketch. First, the given mapping preserves identities: the identity in $\mathbf{Para}(*)$ is trivially parameterized, and is therefore taken to the identity in $\mathcal{C}$. The mapping also preserves composites, by the naturality of the unitors of the symmetric monoidal structure on $\mathcal{C}$. That is, given $f : X \xrightarrow{(\Omega,\mu)} Y$ and $g : Y \xrightarrow{(\Theta,\nu)} Z$, their composite $g \circ f : X \xrightarrow{(\Theta\otimes\Omega,\nu\otimes\mu)} Z$ is taken to

$$X \xrightarrow{\sim} 1 \otimes 1 \otimes X \xrightarrow{\nu\otimes\mu\otimes\mathsf{id}_X} \Theta \otimes \Omega \otimes X \xrightarrow{g\circ f} Z$$

where here $g \circ f$ is treated as a morphism in $\mathcal{C}$. Composing the images of g and f under the given mapping gives

$$X \xrightarrow{\sim} 1 \otimes X \xrightarrow{\mu\otimes\mathsf{id}_X} \Omega \otimes X \xrightarrow{f} Y \xrightarrow{\sim} 1 \otimes Y \xrightarrow{\nu\otimes Y} \Theta \otimes Y \xrightarrow{g} Z$$

which is equal to

$$X \xrightarrow{\sim} 1 \otimes 1 \otimes X \xrightarrow{\nu \otimes \mu \otimes \mathrm{id}_X} \Theta \otimes \Omega \otimes X \xrightarrow{\mathrm{id}_\Theta \otimes f} \Theta \otimes Y \xrightarrow{g} Z$$

which in turn is equal to the image of the composite above.

Since the monoidal structure on $\mathbf{Para}(*)$ is inherited from that on $\mathcal{C}$ (with identical objects), the embedding is strict monoidal. $\qquad\square$

Remark 7.1.5. Note that Push is not an embedding, since the mapping on hom sets need not be injective: pushing forward the noise of two parallel morphisms in $\mathbf{Para}(*)_1$ may yield equal morphisms in $\mathcal{C}$ without the noise sources being isomorphic, and hence without the original morphisms being equivalent in $\mathbf{Para}(*)$; that is to say, the parameterization of noise sources is not generally unique.

We now restrict our attention to Gaussian morphisms in $\mathcal{C} = \mathbf{sfKrn}$.

Definition 7.1.6. We say that $f : X \nrightarrow Y$ in $\mathbf{sfKrn}$ is *Gaussian* if, for any $x : X$, the measure $f(x)$ is Gaussian[1]. Similarly, we say that $f : X \xrightarrow{(\Omega, \mu)} Y$ in $\mathbf{Para}(*)$ is Gaussian if its image under Push is Gaussian. We will write $\mathbf{Gauss}$ to denote the subcategory of $\mathbf{sfKrn}$ generated by Gaussian kernels and their composites; likewise, we will write $\mathbf{Gauss}_*$ to denote the Gaussian subcategory of $\mathbf{Para}(*)$. Given a separable Banach space X, we will write $\mathbf{Gauss}(X)$ for the space of Gaussian states on X.

Example 7.1.7. Random functions of the form $x \mapsto f(x) + \omega$, where $\omega : \Omega$ is distributed according to a Gaussian, are therefore morphisms in $\mathbf{Gauss}_*$. Under the embedding into $\mathbf{Gauss}$, the corresponding kernel emits, for each $x : X$, a Gaussian distribution with mean $f(x) + \mu_\omega$, where μ_ω is the mean of the Gaussian random variable ω, and variance the same as that of ω.

Remark 7.1.8. In general, Gaussian morphisms are not closed under composition: pushing a Gaussian distribution forward along a nonlinear transformation will not generally result in another Gaussian. For instance, consider the Gaussian functions $x \mapsto f(x) + \omega$ and $y \mapsto g(y) + \omega'$. Their composite in $\mathbf{Gauss}_*$ is the morphism $x \mapsto g\big(f(x) + \omega\big) + \omega'$; even if $g\big(f(x) + \omega\big)$ is Gaussian-distributed, the sum of two Gaussians is in general not Gaussian, and so $g\big(f(x) + \omega\big) + \omega'$ will not be Gaussian. This non-closure underlies the power of statistical models such as the variational autoencoder, which are often constructed by pushing a Gaussian forward along a

---

[1] We admit Dirac delta distributions, and therefore deterministic morphisms, as Gaussian, since delta distributions can be seen as Gaussians with infinite precision.

learnt nonlinear transformation [155], in order to approximate an unknown distribution; since sampling from Gaussians is relatively straightforward, this method of approximation can be computationally tractable. The **Gauss** construction here is an abstraction of the Gaussian-preserving transformations of Shiebler [232], and is to be distinguished from the category with the same name introduced by Fritz [109], whose morphisms are affine transformations (which do preserve Gaussianness) and which *are* therefore closed under composition; there is nonetheless an embedding of Fritz's **Gauss** into our **Gauss**.

For Laplacian statistical games (in the image of LFE), and for the associated approximate inference doctrines, we are interested only in Gaussian channels between finite-dimensional Cartesian spaces $\mathbb{R}^n$ for $n : \mathbb{N}$.

Definition 7.1.9. Denote by **FdGauss** the full subcategory of **Gauss** spanned by the objects $\mathbb{R}^n$ for $n : \mathbb{N}$.

Proposition 7.1.10. Every channel $c : X \nrightarrow Y$ in **FdGauss** admits a density function $p_c : Y \times X \to [0, 1]$ with respect to the Lebesgue measure on Y. Moreover, since $Y = \mathbb{R}^n$ for some $n : \mathbb{N}$, this density function is determined by two maps: the *mean* $\mu_c : X \to \mathbb{R}^n$, and the *covariance* $\Sigma_c : X \to \mathbb{R}^{n \times n}$ in $\mathcal{E}$. We call the pair $(\mu_u, \Sigma_c) : X \to \mathbb{R}^n \times \mathbb{R}^{n \times n}$ the *statistical parameters* of c (to be distinguished from any parameterization in the category-theoretic sense of §3.2).

Proof. The density function $p_c : Y \times X \to [0, 1]$ is defined by

$$\log p_c(y|x) = \frac{1}{2} \left\langle \epsilon_c(y, x), \Sigma_c(x)^{-1} \epsilon_c(y, x) \right\rangle - \log \sqrt{(2\pi)^n \det \Sigma_c(x)}$$

where the 'error' function $\epsilon_c : Y \times X \to Y$ is defined by $\epsilon_c(y, x) := y - \mu_c(x)$. □

7.2. Externally parameterized Bayesian lenses and statistical games

The statistical games of Chapter 5 are simply Bayesian lenses equipped with loss functions. Given a statistical game, its lens is therefore fixed, and the only way to a high score on its loss is through its openness to the environment—the dependence on a prior and an observation. But this seems like a strange model of adaptive or cybernetic systems, which should also be free to change themselves in order to improve their performance.

Indeed, this changing-oneself is at the heart of the construction of approximate inference doctrines, and in order to incorporate it into the structure, there must be some more freedom in

the model: the freedom to choose the lens. This freedom is afforded by the use of parameterized statistical games, and in particular, *externally* parameterized statistical games, in the sense of §3.2.2.

Remark 7.2.1. It is of course possible to define an actegorical (internal) parameterization of statistical games, but this seems to prove more complicated than necessary for our purposes.

In advance of our use of external parameterization in the construction of approximate inference doctrines, recall that we denote the external parameterization of an enriched category $\mathcal{C}$ in its base of enrichment $\mathcal{E}$ by $\mathbf{P}\mathcal{C}$. This section is dedicated to exhibiting the external parameterizations $\mathbf{PBayesLens}_2$ and $\mathbf{PSGame}$ of Bayesian lenses and statistical games, and the notion of parameterized loss model.

Remark 7.2.2. Because $\mathbf{BayesLens}_2$ and $\mathbf{SGame}$ are both bicategories, they are weakly enriched in $\mathbf{Cat}$. Consequently, following Remark 3.2.12, $\mathbf{P}$ has the type $\mathbf{Cat}\text{-}\mathbf{Cat} \to (\mathbf{Cat}\text{-}\mathbf{Cat})\text{-}\mathbf{Cat}$, or equivalently, $\mathbf{Bicat} \to \mathbf{Tricat}$. This means that, in full generality, $\mathbf{PBayesLens}_2$ and $\mathbf{PSGame}$ are tricategories: if $\mathcal{B}$ is a bicategory, then the hom-bicategory $\mathbf{P}\mathcal{B}(a,b)$ is the bicategory $\mathbf{Cat}/\mathcal{B}(a,b)$. Because we are now working with weakened structures (weak enrichment, bicategories, lax loss models), we take this to be a *lax* slice of $\mathbf{Cat}$.

We pause to define this new notion, generalizing our earlier Definition 3.2.10 (slice category).

Definition 7.2.3. Suppose X is a 0-cell in a bicategory $\mathcal{B}$. The *lax slice* of $\mathcal{B}$ over X, denoted $\mathcal{B}/X$, is the following bicategory. Its 0-cells are pairs (A, p) where A is a 0-cell and p is a 1-cell $A \to X$ in $\mathcal{B}$. A 1-cell $(A, p) \to (B, q)$ is a pair (f, ϕ) where f is a 1-cell $A \to B$ and ϕ is a 2-cell $p \Rightarrow q \circ f$ in $\mathcal{B}$, as in the diagram

$$
\begin{array}{ccc}
A & \xrightarrow{\quad f \quad} & B \\
 & \underset{p}{\searrow} \overset{\phi}{\Longrightarrow} \underset{q}{\swarrow} & \\
 & X &
\end{array}
\quad .
$$

A 2-cell $(f, \phi) \Rightarrow (g, \gamma)$ is a 2-cell $\alpha : f \Rightarrow g$ in $\mathcal{B}$ such that

$$
p \xrightarrow{\phi} q \circ f \xrightarrow{q \circ \alpha} g \circ g \quad = \quad p \xrightarrow{\gamma} q \circ g \quad .
$$

(In this definition, $\circ$ denotes horizontal composition in $\mathcal{B}$.) The horizontal composition in $\mathcal{B}/X$ is given in the obvious way by the horizontal composition of the relevant 1- and 2-cells. Likewise, vertical composition in $\mathcal{B}/X$ is vertical composition in $\mathcal{B}$. (It is easy to check that these definitions all satisfy the relevant axioms, hence constituting a valid bicategory.)

We will see how this structure works in practice in our examples of parameterization below.

Remark 7.2.4. To avoid venturing into 3- and 4-dimensional category theory, we will restrict the hom-bicategories of $\mathbf{PBayesLens}_2$ and $\mathbf{PSGame}$ to be *locally discrete*, with the parameterizing objects being mere sets (treated as discrete categories). Strictly speaking, our parameterizing sets will be the underlying sets of differential manifolds — specifically, the trivial manifolds $\mathbb{R}^n$ — and we could treat them properly as parameterizing categories by using their groupoidal (path) structure, but we do not pursue this here. (Alternatively, we could follow the idea of Proposition 3.2.8 and truncate the hom-categories by quotienting by connected components; but this turns the 1-cells into equivalence classes of functors, which are again more complicated than we have the need or appetite for here.)

Restricting $\mathbf{P}$ to discrete parameterization means that we instantiate $\mathbf{PBayesLens}_2$ and $\mathbf{PSGame}$ as follows. Both being constructed over $\mathbf{Copara}_2(\mathcal{C})$, we build up from $\mathbf{PCopara}_2(\mathcal{C})$, after first sketching the horizontal composition of externally parameterized bicategories.

Remark 7.2.5. Given a bicategory $\mathcal{B}$, horizontal composition in $\mathbf{P}\mathcal{B}$ is obtained from the strong monoidal structure of the covariant self-indexing (which follows from the universal property of the product of categories) and the horizontal composition in $\mathcal{B}$. For each triple of 0-cells $a, b, c : \mathcal{B}$, the composition pseudofunctor is given by

$$\mathbf{P}\mathcal{B}(b,c) \times \mathbf{P}\mathcal{B}(a,b) = \mathbf{Cat}/\mathcal{B}(b,c) \times \mathbf{Cat}/\mathcal{B}(a,b) \cdots$$

$$\cdots \xrightarrow{\sim} \mathbf{Cat}/\big(\mathcal{B}(b,c) \times \mathcal{B}(a,b)\big) \xrightarrow{\mathbf{Cat}/\circ_{a,b,c}} \mathbf{Cat}/\mathcal{B}(a,c) = \mathbf{P}\mathcal{B}(a,c) \ .$$

Observe that this generalizes the lower-dimensional case of Definition 3.2.11: first, we take the product of the parameterizing functors, and then we compose in their codomain.

Example 7.2.6. The 0-cells of $\mathbf{PCopara}_2(\mathcal{C})$ are the 0-cells of $\mathbf{Copara}_2(\mathcal{C})$, which are in turn the objects of $\mathcal{C}$. A 1-cell from X to Y is a choice of (discrete) parameterizing category (hence a set) Θ, along with a functor $\Theta \to \mathbf{Copara}_2(\mathcal{C})(X,Y)$. More intuitively, we can think of such a 1-cell as a morphism in $\mathcal{C}$ that is both (externally) parameterized and (internally) coparameterized, and write it as $f : X \xrightarrow[M]{\Theta} Y$, denoting a 1-cell with parameter Θ (in the base of enrichment of $\mathcal{C}$), domain X, codomain Y, and coparameter M.

A 2-cell from $f : X \xrightarrow[M]{\Theta} Y$ to $f' : X \xrightarrow[M']{\Theta'} Y$ is a pair (ϕ, φ) of a functor $\phi : \Theta \to \Theta'$ and a natural transformation $\varphi : f \Rightarrow f' \circ \phi$. The functor ϕ changes the parameterization; and the

natural transformation φ permits additionally a compatible change of *co*parameterization, being given by a natural family of 2-cells in $\mathbf{Copara}_2(\mathcal{C})$

$$\varphi^\theta : \left(f^\theta : X \xrightarrow[M]{\theta} Y\right) \implies \left(f'^{\phi(\theta)} : X \xrightarrow[M']{\phi(\theta)} Y\right)$$

indexed by the parameters $\theta : \Theta$. (With discrete parameterization, such a family is trivially natural.) Recalling the definition of $\mathbf{Copara}_2$ in Theorem 5.2.1, this means that each component φ^θ corresponds to a morphism $X \otimes M \otimes Y \to N$ in $\mathcal{C}$ satisfying the change of coparameter axiom with respect to f^θ and $f'^{\phi(\theta)}$.

Horizontal composition in $\mathbf{PCopara}_2(\mathcal{C})$ is as sketched in Remark 7.2.5: given 1-cells $f : X \xrightarrow[M]{\Theta} Y$ and $g : Y \xrightarrow[N]{\Omega} Z$, their composite is the evident $g \circ f : X \xrightarrow[M]{\Theta} Y \xrightarrow[N]{\Omega} Z$ whose parameter is the product $\Omega \times \Theta$ and whose coparameter is the tensor of M and N. The horizontal composition of 2-cells is likewise by first forming the product of their parameters. Vertical composition in $\mathbf{PCopara}_2(\mathcal{C})$ is given by the horizontal composition in each lax slice hom (bi)category.

The structure of $\mathbf{PBayesLens}_2$ and $\mathbf{PSGame}$ follows the same pattern.

Example 7.2.7. The 0-cells of $\mathbf{PBayesLens}_2$ are the same pairs (X, A) as in $\mathbf{BayesLens}_2$. A 1-cell from (X, A) to (Y, B) is a biparameterized Bayesian lens: a pair (c, c') of a biparameterized forwards channel $c : X \xrightarrow[M]{\Theta} Y$ and a biparameterized inversion (state-dependent) channel $c' : B \xrightarrow[M']{\Theta;X} A$; here we have denoted the state-dependence and the parameterization together as $\Theta; X$. (Note that in all our examples, the forwards and backwards coparameters will be equal, *i.e.*, $M = M'$; *cf.* Remark 5.2.21 on dependent optics.)

A 2-cell from $(c, c') : (X, A) \xrightarrow[M,M']{\Theta} (Y, B)$ to $(d, d') : (X, A) \xrightarrow[N,N']{\Omega} (Y, B)$ is a triple $(\alpha, \alpha_1, \alpha')$ such that α is a functor $\Theta \to \Omega$, (α, α_1) is a 2-cell $c \Rightarrow d$ in $\mathbf{PCopara}_2(\mathcal{C})$ (*cf.* Example 7.2.6), and (α, α') is a 2-cell $c' \Rightarrow d'$ in $\mathbf{PStat}_2(X)(B, A)$. The latter means that α' is a family of 2-cells in $\mathbf{Copara}_2(\mathcal{C})(B, A)$

$$\alpha'^\theta_\pi : \left(c'^\theta_\pi : B \xrightarrow[M']{\theta;\pi} A\right) \implies \left(d'^{\alpha(\theta)}_\pi : B \xrightarrow[N']{\alpha(\theta);\pi} A\right)$$

natural in $\theta : \Theta$ and indexed by $\pi : \mathcal{C}(I, X)$. (The preceding example shows how this corresponds to an indexed natural family of change-of-coparameter morphisms in $\mathcal{C}$.)

Horizontal composition in $\mathbf{PBayesLens}_2$ is of course by taking the product of the parameters and then applying horizontal composition in $\mathbf{BayesLens}_2$; and vertical composition is horizontal composition in the lax slices making up each hom (bi)category.

Example 7.2.8. Statistical games are obtained by attaching loss functions to Bayesian lenses, and hence to understand parameterized statistical games having elaborated parameterized Bayesian lenses in the preceding example, it suffices to exhibit parameterized loss functions.

A parameterized statistical game $(X, A) \xrightarrow[M,M']{\Theta} (Y, B)$ consists of a parameterized Bayesian lens $(X, A) \xrightarrow[M,M']{\Theta} (Y, B)$ along with a parameterized loss function $B \xrightarrow{\Theta;X} I$ in $\mathbf{PStat}(X)(B, I)$. Since $\mathsf{Stat}(X)(B, I)$ is a discrete category, such a loss function is given by a function $\Theta_0 \to \mathsf{Stat}(X)(B, I)$, or equivalently (by the Cartesian closure of $\mathbf{Set}$) a function $\Theta_0 \times \mathcal{C}(I, X) \to \mathcal{C}(B, I)$. In the case where $\mathcal{C} = \mathbf{sfKrn}$, this means a function $\Theta_0 \times \mathbf{sfKrn}(1, X) \times B \to \mathbb{R}_+$ which is measurable in B.

A 2-cell from the parameterized statistical game $(c, c', K) : (X, A) \xrightarrow[M,M']{\Theta} (Y, B)$ to $(d, d', L) :$ $(X, A) \xrightarrow[N,N']{\Omega} (Y, B)$ is a quadruple $(\alpha, \alpha_1, \alpha', \tilde{\alpha})$ where $(\alpha, \alpha_1, \alpha')$ is a 2-cell of Bayesian lenses and $\tilde{\alpha}$ is a family of parameterized loss functions $B \xrightarrow{\Theta;X} I$ such that $K^\theta = L^{\alpha(\theta)} + \tilde{\alpha}^\theta$, naturally in $\theta : \Theta$.

Horizontal and vertical composition of parameterized statistical games and their 2-cells follow the pattern of the preceding examples.

Because $\mathbf{P}$ is functorial, we can consider parameterized versions of the inference systems and loss models that we introduced in §5.3.2. We can think of parameterization as introducing a 'hole' in a structure (such as an extra input to a system), and parameterized inference systems and loss models are inference systems and loss models that account for (and possibly modulate) such holes.

Example 7.2.9. Suppose $(\mathcal{D}, \ell)$ is an inference system in $\mathcal{C}$. $\mathbf{P}$ acts on the canonical inclusion $(-)^{\curvearrowleft} : \mathcal{D} \hookrightarrow \mathbf{Copara}_2^l(\mathcal{D})$ to return the inclusion $\mathbf{P}(-)^{\curvearrowleft} : \mathbf{P}\mathcal{D} \hookrightarrow \mathbf{PCopara}_2^l(\mathcal{D})$, which maps a parameterized channel $d : X \xrightarrow{\Theta} Y$ to its trivially coparameterized form $d : X \xrightarrow[I]{\Theta} Y$.

ℓ then maps a channel d to a lens $(d, \ell d)$. If d is parameterized by Θ, then its inversion ℓd under ℓ will be parameterized accordingly, so that the whole lens $(d, \ell d)$ has parameter Θ. This mapping is the action of the pseudofunctor $\mathbf{P}\ell : \mathbf{P}\mathcal{D}^{\curvearrowleft} \to \mathbf{PBayesLens}_2|_{\mathcal{D}}$, induced by the parameterization of ℓ.

However, in the next section, we will want approximate inference systems that do not just preserve an existing parameterization, but which also add to it, equipping (possibly parameterized) morphisms with inversions that may have their own distinct capacity for improvement or learning. For this reason, we make the following definition.

Definition 7.2.10. Suppose $(\mathcal{C}, \otimes, I)$ is a copy-delete category. A *parameterized inference system* in $\mathcal{C}$ is a pair $(\mathcal{B}, \ell)$ of a sub-bicategory $\mathcal{B} \hookrightarrow \mathbf{P}\mathcal{C}$ along with a (strong functorial) section $\ell : \mathcal{B}^\curlyvee \to$ $\mathbf{PBayesLens}_2|_\mathcal{B}$ of the restriction $\mathbf{P}(\pi_{\mathsf{Lens}})|_\mathcal{B}$ to $\mathcal{B}$ of the parameterized 2-fibration $\mathbf{P}(\pi_{\mathsf{Lens}}) :$ $\mathbf{PBayesLens}_2 \to \mathbf{PCopara}_2^l(\mathcal{C})$, where $\mathcal{B}^\curlyvee$ is the essential image of the restriction to $\mathcal{B}$ of the (parameterized) canonical lax inclusion $\mathbf{P}(-)^\curlyvee : \mathbf{P}\mathcal{C} \hookrightarrow \mathbf{PCopara}_2^l(\mathcal{C})$. We say *lax parameterized inference system* when ℓ is a lax functor.

A trivial example of a lax parameterized inference system is obtained by taking the parameters to be hom categories, and the choice functor to be the identity, as the following example shows.

Example 7.2.11. The following data define a lax parameterized inference system ℓ acting on the entirety of $\mathbf{P}\mathcal{C}$. First, let $\mathrm{P}(X, Y, M)$ denote the full subcategory of $\mathsf{Stat}_2(X)(Y, X)$ on those objects (state-dependent morphisms) with coparameter M. Then ℓ is defined as follows.

(i) Each 0-cell X is mapped to the 0-cell (X, X).

(ii) Each 1-cell $c : X \xrightarrow[M]{\Theta} Y$ is mapped to the parameterized lens $(c, c') : (X, X) \xrightarrow[M]{\Theta \times \mathrm{P}(X,Y,M)} (Y, Y)$ whose forward channel is chosen by

$$\Theta \times \mathrm{P}(X, Y, M) \xrightarrow{\mathsf{proj}_1} \Theta \xrightarrow{c} \mathbf{Copara}_2^l(\mathcal{C})(X, Y)$$

and whose inverse channel $c' : Y \xrightarrow[M]{\mathrm{P}(X,Y,M);X} X$ is chosen by

$$\Theta \times \mathrm{P}(X, Y, M) \xrightarrow{\mathsf{proj}_2} \mathrm{P}(X, Y, M) \hookrightarrow \mathsf{Stat}_2(X)(Y, X)$$

(iii) Each 2-cell $(a, \alpha) : \big(c : X \xrightarrow[M]{\Theta} Y\big) \Rightarrow \big(d : X \xrightarrow[M']{\Theta'} Y\big)$ is mapped to the 2-cell $(a \times \alpha_*, \alpha, \alpha)$, where α_* is the functor defined by post-composing with α taken as a family of 2-cells in $\mathbf{Copara}_2^r(\mathcal{C})$ and hence in $\mathrm{P}(X, Y, M)$.

Proof. First, we confirm that the mapping is well-defined on 1-cells (taking it to be evidently so on 0-cells): in general, the coparameters in $\mathbf{PCopara}_2(\mathcal{C})$ may depend on the parameters, but here the parameters arise from the embedding $(-)^\curlyvee : \mathbf{P}\mathcal{C} \to \mathbf{PCopara}_2(\mathcal{C})$. The only coparameters are therefore those that arise by copy-composition, and their type is thus not parameter-dependent. It is therefore legitimate to map a 1-cell $c : X \xrightarrow[M]{\Theta} Y$ to a lens with type $(X, X) \xrightarrow[M]{\Theta \times \mathrm{P}(X,Y,M)} (Y, Y)$.

Next, we check well-definedness on 2-cells. Note that the 2-cell $(a, \alpha) : \big(c : X \xrightarrow[M]{\Theta} Y\big) \Rightarrow \big(d : X \xrightarrow[M']{\Theta'} Y\big)$ in $\mathbf{PCopara}_2^l(\mathcal{C})$ is constituted by a family of morphisms $\alpha^\theta : X \otimes M \otimes Y \rightarrowtail M'$, and that a 2-cell $\big(Y \xrightarrow[M]{} X\big) \Rightarrow \big(Y \xrightarrow[M']{} X\big)$ in $\mathbf{Copara}_2^r(\mathcal{C})$ has an underlying morphism of the same

type; hence each α^θ witnesses such a 2-cell in $\mathbf{Copara}_2^r(\mathcal{C})$. In particular, for each $\pi : I \nrightarrow X$ in $\mathcal{C}$, and for each state-dependent $\rho : Y \xrightarrow[M]{X} X$, α^θ yields a 2-cell from ρ_π to
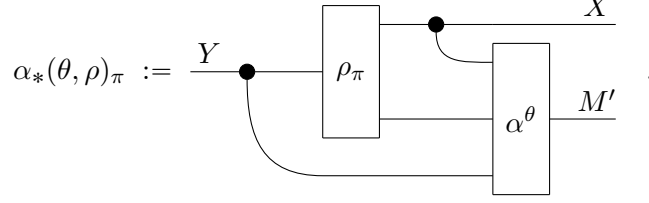
The functor α_* is thus defined by mapping $(\theta, \rho) : \Theta \times \mathrm{P}(X, Y, M)$ to $\alpha_*(\theta, \rho) : \mathrm{P}(X, Y, M')$; its own action on 2-cells is likewise by parameterized post-composition. Finally, note that d' is also given by evaluation, and so α also defines an indexed natural family of 2-cells

$$\alpha_\pi^{\theta,\rho} : \left(c_\pi'^\rho = \rho_\pi : Y \xrightarrow[M]{} X\right) \Rightarrow \left(d_\pi'^{\alpha_*(\theta,\rho)} = \alpha_*(\theta, \rho)_\pi : Y \xrightarrow[M']{} X\right)$$

as required (*cf.* Example 7.2.7). Therefore, $(a \times \alpha_*, \alpha, \alpha)$ defines a 2-cell in $\mathbf{PBayesLens}_2$. This is compatible with ℓ being a section of $\mathbf{P}\pi_{\mathsf{Lens}}$, as $(a \times \alpha_*, \alpha, \alpha) \mapsto (a, \alpha)$.

To establish lax unity, we need to exhibit a family of 2-cells $(i_X, i_{X1}, i_X') : \mathsf{id}_{(X,X)} \Rightarrow (\mathsf{id}_X, \mathsf{id}_X')$ natural in X, where $\mathsf{id}_{(X,X)}$ is the identity lens on (X, X) in $\mathbf{PBayesLens}_2$ with trivial parameter $\mathbf{1}$, id_X is the likewise trivially parameterized identity on X in $\mathbf{P}\mathcal{C}$, and id_X' is the parameterized state-dependent channel $\mathsf{id}_x' : X \xrightarrow[\mathbf{1}]{\mathbf{1} \times \mathrm{P}(X,X,1)} X$ defined by the inclusion

$$\mathbf{1} \times \mathrm{P}(X, X, 1) \xrightarrow{\sim} \mathrm{P}(X, X, 1) \hookrightarrow \mathsf{Stat}_2(X)(X, X) \ .$$

Clearly id_X' is not constantly the identity morphism, and this is why ℓ is only laxly unital. By defining the functor $i_X : \mathbf{1} \to \mathbf{1} \times \mathrm{P}(X, X, 1)$ to pick the element id_X, the 2-cell i_{X1} to be the identity on id_X, and likewise i_x', we obtain the required family of witnesses.

Lax functoriality is witnessed by a family of 2-cells

$$(f_{dc}, f_{dc1}, f_{dc}') : (d, d') \diamond (c, c') \Rightarrow (d \bullet c, (d \bullet c)')$$

natural in $c : X \xrightarrow[M]{\Theta} Y$ and $d : Y \xrightarrow[N]{\Phi} Z$. We define the functor

$$f_{dc} : \Theta \times \mathrm{P}(X, Y, M) \times \Phi \times \mathrm{P}(Y, Z, N) \to \Theta \times \Phi \times \mathrm{P}(X, Z, M \otimes N)$$

by composition, $f_{dc}(\theta, \rho, \phi, \chi) := (\theta, \phi, \rho \circ \chi_{c^\theta})$; it is the fact that not all morphisms $X \nrightarrow Z$ factor through Y that makes ℓ lax functorial. With f_{dc} so defined, we can set both f_{dc1} and f_{dc}' to be identity 2-cells, and thus obtain the requisite witnesses. $\qquad\square$

On the other hand, the only parameterized loss models we encounter will be those of §5.3.3 under the action of $\mathbf{P}$. This is because the ability to change is part of the system itself, rather than part of how we measure the system[2]: we do not seek to "move the goalposts". (In future work, we may consider systems whose performance is dependent on some broader context; but not here.) Therefore, our parameterized loss models will all be of the following form.

Example 7.2.12. If L is a loss model for $\mathcal{B}$, then its parameterization $\mathbf{P}L$ assigns to a parameterized Bayesian lens $(c, c') : (X, A) \xrightarrow[M,M']{\Theta} (Y, B)$ the correspondingly parameterized statistical game $(c, c', L(c))$. The parameterized loss function $L(c)$ thus also has parameter Θ and depends accordingly on it, with type $L(c) : B \xrightarrow{\Theta; X} I$. For each $\theta : \Theta$, its component is the loss function $L(c)^\theta : B \xrightarrow{X} I$ which is assigned to the lens (c^θ, c'^θ) by L (as a loss model applied to an unparameterized lens).

Remark 7.2.13. Before we move on to examples of approximate inference doctrines, let us note the similarity of the notions of externally parameterized lens (Example 7.2.7), cilia (Definition 6.3.8), and differential cilia (Definition 6.3.19): both of the latter can be considered as externally parameterized lenses with extra structure, where the extra structure is a morphism or family of morphisms back into (an algebra of) the parameterizing object: in the case of differential cilia, this 'algebra' is the tangent bundle; for (dynamical) cilia, it is trivial; and forgetting this extra structure returns a mere external parameterization. Notably, the 'input' on the external hom polynomial defining both types of cilia (Definition 6.3.1) corresponds precisely to the domain of the loss function of a statistical game; and so the domains of the update maps of either type of cilia correspond to the domains of parameterized loss functions. We will make use of this correspondence in defining approximate inference doctrines in the next section.

7.3. Approximate inference doctrines

We are at last in a position to build the bridge between abstract statistical models and the dynamical systems that play them: functors from a copy-discard category of parameterized channels to a category of cilia that factorize through an inference system (modelling how the system inverts the channels) and possibly a loss model (encoding how well the system is doing). In the round, we can think of the resulting approximate inference doctrines as "dynamical algebras" for categories of parameterized stochastic channels (considered as statistical models), which take the parameters

---

[2]Physically speaking, we adopt the 'Schrödinger' perspective on change rather than the 'Heisenberg' perspective.

as part of the dynamical state space so that they might improve themselves. This line of thinking leads us to the following definitions.

Definition 7.3.1. Let $(\mathcal{C}, \otimes, I)$ be a copy-discard category, and let $(\mathcal{B}, \ell)$ be a parameterized inference system in $\mathcal{C}$.

(a) An *approximate inference doctrine* is a pseudofunctor $\mathcal{B} \to \mathbf{Cilia}^{\mathbb{T}}$ that factors through ℓ, as

$$\mathcal{B} \xrightarrow{\mathbf{P}(-)^{\mathsf{Y}}|_{\mathcal{B}}} \mathcal{B}^{\mathsf{Y}} \xrightarrow{\ell} \mathsf{im}(\ell) \xrightarrow{\mathsf{D}} \mathbf{Cilia}^{\mathbb{T}} \ .$$

We say *lax approximate inference doctrine* if D is instead a lax functor.

(b) An *approximate inference doctrine with loss L* is an approximate inference doctrine along with a loss model L for $\mathsf{im}(\ell)$, a pseudofunctor $\mathsf{D}^L : \mathsf{im}(L) \to \mathbf{Cilia}^{\mathbb{T}}$, and an icon $\lambda : \mathsf{D} \Rightarrow \mathsf{D}^L \circ L$, as in the diagram

We say *lax approximate inference doctrine with loss* if L and D^L are lax functors.

(c) A *differential approximate inference doctrine with loss L* is an approximate inference doctrine with loss L such that D^L factors through a differential system, as in the diagram

We say *lax differential approximate inference doctrine* when L, ∇ and $\int$ are lax functors.

The different factors of a differential approximate inference doctrine with loss encode the different stages by which a dynamical system is constructed from a statistical model: the parameterized inference system ℓ equips a parameterized channel with a parameterized inversion; the loss model L equips the resulting lens with a loss function; the functor ∇ translates this statistical game to a

differential system, possibly representing gradient descent on the loss; and finally the functor $\int$ turns this differential system into a dynamical system that 'flows', possibly by integration.

With these definitions to hand, we come to our motivating neuroscientific examples. First (§7.3.1), we formalize predictive coding using the Laplace approximation to the free energy [21, 33, 104], which we saw in §5.3.3.4 forms a lax loss model for Gaussian lenses. This approximation allows the resulting dynamical systems to exhibit some biological plausibility, with prediction errors computed linearly and the dynamical updates obtained as affine transformations of prediction errors. We call this the *Laplace doctrine*.

Apart from requiring Gaussian channels, the Laplace doctrine is agnostic about how predictions are actually generated, and it does not produce systems which are able to improve their predictions; they have no 'synaptic' plasticity, and thus do not learn. To remedy this, our second example of an approximate inference doctrine (§7.3.2) is more opinionated about the predictive forward channels, restricting them to be of the form $x \mapsto \theta\,h(x) + \omega$ where θ is a square matrix on Y, h is a differentiable function $X \to Y$, and ω is distributed according to a Gaussian on Y; compare this with the form of the firing rate dynamics of rate-coded neural circuits in Definition 3.3.10. The 'synaptic' parameter (or weight matrix) θ can then be learnt, and this is incorporated into the state space of the systems produced by the corresponding *Hebb-Laplace doctrine*, which formalizes another standard scheme in the neuroscience literature [33]. The name of this doctrine indicates another aspect of the biological plausibility of this scheme: the θ-updates can be seen as a form of Hebbian learning [127].

Remark 7.3.2. In what follows, in order to focus on exemplification, we omit a full treatment of all the higher-categorical structure. This means that we do not consider the action of the doctrines on 2-cells, and leave leave the full elaboration of the 2-dimensional structure to future work. Our main concern in this final part is the scientific question of the compositional structure of predictive coding, and one further mathematical consequence of this is that the inference systems on which the doctrines are based will not be unital: the schemes that are presented in the literature involve mappings which do not preserve identity channels.

7.3.1. Predictive coding circuits and the Laplace doctrine

Notation 7.3.3. Any category $\mathcal{C}$ embeds into its external parameterization $\mathbf{P}\mathcal{C}$ by mapping every morphism to its trivially parameterized form; in a mild abuse of notation, we will denote the image of this embedding simply by $\mathcal{C}$. In this section, we will work with the trivial parameterization of

the subcategory **FdGauss** of **sfKrn** of Gaussian kernels between finite-dimensional Cartesian spaces (Definition 7.1.9). Hence, when we write **FdGauss**$^{\curlyvee}$, it denotes the image of **FdGauss** under $\mathbf{P}(-)^{\curlyvee}$.

We begin by presenting the action of the Laplace doctrine on a (non-coparameterized[3]) Gaussian channel c. Below, we will see how the resulting system is obtained from a differential approximate inference doctrine with the Laplacian free energy loss.

Proposition 7.3.4. Suppose $c : X \nrightarrow Y$ is a morphism in **FdGauss**, and fix a "learning rate" $\lambda : \mathbb{R}$. Then the following data define a system $\mathsf{L}_\lambda(c) : (X, X) \rightarrow (Y, Y)$ in **Cilia**$^{\mathbb{N}}$, following the representation of Proposition 6.3.9.

(i) the state space is X;

(ii) the forwards output map $\mathsf{L}_\lambda(c)_1^o : X \times X \rightarrow \mathbf{Gauss}(Y)$ is defined by

$$\mathsf{L}_\lambda(c)_1^o : X \times X \xrightarrow{\mathsf{proj}_2} X \xrightarrow{c} \mathbf{Gauss}(Y) \ ;$$

(iii) the inverse output map $\mathsf{L}_\lambda(c)_2^o : X \times \mathbf{Gauss}(X) \times Y \rightarrow \mathbf{Gauss}(X)$ is defined by

$$\mathsf{L}_\lambda(c)_2^o : X \times \mathbf{Gauss}(X) \times Y \rightarrow \mathbb{R}^{|X|} \times \mathbb{R}^{|X| \times |X|} \hookrightarrow \mathbf{Gauss}(X)$$
$$(x, \pi, y) \mapsto \big(x, \Sigma_{c'}(x, \pi, y)\big)$$

where the inclusion picks the Gaussian state with the indicated statistical parameters, whose covariance $\Sigma_{c'}(x, \pi, y) := \big(\partial_x^2 E_{(c,\pi)}\big)(x, y)^{-1}$ is defined following equation (5.8) of Lemma 5.3.31 (with trivial coparameterization $M = 1$);

(iv) the update map $\mathsf{L}_\lambda(c)^u : X \times \mathbf{Gauss}(X) \times Y \rightarrow \mathcal{G}(X)$ is defined by

$$\mathsf{L}_\lambda(c)^u : X \times \mathbf{Gauss}(X) \times Y \rightarrow X \hookrightarrow \mathcal{G}(X)$$
$$(x, \pi, y) \mapsto x + \lambda \, \partial_x \mu_c(x, y)^T \eta_c(x, y) - \lambda \, \eta_\pi(x)$$

where the inclusion $X \hookrightarrow \mathcal{G}(X)$ is given by the unit of the Giry monad $\mathcal{G}$ which takes each $x : X$ to the corresponding delta distribution, and where $\eta_c(x, y) := \Sigma_c(x)^{-1} \, \epsilon_c(y, x)$ and $\eta_\pi(x) := \Sigma_\pi^{-1} \, \epsilon_\pi(x)$.

---

[3]Note that all coparameterized channels of interest are obtained as the copy-composites of non-coparameterized channels.

Remark 7.3.5. Note that the update map $\mathsf{L}_\lambda(c)^u$ is actually deterministic, in the sense that it is defined as a deterministic map followed by the unit of the probability monad. However, the general stochastic setting is necessary, because the composition of system depends on the composition of Bayesian lenses; recall Definition 6.3.6, which defines the bidirectional composition of cilia. Intuitively, we can consider a composite system $\mathsf{L}_\lambda(d) \circ \mathsf{L}_\lambda(c)$ and note that the forward inputs to the d component and the backward inputs to the c component will be sampled from the stochastic outputs of c and d respectively. Because these inputs are passed to the corresponding update maps, the updates inherit this stochasticity.

Remark 7.3.6. The terms $\eta_c(x, y) = \Sigma_c(x)^{-1} \, \epsilon_c(y, x)$ in the update map of the Laplace doctrine can be understood as *precision-weighted error* terms: the inverse covariance $\Sigma_c(x)^{-1}$ encodes the 'precision' of the distribution (consider the univariate case); and the term $\epsilon_c(y, x) = y - \mu_c(x)$ encodes the 'error' between the observation y and the predicted mean $\mu_c(x)$. The representation of prediction errors is a hallmark of predictive coding schemes.

To define an approximate inference doctrine, we need a (parameterized) inference system. For predictive coding, this will be obtained by assigning to each channel c an inversion whose parameter represents the mean of the emitted posterior; this parameter will later be learned by the resulting doctrine. In order for this assignment to be functorial, we restrict the posteriors emitted by this inference system to have diagonal covariance, meaning that there will be no correlations between dimensions. This formalizes what is known in the literature as a *mean field* assumption [48, 101], without which those earlier works would not have been able to make implicit use of functoriality.

Proposition 7.3.7 (Mean field Laplace)**.** As long as $\bullet$ denotes copy-composition, the following data define a (non-unital) strong parameterized inference system ℓ on **FdGauss**. Each 0-cell X is mapped to (X, X). Each 1-cell $c \; : \; X \underset{M}{\rightsquigarrow} Y$ is mapped to the parameterized lens $(c, c') \; :$ $(X, X) \xrightarrow[M]{X \times M} (Y, Y)$ whose forward channel is c and whose parameterized backward channel $c' : Y \xrightarrow[M]{X \times M; X} X$ emits the Gaussian with mean $(x, m) : X \times M$ determined by the parameter and which minimizes the (mean-field) Laplacian free energy. Thus, writing $\left(\mu_{c'_\pi}^{x,m}(y), \Sigma_{c'_\pi}^{x,m}(y) \right)$ for the statistical parameters of $c'^{x,m}_\pi(y)$, ℓ assigns

$$\mu_{c'_\pi}^{x,m}(y) := (x, m) \qquad \text{and} \qquad \Sigma_{c'_\pi}^{x,m}(y) := \begin{pmatrix} \left(\overline{\partial_x^2} E_{(c,\pi)} \right)(x, m, y)^{-1} & 0 \\ 0 & \left(\overline{\partial_m^2} E_{(c,\pi)} \right)(x, m, y)^{-1} \end{pmatrix} .$$

where $\overline{\partial^2}$ denotes the diagonal Hessian[4] It is the diagonal structure of $\Sigma^{x,m}_{c'_\pi}$ that justifies the 'mean-field' moniker.

Proof. First, we note that ℓ fails to be unital because, for each identity channel $\mathrm{id}_X : X \overset{X}{\nrightarrow}$, the mean of the assigned inversion id'_X is determined by the parameter X, rather than the input. If this parameter happens to equal to the input, then id'_X will actually act as the identity channel. This is because we can understand the identity channel as the limit as $\sigma \to 0$ of a Gaussian with mean equal to the input x and variance $\sigma \mathbf{1}_X$ (where $\mathbf{1}_X$ is the identity matrix on X). Informally, we have $\Sigma^x_{(\mathrm{id}'_X)_\pi}(x') = \big(\overline{\partial^2_x} E_{(\mathrm{id}_X,\pi)}(x,x')^{-1} = 0$, and so $(\mathrm{id}'_X)^x$ acts as the Dirac delta distribution on the parameter x; but of course in general the parameter x need not equal the forward input.

Next, we show that ℓ is strongly functorial (as long as $\bullet$ is always interpreted as copy-composition). If $c : X \underset{M}{\nrightarrow} Y$ and $d : Y \underset{N}{\nrightarrow} Z$ are composable Gaussian channels, then the statistical parameters of the composite approximate inversion $c' \circ d'_c : Z \xrightarrow[MYN]{YNXM;X} X$ are $\mu^{y,n,x,m}_{(c' \circ d'_c)_\pi}(z) = (x,m,y,n)$ and

$$\Sigma^{y,n,x,m}_{(c' \circ d'_c)_\pi}(z) = \mathrm{diag} \begin{bmatrix} \big(\overline{\partial^2_x} E_{(c,\pi)}\big)(x,m,y)^{-1} \\ \big(\overline{\partial^2_m} E_{(c,\pi)}\big)(x,m,y)^{-1} \\ \big(\overline{\partial^2_y} E_{(d,c\bullet\pi)}\big)(x,m,y,n,z)^{-1} \\ \big(\overline{\partial^2_n} E_{(d,c\bullet\pi)}\big)(x,m,y,n,z)^{-1} \end{bmatrix} .$$

Note that, by interpreting $\bullet$ as copy-composition, we have

$$E_{(d,c\bullet\pi)}(x,m,y,n,z) = -\log p_d(n,z|y) - \log p_c(m,y|x) - \log p_\pi(x) .$$

On the other hand, ℓ assigns to $d \bullet c : X \xrightarrow[MYN]{\bullet} Z$ the lens $\big(d \bullet c, (d \bullet c)'\big)$ whose inversion $(d \bullet c)' : Z \xrightarrow[MYN]{XMYN;X} X$ is defined by the statistical parameters $\mu^{x,m,y,n}_{(d\bullet c)'_\pi}(z) = (x,m,y,n)$ and

$$\Sigma^{x,m,y,n}_{(d\bullet c)'_\pi}(z) = \mathrm{diag} \begin{bmatrix} \big(\overline{\partial^2_x} E_{(d\bullet c,\pi)}\big)(x,m,y,n,z)^{-1} \\ \big(\overline{\partial^2_m} E_{(d\bullet c,\pi)}\big)(x,m,y,n,z)^{-1} \\ \big(\overline{\partial^2_y} E_{(d\bullet c,\pi)}\big)(x,m,y,n,z)^{-1} \\ \big(\overline{\partial^2_n} E_{(d\bullet c,\pi)}\big)(x,m,y,n,z)^{-1} \end{bmatrix}$$

where

$$E_{(d\bullet c,\pi)}(x,m,y,n,z) = -\log p_d(n,z|y) - \log p_c(m,y|x) - \log p_\pi(x)$$

$$= E_{(d,c\bullet\pi)}(x,m,y,n,z) .$$

Consequently, $\Sigma^{x,m,y,n}_{(d\bullet c)'_\pi}(z) = \Sigma^{y,n,x,m}_{(c' \circ d'_c)_\pi}(z)$. It therefore suffices to take the laxator $\ell(d) \mathbin{\Phi} \ell(c) \Rightarrow \ell(d \bullet c)$ to be defined by the isomorphism $(Y \times N) \times (X \times M) \overset{\sim}{\to} (X \times M) \times (Y \times N)$. $\qquad\square$

---

[4] That is, $\overline{\partial^2_x} f(x)$ can be represented as the matrix with diagonal equal to the diagonal of the Hessian matrix $\partial^2_x f(x)$ and with all other coefficients 0.

Remark 7.3.8. Note that the preceding inference system requires $\bullet$ to be interpreted as copy-composition everywhere, which is not strictly in accordance with our earlier usage (which mixed copy-composition with ordinary composition in the state-dependence). Resolving this irregularity is the subject of ongoing work.

Proposition 7.3.9. Stochastic gradient descent with respect to the mean parameter of Laplacian free energy games in the image of ℓ yields a strong functor $\nabla : \mathcal{L} \to \mathbf{DiffCilia}$, where $\mathcal{L}$ is the essential image of LFE restricted to the essential image of ℓ. If $c := (c, c', L^c) : (X, X) \xrightarrow[M]{X \times M} (Y, Y)$ is such a game (a 1-cell) in $\mathcal{L}$, then ∇c is the differential cilium $(X, X) \xrightarrow{X \times M} (Y, Y)$ with state space equal to the parameter $X \times M$ defined as follows.

For each $(x, m) : X \times M$, ∇c outputs the non-coparameterized Bayesian lens $\ell(c)_{\dotplus}^{x,m} : (X, X) \nrightarrow (Y, Y)$ obtained by taking the dynamical state (x, m) as the parameter of the lens and discarding any coparameters.

The 'update' vector field $(\nabla c)^u : (X \times M) \to \mathbf{Gauss}(X) \to Y \nrightarrow \mathsf{T}(X \times M)$ is obtained by taking the negative gradient of the loss function $L^c : Y \xrightarrow{\mathbf{Gauss}(X \times M);X} I$ with respect to the posterior mean parameter, evaluated at the posterior mean:

$$(X \times M) \to \mathbf{Gauss}(X) \to Y \to \mathsf{T}(X \times M)$$
$$(x, m, \pi, y) \mapsto - \left(\partial_{(x,m)} E_{(c,\pi)} \right) (x, m, y) \ .$$

(This yields a morphism in $\mathbf{sfKrn}$ via the embedding $\mathbf{Meas} \hookrightarrow \mathbf{sfKrn}$; it is clearly measurable as it is a continuous function between Cartesian spaces.)

Proof. Since the state space $X \times M$ is the space of means of the Laplacian posteriors, the 'update' action of ∇c, the open vector field $(\nabla c)^u$, is defined as the (negative) gradient of L^c with respect to these means (so that the associated flow performs gradient descent). The parameterized loss function $L^c : Y \xrightarrow{X \times M; X} I$ encodes the Laplacian free energy associated to the parameterized lens (c, c'), and corresponds (by Example 5.3.4) to the function

$$X \times M \to \mathbf{Gauss}(X) \to Y \to [0, \infty]$$
$$(x, m, \pi, y) \mapsto \mathsf{LFE}(c, c'^{x,m})_\pi(y)$$

where

$$\mathsf{LFE}(c, c'^{x,m})_\pi(y) = E_{(c,\pi)}(x, m, y) - S_{X \otimes M}[c_\pi'^{x,m}(y)] \ .$$

The entropy $S_{X \otimes M}[c'^{x,m}_\pi(y)]$ does not depend on the mean of $c'^{x,m}_\pi(y)$, and so the gradient of $\mathsf{LFE}(c, c'^{x,m})_\pi(y)$ with respect to (x, m) is simply $\left(\partial_{(x,m)} E_{(c,\pi)}\right)(x, m, y)$. Hence defining $(\nabla c)^u$ as stated yields

$$(\nabla c)^u : (x, m, \pi, y) \mapsto - \left(\partial_{(x,m)} E_{(c,\pi)}\right)(x, m, y) .$$

We now show that ∇ is strongly functorial with respect to composition of 1-cells in $\mathcal{L}$. First, we check that ∇ satisfies the strong unity axiom, which means we need a 2-isomorphism $\mathsf{id}_{(X,X)} \Rightarrow \nabla(\mathsf{id}_{(X,X)})$ in **DiffCilia**. Note that the cilium $\mathsf{id}_{(X,X)}$ has trivial state space 1, trivial update map, and outputs the identity lens $(X, X) \twoheadrightarrow (X, X)$. Likewise, the identity game $\mathsf{id}_{(X,X)}$ has trivial parameter 1, loss function equal to 0, and lens being the (trivially coparameterized copy-composite) identity lens $(X, X) \twoheadrightarrow (X, X)$. Since the loss function is constantly 0 with trivial parameter, ∇ acts to return a cilium $(X, X) \xrightarrow{1} (X, X)$ again with trivial state space and which constantly outputs the identity lens; its update map is likewise trivial. Therefore we take the 2-cell $\mathsf{id}_{(X,X)} \Rightarrow \nabla(\mathsf{id}_{(X,X)})$ to be witnessed by the identity $\mathsf{id}_1 : 1 \to 1$, which satisfies strong unity *a fortiori*.

Finally, we check that ∇ satisfies the strong functoriality axiom, meaning that we seek a 2-isomorphism $\nabla(d, d', L^d) \circ \nabla(c, c', L^c) \Rightarrow \nabla\big((d, d', L^d) \circ (c, c', L^c)\big)$ for each pair of composable Laplacian free energy games $(c, c', L^c) : (X, X) \xrightarrow[M]{X \times M} (Y, Y)$ and $(d, d', L^d) : (Y, Y) \xrightarrow[N]{Y \times N} (Z, Z)$. Note that the composite game has the type $(X, X) \xrightarrow[MYN]{(Y \times N) \times (X \times M)} (Z, Z)$, that by the universal property of $\times$ we have an isomorphism $(Y \times N) \times (X \times M) \cong (X \times M) \times (Y \times N)$, and that the product of Gaussians is again Gaussian. Note also that the parameterized loss function $L^d \circ L^c$ equals

$$
\begin{array}{ccccc}
(Y \times N) \times (X \times M) & \to & \mathbf{Gauss}(X) & \to & Z & \to & [0, \infty] \\
(y, n, x, m, & & \pi, & & z) & \mapsto & (L^c)^{x,m}_\pi \circ d'^{y,n}_{c \bullet \pi}(z) + (L^d)^{y,n}_{c \bullet \pi}(z) .
\end{array}
$$

On the other hand, the update map of the composite cilium $\big(\nabla(d, d', L^d) \circ \nabla(c, c', L^c)\big)^u$ equals

$$
\begin{array}{ccccc}
(X \times M) \times (Y \times N) & \to & \mathbf{Gauss}(X) & \twoheadrightarrow & Z & \to & \mathsf{T}\big((X \times M) \times (Y \times N)\big) \\
(x, m, y, n, & & \pi, & & z) & \mapsto & \big((\nabla c^u)^{x,m}_\pi \bullet d'^{y,n}_{c \bullet \pi}(z), (\nabla d^u)^{y,n}_{c \bullet \pi}(z)\big) .
\end{array}
$$

The desired 2-isomorphism $\nabla(d, d', L^d) \circ \nabla(c, c', L^c) \Rightarrow \nabla\big((d, d', L^d) \circ (c, c', L^c)\big)$ is thus witnessed by a map $(Y \times N) \to (X \times M) \to (X \times M) \times (Y \times N)$, which we take to be the symmetry swap of the categorical product. Computing the gradient of the L terms in $L^d \circ L^c$ with respect to the mean of the joint Gaussian (χ, ρ) yields the update map

$$
\begin{array}{ccccc}
(Y \times N) \times (X \times M) & \to & \mathbf{Gauss}(X) & \twoheadrightarrow & Z & \to & \mathsf{T}\big((Y \times N) \times (X \times M)\big) \\
(y, n, x, m, & & \pi, & & z) & \mapsto & \big((\nabla d^u)^{y,n}_{c \bullet \pi}(z), (\nabla c^u)^{x,m}_\pi \bullet \overleftarrow{d}^{y,n}_{c \bullet \pi}(z)\big)
\end{array}
$$

which is clearly equal to $\left(\nabla(d, d', L^d) \circ \nabla(c, c', L^c)\right)^u$ upon composition with swap. It therefore only remains to check that the two cilia output the same Bayesian lenses $(X, X) \rightarrow (Z, Z)$, up to swap. This follows from the strong functoriality of ℓ. $\square$

Remark 7.3.10. Although we have defined ∇ manually, we expect that it can alternatively be obtained more abstractly, from a proper treatment of stochastic gradient descent applied to statistical games. We leave this to future work.

Finally, to obtain the dynamical systems with which we started this subsection (in Proposition 7.3.4), we use Euler integration, using the functor Euler_λ of Remark 6.3.25.

Corollary 7.3.11. Fix a real number $\lambda : \mathbb{R}$. By defining $\mathsf{L}_\lambda := \mathsf{Euler}_\lambda \circ \nabla \circ \mathsf{LFE}$ one obtains Laplacian predictive coding as a differential approximate inference doctrine, the *Laplace doctrine* for the mean field Laplace inference system ℓ. The systems of Proposition 7.3.4 are obtained in its image.

Proof. Suppose $c : X \rightarrow Y$ is a morphism in **FdGauss**. It is not coparameterized, so ℓ assigns to it the parameter space X, which becomes the state space of the cilium $\mathsf{L}_\lambda(c)$. By definition, this cilium emits the same lens — and therefore has the same output maps — as those given in Proposition 7.3.4. We therefore only need to check that

$$\left(\nabla(c)^u\right)^x_\pi(y) = -\left(\partial_x E_{(c,\pi)}\right)(x, y)$$
$$= \partial_x \mu_c(x, y)^T \eta_c(x, y) - \eta_\pi(x) .$$

Recall from Proposition 5.3.31 that

$$E_{(c,\pi)}(x, y) = -\log p_c(y|x) - \log p_\pi(x)$$
$$= -\frac{1}{2}\left\langle \epsilon_c(y, x), \Sigma_c^{-1}\epsilon_c(y, x)\right\rangle - \frac{1}{2}\left\langle \epsilon_\pi(x), \Sigma_\pi^{-1}\epsilon_\pi(x)\right\rangle$$
$$+ \log\sqrt{(2\pi)^{|Y|}\det\Sigma_c} + \log\sqrt{(2\pi)^{|X|}\det\Sigma_\pi} .$$

It is then a simple exercise in vector calculus to show that

$$-\left(\partial_x E_{(c,\pi)}\right)(x, y) = \partial_x \mu_c(x, y)^T \eta_c(x, y) - \eta_\pi(x)$$

as required. $\square$

7.3.2. Synaptic plasticity with the Hebb-Laplace doctrine

The Laplace doctrine constructs dynamical systems that produce progressively better posterior approximations given a fixed forwards channel, but natural adaptive systems—brains in particular—do more than this: they also refine the forwards channels themselves, in order to produce better predictions. In doing so, these systems better realize the abstract nature of free energy games, for which improving performance means improving both prediction as well as inversion. To be able to improve the forwards channel requires allowing some freedom in its choice, which means giving it a nontrivial parameterization.

The Hebb-Laplace doctrine that we introduce in this section therefore modifies the Laplace doctrine by fixing a class of parameterized forwards channels and performing stochastic gradient descent with respect to both these parameters as well as the posterior means; we call it the *Hebb-Laplace* doctrine as the particular choice of forwards channels results in their parameter-updates resembling the 'local' Hebbian plasticity known from neuroscience, in which the strength of the connection between two neurons is adjusted according to their correlation [79, 118, 127, 218, 233]. (Here, we could think of the 'neurons' as encoding the level of activity along a basis vector.)

We begin by defining the category of these parameterized forwards channels, after which we proceed by modifying the mean-field Laplace inference system and the Laplace doctrine accordingly.

Definition 7.3.12 ('Neural' channels)**.** Let $\mathcal{H}$ denote the subbicategory of $\mathbf{PFdGauss}_*$ generated by 1-cells $X \rightarrowtail Y$ of the form

$$\Theta_X \to \mathbf{Gauss}_*(X, Y)$$
$$\theta \; \mapsto \; \Big(x \mapsto \theta\, h(x) + \omega \Big)$$

where X and Y are finite-dimensional Cartesian spaces, h is a differentiable map $X \to Y$, Θ_X is the vector space of square matrices on X, and ω is sampled from a Gaussian distribution on Y.

Proposition 7.3.13 (Mean field Hebb-Laplace)**.** Taking $\bullet$ as copy-composition, the following data define a (non-unital) strong parameterized inference system ℓ on $\mathcal{H}$. Each 0-cell X is mapped to (X, X). Each (parameterized) 1-cell $c : X \overset{\Theta}{\underset{M}{\bullet\!\!\!\to}} Y$ is mapped to the parameterized lens $(c, c') : (X, X) \xrightarrow[M]{\Theta \times (X \times M)} (Y, Y)$ whose forward channel is given by projecting Θ from $\Theta \times (X \times M)$ and applying c, and whose backward channel is defined as in Proposition 7.3.7 (mean-field Laplacian inference).

Proof. The only difference from Proposition 7.3.7 is in the forward channel; but these are just taken from $\mathcal{H}$, and so they compose strongly by assumption. $\qquad\square$

Like the Laplace doctrine, the Hebb-Laplace doctrine is obtained by stochastic gradient descent with respect to the parameters.

Proposition 7.3.14. Let $\mathcal{L}$ denote the essential image of LFE restricted to the essential image of ℓ. Let $c := (c, c', L^c) : (X, X) \xrightarrow[M]{\Theta \times (X \times M)} (Y, Y)$ be a 1-cell in $\mathcal{L}$. Then stochastic gradient descent yields an identity-on-objects strong functor $\nabla : \mathcal{L} \to \mathbf{DiffCilia}$ mapping c to the differential cilium $\nabla(c) : (X, X) \xrightarrow{\Theta \times (X \times M)} (Y, Y)$ defined as follows.

For each triple of parameters $(\theta, x, m) : \Theta \times (X \times M)$, ∇c outputs the non-coparameterized Bayesian lens $\ell(c)_{\maltese}^{\theta,x,m} : (X, X) \nrightarrow (Y, Y)$ obtained by taking the dynamical state (θ, x, m) as the parameter of the lens and discarding any coparameters.

The vector field $(\nabla c)^u$ is obtained by taking the gradient of the loss function L^c with respect to the 'synaptic' parameter $\theta : \Theta$ and the posterior mean $(x, m) : X \times M$, evaluated at (θ, x, m):

$$\Theta \times (X \times M) \to \mathbf{Gauss}(X) \to Y \to \mathsf{T}(\Theta \times (X \times M))$$
$$(\theta, x, m, \pi, y) \mapsto - \left(\partial_{(\theta,x,m)} E_{(c,\pi)} \right) (\theta, x, m, y) \ .$$

Proof. The proof is almost identical to that of Proposition 7.3.9: the sole difference is that now we also take gradients with respect to the synaptic parameter $\theta : \Theta$, but the reasoning is otherwise the same. $\qquad\square$

Finally, we obtain dynamical systems by Euler integration.

Definition 7.3.15. Fix a real number $\lambda : \mathbb{R}$. The *Hebb-Laplace doctrine* is obtained as the composite $\mathsf{H}_\lambda := \mathsf{Euler}_\lambda \circ \nabla \circ \mathsf{LFE}$, yielding a differential approximate inference doctrine for the mean field Hebb-Laplace inference system ℓ.

Corollary 7.3.16. Suppose $c : X \xrightarrow{\Theta} Y$ is a channel in $\mathcal{H}$ defined by $c^\theta(x) = \theta\, h(x) + \omega$, for some differentiable h and Gaussian noise ω. Then the update map $\mathsf{H}_\lambda(c)^u$ is given by

$$\begin{array}{ccccc} \Theta \times X & \to & \mathbf{Gauss}(X) & \to & Y \to \\ (\theta, x, & & \pi, & & y) \ \mapsto \end{array} \begin{pmatrix} \theta - \lambda\, \eta_{c^\theta}(x, y)\, h(x)^T \\ x + \lambda\, \partial_x h(x)^T \theta^T \eta_{c^\theta}(x, y) - \lambda\, \eta_\pi(x) \end{pmatrix}$$

where $\eta_{c^\theta}(x, y) = \Sigma_{c^\theta}^{-1} \epsilon_{c^\theta}(y, x)$ and $\eta_\pi(x) = \Sigma_\pi^{-1} \epsilon_\pi(x)$ are the precision-weighted error terms.

Proof. Following Corollary 7.3.11 (the Laplace doctrine), We just need to check that

$$\left(\partial_{(\theta,x)} E_{(c^\theta,\pi)}\right)(x,y) = \begin{pmatrix} \partial_\theta E_{(c^\theta,\pi)} \\ \partial_x E_{(c^\theta,\pi)} \end{pmatrix}(x,y) = \begin{pmatrix} \eta_{c^\theta}(x,y)\,h(x)^T \\ -\partial_x h(x)^T \theta^T \eta_{c^\theta}(x,y) + \lambda\,\eta_\pi(x) \end{pmatrix} .$$

This amounts to verifying that $\partial_x \mu_{c^\theta}(x) = \theta\,\partial_x h(x)$ and that $\partial_\theta E_{(c^\theta,\pi)}(x,y) = \eta_{c^\theta}(x,y)\,h(x)^T$. The former holds by the linearity of derivation since $\mu_{c^\theta}(x) = \theta\,h(x)$ by definition; and the latter holds because

$$\begin{aligned}
\partial_\theta E_{(c^\theta,\pi)}(x,y) &= \frac{-\partial_\theta}{2} \left\langle \epsilon_{c^\theta}(y,x), \Sigma_{c^\theta}^{-1}\,\epsilon_{c^\theta}(y,x) \right\rangle \\
&= \frac{-\partial_\theta}{2} \left\langle y - \theta\,h(x), \Sigma_{c^\theta}^{-1}\left(y - \theta\,h(x)\right) \right\rangle \\
&= \Sigma_{c^\theta}^{-1}\left(y - \theta\,h(x)\right)h(x)^T \\
&= \Sigma_{c^\theta}^{-1}\,\epsilon_{c^\theta}(y,x)\,h(x)^T \\
&= \eta_{c^\theta}(x,y)\,h(x)^T
\end{aligned}$$

as required. $\qquad\square$

Remark 7.3.17. From a biophysical point of view, the Hebb-Laplace doctrine so defined has a notably suboptimal feature: the 'synaptic' forwards parameter $\theta : \Theta$ is updated on the same timescale λ as the parameter $x : X$ that encodes the posterior mean, even though the latter parameter is typically interpreted as encoding the activity of a population of neurons, which therefore changes on a faster timescale than those neurons' synapses. Not only is this important for reasons of biological plausibility, but also for mathematical reasons: we should understand the backwards activity as bundled over the forwards synapses, and a change in the parameter θ should induce a corresponding 'transport' of x. An appropriately geometric treatment of compositional approximate inference and predictive coding, resulting in bundles of open dynamical systems, is again something that we leave to future work.

8. Future directions

A powerful motivation propelling the development of this thesis was the belief that science, and particularly the cognitive sciences, will benefit from being supplied with well-typed compositional foundations. In this final chapter, we survey a number of new vistas that we have glimpsed from the vantage point of our results, and indicate routes that we might climb in order to see them better.

One important benefit of the categorical framework is that it helps us express ideas at a useful level of abstraction, and thereby compare patterns across systems and phenomena of interest. As a result, although our primary system of interest is the brain, we are aware that much of our work is more diversely applicable, and so our survey here is similarly not restricted to neural systems. At the same time, as neural systems are our finest examples of natural intelligence, we attempt to stay grounded in current neuroscience.

Beyond the evident shortcomings of the work that we have presented—which we review momentarily—we first consider how to use the categorical language of structure to incorporate structure better into our models themselves (§8.1), with a particular focus on the brain's "cognitive maps" (§8.1.3). We will see that the compositional consideration of the structure of open systems naturally leads us to consider *societies* of systems (§8.2), and hence the relationships between compositional active inference and single- and multi-agent reinforcement learning and economic game theory (§8.2.3), although our first priority in this section is the incorporation of action (§8.2.1) and planning (§8.2.2) into the framework of statistical games. From our abstract vantage point, there is little difference between societies of agents and collective natural systems such as ecosystems[1], and so we then consider the prospects for a compositional mathematics of life (§8.3). Finally, we close with some thoughts on matters of fundamental theory (§8.4).

Before we wade into the thick of it, let us note three prominent examples of the aforementioned evident shortcomings.

Firstly, the current presentation of copy-composite stochastic channels, and the bicategories of lenses and statistical games that result from them, is quite inelegant: the necessity of coparameters

---

[1]After all, a single multicellular organism is itself a kind of society of agents.

introduces substantial complexity that is never repaid, because all coparameters arise from the copy-composition of ordinary channels. This complexity infects the notion of approximate inference doctrine, which could benefit both from simplification and from further exemplification, ideally by examples drawn from beyond neuroscience.

Secondly, the generative models that we have considered are somehow 'static', despite our interest in dynamical systems, and this warrants a satisfactory exploration of *dynamical* generative models.

Thirdly, although we considered "lower level" neural circuit models in §3.3, we did not explicitly connect our approximate inference doctrines to these more 'biological' models. A satisfactory account of the Bayesian brain would of course span from abstract principles to detailed biology, a relationship the elaboration of which we sadly we leave to future work.

Fortunately, although these three shortcomings may be pressing, we expect that the pursuit of a research programme akin to that sketched below would result in overcoming them.

8.1. Structured worlds

8.1.1. Bayesian sensor fusion

A situation that is common in natural embodied systems but which is not yet well treated by current statistical and machine learning methods[2], particularly those that are most popular in computational neuroscience, is that of sensor fusion. In this situation, one has a number of sensors (such as cameras or retinal ganglion cells) which report spatially situated data, and where the sensor fields *overlap* in the space; the problem is then how to combine these "local views" of the space into a coherent global picture. Mathematically, fusing 'local' data into a 'global' representation is the job of *sheaves*: a sheaf is a "spatially coherent data type"—something like a bundle for which 'local' sections can always be uniquely glued together into a global section—and sheaf theory and the related fields of applied topology and cohomology allow us to judge when it is possible to form a consensus, and quantify the obstructions to the formation of consensus; recent work has also begun to suggest algorithms and dynamics by which we can construct consensus-forming distributed sensor systems [123].

Sheaves therefore allow us to construct and to measure spatially structured data types, but missing from the current sheaf-theoretic understanding of sensor fusion is a thorough treatment of belief and uncertainty, especially from a Bayesian perspective. Since biological systems contain many

---

[2]This is beginning to change: recently, the use of sheaf-theoretic and other applied-topological devices has started to penetrate machine learning [31, 266].

distributed sensor types, and each of these systems is constituted by many cells, the mathematics of neural representations may be expected to be sheaf-theoretic. A first possible extension of the work presented here, therefore, is to extend statistical games and approximate inference doctrines (and hence the classes of model that they encompass) to structured data types such as sheaves. Because statistical games and approximate inference doctrines are defined using lenses over an abstract category of stochastic channels, we expect that the first step will be to consider categories of channels between sheaves; recently, there has been work on categorifying lenses [63, 64], and we expect that this may prove relevant here. We also expect that at this point the fibrational structure of statistical games will again prove utile in order that losses may be correctly counted on any overlaps. Fortunately, being similarly structured, sheaves and fibrations are natural partners, and so we expect that a second spatial extension of the present work will be to exploit the latent geometric structure of fibrations of statistical games.

In this context, we may also encounter connections to sheaf-theoretic approaches to 'contextuality', in which answers to questions depend on (the topology of) how they are asked, and which seems to lie at the heart of quantum nonlocality. It is notable that lenses originated in database theory [34, 99] and that contextuality can also be observed in database systems [1, 58], and so at this point, it may be possible to uncover the mathematical origins of 'quantum-like' psychological effects [6, 49, 153], and relate them formally to other kinds of perceptual bistability that have been interpreted in a Bayesian context [144, 170]. Sheaves come with a cohomology theory that permits the quantification of the 'disagreements' that underlie such paradoxes [5, 45, 75], and dynamical systems can be designed accordingly to minimize disagreements and thus seek consensus [123–125]. We hope that these tools will supply new and mathematically enlightening models of these psychological phenomena, while at the same time also suggesting new connections to work on quantum-theoretic formulations of the free-energy framework itself [90, 91].

The adoption of a sheaf-theoretic framework in this way may furthermore illuminate connections between computational neuroscience and machine learning. Graph neural networks [156, 157, 285], and their generalization in 'geometric' deep learning [44], are increasingly used to apply the techniques of deep learning to arbitrarily structured domains, and, as indicated above, recent work has found sheaves to supply a useful language for their study [31]. In a similar fashion, we expect connections here to the structure of message passing algorithms [83, 190, 208, 283, 285] (also hinted at by Sergeant-Perthuis [229]) and less conventional structured machine learning architectures such as capsule networks [223]. Finally, each category of sheaves is naturally a topos [177], and

hence comes with its own rich internal language, modelling dependent type theory (*cf.* §2.3.4).

8.1.2. Learning structure and structured learning

Having considered the incorporation of structured data into the process of inference, we can consider the incorporation of structure into the process of learning, and here we make an important distinction between *structured learning* and *learning structure*. By the former, we mean extending the process of learning to a structured setting (such as the sheaf-theoretic one of the preceding section), whereas by the latter, we mean learning the underlying structures themselves. This latter process is also known in the literature as *structure learning* [143, 260, 262], but in order to avoid ambiguity, we swap the order of the two words.

The observation at the end of the preceding section, that each category of sheaves forms a topos, is pertinent here, as dependent type theory formalizes a notion of logical 'context', containing the "axioms that are valid in the present situation", and determining which (non-tautological) statements can be derived. In the categorical semantics of dependent type theory, the context is witnessed by the object over which a slice category is defined, and so in some sense it defines the "shape of the universe". By the Grothendieck construction, there is a correspondence between sheaves and certain bundles (objects of slice categories), and so (very roughly speaking) we can think of structured inference and learning as taking place in appropriate slice categories.

In the same way that we motivated random dynamical systems (*qua* bundles, §6.2.3) through "parameterization by a noise source", we can think of bundle morphisms as generalized parameterized maps. The problem of learning structure then becomes a problem of generalized parameter-learning, and much like this can be formalized by a 'reparameterization' in the **Para** construction (§3.2.1), in this more general setting it is formalized by the "generalized reparameterization" of base-change between topoi (*cf.* Remark 4.2.29). Base-change synthesizes notions of parallel transport, allowing us to translate spatially-defined data coherently between spaces—and, in particular, along changes of structure; recall our earlier remark about the importance of parallel transport to a biophysically-plausible Hebb-Laplace doctrine (Remark 7.3.17). In this setting therefore, we expect that work on functorial lenses [63], as well as work on functorial data migration [236, 242], may prove relevant.

At the same time, we expect this line of enquiry to clarify the relationships between our formalism of approximate inference and other related work on the categorical foundations of cybernetics [51, 54], which have typically been studied in a differential rather than probabilistic setting [74]. We expect the connection to be made via information geometry [9, 10, 195], where Bayesian inference can be understood both using gradient descent [199] and as a kind of parallel transport [225].

8.1.3. Compositional cognitive cartography

Natural systems such as animals learn the structure of their environments as they explore them. We will come below (§8.2.1) to the question of how to incorporate action—and hence exploration—into the compositional framework that we have developed here, but meanwhile we note that the topos-theoretic developments sketched above may provide a suitable setting in which to understand the neural basis for navigation, and help explain how ostensibly 'spatial' navigation processes and circuits are invariably involved in more abstract problem solving [24, 26, 27, 116, 178].

There are two key observations underlying this proposal. Firstly, a topos is not only a richly structured category of spaces (or spatial types), but it can also be understood as a categorified space itself [234]: in this context, we can call each categorified space a 'little' topos, and the category of spaces itself is the corresponding 'big' topos; changes in spatial structure—witnessed by base-change between little topoi—thus correspond to trajectories within the space represented by the big topos.

Secondly, under the free energy principle, there is a close relationship between beliefs about the geometry of an environment and beliefs about expected future trajectories in that environment [148]: fundamentally, this is also the idea underlying the "successsor representation" [78] of the cognitive map, which says roughly that the brain's representation of where it is is equivalently a representation of where it soon expects to be [47, 256, 257]. Although there have been studies in the informal scientific literature attempting to connect free-energy models of navigation, exploration, and the cognitive map with the successor representation [185], and to place both of these in less overtly spatial contexts [47, 205], there has not yet been a comprehensive mathematical treatment explaining the structures that underlie this nexus.

By placing such a mathematical treatment in a topos-theoretic context, it may be possible to make sense of the "logic of space" of topoi to explain why animals' abstract problem-solving makes use of their abilities for spatial navigation: in particular, proving a proposition is mathematically analogous to finding a path from premise to conclusion. Moreover, in a spatial topos, the "truth values" are no longer simply binary, but encode *where* a proposition is (believed to be) true; the (sub)object classifier of a spatial topos encodes something like the "internal universe" of that topos, or "the universe according to the system".

To be successful, this mathematical treatment should be attentive to the results and proposals of computational and theoretical neuroscience, and so we now turn to our second key observation: the relationship between (believed) geometry and (expected) dynamics. This will require an extension of statistical games and approximate inference to *dynamical* generative models; until this point,

our treatment has merely supplied inference (or 'recognition' [48]) dynamics to static models. Through this extension, we should expect a connection to other work on dynamical inference, such as filtering [105, 147] and particularly its emerging compositional treatment [271, 272].

Under the free-energy principle, and similarly under the successor representation, the expected dynamics is a geodesic flow, which is by geodesy determined by beliefs about the spatial geometry. But these beliefs in turn are not static: they depend on what the agent believes will happen [76, 186], and this has been suggested as an explanation for the 'predictive' nature of the cognitive map [148]. The cognitive map has its central locus in the hippocampus [88, 193, 198], which we may therefore understand as representing the base space over which the big topos is sliced; and since changes-of-plan seem therefore to induce changes-of-base, we might see the 'functional' connectivity of the brain [245] as witnessing this mathematical structure.

Because the internal universe of the topos represented by the cognitive map is inherently context-dependent, it seems to fit naturally with the subjectivist metaphysics implied by the free energy framework—that the universe as experienced by an agent is a construction of that agent's internal model, as updated by approximate inference—and thus to provide a natural setting for the mathematical study of phenomenology. Moreover, as categories of sheaves, agents' internal topoi encode the consensus formed by the distributed circuits and sensors that constitute their beliefs, and this points a way towards understanding how societies of agents are able to inhabit shared spaces about which they form a consensus between themselves: the mathematics of this shared universe should be little different from the mathematics of a single agent's internal universe.

Such multi-agent adaptive systems have been studied in the context of reinforcement learning (of which more below), but this potential for the formation of 'nested' systems with shared universes implied by consensus is not the only connection between cognitive cartography and reinforcement learning, as it is in reinforcement learning that the successor representation originates. We therefore hope that this line of enquiry may illuminate the relationship between reinforcement learning and compositional active inference, to the basis of which we now turn.

8.2. Societies of systems

Adaptive agents being necessarily in interaction with an external environment, we saw in the previous section how consideration of the compositional structure of agents' internal maps of their worlds easily leads to the consideration of societies of agents. However, in order for us to study

these, we first need to make the more basic step of incorporating action into the compositional framework: a collection of purely passive agents is no society.

8.2.1. *Active* inference

The doctrines of approximate inference introduced in this thesis are inherently perceptual. As we described in Remark 4.0.1, the forwards channel of a statistical game points "towards the environment", predicting the expected incoming sense-data, whereas the backwards channel points from the environment into the agent, terminating in the agent's most causally abstract beliefs. In other contemporary work on categorical cybernetics, the orientation appears different: the forwards channel of an open (economic) game, for instance, points along the direction of interaction in the environment, in the direction of causality, from observations to actions [36, 119]; there is no room for prediction and its inversion, and the two kinds of game seem somehow perpendicular.

In resolution of this apparent disagreement, we can observe that an open economic game does have a perpendicular direction: a second[3] dimension inhabited by the strategies. That is to say, an open economic game is a lens externally parameterized by strategies, a function from the latter to the former, and therefore formally much like our cilia (§6.3). This resemblence becomes even closer when one considers the recent 'diegetic' formulation of open games, in which strategies themselves can be updated using a backwards map from the arena of the game back into strategies (or rather, strategy updates).

This suggests one way in which we can incorporate action and thereby shape the framework of this thesis into a framework for *active* inference: the forwards channel should predict not only sense-data incoming from the environment, but also the actions to be taken by the agent. Indeed this matches the usual informal presentation of active inference, which adopts a channel of the form $X \rightarrow S \otimes A$, where S is the space of sense-data, A the space of possible actions, and X the 'latent' space.

Yet at this point the formal similarity between compositional active inference and compositional game theory again begins to recede, as a channel $X \rightarrow S \otimes A$ is more like a "stochastic span" than an open economic game's player model $\Sigma \rightarrow [S, A]$. Moreover, we expect our active inference systems to have a richer variety of patterns of interaction, being embodied in a world—in part, this motivated our adoption of polynomial functors for structuring interaction. We therefore expect the compositional theory of active inference to have forwards channels rather of the form

---

[3]Or third, if one remembers the monoidal structure.

$X \nrightarrow \sum_{a:A} S[a]$, so that an agent's sensorium depends on the configuration (or 'action') that it has chosen.

This was the approach we sketched in our earlier work-in-progress on *Polynomial Life* [252], where we suggested that polynomial functors supply a formalization of the notion of "Markov blanket" used in the informal active inference literature to characterize the interaction boundaries of adaptive systems [102, 158, 203] (a formalization that is situated at a useful level of technical flexibility, being neither as abstract as the very general notion of interface adopted by categorical systems theory [192], nor as concrete as simple products of spaces). In this way, we believe that a fruitful direction in which to pursue a compositional theory of active inference is, like our theory of open dynamical systems, as a **Poly**-algebra of statistical games. Fortunately, although the types prove somewhat different, the structural resemblence between active inference and economic games is maintained: in both cases, one has categories of lenses into the arena of interaction, indexed by a category of interfaces, and thus in philosophical (and thus we expect also mathematical) concordance with Myers' double-categorical view of systems theory [192].

Once again, this line of enquiry naturally leads on to the consideration of multi-agent systems. But before we come to that, there remain important questions about single-agent systems, and the connection between single-agent active inference and the cousin of economic games, reinforcement learning.

8.2.2. What is the type of a plan?

Each active inference system has an internal 'latent' state space equipped (by its generative model) with a prior distribution, which represents the systems's initial beliefs about the likelihood of those states. As we have seen, the system can perceive, changing that distribution better to match incoming sense data. And as we hope to see, it should also be able to act, affecting its environment so that future states better match its initial beliefs. Perception and action are thus in general the two dual ways in which a system can minimize its free energy, akin to the two degrees of freedom available in a free energy game.

But a system that acts must necessarily be motivated towards some goal, even if that goal is simply "stay alive" or "perform action a", and even though this goal may be adjusted by the system's perceptions. In order to realize its goal, whatever it may be, the system must enact a plan, however trivial—and the informal literature on active inference encodes the plan into the system's latent prior. When it comes to static models, the prior may be simply a (static) distribution over the state space itself; but in the dynamical case, it is typically a distribution over *trajectories* of states.

Such a distribution is often [76, 148] taken to encode likelihoods of hypothetical courses of action, which one might call a *policy*[4]: the system then perceives and acts in order to implement its policy. But the construction of this policy may involve a lot of data, such as the specification of goal states and the accumulation of the "expected free energy" of trajectories in the context of those goals, and so it seems unnecessarily crude to hide all of this data inside a single undifferentiated choice of prior distribution.

This prompts us to ask, what is the form of this data, and how can we incorporate it into the compositional framework? In other words, what is the type of a plan? These seem to us to be key questions for future work.

8.2.3. Reinforcement learning, open games, and ecosystems

There is known to be a close relationship between active inference in Markov decision problems (MDPs) and reinforcement learning [73], and it is through this relationship that one sees particularly clearly the strangeness of encoding all the data of an agent's policy in a single 'prior' state. This relationship is seemingly not superficial, as there are hints of a deep structural connection.

First, recall that the standard algorithm for obtaining a Bellman-optimal policy for an MDP is *backward induction* (otherwise known as *dynamic programming*) [214, 284][5]. It is now known that backward induction is structured according a similar bidirectional pattern (the *optic* pattern) to that of both Bayesian inference and reverse differentiation [128], and that MDPs themselves fit into the associated general framework of open games [36] (which are governed by the same pattern). Second, in the informal active inference approach to MDPs, the system in question counterfactually evaluates policies using a backward-induction-like process, accumulating free energies in order to score them [73]. It is this process that results in the prior discussed above, which is then updated by the agent's inference process. Future work will need to untangle this knot of interrelated bidirectional processes; and as usual in categorical modelling, this means first writing them all down precisely. We hope that, having done so, we will see how the whole picture emerges, and how it relates to the developing geometric (or 'diegetic') framework in categorical cybernetics [51] (possibly involving the further development of our notion of 'cilia' from §6.3). In particular, since the free energy principle underlying active inference asserts a certain informal universality (on which more in §8.3.1), we might also hope that the satisfactory development of compositional

---

[4]In the language of polynomial functors, this seems to be something like a distribution over the cofree comonad on the system's polynomial interface.

[5]Also see [29, 89, 106, 128, 261] for other presentations.

active inference might exhibit a universal property: that any other doctrine of cybernetic systems factors uniquely through it.

The story of these connections will initially be told from the perspective of a single agent, as backward induction only considers how to find a single policy for a single MDP; although this policy may involve multiple agents, the implied global search entails a common controller: the procedure doesn't consider the factorisation of the agents. But casting this account into the emerging framework of compositional active inference will point towards a bridge to multi-agent reinforcement learning. For example, multi-agent RL often studies the emergence of collaboration, and we might expect to see this represented in the formal structure, thereby understanding how to incorporate the factorisation of agents into the compositional framework for backward induction (which in turn may be helpful for designing collaborative 'edge' AI systems).

The resulting general account of multi-agent intelligence will encompass both reinforcement learning and active inference, allowing us to understand their relative strengths and differences. One seeming difference (at this early stage, and following our thinking above) is that compositional active inference envisages the latent state spaces of agents as their "internal universes", which come along with sufficient structure that we might consider them as Umwelten (*i.e.*, their subjective worlds, in the sense of biosemiotics; see §8.3.2 below). Consequently, we should be able to study how agents might come to consensus, thereby resolving their disagreements. And because agents are embodied in a shared world within which they act, this process might involve planning cooperation, at which point the teleological structure of compositional game theory may become important, as cooperating agents will have to bet on spatiotemporally distributed actions. We hope therefore that one distal outcome of this work will be a new and beneficial understanding of corporate activity.

Below, in §8.3, we will discuss how active inference and the free energy principle aim not only to be theories of brains or other prominent intelligent systems, but rather universal theories of all adaptive things. Consequently, their compositional treatment should extend in the 'multi-agent' case not just to corporate activity, but to ecosystems more broadly. And, following the multicategorical algebra latent throughout this thesis, it will undoubtedly prove natural, once we have considered a single level of nesting of systems into ecosystems, to let the hierarchy continue to infinity, producing a fractal-like structure. At this point, we should expect once more to make contact with topics such as higher categories and type theory, particularly in operadic or opetopic (*i.e.*, 'directed') forms; synthetic approaches to mathematical physics; and iterated parameterization in categorical systems theory.

It almost goes without saying that we should expect any framework resulting from this work to capture existing models of collective active inference, such as recent work on spin glasses [129].

8.3. The mathematics of life

We move on to consider the relationships between compositional active inference and the contemporary mathematics of life. We hope that compositional active inference may supply part of the story of a modern theory of autopoiesis, the ability for life to recreate itself [274].

8.3.1. Bayesian mechanics and the free energy principle

Recently, it has been suggested in various venues [102, 204] that the free energy framework provides a 'universal' way to understand the behaviour of adaptive systems, in the sense that, given a random dynamical system, it may be possible to write down a generative model such that the dynamics of the system can be modeled as performing inference on this model. In the language of the conjectured compositional framework for active inference, we may be able to describe a canonical statistical game that each given random dynamical system can be seen as playing.

If this is true, we should be able to express this canonicity precisely: in particular, it should correspond to a *universal property*. Since approximate inference doctrines already gives us functorial ways to turn statistical games into dynamical systems, this suggests we should seek functors that associate to each random dynamical system a statistical game; and we should expect these functors to be adjoint (as morphisms of categories indexed by the systems' interfaces). The desired universal property would then be expressed by the adjunction. (Notably, adjunctions are at the heart of recent synthetic approaches to mathematical physics [228].) This would constitute an important mathematical step to establishing the universality of the free energy principle, or to establishing the conditions that must be satisfied by any satisfactory successor.

Bayesian mechanics promises to build upon the nascent understanding of random dynamics via inference [224] to supply a new theory of mechanics for statistical systems [215]. The present formulation of Bayesian mechanics is constructed using mathematical tools from physics, but not (yet) the kinds of compositional tool promoted in this thesis and further described above. We expect that developments along the lines sketched here will unify the on-going development of Bayesian mechanics (and the resulting understanding of non-equilibrium systems) with the new synthetic understanding of mathematical physics. By casting all dynamics as abstract inference, we should

also expect this line of enquiry to begin to quantify the persistence of things and imbue much of physics with an *élan vital*.

8.3.2. Biosemiotics

It is increasingly acknowledged that biological systems are characterized not only by information-processing, but by *communication* [20]: an often overlooked fact about 'information' in the strict mathematical sense is that it is only meaningful in context. In the original Nyquist-Hartley-Shannon conception of information, this context is the communication of a predefined message over a noisy channel [126, 197, 230]; but more generally, we might think of this context as simply "a question", in which case it is easy to see that information answering one question may not be useful in answering another; or, in a more computational setting, we can see that the bits of an encrypted signal are only useful in revealing the message if one has the decryption key.

Still, one often encounters descriptions of signals as containing n bits of information, without a clear specification of *about what*. Mathematically, the confusion arises because information theory is framed by classical probability, and the assumed context is always the problem of trying to communicate a probability distribution over a pre-defined space X; and once the space is fixed, the only question that can be asked is "what is the distribution?" (Mathematically, this is to say that in the Markov category of classical stochastic channels, there are no non-trivial effects or costates.)

Yet, in the shared universe that we inhabit, there are more questions than this: in quantum theory, for instance, one can ask many questions of the state of a system, by projecting the state onto the subspace representing one's question. (These projections are the non-trivial effects or costates of quantum probability.) This act of projection is an act of *interpretation* of the message encoded by the state at hand.

The emerging 'biosemiotic' reconceptualization of life explicitly acknowledges the importance and universality of communication in context [20], proposing that in any such situation the interpreting system necessarily has an internal representation of the external world (its *Umwelt*) which is updated by interpreting incoming signals. We can in principle reconstruct the external world by understanding it as "that which a collection of systems agrees about": perhaps, then, the shared universe (as determined topos-theoretically) of a fusion of active inference agents is a good model of this 'semiosphere'. It seems therefore that the mathematics resulting from our work on internal universes and their interactions — and, more broadly, many of the formal ingredients of compositional active inference — is well aligned with the informal structures of biosemiotics, and so it may be desirable to re-express biosemiotics accordingly. In doing so, perhaps

the mathematics for a modern Bayesian subjectivist metaphysics will be found[6]: for instance, by expressing communication and its phenomenology as a geometric morphism (a generalized base-change) between agents' internal universes. More pragmatically, perhaps we will be able to say precisely when some object may act as a symbol, and how systems may (learn to) manipulate such symbols.

8.4. Fundamental theory

Future work connected to this thesis need not only be in applications; a number of purely theoretical questions raise themselves, too.

8.4.1. Geometric methods for (structured) belief updating

The mathematics of 'belief' is in large part about replacing definite points with 'fuzzier' distributions over them. In dependent type theory, we replace points with 'terms' (non-dependent terms are exactly points): so a type theory with belief should somehow encompass "fuzzy terms". Just as we can replace points with distributions, we can replace dependent points with dependent distributions. However, the standard replacement (moving from a category of functions to a category of stochastic channels) obscures some of the 'universal' categorical structure that underpins the rules of type theory. This standard replacement also misses something else: while it does allow for fuzzy terms, it omits a model of fuzzy *types*; and we might well want to express beliefs about things whose identity we are not quite sure. (This omission also seems to be related to the loss of universal structure.)

There seem to be a couple of related resolutions to this puzzle. The first is to notice that replacing points by distributions yields another space: the space of distributions over the original space; this is akin to the move in dynamics from working with the random motion of states to working with the deterministic motion of the distribution over states. This space of distributions has a particular geometry (its *information geometry*), and hence we should expect corresponding flavours of topos and type theory. As we have indicated above, there is a move in fundamental mathematical physics (*cf.* Schreiber [228]) to work 'synthetically', expressing concepts using the universal structures of higher topoi. This has proven particularly fruitful in the context of differential systems, but it is interesting that stochastic and differential structures bear a number of similarities[7]: what

---

[6] Perhaps getting to the structural heart of the theory known as *QBism* [112, 113].

[7] Both conditional probability and differential calculus exhibit "chain rules" of similar types, which give rise to backwards actions that compose via the lens rule: in the former case, Bayesian inversion; in the latter, reverse differentiation. Categories that admit a differentiation operation have begun to be axiomatized (as differential categories [30]

are we to make of this? Does Bayesian inversion induce a canonical geometric morphism, by which structured models may be coherently updated? We have already indicated above signs of a relationship between inference and parallel transport; it seems that it may at least be fruitful to consider 'metric' topoi, appropriately enriched.

The second resolution is to work with topoi as we work with random dynamical systems, by noticing that randomness is often like "an uncertain parameterization". By parameterizing a topos with a category of noise sources, we may obtain a notion of "stochastic topos" in which the standard operations of dependent type theory are available, but where each type and term may depend on the realization of the noise source, thereby giving us notions of fuzzy term and fuzzy type. The mathematics of such uncertainly parameterized topoi is as yet undeveloped, although we expect that they should bear a relationship to the "topoi of beliefs" of the foregoing first resolution similar to the relationship of Fokker-Planck to random dynamical systems.

Finally, we note that higher topoi behave abstractly somewhat like vector spaces (with sheaves like categorified functionals). Since distributions are themselves like vectors, perhaps this observation is a first step towards relating the resolutions.

8.4.2. Dynamics

Chapter 6 has supplied the beginnings of a compositional coalgebraic theory for open stochastic and random dynamical systems in general time, and we hope that this theory could provide a home for a modern account of non-equilibrium systems, with the category of polynomial functors supplying a satisfactory account of these systems' interfaces (*i.e.*, the boundaries across which information flows, along which they compose, and through which they interact).

In this context, and in parallel to the abstract questions above, there are similar questions to be asked specifically of dynamical systems. For instance, what is the precise relationship between the category of Markov processes on an interface, and the category of random dynamical systems on that interface? We know that categories of deterministic discrete-time polynomial coalgebras are topoi [240], so does the same hold in general time? To what extent is the logic of our systems related to coalgebraic logics [72, 138, 162, 207, 275]?

Besides these 'parallel' questions, there are a number of more technical ones. For instance, our current definition of "Markov process on a polynomial interface" is somewhat inelegant, and we

---

and reverse-derivative categories [66]), and categories whose morphisms behave like stochastic channels are also presently being axiomatized (in the framework of Markov categories [109]), but the connections between these various formalisms are not yet clear. The similar structures indicate that the two families of axiomatisation may have a common generalization.

seek to simplify it. Similarly, we believe that there is a better definition of "random dynamical system on a polynomial interface" that may be obtained by a (different) generalization of the category of polynomial functors, using random variables. And we know that a topology for the cofree comonoid on an interface can be generated by the corresponding free monoid, which may be relevant for understanding the topological structure of open systems. An important set of open questions about open random dynamical systems in this framework comes from attempting to import notions about random systems from the classical 'closed' setting: fundamentally, we ask, does this framework indeed supply a satisfactory setting in which to understand stochastic systems away from equilibrium?

8.4.3. Computation

The early 21st century understanding of biological systems as information-processing involves treating them as *computational*, but remarkably lacks a precise concept of what it means for a system to compute, other than in the context of artificial machines. To us, it seems that a crisper understanding of computation in general might begin with the slogan that "computation is dynamics plus semantics", which is philosophically aligned with the semiotic understanding of biological information-processing sketched above: for example, we know that attractor networks in the brain can informally be understood as computational [11], but these are 'continuous' systems for which we do not yet have a good corresponding concept of *algorithm* (and it is upon algorithms that our current understanding is built). But what more is an algorithm than a description of a discrete-time open dynamical system? The quality that makes an algorithm computational is that its states or its outputs correspond to some quantity of interest, and that it reaches a fixed point (it halts) at the target quantity when the computation is complete. If this intuition is correct, then a new understanding of computation may follow the semiotic understanding of information-processing that we propose above: perhaps we could say more precisely that computation is the dynamics of semiosis? The time is right for such a reconceptualization, as human-made systems increasingly move away from von Neumann architectures towards more biosimilar ones (such as memristors, optical processors, neuromorphic technology, graph processors, or even many-core and mesh-based evolutions of classical processors).

A. Auxiliary material

A.1. From monads to multicategories

The assignment of domain and codomain to the morphisms of a small category $\mathcal{C}$ constitutes a pair of functions $\mathcal{C}_1 \to \mathcal{C}_0$, which we can write as a *span*, $\mathcal{C}_0 \xleftarrow{\text{cod}} \mathcal{C}_1 \xrightarrow{\text{dom}} \mathcal{C}_0$. Similarly, the assignment of domain and codomain to the morphisms of a multicategory $\mathcal{M}$ constitutes a span $\mathcal{M}_0 \xleftarrow{\text{cod}} \mathcal{M}_1 \xrightarrow{\text{dom}} \mathsf{List}(\mathcal{M}_0)$. This observation was used by Leinster [168] to construct a general framework for constructing multicategories, replacing List with an arbitrary 'Cartesian' monad T, which opens the way to a connection between monad algebras and multicategory algebras. In this section, we explore this connection, starting by defining categories of spans.

Definition A.1.1. Suppose A and B are two objects in a category $\mathcal{C}$. We will write a *span* from A to B as $(X, x) : A \xleftarrow{x_A} X \xrightarrow{x_B} B$, and call X the *apex* of the span and x_A, x_B its *legs* or *projections*. The *category of spans from A to B*, denoted $\mathbf{Span}(A, B)$ has spans (X, x) as objects, and the morphisms $f : (X, x) \to (X', x')$ are morphisms $f : X \to X'$ in $\mathcal{C}$ that commute with the spans, as in the following diagram:

$$
\begin{array}{ccccc}
 & & X & & \\
 & {}^{x_A}\swarrow & \downarrow{\scriptstyle f} & \searrow{}^{x_B} & \\
A & & & & B \\
 & {}_{x'_A}\nwarrow & \downarrow & \nearrow{}_{x'_B} & \\
 & & X' & &
\end{array}
$$
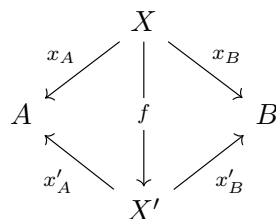
We can treat the categories $\mathbf{Span}(A, B)$ as the hom categories of a bicategory.

Definition A.1.2. Suppose $\mathcal{C}$ is a category with all pullbacks. The *bicategory of spans* in $\mathcal{C}$, denoted $\mathbf{Span}$, has for objects the objects of $\mathcal{C}$, and for hom-categories the categories $\mathbf{Span}(A, B)$ of spans from A to B. Given spans $(X, x) : A \to B$ and $(Y, y) : B \to C$, their horizontal composite

$(Y, y) \circ (X, x) : A \to C$ is the pullback span defined by

$$
\begin{array}{c}
X \times_B Y \\
{\scriptstyle \mathrm{proj}_X} \swarrow \quad \searrow {\scriptstyle \mathrm{proj}_Y} \\
X \qquad\qquad Y \\
{\scriptstyle x_A} \swarrow \quad \searrow {\scriptstyle x_B} \qquad {\scriptstyle y_B} \swarrow \quad \searrow {\scriptstyle y_C} \\
A \qquad\qquad B \qquad\qquad C
\end{array}
$$
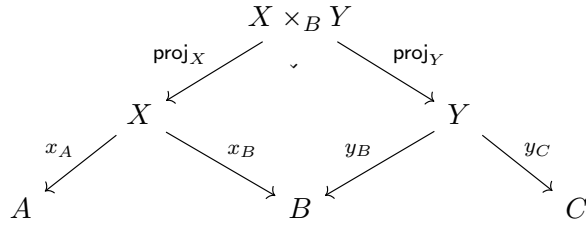
If $(X', x') : A \to B$ and $(Y', y') : B \to C$ are also spans with $f : (X, x) \Rightarrow (X', x')$ and $g : (Y, y) \Rightarrow (Y', y')$ vertical morphisms, the horizontal composite of f and g is also defined by pullback as $f \times_B g : (Y, y) \circ (X, x) \Rightarrow (Y', y') \circ (X', x')$. The identity span on an object A is $(A, \mathsf{id}) : A = A = A$.

If the ambient category $\mathcal{C}$ is not clear from the context, we will write $\mathbf{Span}_{\mathcal{C}}$ to denote the bicategory of spans in $\mathcal{C}$.

Remark A.1.3. Note that $\mathbf{Span}$ really is a bicategory rather than a 2-category: since the horizontal composition of spans is defined by pullback, it is only defined up to isomorphism. Consequently, the composition of spans can in general only be associative and unital up to isomorphism, rather than the strict equality required by a 2-category.

Now, recall that 'monad' is another name for "monoid in a bicategory", where the bicategory has so far been taken to be $\mathbf{Cat}$: but it need not be.

Remark A.1.4. Since $\mathcal{C}^{\mathcal{C}}$ is the endomorphism monoid on $\mathcal{C}$ in the bicategory $\mathbf{Cat}$, we can generalize the preceding definition of monad to any bicategory $\mathcal{B}$: a *monad in a bicategory* $\mathcal{B}$ is simply a monoid in $\mathcal{B}$, as defined in Remark 3.4.9. That is, a monad in $\mathcal{B}$ is a monoid object in the monoidal category $\big(\mathcal{B}(b, b), \circ, \mathsf{id}_b\big)$ for some choice of 0-cell $b : \mathcal{B}$, where $\circ$ denotes horizontal composition. Explicitly, a monad (t, μ, η) in $\mathcal{B}$ is a 1-cell $t : b \to b$, a multiplication 2-cell $\mu : t \circ t \Rightarrow t$, and a unit 2-cell $\eta : \mathsf{id}_b \Rightarrow t$, such that the associativity and unitality diagrams commute in $\mathcal{B}(b, b)$:

$$
\begin{array}{ccc}
ttt & \xrightarrow{\;t\mu\;} & tt \\
{\scriptstyle \mu_t}\downarrow & & \downarrow{\scriptstyle \mu} \\
tt & \xrightarrow{\;\mu\;} & t
\end{array}
\qquad \text{and} \qquad
\begin{array}{ccc}
t & \xrightarrow{\;\eta_t\;} tt \xleftarrow{\;t\eta\;} & t \\
& \searrow\;\downarrow{\scriptstyle \mu}\;\swarrow & \\
& t &
\end{array}
$$
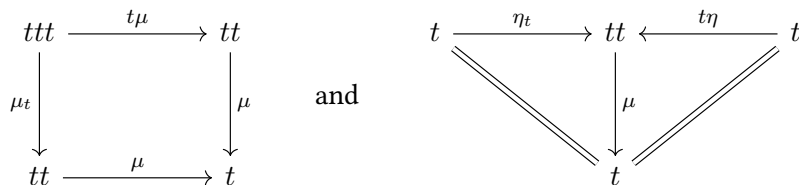
With this more general notion of monad, we obtain another monadic definition of "small category", to add to the explicit Definition 2.1.2 and the monad-algebraic Example 3.4.19.

Proposition A.1.5. Small categories are monads in $\mathbf{Span}_{\mathbf{Set}}$.

Proof. A monad in $\mathbf{Span_{Set}}$ is a choice of object $\mathcal{C}_0$ and monoid in $\mathbf{Span_{Set}}(\mathcal{C}_0, \mathcal{C}_0)$. Such a monoid is a span of sets $\mathcal{C} : \mathcal{C}_0 \xleftarrow{\text{cod}} \mathcal{C}_1 \xrightarrow{\text{dom}} \mathcal{C}_0$ along with functions $\bullet : \mathcal{C}_1 \times_{\mathcal{C}_0} \mathcal{C}_1 \to \mathcal{C}_1$ and $\text{id} : \mathcal{C}_0 \to \mathcal{C}_1$. The set $\mathcal{C}_1 \times_{\mathcal{C}_0} \mathcal{C}_1$ is the apex of the pullback span $\mathcal{C} \circ \mathcal{C}$ as in

$$
\begin{array}{ccccc}
& & \mathcal{C}_1 \times_{\mathcal{C}_0} \mathcal{C}_1 & & \\
& \swarrow & & \searrow & \\
\mathcal{C}_1 & & & & \mathcal{C}_1 \\
\text{cod} \swarrow & \searrow \text{dom} & & \text{cod} \swarrow & \searrow \text{dom} \\
\mathcal{C}_0 & & \mathcal{C}_0 & & \mathcal{C}_0
\end{array}
$$
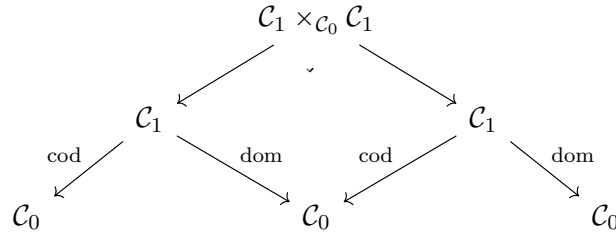
so that $\bullet$ and id make the following diagrams commute:

$$
\begin{array}{ccc}
& \mathcal{C}_1 \times_{\mathcal{C}_0} \mathcal{C}_1 & \\
\swarrow \quad \downarrow \bullet & & \searrow \\
\mathcal{C}_1 \qquad\qquad & & \mathcal{C}_1 \\
\text{cod} \downarrow \qquad & & \downarrow \text{dom} \\
\mathcal{C}_0 \xleftarrow{\text{cod}} \mathcal{C}_1 \xrightarrow{\text{dom}} & & \mathcal{C}_0
\end{array}
\quad \text{and} \quad
\begin{array}{ccc}
& \mathcal{C}_0 & \\
\swarrow \quad \downarrow \text{id} & & \searrow \\
\mathcal{C}_0 \xleftarrow{\text{cod}} \mathcal{C}_1 & & \xrightarrow{\text{dom}} \mathcal{C}_0
\end{array}
$$
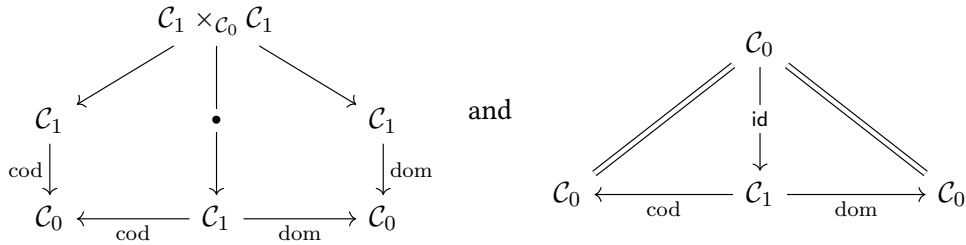
This means that $\text{cod}(g \bullet f) = \text{cod}(g)$ and $\text{dom}(g \bullet f) = \text{dom}(f)$, and $\text{cod}(\text{id}_x) = \text{dom}(\text{id}_x) = x$. It is easy to check that $(\mathcal{C}, \bullet, \text{id})$ therefore constitutes the data of a small category; moreover, the functions $\bullet$ and id must satisfy the monoid axioms of associativity and (right and left) unitality, which correspond directly to the categorical axioms of associativity and unitality. $\qquad\square$

As we indicated at the opening of this section, by generalizing to a category of 'spans' of the form $A \leftarrow X \to TB$, we can use the preceding result to produce generalized multicategories whose morphisms have domains "in the shape of T". Since the horizontal composition of spans is by pullback, we need an extra condition on the monad T to ensure that pullbacks of T-spans are well-defined. This condition is known as 'Cartesianness'.

Definition A.1.6. A *Cartesian natural transformation* between functors F and G is a natural transformation $\alpha : F \Rightarrow G$ for which every naturality square is a pullback:

$$
\begin{array}{ccc}
Fa & \xrightarrow{\alpha_a} & Ga \\
Ff \downarrow & & \downarrow Gf \\
Fb & \xrightarrow{\alpha_b} & Gb
\end{array}
$$
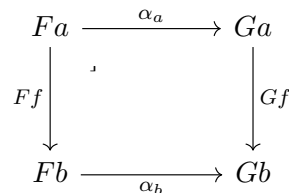
A *Cartesian monad* is a monad $(T : \mathcal{C} \to \mathcal{C}, \mu, \eta)$ such that $\mathcal{C}$ has all pullbacks, T preserves these pullbacks (sending pullback squares to pullback squares), and μ and η are Cartesian natural transformations.

Definition A.1.7. Suppose T is a monad on $\mathcal{C}$. A T-*span* from A to B is a span from A to TB in $\mathcal{C}$. The *category of T-spans* from A to B, denoted $\mathbf{Span}_T(A, B)$ has T-spans as objects, and the morphisms $f : (X, x) \to (X', x')$ are morphisms $f : X \to X'$ in $\mathcal{C}$ that commute with the spans, as in the diagram

$$
\begin{array}{ccc}
 & X & \\
{}^{x_A}\swarrow & \downarrow f & \searrow^{x_B} \\
A & & TB \\
{}_{x'_A}\nwarrow & \downarrow & \nearrow_{x'_B} \\
 & X' &
\end{array}
\qquad .
$$
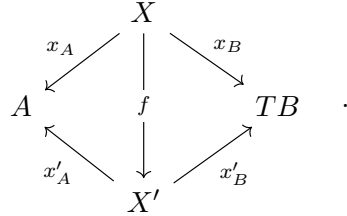
Definition A.1.8. Suppose (T, μ, η) is a Cartesian monad on $\mathcal{C}$. The *bicategory of T-spans* in $\mathcal{C}$, denoted $\mathbf{Span}_T$, has for objects the objects of $\mathcal{C}$, and for hom-categories the categories $\mathbf{Span}_T(A, B)$ of T-spans from A to B. Given T-spans $(X, x) : A \to B$ and $(Y, y) : B \to C$, their horizontal composite $(Y, y) \circ (X, x) : A \to C$ is the outer T-span in the diagram

$$
\begin{array}{ccccccc}
 & & X \times_{TB} TY & & & & \\
 & {}^{\mathsf{proj}_X}\swarrow & \llcorner & \searrow^{\mathsf{proj}_{TY}} & & & \\
 & X & & & TY & & \\
{}^{x_A}\swarrow & & \searrow^{x_B} & {}^{Ty_B}\swarrow & & \searrow^{Ty_C} & \\
A & & TB & & & TTC & \\
 & & & & & & \searrow^{\mu_C} \\
 & & & & & & TC
\end{array}
\qquad .
$$
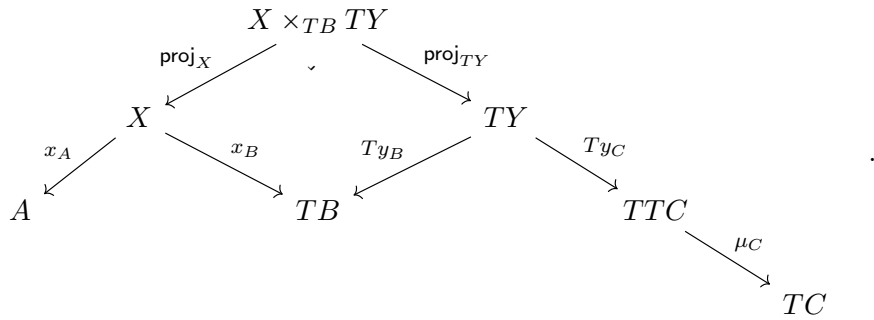
If $(X', x') : A \to B$ and $(Y', y') : B \to C$ are also T-spans with $f : (X, x) \Rightarrow (X', x')$ and $g : (Y, y) \Rightarrow (Y', y')$ vertical morphisms, the horizontal composite of f and g is defined as $f \times_{TB} Tg$ accordingly. The identity span on an object A is $A \xleftarrow{\mathsf{id}_A} A \xrightarrow{\eta_A} TA$.

With these notions to hand, the general concept of T-multicategory is easy to define.

Definition A.1.9 (Leinster [168, Def. 4.2.2]). Suppose T is a Cartesian monad on $\mathcal{C}$. A T-*multicategory* is a monad in the bicategory $\mathbf{Span}_T$ of T-spans.

And of course we can recover our earlier examples of category shapes accordingly.

Example A.1.10. The identity monad on a category with all pullbacks is trivially a Cartesian monad. Therefore, taking $T = \mathsf{id}_{\mathbf{Set}}$ to be the identity monad on $\mathbf{Set}$, we immediately see that an $\mathsf{id}_{\mathbf{Set}}$-multicategory is a small category.

Example A.1.11 (Leinster [168, Examples 4.1.4 and 4.2.7]). The free monoid monad List : $\mathbf{Set} \to \mathbf{Set}$ is Cartesian. Unpacking the definitions, we find that a List-multicategory is precisely a multicategory as in Definition 3.3.1.

At this point, we can sketch how multicategory algebras correspond to monad algebras, referring the reader to Leinster [168, §4.3] for the details. The basic picture is that, if $T : \mathcal{C} \to \mathcal{C}$ is a Cartesian monad and $\mathcal{M}$ is a T-multicategory, then one can obtain functorially a monad $T_\mathcal{M}$ on the slice $\mathcal{C}/\mathcal{M}_0$ of $\mathcal{C}$ over the object $\mathcal{M}_0$ of $\mathcal{M}$-objects. The algebras $\alpha : T_\mathcal{M}(X, p) \to (X, p)$ of this monad are morphisms $\alpha : T_\mathcal{M}X \to X$ as in the commuting diagram

$$
\begin{array}{ccc}
T_\mathcal{M}X & \xrightarrow{\quad\alpha\quad} & X \\
\swarrow \qquad \searrow & & \downarrow p \\
TX \qquad\qquad \mathcal{M}_1 & & \\
\searrow^{Tp} \quad \swarrow_{\mathrm{dom}} \quad \searrow^{\mathrm{cod}} & & \downarrow \\
T\mathcal{M}_0 & & \mathcal{M}_0
\end{array}
$$

where $T_\mathcal{M}(X, p)$ is defined as the bundle $T_\mathcal{M}X \to \mathcal{M}_0$ on the right leg of the pullback square. To get a sense for how this works, consider the case where $T = \mathrm{id}_{\mathbf{Set}}$: a T-multicategory is then simply a small category $\mathcal{C}$, and as Leinster [168, Example 4.3.2] shows, its algebras are functors $\mathcal{C} \to \mathbf{Set}$.

B. Bibliography

[1] Samson Abramsky and Giovanni Carù. "Non-Locality, Contextuality and Valuation Algebras: A General Theory of Disagreement". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 377.2157 (09/2019), p. 20190036. DOI: `10.1098/rsta.2019.0036`.

[2] Samson Abramsky and Bob Coecke. "A Categorical Semantics of Quantum Protocols". In: *Logic in Computer Science, 2004. Proceedings of the 19th Annual IEEE Symposium On*. IEEE. 2004, pp. 415–425.

[3] Samson Abramsky and Viktor Winschel. "Coalgebraic Analysis of Subgame-perfect Equilibria in Infinite Games without Discounting". 10/16/2012. DOI: `10.1017/S0960129515000365`. arXiv: `1210.4537 [cs.GT]`.

[4] Samson Abramsky et al. "Categorical Methods at the Crossroads (Dagstuhl Perspectives Workshop 14182)". In: (2014). In collab. with Marc Herbstritt, 15 pages. DOI: `10.4230/DAGREP.4.4.49`. URL: `http://drops.dagstuhl.de/opus/volltexte/2014/4618/` (visited on 12/23/2022).

[5] Samson Abramsky et al. "Contextuality, Cohomology and Paradox". In: *24th EACSL Annual Conference on Computer Science Logic (CSL 2015), Leibniz International Proceedings in Informatics (LIPIcs), 41: 211-228, 2015* (02/10/2015). DOI: `10.4230/LIPIcs.CSL.2015.211`. arXiv: `1502.03097 [quant-ph]`.

[6] Diederik Aerts et al. "Quantum Cognition Beyond Hilbert Space: Fundamentals and Applications". In: *Lecture Notes in Computer Science* (2017), pp. 81–98. ISSN: 1611-3349. DOI: `10.1007/978-3-319-52289-0_7`.

[7] Danel Ahman and Tarmo Uustalu. "Directed Containers as Categories". In: *EPTCS 207, 2016, pp. 89-98* (04/05/2016). DOI: `10.4204/EPTCS.207.5`. arXiv: `1604.01187 [cs.LO]`.

[8] Thosten Altenkirch, James Chapman, and Tarmo Uustalu. "Monads Need Not Be Endofunctors". In: *Logical Methods in Computer Science* Volume 11, Issue 1 (03/06/2015), p. 928. ISSN: 1860-5974. DOI: `10.2168/LMCS-11(1:3)2015`. URL: `https://lmcs.episciences.org/928` (visited on 10/16/2023).

[9] Shun-ichi Amari. "Information Geometry". In: *Contemporary Mathematics*. Ed. by Hanna Nencka and Jean-Pierre Bourguignon. Vol. 203. Providence, Rhode Island: American Mathematical Society, 1997, pp. 81–95. ISBN: 978-0-8218-0607-4 978-0-8218-7794-4. DOI:

10.1090/conm/203/02554. URL: http://www.ams.org/conm/203/ (visited on 12/23/2022).

[10]   Shun'ichi Amari. *Information Geometry and Its Applications*. Applied Mathematical Sciences volume 194. Japan: Springer, 2016. 374 pp. ISBN: 978-4-431-55977-1.

[11]   Daniel J Amit. *Modeling Brain Function: The World of Attractor Neural Networks*. Cambridge University Press, 1992.

[12]   Britt Anderson et al. "Category Theory for Cognitive Science". In: *Proceedings of the Annual Meeting of the Cognitive Science Society*. Vol. 44. 2022.

[13]   Ludwig Arnold. *Random Dynamical Systems*. Springer Berlin Heidelberg, 1998. 612 pp. ISBN: 978-3-540-63758-5. DOI: 10.1007/978-3-662-12878-7. URL: https://doi.org/10.1007/978-3-662-12878-7.

[14]   Robert Atkey. "Syntax and Semantics of Quantitative Type Theory". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. ACM, 07/2018. DOI: 10.1145/3209108.3209189.

[15]   Robert J. Aumann. "Borel Structures for Function Spaces". In: *Illinois Journal of Mathematics* 5.4 (12/01/1961). ISSN: 0019-2082. DOI: 10.1215/ijm/1255631584. URL: https://projecteuclid.org/journals/illinois-journal-of-mathematics/volume-5/issue-4/Borel-structures-for-function-spaces/10.1215/ijm/1255631584.full (visited on 06/20/2023).

[16]   John C. Baez and James Dolan. "Higher-dimensional Algebra and Topological Quantum Field Theory". In: *Journal of Mathematical Physics* 36.11 (11/1995), pp. 6073–6105. ISSN: 0022-2488, 1089-7658. DOI: 10.1063/1.531236. URL: http://aip.scitation.org/doi/10.1063/1.531236 (visited on 12/23/2022).

[17]   John C. Baez and James Dolan. "Higher-Dimensional Algebra III: N-Categories and the Algebra of Opetopes". In: *Adv. Math. 135 (1998), 145-206.* (02/10/1997). arXiv: q-alg/9702014.

[18]   John C. Baez et al. "Network Models". In: *Theory and Applications of Categories, Vol. 35, 2020, No. 20, pp 700-744* (10/31/2017). arXiv: 1711.00037 [math.CT].

[19]   Igor Baković. "Fibrations of Bicategories". In: (2010). URL: https://www2.irb.hr/korisnici/ibakovic/groth2fib.pdf.

[20]   Marcello Barbieri, ed. *Introduction to Biosemiotics The New Biological Synthesis. The New Biological Synthesis*. Springer, 2007. ISBN: 978-1-4020-4814-2.

[21]   A. M. Bastos et al. "Canonical Microcircuits for Predictive Coding". In: *Neuron* 76.4 (11/2012), pp. 695–711. DOI: 10.1016/j.neuron.2012.10.038. pmid: 23177956.

[22] Matthew J. Beal and Zoubin Ghahramani. "The Variational Bayesian EM Algorithm for Incomplete Data: With Application to Scoring Graphical Model Structures". In: Bayesian Statistics. Vol. 7. Valencia: Oxford University Press, 2003.

[23] C. Glenn Begley and Lee M. Ellis. "Raise Standards for Preclinical Cancer Research". In: *Nature* 483.7391 (03/2012), pp. 531–533. DOI: `10.1038/483531a`.

[24] Timothy EJ Behrens et al. "What Is a Cognitive Map? Organizing Knowledge for Flexible Behavior". In: *Neuron* 100.2 (2018), pp. 490–509.

[25] Richard Bellman. "The Theory of Dynamic Programming". In: *Bulletin of the American Mathematical Society* 60.6 (1954), pp. 503–515. ISSN: 0273-0979, 1088-9485. DOI: `10.1090/S0002-9904-1954-09848-8`. URL: `https://www.ams.org/bull/1954-60-06/S0002-9904-1954-09848-8/` (visited on 05/17/2023).

[26] Jacob LS Bellmund et al. "Navigating Cognition: Spatial Codes for Human Thinking". In: *Science (New York, N.Y.)* 362.6415 (2018, 2018-11), eaat6766. DOI: `10.1126/science.aat6766`.

[27] Silvia Bernardi et al. "The Geometry of Abstraction in Hippocampus and Pre-Frontal Cortex". In: (09/2018). DOI: `10.1101/408633`.

[28] Martin Biehl, Felix A. Pollock, and Ryota Kanai. "A Technical Critique of the Free Energy Principle as Presented in "Life as We Know It" and Related Works". 01/12/2020. arXiv: `2001.06408v2 [q-bio.NC]`.

[29] Achim Blumensath and Viktor Winschel. "A Compositional Coalgebraic Semantics of Strategic Games". 12/22/2017. arXiv: `1712.08381v1 [cs.GT]`.

[30] R. F. Blute, J. R. B. Cockett, and R. A. G. Seely. "Differential Categories". In: *Mathematical Structures in Computer Science* 16.06 (11/2006), p. 1049. DOI: `10.1017/s0960129506005676`.

[31] Cristian Bodnar et al. *Neural Sheaf Diffusion: A Topological Perspective on Heterophily and Oversmoothing in GNNs*. 10/21/2022. arXiv: `2202.04579 [cs, math]`. URL: `http://arxiv.org/abs/2202.04579` (visited on 12/23/2022). preprint.

[32] M Boerlin et al. "Predictive Coding of Dynamical Variables in Balanced Spiking Networks". In: *PLoS computational biology* 9.11 (2013), e1003258. URL: `http://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1003258`.

[33] Rafal Bogacz. "A Tutorial on the Free-Energy Framework for Modelling Perception and Learning". In: *Journal of Mathematical Psychology* 76 (02/2017), pp. 198–211. DOI: `10.1016/j.jmp.2015.11.003`.

[34] Aaron Bohannon, Benjamin C Pierce, and Jeffrey A Vaughan. "Relational Lenses: A Language for Updatable Views". In: *Proceedings of the Twenty-Fifth ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems*. ACM. 2006, pp. 338–347.

[35] Guillaume Boisseau. "String Diagrams for Optics". 02/11/2020. arXiv: `2002.11480v1 [math.CT]`.

[36] Joe Bolt, Jules Hedges, and Philipp Zahn. "Bayesian Open Games". 10/08/2019. arXiv: `1910.03656v1 [cs.GT]`.

[37] Joe Bolt et al. "Interacting Conceptual Spaces I: Grammatical Composition of Concepts". In: Springer, 2019, 2017, pp. 151–181. arXiv: `1703.08314 [cs.LO]`.

[38] Evert A. Boonstra and Heleen A. Slagter. "The Dialectics of Free Energy Minimization". In: *Frontiers in Systems Neuroscience* 13 (09/10/2019), p. 42. ISSN: 1662-5137. DOI: `10.3389/fnsys.2019.00042`. URL: `https://www.frontiersin.org/article/10.3389/fnsys.2019.00042/full` (visited on 12/23/2022).

[39] Francis Borceux. *Handbook of Categorical Algebra 2. Categories and Structures*. Vol. 51. Encyclopedia of Mathematics and Its Applications. Cambridge University Press, Cambridge, 1994. ISBN: 978-0-521-44179-7.

[40] Tai-Danae Bradley. "Entropy as a Topological Operad Derivation". In: *Entropy* 23.9 (09/09/2021), p. 1195. ISSN: 1099-4300. DOI: `10.3390/e23091195`. URL: `https://www.mdpi.com/1099-4300/23/9/1195` (visited on 05/02/2023).

[41] Dylan Braithwaite and Jules Hedges. *Dependent Bayesian Lenses: Categories of Bidirectional Markov Kernels with Canonical Bayesian Inversion*. 09/29/2022. arXiv: `2209.14728 [cs, math, stat]`. URL: `http://arxiv.org/abs/2209.14728` (visited on 12/17/2022). preprint.

[42] Dylan Braithwaite, Jules Hedges, and Toby St Clere Smithe. *The Compositional Structure of Bayesian Inference*. 05/10/2023. arXiv: `2305.06112 [cs, math]`. URL: `http://arxiv.org/abs/2305.06112` (visited on 05/11/2023). preprint.

[43] Dylan Braithwaite et al. *Fibre Optics*. 12/21/2021. arXiv: `2112.11145 [math.CT]`. preprint.

[44] Michael M. Bronstein et al. *Geometric Deep Learning: Grids, Groups, Graphs, Geodesics, and Gauges*. 05/02/2021. arXiv: `2104.13478 [cs, stat]`. URL: `http://arxiv.org/abs/2104.13478` (visited on 12/23/2022). preprint.

[45] Kenneth S. Brown. "Abstract Homotopy Theory and Generalized Sheaf Cohomology". In: *Transactions of the American Mathematical Society* 186 (1973), pp. 419–419. DOI: `10.1090/s0002-9947-1973-0341469-9`. JSTOR: 1996573. URL: `https://www.jstor.org/stable/1996573`.

[46] N. G. de Bruijn. *Asymptotic Methods in Analysis*. Dover ed. New York: Dover Publications, 1981. 200 pp. ISBN: 978-0-486-64221-5.

[47] Iva K. Brunec and Ida Momennejad. "Predictive Representations in Hippocampal and Prefrontal Hierarchies". In: *bioRxiv : the preprint server for biology* (2019). DOI: `10.1101/`

786434. eprint: `https://www.biorxiv.org/content/early/2019/09/30/786434.full.pdf`. URL: `https://www.biorxiv.org/content/early/2019/09/30/786434`.

[48] Christopher L Buckley et al. "The Free Energy Principle for Action and Perception: A Mathematical Review". In: *Journal of Mathematical Psychology* 81 (05/24/2017), pp. 55–79. DOI: `10.1016/j.jmp.2017.09.004`. arXiv: `1705.09156v1 [q-bio.NC]`.

[49] Jerome R Busemeyer and Peter D Bruza. *Quantum Models of Cognition and Decision*. Cambridge University Press, 2012.

[50] Matteo Capucci. *Seeing Double through Dependent Optics*. 04/22/2022. arXiv: `2204.10708 [math.CT]`. preprint.

[51] Matteo Capucci. "Diegetic Representation of Feedback in Open Games". In: *Electronic Proceedings in Theoretical Computer Science*. Applied Category Theory 2022. Vol. 380. 08/07/2023, pp. 145–158. DOI: `10.4204/EPTCS.380.9`. URL: `http://arxiv.org/abs/2206.12338v3` (visited on 09/29/2023).

[52] Matteo Capucci and Bruno Gavranović. "Actegories for the Working Amthematician". 03/30/2022. arXiv: `2203.16351 [math.CT]`.

[53] Matteo Capucci, Bruno Gavranović, and Toby St Clere Smithe. "Parameterized Categories and Categories by Proxy". In: *Category Theory 2021*. Category Theory. Genoa, 2021. URL: `https://www.youtube.com/watch?v=lYs6Bs6JK9Q`.

[54] Matteo Capucci et al. "Towards Foundations of Categorical Cybernetics". In: *Electronic Proceedings in Theoretical Computer Science*. Applied Category Theory 2021. Vol. 372. 11/03/2022, pp. 235–248. DOI: `10.4204/EPTCS.372.17`. URL: `http://arxiv.org/abs/2105.06332v2` (visited on 09/29/2023).

[55] Olivia Caramello. *A Topos-Theoretic Approach to Stone-type Dualities*. 03/17/2011. arXiv: `1103.3493 [math]`. URL: `http://arxiv.org/abs/1103.3493` (visited on 05/17/2023). preprint.

[56] Henri Cartan. "Variétés Analytiques Complexes et Cohomologie". In: Colloque Sur Les Fonctions de Plusieurs Variables Tenu a Bruxelles. 1953. URL: `https://www.inp.nsk.su/~silagadz/Cartan.pdf`.

[57] Henri Paul Cartan and Samuel Eilenberg. *Homological Algebra*. Princeton Landmarks in Mathematics and Physics. Princeton, N.J: Princeton University Press, 1956. 390 pp. ISBN: 978-0-691-04991-5.

[58] Giovanni Carù. "Logical and Topological Contextuality in Quantum Mechanics and Beyond". University of Oxford / University of Oxford / University of Oxford, 2019. URL: `https://ora.ox.ac.uk/objects/uuid:9bc2335a-b627-463b-9526-f4b881b0fbbf`.

[59] Eugenia Cheng. "Weak N-Categories: Opetopic and Multitopic Foundations". 04/21/2003. arXiv: math/0304277.

[60] Kenta Cho and Bart Jacobs. "Disintegration and Bayesian Inversion via String Diagrams". In: *Math. Struct. Comp. Sci. 29 (2019) 938-971* (08/29/2017). DOI: 10.1017/S0960129518000488. arXiv: 1709.00322v3 [cs.AI].

[61] Kenta Cho et al. "An Introduction to Effectus Theory". 2015. arXiv: 1512.05813 [cs.LO].

[62] Corina Cirstea. "An Algebra-Coalgebra Framework for System Specification". In: *Electronic Notes in Theoretical Computer Science* 33 (2000), pp. 80–110.

[63] Bryce Clarke. "Internal Lenses as Functors and Cofunctors". In: *Electronic Proceedings in Theoretical Computer Science* 323 (09/15/2020), pp. 183–195. ISSN: 2075-2180. DOI: 10.4204/EPTCS.323.13. URL: http://arxiv.org/abs/2009.06835v1 (visited on 12/23/2022).

[64] Bryce Clarke and Matthew Di Meglio. *An Introduction to Enriched Cofunctors*. 09/02/2022. arXiv: 2209.01144 [math]. URL: http://arxiv.org/abs/2209.01144 (visited on 12/23/2022). preprint.

[65] Bryce Clarke et al. "Profunctor Optics, a Categorical Update". 01/21/2020. arXiv: 2001.07488v1 [cs.PL].

[66] Robin Cockett et al. "Reverse Derivative Categories". In: *CSL 2020*. 28th International Conference on Computer Science Logic. 2019-10-15, 2020. arXiv: 1910.07065v1 [cs.LO].

[67] Bob Coecke. "The Mathematics of Text Structure". 2019. arXiv: 1904.03478.

[68] Bob Coecke and Aleks Kissinger. "Categorical Quantum Mechanics I: Causal Quantum Processes". 10/19/2015. arXiv: 1510.05468v3 [quant-ph].

[69] Bob Coecke and Aleks Kissinger. "Categorical Quantum Mechanics II: Classical-Quantum Interaction". 05/27/2016. arXiv: 1605.08617v1 [quant-ph].

[70] Bob Coecke, Mehrnoosh Sadrzadeh, and Stephen Clark. "Mathematical Foundations for a Compositional Distributional Model of Meaning". In: *Lambek Festschirft, special issue of Linguistic Analysis,* 2010. (03/2010). eprint: 1003.4394. URL: https://arxiv.org/abs/1003.4394.

[71] Matteo Colombo and Cory Wright. "First Principles in the Life Sciences: The Free-Energy Principle, Organicism, and Mechanism". In: *Synthese* 198.S14 (06/2021), pp. 3463–3488. ISSN: 0039-7857, 1573-0964. DOI: 10.1007/s11229-018-01932-w. URL: https://link.springer.com/10.1007/s11229-018-01932-w (visited on 12/23/2022).

[72] David Corfield. *Coalgebraic Modal Logic*. 2009. URL: `https://golem.ph.utexas.edu/category/2009/09/coalgebraic_modal_logic.html`.

[73] Lancelot Da Costa et al. "Reward Maximisation through Discrete Active Inference". 07/11/2022. arXiv: `2009.08111 [cs.AI]`.

[74] Geoffrey S. H. Cruttwell et al. "Categorical Foundations of Gradient-Based Learning". In: *Programming Languages and Systems*. Springer International Publishing, 2022, pp. 1–28. DOI: `10.1007/978-3-030-99336-8_1`.

[75] Justin Curry. "Sheaves, Cosheaves and Applications". 03/13/2013. arXiv: `1303.3255v2 [math.AT]`.

[76] Lancelot Da Costa et al. "Active Inference on Discrete State-Spaces: A Synthesis". 01/20/2020. arXiv: `2001.07203 [q-bio.NC]`.

[77] Brian Day and Ross Street. "Monoidal Bicategories and Hopf Algebroids". In: *Advances in Mathematics* 129.1 (07/1997), pp. 99–157. ISSN: 00018708. DOI: `10.1006/aima.1997.1649`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0001870897916492` (visited on 12/17/2022).

[78] Peter Dayan. "Improving Generalization for Temporal Difference Learning: The Successor Representation". In: *Neural Computation* 5.4 (07/1993), pp. 613–624. ISSN: 1530-888X. DOI: `10.1162/neco.1993.5.4.613`.

[79] Peter Dayan and Laurence F Abbott. *Theoretical Neuroscience*. Vol. 806. Cambridge, MA: MIT Press, 2001.

[80] Peter Dayan et al. "The Helmholtz Machine". In: *Neural Computation* 7.5 (09/1995), pp. 889–904. DOI: `10.1162/neco.1995.7.5.889`. URL: `http://dx.doi.org/10.1162/neco.1995.7.5.889`.

[81] Sophie Deneve. "Making Decisions with Unknown Sensory Reliability". In: *Frontiers in neuroscience* 6 (2012). URL: `http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3367295/`.

[82] Sophie Denève and Christian K Machens. "Efficient Codes and Balanced Networks". In: *Nature neuroscience* 19.3 (02/2016), pp. 375–382. ISSN: 1546-1726. DOI: `10.1038/nn.4243`.

[83] Bert de Vries and Karl J. Friston. "A Factor Graph Description of Deep Temporal Active Inference". In: *Frontiers in Computational Neuroscience* 11 (10/2017). DOI: `10.3389/fncom.2017.00095`.

[84] A Ehresmann and J Vanbremeersch. "Hierarchical Evolutive Systems: A Mathematical Model for Complex Systems". In: *Bulletin of Mathematical Biology* 49.1 (1987), pp. 13–50. ISSN: 00928240. DOI: `10.1016/S0092-8240(87)80033-2`. URL: `http://link.springer.com/10.1016/S0092-8240(87)80033-2` (visited on 12/23/2022).

[85] A.C. Ehresmann and J.P. Vanbremeersch. *Memory Evolutive Systems; Hierarchy, Emergence, Cognition*. Studies in Multidisciplinarity. Elsevier Science, 2007. ISBN: 978-0-08-055541-6. URL: `https://books.google.co.uk/books?id=OqcYQbY79GMC`.

[86] Andrée C. Ehresmann and Jean-Paul Vanbremeersch. "The Memory Evolutive Systems as a Model of Rosen's Organisms – (Metabolic, Replication) Systems". In: *Axiomathes* 16.1-2 (03/2006), pp. 137–154. ISSN: 1572-8390. DOI: `10.1007/s10516-005-6001-0`. URL: `http://link.springer.com/10.1007/s10516-005-6001-0`.

[87] Samuel Eilenberg and Saunders MacLane. "General Theory of Natural Equivalences". In: *Transactions of the American Mathematical Society* 58.0 (1945), pp. 231–294. ISSN: 0002-9947, 1088-6850. DOI: `10.1090/S0002-9947-1945-0013131-6`. URL: `https://www.ams.org/tran/1945-058-00/S0002-9947-1945-0013131-6/` (visited on 12/23/2022).

[88] Russell A Epstein et al. "The Cognitive Map in Humans: Spatial Navigation and Beyond". In: *Nature Neuroscience* 20.11 (10/2017), pp. 1504–1513. DOI: `10.1038/nn.4656`.

[89] Martín Escardó and Paulo Oliva. "Selection Functions, Bar Recursion and Backward Induction". In: *Mathematical Structures in Computer Science* 20.2 (03/2010), pp. 127–168. DOI: `10.1017/s0960129509990351`.

[90] Chris Fields, James F. Glazebrook, and Antonino Marciano. "The Physical Meaning of the Holographic Principle". In: *Quanta* 11.1 (11/21/2022), pp. 72–96. ISSN: 1314-7374. DOI: `10.12743/quanta.v11i1.206`. arXiv: `2210.16021 [gr-qc, physics:hep-th, physics:quant-ph]`. URL: `http://arxiv.org/abs/2210.16021` (visited on 12/23/2022).

[91] Chris Fields et al. "A Free Energy Principle for Generic Quantum Systems". In: *Progress in Biophysics and Molecular Biology* 173 (09/2022), pp. 36–59. DOI: `10.1016/j.pbiomolbio.2022.05.006`.

[92] John Rupert Firth. "A Synopsis of Linguistic Theory, 1930-1955". In: *Studies in Linguistic Analysis*. Oxford: Basil Blackwell, 1957, pp. 1–32.

[93] Brendan Fong. "Decorated Cospans". In: *Theory and Applications of Categories* 30.33 (2015), pp. 1096–1120. arXiv: `1502.00872`. URL: `http://www.tac.mta.ca/tac/volumes/30/33/30-33abs.html`.

[94] Brendan Fong. "The Algebra of Open and Interconnected Systems". University of Oxford, 2016.

[95] Brendan Fong. "Causal Theories: A Categorical Perspective on Bayesian Networks". University of Oxford, 2013-01-26, 2013. arXiv: `1301.6201v1 [math.PR]`. URL: `http://arxiv.org/abs/1301.6201`.

[96] Brendan Fong and Michael Johnson. "Lenses and Learners". In: *In: J. Cheney, H-S. Ko (eds.): Proceedings of the Eighth International Workshop on Bidirectional Transformations (Bx*

2019), Philadelphia, PA, USA, June 4, 2019, published at http://ceur-ws.org (03/05/2019). arXiv: `1903.03671v2 [cs.LG]`.

[97]   Brendan Fong and David I. Spivak. *Seven Sketches in Compositionality: An Invitation to Applied Category Theory.* 2018. arXiv: `1803.05316v3 [math.CT]`.

[98]   Brendan Fong and David I. Spivak. "Hypergraph Categories". In: *Journal of Pure and Applied Algebra* 223.11 (11/2019), pp. 4746–4777. ISSN: 00224049. DOI: `10.1016/j.jpaa.2019.02.014`. arXiv: `1806.08304v3 [math.CT]`. URL: `https://linkinghub.elsevier.com/retrieve/pii/S0022404919300489` (visited on 12/17/2022).

[99]   J. Nathan Foster et al. "Combinators for Bidirectional Tree Transformations". In: *ACM Transactions on Programming Languages and Systems* 29.3 (05/2007), p. 17. DOI: `10.1145/1232420.1232424`.

[100]   K. Friston. "A Theory of Cortical Responses". In: *Philos. Trans. R. Soc. Lond., B, Biol. Sci.* 360.1456 (04/2005), pp. 815–836. DOI: `10.1098/rstb.2005.1622`. PMID: `15937014`.

[101]   K. Friston et al. "Variational Free Energy and the Laplace Approximation". In: *Neuroimage* 34.1 (01/2007), pp. 220–234. DOI: `10.1016/j.neuroimage.2006.08.035`. PMID: `17055746`.

[102]   Karl Friston. "A Free Energy Principle for a Particular Physics". 06/24/2019. arXiv: `1906.10184v1 [q-bio.NC]`.

[103]   Karl Friston, Lancelot Da Costa, and Thomas Parr. "Some Interesting Observations on the Free Energy Principle". 02/05/2020. arXiv: `2002.04501v1 [q-bio.NC]`.

[104]   Karl Friston and Stefan Kiebel. "Predictive Coding under the Free-Energy Principle". In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 364.1521 (2009), pp. 1211–1221. DOI: `10.1098/rstb.2008.0300`. URL: `http://m.rstb.royalsocietypublishing.org/content/364/1521/1211`.

[105]   Karl Friston et al. "Generalised Filtering". In: *Mathematical Problems in Engineering* 2010 (2010), pp. 1–34. DOI: `10.1155/2010/621670`.

[106]   Karl J. Friston, Jean Daunizeau, and Stefan J. Kiebel. "Reinforcement Learning or Active Inference?" In: *PLoS ONE* 4.7 (07/2009). Ed. by Olaf Sporns, e6421. DOI: `10.1371/journal.pone.0006421`.

[107]   Karl J. Friston and Klaas E. Stephan. "Free-Energy and the Brain". In: *Synthese. An International Journal for Epistemology, Methodology and Philosophy of Science* 159.3 (09/2007), pp. 417–458. DOI: `10.1007/s11229-007-9237-y`. URL: `http://dx.doi.org/10.1007/s11229-007-9237-y`.

[108] Karl J. Friston et al. "Action and Behavior: A Free-Energy Formulation". In: *Biological Cybernetics* 102.3 (02/2010), pp. 227–260. DOI: `10.1007/s00422-010-0364-z`.

[109] Tobias Fritz. "A Synthetic Approach to Markov Kernels, Conditional Independence and Theorems on Sufficient Statistics". In: *Advances in Mathematics* 370.107239 (08/19/2019). DOI: `10.1016/j.aim.2020.107239`. arXiv: `1908.07021v3 [math.ST]`.

[110] Tobias Fritz and Paolo Perrone. "Bimonoidal Structure of Probability Monads". In: *Electronic Notes in Theoretical Computer Science* 341 (12/2018), pp. 121–149. ISSN: 15710661. DOI: `10.1016/j.entcs.2018.11.007`. arXiv: `1804.03527 [cs, math]`. URL: `http://arxiv.org/abs/1804.03527` (visited on 06/20/2023).

[111] Peng Fu, Kohei Kishida, and Peter Selinger. "Linear Dependent Type Theory for Quantum Programming Languages". 04/28/2020. arXiv: `2004.13472 [cs.PL]`.

[112] Christopher A. Fuchs. "Notwithstanding Bohr, the Reasons for QBism". In: *Mind and Matter 15(2), 245-300 (2017)* (05/09/2017). arXiv: `1705.03483v2 [quant-ph]`.

[113] Christopher A. Fuchs, N. David Mermin, and Ruediger Schack. "An Introduction to QBism with an Application to the Locality of Quantum Mechanics". In: *Am. J. Phys., Vol. 82,* (11/2013), No.8, August2014, 749–754. eprint: `1311.5253`.

[114] Christopher A. Fuchs and Ruediger Schack. "A Quantum-Bayesian Route to Quantum-State Space". 12/2009. DOI: `10.1007/s10701-009-9404-8`. arXiv: `0912.4252 [quant-ph]`.

[115] Maria Carla Galavotti. "Subjectivism, Objectivism and Objectivity in Bruno de Finetti's Bayesianism". In: *Foundations of Bayesianism*. Ed. by David Corfield and Jon Williamson. Red. by Dov M. Gabbay and Jon Barwise. Vol. 24. Dordrecht: Springer Netherlands, 2001, pp. 161–174. ISBN: 978-90-481-5920-8 978-94-017-1586-7. DOI: `10.1007/978-94-017-1586-7_7`. URL: `http://link.springer.com/10.1007/978-94-017-1586-7_7` (visited on 12/23/2022).

[116] Mona M Garvert, Raymond J Dolan, and Timothy EJ Behrens. "A Map of Abstract Relational Knowledge in the Human Hippocampal–Entorhinal Cortex". In: *eLife* 6 (04/2017). ISSN: 2050-084X. DOI: `10.7554/elife.17086`.

[117] Wulfram Gerstner. "Spike-Response Model". In: *Scholarpedia* 3.12 (2008), p. 1343.

[118] Wulfram Gerstner and Werner M. Kistler. "Mathematical Formulations of Hebbian Learning". In: *Biological Cybernetics* 87.5-6 (12/2002), pp. 404–415. ISSN: 0340-1200. DOI: `10.1007/s00422-002-0353-y`. URL: `http://dx.doi.org/10.1007/s00422-002-0353-y`.

[119] Neil Ghani et al. "Compositional Game Theory". In: *Proceedings of the 33rd Annual ACM/IEEE Symposium on Logic in Computer Science*. LICS '18: 33rd Annual ACM/IEEE Symposium on Logic in Computer Science. Oxford United Kingdom: ACM, 07/09/2018, pp. 472–481. ISBN:

978-1-4503-5583-4. DOI: 10.1145/3209108.3209165. URL: https://dl.acm.org/doi/10.1145/3209108.3209165 (visited on 09/29/2023).

[120] Michèle Giry. "A Categorical Approach to Probability Theory". In: *Categorical Aspects of Topology and Analysis*. Ed. by B. Banaschewski. Red. by A. Dold and B. Eckmann. Vol. 915. Berlin, Heidelberg: Springer Berlin Heidelberg, 1982, pp. 68–85. ISBN: 978-3-540-11211-2 978-3-540-39041-1. DOI: 10.1007/BFb0092872. URL: http://link.springer.com/10.1007/BFb0092872 (visited on 06/20/2023).

[121] Alexander Grothendieck. "Sur Quelques Points d'algèbre Homologique". In: *Tohoku Mathematical Journal* 9.2 (01/01/1957). ISSN: 0040-8735. DOI: 10.2748/tmj/1178244839. URL: https://projecteuclid.org/journals/tohoku-mathematical-journal/volume-9/issue-2/Sur-quelques-points-dalg%c3%a8bre-homologique-I/10.2748/tmj/1178244839.full (visited on 12/23/2022).

[122] Micah Halter et al. *Compositional Scientific Computing with Catlab and SemanticModels*. 06/29/2020. arXiv: 2005.04831 [cs, math]. URL: http://arxiv.org/abs/2005.04831 (visited on 12/23/2022). preprint.

[123] Jakob Hansen. "Laplacians of Cellular Sheaves: Theory and Applications". PhD thesis. University of Pennsylvania, 2020.

[124] Jakob Hansen and Robert Ghrist. "Learning Sheaf Laplacians from Smooth Signals". In: *ICASSP 2019 - 2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 05/2019. DOI: 10.1109/icassp.2019.8683709.

[125] Jakob Hansen and Robert Ghrist. "Opinion Dynamics on Discourse Sheaves". 05/26/2020. arXiv: 2005.12798 [math.DS].

[126] R. V. L. Hartley. "Transmission of Information [1]". In: *Bell System Technical Journal* 7.3 (07/1928), pp. 535–563. ISSN: 00058580. DOI: 10.1002/j.1538-7305.1928.tb01236.x. URL: https://ieeexplore.ieee.org/document/6769394 (visited on 12/23/2022).

[127] Donald Olding Hebb. *The Organization of Behavior: A Neuropsychological Approach*. John Wiley & Sons, 1949.

[128] Jules Hedges and Riu Rodríguez Sakamoto. "Value Iteration Is Optic Composition". 06/09/2022. arXiv: 2206.04547 [math.CT].

[129] Conor Heins et al. "Spin Glass Systems as Collective Active Inference". 07/14/2022. arXiv: 2207.06970 [cond-mat.dis-nn].

[130] Chris Heunen, Mehrnoosh Sadrzadeh, and Edward Grefenstette, eds. *Quantum Physics and Linguistics: A Compositional, Diagrammatic Discourse*. Oxford University Press, 02/2013. ISBN: 978-0-19-964629-6. URL: http://ukcatalogue.oup.com/product/9780199646296.do#.UPAFH4mLLqp.

[131] Chris Heunen et al. "A Convenient Category for Higher-Order Probability Theory". In: *2017 32nd Annual ACM/IEEE Symposium on Logic in Computer Science (LICS)* (2017-01-10, 2017-06). DOI: 10.1109/lics.2017.8005137. arXiv: 1701.02547 [cs.PL].

[132] Sepp Hochreiter and Jürgen Schmidhuber. "Long Short-Term Memory". In: *Neural Computation* 9.8 (11/01/1997), pp. 1735–1780. ISSN: 0899-7667, 1530-888X. DOI: 10.1162/neco.1997.9.8.1735. URL: https://direct.mit.edu/neco/article/9/8/1735-1780/6109 (visited on 12/17/2022).

[133] A. L. Hodgkin and A. F. Huxley. "A Quantitative Description of Membrane Current and Its Application to Conduction and Excitation in Nerve". In: *The Journal of Physiology* 117.4 (08/1952), pp. 500–544. ISSN: 0022-3751. DOI: 10.1113/jphysiol.1952.sp004764. URL: http://dx.doi.org/10.1113/jphysiol.1952.sp004764.

[134] Matthew Hutson. "Artificial Intelligence Faces Reproducibility Crisis". In: *Science (New York, N.Y.)* 359.6377 (02/2018), pp. 725–726. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.359.6377.725. URL: https://www.science.org/doi/10.1126/science.359.6377.725.

[135] John P. A. Ioannidis. "Contradicted and Initially Stronger Effects in Highly Cited Clinical Research". In: *JAMA* 294.2 (07/13/2005), p. 218. ISSN: 0098-7484. DOI: 10.1001/jama.294.2.218. URL: http://jama.jamanetwork.com/article.aspx?doi=10.1001/jama.294.2.218 (visited on 12/23/2022).

[136] Eugene M Izhikevich. "Neural Excitability, Spiking and Bursting". In: *International Journal of Bifurcation and Chaos* 10.06 (06/2000), pp. 1171–1266. ISSN: 1793-6551. DOI: 10.1142/s0218127400000840. URL: http://dx.doi.org/10.1142/S0218127400000840.

[137] Bart Jacobs. "Convexity, Duality and Effects". In: *Theoretical Computer Science*. Ed. by Cristian S. Calude and Vladimiro Sassone. Vol. 323. Berlin, Heidelberg: Springer Berlin Heidelberg, 2010, pp. 1–19. ISBN: 978-3-642-15239-9 978-3-642-15240-5. DOI: 10.1007/978-3-642-15240-5_1. URL: http://link.springer.com/10.1007/978-3-642-15240-5_1 (visited on 05/12/2023).

[138] Bart Jacobs. *Introduction to Coalgebra*. Vol. 59. Cambridge University Press, 2017. ISBN: 978-1-107-17789-5. URL: https://www.cambridge.org/gb/academic/subjects/mathematics/logic-categories-and-sets/introduction-coalgebra-towards-mathematics-states-and-observation?format=HB&isbn=9781107177895.

[139] Bart Jacobs. "From Probability Monads to Commutative Effectuses". In: *Journal of Logical and Algebraic Methods in Programming* 94 (01/2018), pp. 200–237. DOI: 10.1016/j.jlamp.2016.11.006.

[140]   Bart Jacobs. "A Recipe for State-and-Effect Triangles". In: *Logical Methods in Computer Science* 13.2 (2017-05-17, 2017), pp. 1860–5974. DOI: 10.23638/LMCS-13(2:6)2017. arXiv: 1703.09034 [cs.LO].

[141]   Bart Jacobs and Bram Westerbaan. "An Effect-Theoretic Account of Lebesgue Integration". In: *Electronic Notes in Theoretical Computer Science* 319 (12/2015), pp. 239–253. DOI: 10.1016/j.entcs.2015.12.015.

[142]   Bart Jacobs and Fabio Zanasi. "The Logical Essentials of Bayesian Reasoning". In: *Foundations of Probabilistic Programming*. Ed. by Gilles Barthe, Joost-Pieter Katoen, and Alexandra Silva. 1st ed. Cambridge University Press, 12/03/2020, pp. 295–332. ISBN: 978-1-108-77075-0 978-1-108-48851-8. DOI: 10.1017/9781108770750.010. URL: https://www.cambridge.org/core/product/identifier/9781108770750%23c9/type/book_part (visited on 09/29/2023).

[143]   Amirhossein Jafarian et al. "Structure Learning in Coupled Dynamical Systems and Dynamic Causal Modelling". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 377.2160 (10/2019), p. 20190048. DOI: 10.1098/rsta.2019.0048.

[144]   Renaud Jardri and Sophie Deneve. "Computational Models of Hallucinations". In: *The Neuroscience of Hallucinations*. Springer, 2013, pp. 289–313.

[145]   Niles Johnson and Donald Yau. *2-Dimensional Categories*. 06/17/2020. arXiv: 2002.06055 [math]. URL: http://arxiv.org/abs/2002.06055 (visited on 11/03/2022). preprint.

[146]   Renaud Jolivet, Timothy J., and Wulfram Gerstner. "The Spike Response Model: A Framework to Predict Neuronal Spike Trains". In: *Lecture Notes in Computer Science* (2003), pp. 846–853. ISSN: 0302-9743. DOI: 10.1007/3-540-44989-2_101. URL: http://dx.doi.org/10.1007/3-540-44989-2_101.

[147]   R. E. Kalman. "A New Approach to Linear Filtering and Prediction Problems". In: *Journal of Basic Engineering* 82.1 (1960), p. 35. DOI: 10.1115/1.3662552. URL: http://dx.doi.org/10.1115/1.3662552.

[148]   Raphael Kaplan and Karl J. Friston. "Planning and Navigation as Active Inference". In: *Biological Cybernetics* 112.4 (03/2018), pp. 323–343. DOI: 10.1007/s00422-018-0753-2.

[149]   Sayash Kapoor and Arvind Narayanan. *Leakage and the Reproducibility Crisis in ML-based Science*. 07/14/2022. arXiv: 2207.07048 [cs, stat]. URL: http://arxiv.org/abs/2207.07048 (visited on 12/23/2022). preprint.

[150]   Robert E Kass, Luke Tierney, and Joseph B Kadane. "The Validity of Posterior Expansions Based on Laplace's Method". In: *Bayesian and Likelihood Methods in Statistics and Econometrics: Essays in Honor of George A. Barnard*. Ed. by S Geisser et al. Elsevier Science Publishers B.V. (North-Holland), 1990.

[151] Robert E. Kass and Adrian E. Raftery. "Bayes Factors". In: *Journal of the American Statistical Association* 90.430 (06/1995), pp. 773–795. ISSN: 0162-1459, 1537-274X. DOI: `10.1080/01621459.1995.10476572`. URL: `http://www.tandfonline.com/doi/abs/10.1080/01621459.1995.10476572` (visited on 05/15/2023).

[152] Richard A. Kelsey. "A Correspondence between Continuation Passing Style and Static Single Assignment Form". In: *Papers from the 1995 ACM SIGPLAN Workshop on Intermediate Representations*. POPL95: 22nd ACM Symposium on Principles of Programming Languages. San Francisco California USA: ACM, 03/1995, pp. 13–22. ISBN: 978-0-89791-754-4. DOI: `10.1145/202529.202532`. URL: `https://dl.acm.org/doi/10.1145/202529.202532` (visited on 05/02/2023).

[153] Andrei Khrennikov et al. "Quantum Models for Psychological Measurements: An Unsolved Problem". In: *PLoS ONE* 9.10 (10/2014). Ed. by Zhong-LinEditor Lu, e110909. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0110909`. URL: `http://dx.doi.org/10.1371/journal.pone.0110909`.

[154] Diederik P Kingma and Max Welling. *Auto-Encoding Variational Bayes*. 12/20/2013. arXiv: `1312.6114 [stat.ML]`. preprint.

[155] Diederik P. Kingma. "Variational Inference & Deep Learning. A New Synthesis". PhD thesis. University of Amsterdam, 2017. URL: `https://hdl.handle.net/11245.1/8e55e07f-e4be-458f-a929-2f9bc2d169e8`.

[156] Thomas N. Kipf and Max Welling. *Variational Graph Auto-Encoders*. 11/21/2016. arXiv: `1611.07308 [cs, stat]`. URL: `http://arxiv.org/abs/1611.07308` (visited on 12/23/2022). preprint.

[157] Thomas N. Kipf and Max Welling. *Semi-Supervised Classification with Graph Convolutional Networks*. 02/22/2017. arXiv: `1609.02907 [cs, stat]`. URL: `http://arxiv.org/abs/1609.02907` (visited on 12/23/2022). preprint.

[158] Michael Kirchhoff et al. "The Markov Blankets of Life: Autonomy, Active Inference and the Free Energy Principle". In: *Journal of The Royal Society Interface* 15.138 (01/2018), p. 20170792. DOI: `10.1098/rsif.2017.0792`.

[159] Michael D. Kirchhoff. "Autopoiesis, Free Energy, and the Life–Mind Continuity Thesis". In: *Synthese* 195.6 (2018), pp. 2519–2540. ISSN: 0039-7857, 1573-0964. DOI: `10.1007/s11229-016-1100-6`. URL: `http://link.springer.com/10.1007/s11229-016-1100-6` (visited on 12/23/2022).

[160] David C Knill and Alexandre Pouget. "The Bayesian Brain: The Role of Uncertainty in Neural Coding and Computation". In: *TRENDS in Neurosciences* 27.12 (2004), pp. 712–719. DOI: `10.1016/j.tins.2004.10.007`. URL: `http://www.sciencedirect.com/science/article/pii/S0166223604003352`.

[161] Mark A. Kramer. "Nonlinear Principal Component Analysis Using Autoassociative Neural Networks". In: *AIChE Journal* 37.2 (02/1991), pp. 233–243. ISSN: 0001-1541, 1547-5905. DOI:

10.1002/aic.690370209. URL: `https://onlinelibrary.wiley.com/doi/10.1002/aic.690370209` (visited on 05/02/2023).

[162]   Alexander Kurz. *Logic Column 15: Coalgebras and Their Logics*. 05/28/2006. arXiv: `cs/0605128`. URL: `http://arxiv.org/abs/cs/0605128` (visited on 12/17/2022). preprint.

[163]   Pierre Simon Laplace. "Memoir on the Probability of the Causes of Events". In: *Statistical Science* 1.3 (1986), pp. 364–378. ISSN: 08834237. JSTOR: `2245476`. URL: `http://www.jstor.org/stable/2245476` (visited on 05/15/2023).

[164]   F. W. Lawvere and S. H. Schnauel. *Conceptual Mathematics : A First Introduction to Categories*. Cambridge, UK New York: Cambridge University Press, 2009. ISBN: 978-0-511-80419-9. DOI: `10.1017/CBO9780511804199`.

[165]   F. William Lawvere. "Functorial Semantics of Algebraic Theories". In: *Proceedings of the National Academy of Sciences* 50.5 (11/1963), pp. 869–872. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.50.5.869`. URL: `https://pnas.org/doi/full/10.1073/pnas.50.5.869` (visited on 12/23/2022).

[166]   F. William Lawvere. "An Elementary Theory of the Category of Sets". In: *Proceedings of the National Academy of Sciences* 52.6 (12/1964), pp. 1506–1511. ISSN: 0027-8424, 1091-6490. DOI: `10.1073/pnas.52.6.1506`. URL: `https://pnas.org/doi/full/10.1073/pnas.52.6.1506` (visited on 12/23/2022).

[167]   John M. Lee. "Smooth Manifolds". In: *Introduction to Smooth Manifolds*. New York, NY: Springer New York, 2012, pp. 1–31. ISBN: 978-1-4419-9982-5. DOI: `10.1007/978-1-4419-9982-5_1`. URL: `https://doi.org/10.1007/978-1-4419-9982-5_1`.

[168]   Tom Leinster. *Higher Operads, Higher Categories*. London Mathematical Society Lecture Note Series 298. Cambridge University Press, Cambridge, 2004. ISBN: 0-521-53215-9. DOI: `10.1017/CBO9780511525896`.

[169]   Tom Leinster. *Entropy and Diversity: The Axiomatic Approach*. 1st ed. Cambridge University Press, 04/30/2021. ISBN: 978-1-108-96355-8 978-1-108-83270-0 978-1-108-96557-6. DOI: `10.1017/9781108963558`. URL: `https://www.cambridge.org/core/product/identifier/9781108963558/type/book` (visited on 09/29/2023).

[170]   Pantelis Leptourgos et al. "Circular Inference in Bistable Perception". In: *Journal of Vision* 20.4 (04/21/2020), p. 12. ISSN: 1534-7362. DOI: `10.1167/jov.20.4.12`. URL: `https://jov.arvojournals.org/article.aspx?articleid=2765046` (visited on 12/23/2022).

[171]   Eugene Lerman and David I. Spivak. "An Algebra of Open Continuous Time Dynamical Systems and Networks". 02/02/2016. arXiv: `1602.01017v2` `[math.DS]`.

[172] Paul Blain Levy. "Locally Graded Categories". University of Birmingham. 02/17/2019. URL: https://www.cs.bham.ac.uk/~pbl/papers/locgrade.pdf.

[173] Fosco Loregian and Emily Riehl. "Categorical Notions of Fibration". 06/15/2018. arXiv: 1806.06129v2 [math.CT].

[174] Martin Lundfall. "A Diagram Model of Linear Dependent Type Theory". 06/25/2018. arXiv: 1806.09593 [math.LO].

[175] Saunders Mac Lane. *Categories for the Working Mathematician*. 2nd ed. Graduate Texts in Mathematics 5. New York: Springer-Verlag, 1998. ISBN: 0-387-98403-8.

[176] David J. C. MacKay. *Information Theory, Inference, and Learning Algorithms*. Cambridge, UK ; New York: Cambridge University Press, 2003. 628 pp. ISBN: 978-0-521-64298-9.

[177] Saunders MacLane and Ieke Moerdijk. *Sheaves in Geometry and Logic: A First Introduction to Topos Theory*. Springer, 1992. ISBN: 0-387-97710-4.

[178] Shirley Mark et al. "Transferring Structural Knowledge across Cognitive Maps in Humans and Models". In: *Nature Communications* 11.1 (09/2020). DOI: 10.1038/s41467-020-18254-6.

[179] David Marr. *Vision: A Computational Investigation into the Human Representation and Processing of Visual Information*. Henry Holt and Co, 1982.

[180] Yoshihiro Maruyama. "Meaning and Duality: From Categorical Logic to Quantum Physics". PhD thesis. University of Oxford / University of Oxford, 2016. URL: https://ora.ox.ac.uk/objects/uuid:440a291d-7533-493d-b5aa-f6db30ca03cf.

[181] Michael V Mascagni, Arthur S Sherman, et al. "Numerical Methods for Neuronal Modeling". In: *Methods in neuronal modeling* 2 (1989). URL: http://cox.iwr.uni-heidelberg.de/teaching/numsimneuro_ss2011/mascagni_sherman.pdf.

[182] Conor McBride. "I Got Plenty o' Nuttin'". In: *A List of Successes That Can Change the World*. Springer International Publishing, 2016, pp. 207–233. DOI: 10.1007/978-3-319-30936-1_12.

[183] Yaared Al-Mehairi, Bob Coecke, and Martha Lewis. "Compositional Distributional Cognition". In: *International Symposium on Quantum Interaction*. Springer, 08/12/2016, pp. 122–134. DOI: 10.1007/978-3-319-52289-0_10. arXiv: 1608.03785 [cs.AI].

[184] Bartosz Milewski. *Profunctor Optics: The Categorical View*. 2017. URL: https://bartoszmilewski.com/2017/07/07/profunctor-optics-the-categorical-view/.

[185] Beren Millidge and Christopher L. Buckley. *Successor Representation Active Inference*. 07/20/2022. arXiv: `2207.09897 [cs]`. URL: `http://arxiv.org/abs/2207.09897` (visited on 12/23/2022). preprint.

[186] Beren Millidge, Alexander Tschantz, and Christopher L Buckley. "Whence the Expected Free Energy?" 04/17/2020. arXiv: `2004.08128 [cs.AI]`.

[187] Beren Millidge et al. *Predictive Coding: Towards a Future of Deep Learning beyond Backpropagation?* 02/18/2022. arXiv: `2202.09467 [cs]`. URL: `http://arxiv.org/abs/2202.09467` (visited on 12/23/2022). preprint.

[188] Aaron Mobley et al. "A Survey on Data Reproducibility in Cancer Research Provides Insights into Our Limited Ability to Translate Findings from the Laboratory to the Clinic". In: *PLoS ONE* 8.5 (05/15/2013). Ed. by Hirofumi Arakawa, e63221. ISSN: 1932-6203. DOI: `10.1371/journal.pone.0063221`. URL: `https://dx.plos.org/10.1371/journal.pone.0063221` (visited on 12/23/2022).

[189] Joe Moeller and Christina Vasilakopoulou. "Monoidal Grothendieck Construction". In: *Theory and Applications of Categories* 35.31 (2020), pp. 1159–1207. arXiv: `1809.00727v2 [math.CT]`.

[190] Jason Morton. "Belief Propagation in Monoidal Categories". In: *Proceedings of the 11th Workshop on Quantum Physics and Logic*. Vol. 172. EPTCS, 12/28/2014, pp. 262–269. DOI: `10.4204/EPTCS.172.18`. arXiv: `1405.2618 [math]`. URL: `http://arxiv.org/abs/1405.2618` (visited on 12/23/2022).

[191] David Jaz Myers. "Double Categories of Open Dynamical Systems (Extended Abstract)". In: *Electronic Proceedings in Theoretical Computer Science* 333 (02/08/2021), pp. 154–167. ISSN: 2075-2180. DOI: `10.4204/EPTCS.333.11`. arXiv: `2005.05956 [math]`. URL: `http://arxiv.org/abs/2005.05956` (visited on 11/16/2022).

[192] David Jaz Myers. *Categorical Systems Theory (Draft)*. 2022. URL: `http://davidjaz.com/Papers/DynamicalBook.pdf`.

[193] Lynn Nadel and Lloyd MacDonald. "Hippocampus: Cognitive Map or Working Memory?" In: *Behavioral and Neural Biology* 29.3 (1980), pp. 405–409. ISSN: 0163-1047. DOI: `10.1016/S0163-1047(80)90430-6`. URL: `http://www.sciencedirect.com/science/article/pii/S0163104780904306`.

[194] Jet Nestruev. *Smooth Manifolds and Observables*. Second edition. Graduate Texts in Mathematics 220. New York: Springer, 2020. 433 pp. ISBN: 978-3-030-45649-8. DOI: `10.1007/978-3-030-45650-4`.

[195] Frank Nielsen. "An Elementary Introduction to Information Geometry". 08/17/2018. arXiv: `1808.08271 [cs.LG]`.

[196] nLab authors. *Essential Image*. 05/2023. URL: `https://ncatlab.org/nlab/show/essential+image`.

[197] H. Nyquist. "Certain Topics in Telegraph Transmission Theory". In: *Transactions of the American Institute of Electrical Engineers* 47.2 (04/1928), pp. 617–644. ISSN: 0096-3860. DOI: 10.1109/T-AIEE.1928.5055024. URL: http://ieeexplore.ieee.org/document/5055024/ (visited on 12/23/2022).

[198] John O'Keefe. "Place Units in the Hippocampus of the Freely Moving Rat". In: *Experimental neurology* 51.1 (1976), pp. 78–109. URL: http://www.sciencedirect.com/science/article/pii/0014488676900558.

[199] Yann Ollivier. "The Extended Kalman Filter Is a Natural Gradient Descent in Trajectory Space". 01/03/2019. arXiv: 1901.00696v1 [math.OC].

[200] Frank W. J. Olver. *Asymptotics and Special Functions.* AKP Classics. Wellesley, Mass: A.K. Peters, 1997. 572 pp. ISBN: 978-1-56881-069-0.

[201] Open Science Collaboration. "Estimating the Reproducibility of Psychological Science". In: *Science* 349.6251 (08/28/2015), aac4716. ISSN: 0036-8075, 1095-9203. DOI: 10.1126/science.aac4716. URL: https://www.science.org/doi/10.1126/science.aac4716 (visited on 12/23/2022).

[202] Srdjan Ostojic and Nicolas Brunel. "From Spiking Neuron Models to Linear-Nonlinear Models". In: *PLoS computational biology* 7.1 (01/2011). Ed. by Peter E. Latham, e1001056. DOI: 10.1371/journal.pcbi.1001056. URL: http://dx.doi.org/10.1371/journal.pcbi.1001056.

[203] Ensor Rafael Palacios et al. "Biological Self-organisation and Markov Blankets". In: (11/2017). DOI: 10.1101/227181.

[204] Thomas Parr, Lancelot Da Costa, and Karl Friston. "Markov Blankets, Information Geometry and Stochastic Thermodynamics". In: *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences* 378.2164 (12/2019), p. 20190159. DOI: 10.1098/rsta.2019.0159.

[205] Thomas Parr, Giovanni Pezzulo, and Karl J. Friston. *Active Inference. The Free Energy Principle in Mind, Brain, and Behavior.* MIT Press, 2022, p. 288. ISBN: 978-0-262-04535-3.

[206] Evan Patterson, David I. Spivak, and Dmitry Vagner. "Wiring Diagrams as Normal Forms for Computing in Symmetric Monoidal Categories". In: *Electronic Proceedings in Theoretical Computer Science* 333 (02/08/2021), pp. 49–64. ISSN: 2075-2180. DOI: 10.4204/EPTCS.333.4. arXiv: 2101.12046 [cs]. URL: http://arxiv.org/abs/2101.12046 (visited on 10/27/2022).

[207] Dusko Pavlovic, Michael Mislove, and James B Worrell. "Testing Semantics: Connecting Processes and Process Logics". In: *International Conference on Algebraic Methodology and Software Technology.* Springer. Springer Berlin Heidelberg, 2006, pp. 308–322. DOI: 10.1007/11784180_24.

[208] Judea Pearl. "Reverend Bayes on Inference Engines: A Distributed Hierarchical Approach". In: *Probabilistic and Causal Inference*. Ed. by Hector Geffner, Rina Dechter, and Joseph Y. Halpern. 1st ed. New York, NY, USA: ACM, 1982, pp. 129–138. ISBN: 978-1-4503-9586-1. DOI: `10.1145/3501714.3501727`. URL: `https://dl.acm.org/doi/10.1145/3501714.3501727` (visited on 12/23/2022).

[209] Steven Phillips and William H. Wilson. "Categorial Compositionality: A Category Theory Explanation for the Systematicity of Human Cognition". In: *PLoS Computational Biology* 6.7 (07/2010). Ed. by Karl J. Friston, e1000858. DOI: `10.1371/journal.pcbi.1000858`.

[210] Benjamin C. Pierce. *Basic Category Theory for Computer Scientists*. MIT Press, 1991.

[211] V I Piterbarg and V R Fatalov. "The Laplace Method for Probability Measures in Banach Spaces". In: *Russian Mathematical Surveys* 50.6 (12/31/1995), pp. 1151–1239. ISSN: 0036-0279, 1468-4829. DOI: `10.1070/RM1995v050n06ABEH002635`. URL: `https://iopscience.iop.org/article/10.1070/RM1995v050n06ABEH002635` (visited on 05/16/2023).

[212] Tomaso Poggio and Thomas Serre. "Models of Visual Cortex". In: *Scholarpedia* 8.4 (2013), p. 3516. ISSN: 1941-6016. DOI: `10.4249/scholarpedia.3516`. URL: `http://www.scholarpedia.org/article/Models_of_visual_cortex` (visited on 12/23/2022).

[213] The Univalent Foundations Program. *Homotopy Type Theory: Univalent Foundations of Mathematics*. Institute for Advanced Study: https://homotopytypetheory.org/book, 2013. eprint: `http://saunders.phil.cmu.edu/book/hott-a4.pdf`. URL: `https://homotopytypetheory.org/book/`.

[214] Martin L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. 1st ed. Wiley Series in Probability and Statistics. Wiley, 04/15/1994. ISBN: 978-0-471-61977-2 978-0-470-31688-7. DOI: `10.1002/9780470316887`. URL: `https://onlinelibrary.wiley.com/doi/book/10.1002/9780470316887` (visited on 12/23/2022).

[215] Maxwell J D Ramstead et al. "On Bayesian Mechanics: A Physics of and by Beliefs". 05/23/2022. arXiv: `2205.11543 [cond-mat.stat-mech]`.

[216] R. P. Rao and D. H. Ballard. "Predictive Coding in the Visual Cortex: A Functional Interpretation of Some Extra-Classical Receptive-Field Effects". In: *Nature Neuroscience* 2.1 (01/1999), pp. 79–87. DOI: `10.1038/4580`. pmid: `10195184`.

[217] Mitchell Riley. "Categories of Optics". 09/03/2018. arXiv: `1809.00738v2 [math.CT]`.

[218] Edmund T. Rolls and Alessandro Treves. *Neural Networks and Brain Function*. 1st ed. Oxford University Press, USA, 01/15/1998. ISBN: 0-19-852432-3.

[219] Mario Román. "Open Diagrams via Coend Calculus". 04/09/2020. arXiv: `2004.04526v2 [math.CT]`.

[220] Robert Rosen. "The Representation of Biological Systems from the Standpoint of the Theory of Categories". In: *The Bulletin of Mathematical Biophysics* 20.4 (12/1958), pp. 317–341. ISSN: 0007-4985, 1522-9602. DOI: 10.1007/BF02477890. URL: http://link.springer.com/10.1007/BF02477890 (visited on 12/23/2022).

[221] Robert Rosenbaum. "On the Relationship between Predictive Coding and Backpropagation". In: *PLOS ONE* 17.3 (03/31/2022). Ed. by Gennady S. Cymbalyuk, e0266102. ISSN: 1932-6203. DOI: 10.1371/journal.pone.0266102. URL: https://dx.plos.org/10.1371/journal.pone.0266102 (visited on 12/23/2022).

[222] Dylan Rupel and David I. Spivak. *The Operad of Temporal Wiring Diagrams: Formalizing a Graphical Language for Discrete-Time Processes*. 07/25/2013. arXiv: 1307.6894 [cs, math, q-bio]. URL: http://arxiv.org/abs/1307.6894 (visited on 09/19/2023). preprint.

[223] Sara Sabour, Nicholas Frosst, and Geoffrey E Hinton. "Dynamic Routing Between Capsules". In: *Advances in Neural Information Processing Systems*. Ed. by I. Guyon et al. Vol. 30. Curran Associates, Inc., 2017. URL: https://proceedings.neurips.cc/paper/2017/file/2cad8fa47bbef282badbb8de5374b894-Paper.pdf.

[224] Dalton A R Sakthivadivel. "Towards a Geometry and Analysis for Bayesian Mechanics". 04/25/2022. arXiv: 2204.11900 [math-ph].

[225] Dalton A. R. Sakthivadivel. *A Constraint Geometry for Inference and Integration*. 04/18/2022. arXiv: 2203.08119 [cond-mat, physics:math-ph]. URL: http://arxiv.org/abs/2203.08119 (visited on 12/23/2022). preprint.

[226] Ausra Saudargiene, Bernd Porr, and Florentin Wörgötter. "How the Shape of Pre- and Postsynaptic Signals Can Influence STDP: A Biophysical Model." In: *Neural computation* 16.3 (03/2004), pp. 595–625. DOI: 10.1162/089976604772744929. pmid: 15006093. URL: http://dx.doi.org/10.1162/089976604772744929.

[227] Evan S. Schaffer, Srdjan Ostojic, and L. F. Abbott. "A Complex-Valued Firing-Rate Model That Approximates the Dynamics of Spiking Networks". In: *PLoS computational biology* 9.10 (10/2013). Ed. by Bard Ermentrout, e1003301. DOI: 10.1371/journal.pcbi.1003301. URL: http://dx.doi.org/10.1371/journal.pcbi.1003301.

[228] Urs Schreiber. "Differential Cohomology in a Cohesive Infinity-Topos". 10/29/2013. arXiv: 1310.7930 [math-ph].

[229] Grégoire Sergeant-Perthuis. *Regionalized Optimization*. 05/19/2022. arXiv: 2201.11876 [cs, math]. URL: http://arxiv.org/abs/2201.11876 (visited on 12/23/2022). preprint.

[230] C. E. Shannon. "A Mathematical Theory of Communication". In: *Bell System Technical Journal* 27.3 (07/1948), pp. 379–423. ISSN: 00058580. DOI: 10.1002/j.1538-7305.1948.tb01338.x. URL: https://ieeexplore.ieee.org/document/6773024 (visited on 12/23/2022).

[231] Brandon Shapiro and David I. Spivak. "Dynamic Categories, Dynamic Operads: From Deep Learning to Prediction Markets". 05/08/2022. arXiv: `2205.03906 [math.CT]`.

[232] Dan Shiebler. "Categorical Stochastic Processes and Likelihood". In: *Compositionality 3, 1 (2021)* (05/10/2020). DOI: `10.32408/compositionality-3-1`. arXiv: `2005.04735 [cs.AI]`.

[233] Harel Shouval. "Models of Synaptic Plasticity". In: *Scholarpedia* 2.7 (2007), p. 1605. ISSN: 1941-6016. DOI: `10.4249/scholarpedia.1605`. URL: `http://dx.doi.org/10.4249/scholarpedia.1605`.

[234] Michael Shulman. "Homotopy Type Theory: The Logic of Space". In: *New Spaces for Mathematics and Physics*. Ed. by M. Anel and G. Catren. Cambridge University Press, 2017-03-08, 2017. arXiv: `1703.03007 [math.CT]`. URL: `http://arxiv.org/abs/1703.03007`.

[235] David I Spivak and Nelson Niu. *Polynomial Functors: A General Theory of Interaction*. (In press), 2021. URL: `https://raw.githubusercontent.com/ToposInstitute/poly/main/Book-Poly.pdf`.

[236] David I. Spivak. "Functorial Data Migration". In: *Information and Computation* 217 (2012), pp. 31–51. ISSN: 0890-5401. DOI: `10.1016/j.ic.2012.05.001`. URL: `https://doi.org/10.1016/j.ic.2012.05.001`.

[237] David I. Spivak. *The Operad of Wiring Diagrams: Formalizing a Graphical Language for Databases, Recursion, and Plug-and-Play Circuits*. 05/01/2013. arXiv: `1305.0297 [cs, math]`. URL: `http://arxiv.org/abs/1305.0297` (visited on 12/23/2022). preprint.

[238] David I. Spivak. *Generalized Lens Categories via Functors C Op → Cat*. 08/06/2019. arXiv: `1908.02202 [math.CT]`. preprint.

[239] David I. Spivak. *Poly: An Abundant Categorical Setting for Mode-Dependent Dynamics*. 05/05/2020. arXiv: `2005.01894 [math.CT]`. preprint.

[240] David I. Spivak. "Learners' Languages". In: *Electronic Proceedings in Theoretical Computer Science* 372 (03/01/2021), pp. 14–28. DOI: `10.4204/EPTCS.372.2`. arXiv: `2103.01189 [math.CT]`.

[241] David I. Spivak. *A Reference for Categorical Structures on Poly*. 02/01/2022. arXiv: `2202.00534 [math.CT]`. preprint.

[242] David I. Spivak. *Functorial Aggregation*. 01/31/2022. arXiv: `2111.10968 [cs, math]`. URL: `http://arxiv.org/abs/2111.10968` (visited on 12/23/2022). preprint.

[243] David I. Spivak, Patrick Schultz, and Dylan Rupel. "String Diagrams for Traced and Compact Categories Are Oriented 1-Cobordisms". In: *J. Pure Appl. Algebra 221 (2017), no. 8, pp. 2064-*

2110 (08/05/2015). DOI: 10.1016/j.jpaa.2016.10.009. arXiv: 1508.01069 [math.CT].

[244] David I. Spivak and Joshua Tan. "Nesting of Dynamic Systems and Mode-Dependent Networks". 02/25/2015. arXiv: 1502.07380 [math.DS].

[245] Olaf Sporns. "Brain Connectivity". In: *Scholarpedia* 2.10 (2007), p. 4695. ISSN: 1941-6016. DOI: 10.4249/scholarpedia.4695. URL: http://dx.doi.org/10.4249/scholarpedia.4695.

[246] Mandyam Veerambudi Srinivasan, Simon Barry Laughlin, and A Dubs. "Predictive Coding: A Fresh View of Inhibition in the Retina". In: *Proceedings of the Royal Society of London. Series B. Biological Sciences* 216.1205 (11/22/1982), pp. 427–459. ISSN: 0080-4649, 2053-9193. DOI: 10.1098/rspb.1982.0085. URL: https://royalsocietypublishing.org/doi/10.1098/rspb.1982.0085 (visited on 12/23/2022).

[247] Toby St Clere Smithe. "Radically Compositional Cognitive Concepts". 11/14/2019. arXiv: 1911.06602 [q-bio.NC].

[248] Toby St Clere Smithe. *Bayesian Updates Compose Optically*. 05/31/2020. arXiv: 2006.01631 [math.CT]. preprint.

[249] Toby St Clere Smithe. "Compositional Active Inference". In: Finding the Right Abstractions. Topos Institute, 05/12/2021.

[250] Toby St Clere Smithe. *Compositional Active Inference I: Bayesian Lenses. Statistical Games*. 09/09/2021. arXiv: 2109.04461 [math.ST]. preprint.

[251] Toby St Clere Smithe. "Cyber Kittens, or Some First Steps Towards Categorical Cybernetics". In: *Electronic Proceedings in Theoretical Computer Science*. Applied Category Theory 2020. Vol. 333. 02/08/2021, pp. 108–124. DOI: 10.4204/EPTCS.333.8. URL: http://arxiv.org/abs/2101.10483v1 (visited on 09/29/2023).

[252] Toby St Clere Smithe. "Polynomial Life: The Structure of Adaptive Systems". In: *Fourth International Conference on Applied Category Theory (ACT 2021)*. Ed. by K. Kishida. Vol. EPTCS 370. 2021, pp. 133–147. DOI: 10.4204/EPTCS.370.28.

[253] Toby St Clere Smithe. "Compositional Active Inference II: Polynomial Dynamics. Approximate Inference Doctrines". 08/25/2022. arXiv: 2208.12173 [nlin.AO].

[254] Toby St Clere Smithe. "Open Dynamical Systems as Coalgebras for Polynomial Functors, with Application to Predictive Processing". 06/08/2022. arXiv: 2206.03868 [math.CT].

[255] Toby St Clere Smithe and Simon M Stringer. "The Role of Idiothetic Signals, Landmarks, and Conjunctive Representations in the Development of Place and Head-Direction Cells: A Self-Organizing Neural Network Model". In: *Cerebral Cortex Communications* 3.1 (01/01/2022), tgab052. ISSN: 2632-7376. DOI: 10.1093/texcom/tgab052. URL: https://

`academic.oup.com/cercorcomms/article/doi/10.1093/texcom/tgab052/6358621` (visited on 12/23/2022).

[256] Kimberly L Stachenfeld, Matthew Botvinick, and Samuel J Gershman. "Design Principles of the Hippocampal Cognitive Map". In: *Advances in Neural Information Processing Systems 27*. Ed. by Z. Ghahramani et al. Curran Associates, Inc., 2014, pp. 2528–2536. URL: `http://papers.nips.cc/paper/5340-design-principles-of-the-hippocampal-cognitive-map.pdf`.

[257] Kimberly Lauren Stachenfeld, Matthew M Botvinick, and Samuel J Gershman. "The Hippocampus as a Predictive Map". In: (12/2016). DOI: `10.1101/097170`. URL: `http://dx.doi.org/10.1101/097170`.

[258] Sam Staton. "Commutative Semantics for Probabilistic Programming". In: *Programming Languages and Systems*. Springer Berlin Heidelberg, 2017, pp. 855–879. DOI: `10.1007/978-3-662-54434-1_32`.

[259] Dario Maxmilian Stein. "Structural Foundations for Probabilistic Programming Languages". University of Oxford, 2021. URL: `https://dario-stein.de/notes/thesis.pdf`.

[260] Christopher Summerfield, Fabrice Luyckx, and Hannah Sheahan. "Structure Learning and the Posterior Parietal Cortex". In: *Progress in Neurobiology* (10/2019), p. 101717. DOI: `10.1016/j.pneurobio.2019.101717`.

[261] Richard S Sutton and Andrew G Barto. *Reinforcement Learning: An Introduction*. 2nd ed. Vol. 1. 1. MIT press Cambridge, 2018.

[262] D Gowanlock R Tervo, Joshua B Tenenbaum, and Samuel J Gershman. "Toward the Neural Implementation of Structure Learning". In: *Current Opinion in Neurobiology* 37 (04/2016), pp. 99–105. ISSN: 0959-4388. DOI: `10.1016/j.conb.2016.01.014`. URL: `http://dx.doi.org/10.1016/j.conb.2016.01.014`.

[263] Luke Tierney and Joseph B. Kadane. "Accurate Approximations for Posterior Moments and Marginal Densities". In: *Journal of the American Statistical Association* 81.393 (03/1986), pp. 82–86. ISSN: 0162-1459, 1537-274X. DOI: `10.1080/01621459.1986.10478240`. URL: `http://www.tandfonline.com/doi/abs/10.1080/01621459.1986.10478240` (visited on 05/15/2023).

[264] Alexander Tschantz, Anil K. Seth, and Christopher L. Buckley. "Learning Action-Oriented Models through Active Inference". In: (09/2019). DOI: `10.1101/764969`.

[265] Kai Ueltzhöffer. "Deep Active Inference". In: *Biological Cybernetics* 112.6 (2017-09-07, 2018-10), pp. 547–573. DOI: `10.1007/s00422-018-0785-7`. arXiv: `1709.02341 [q-bio.NC]`.

[266] Dániel Unyi et al. *Utility of Equivariant Message Passing in Cortical Mesh Segmentation.* 06/15/2022. arXiv: 2206.03164 [cs]. URL: http://arxiv.org/abs/2206.03164 (visited on 12/23/2022). preprint.

[267] Aad W. van der Vaart. *Asymptotic Statistics.* 1. paperback ed., 8. printing. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge: Cambridge Univ. Press, 2007. 443 pp. ISBN: 978-0-521-78450-4 978-0-521-49603-2.

[268] Dmitry Vagner, David I. Spivak, and Eugene Lerman. "Algebras of Open Dynamical Systems on the Operad of Wiring Diagrams". In: *Theory and Applications of Categories* 30 (2015), Paper No. 51, 1793–1822. ISSN: 1201-561X.

[269] Matthijs Vákár. *Syntax and Semantics of Linear Dependent Types.* 01/16/2015. arXiv: 1405.0033 [cs, math]. URL: http://arxiv.org/abs/1405.0033 (visited on 12/17/2022). preprint.

[270] Matthijs Vákár and Luke Ong. *On S-Finite Measures and Kernels.* 10/03/2018. arXiv: 1810.01837 [math.PR]. preprint.

[271] Frank van der Meulen. *Introduction to Automatic Backward Filtering Forward Guiding.* 10/31/2022. arXiv: 2203.04155 [math, stat]. URL: http://arxiv.org/abs/2203.04155 (visited on 12/23/2022). preprint.

[272] Frank van der Meulen and Moritz Schauer. *Automatic Backward Filtering Forward Guiding for Markov Processes and Graphical Models.* 10/31/2022. arXiv: 2010.03509 [stat]. URL: http://arxiv.org/abs/2010.03509 (visited on 12/23/2022). preprint.

[273] Carl van Vreeswijk and Haim Sompolinsky. "Chaos in Neuronal Networks with Balanced Excitatory and Inhibitory Activity". In: *Science (New York, N.Y.)* 274.5293 (1996), pp. 1724–1726. DOI: 10.1126/science.274.5293.1724. pmid: 8939866. URL: http://www.ncbi.nlm.nih.gov/pubmed/8939866.

[274] F. G. Varela, H. R. Maturana, and R. Uribe. "Autopoiesis: The Organization of Living Systems, Its Characterization and a Model". In: George J. Klir. *Facets of Systems Science.* Boston, MA: Springer US, 1991, pp. 559–569. ISBN: 978-1-4899-0720-2 978-1-4899-0718-9. DOI: 10.1007/978-1-4899-0718-9_40. URL: http://link.springer.com/10.1007/978-1-4899-0718-9_40 (visited on 12/23/2022).

[275] Yde Venema. "Automata and Fixed Point Logic: A Coalgebraic Perspective". In: *Information and Computation* 204.4 (04/2006), pp. 637–678. DOI: 10.1016/j.ic.2005.06.003.

[276] Pietro Vertechi. "Dependent Optics". In: *Electronic Proceedings in Theoretical Computer Science.* Applied Category Theory 2022. Vol. 380. 08/07/2023, pp. 128–144. DOI: 10.4204/EPTCS.380.8. arXiv: 2204.09547 [math.CT]. URL: http://arxiv.org/abs/2204.09547v4 (visited on 09/29/2023).

[277]   Andre Videla and Matteo Capucci. *Lenses for Composable Servers*. 03/29/2022. arXiv: `2203. 15633 [cs]`. URL: `http : / / arxiv . org / abs / 2203 . 15633` (visited on 12/17/2022). preprint.

[278]   James C. R. Whittington and Rafal Bogacz. "An Approximation of the Error Backpropagation Algorithm in a Predictive Coding Network with Local Hebbian Synaptic Plasticity". In: *Neural Computation* 29.5 (05/01/2017), pp. 1229–1262. ISSN: 0899-7667, 1530-888X. DOI: `10 . 1162 / NECO _ a _ 00949`. URL: `https : / / direct . mit . edu / neco / article / 29 / 5 / 1229 / 8261 / An - Approximation - of - the - Error - Backpropagation` (visited on 12/23/2022).

[279]   James C. R. Whittington et al. "The Tolman-Eichenbaum Machine: Unifying Space and Relational Memory through Generalisation in the Hippocampal Formation". In: (09/2019). DOI: `10 . 1101 / 770495`. eprint: `https : / / www . biorxiv . org / content / biorxiv / early / 2019 / 09 / 24 / 770495 . full - text . pdf`.

[280]   C.K.I. Williams and D. Barber. "Bayesian Classification with Gaussian Processes". In: *IEEE Transactions on Pattern Analysis and Machine Intelligence* 20.12 (12/1998), pp. 1342–1351. ISSN: 01628828. DOI: `10 . 1109 / 34 . 735807`. URL: `http : / / ieeexplore . ieee . org / document / 735807 /` (visited on 05/15/2023).

[281]   Daniel Williams. "Is the Brain an Organ for Free Energy Minimisation?" In: *Philosophical Studies* 179.5 (05/2022), pp. 1693–1714. ISSN: 0031-8116, 1573-0883. DOI: `10 . 1007 / s11098 - 021 - 01722 - 0`. URL: `https : / / link . springer . com / 10 . 1007 / s11098 - 021 - 01722 - 0` (visited on 12/23/2022).

[282]   Donald Yau. *Operads of Wiring Diagrams*. Vol. 2192. Lecture Notes in Mathematics. Cham: Springer International Publishing, 2018. ISBN: 978-3-319-95000-6 978-3-319-95001-3. DOI: `10 . 1007 / 978 - 3 - 319 - 95001 - 3`. arXiv: `1512 . 01602 [math]`. URL: `http : / / link . springer . com / 10 . 1007 / 978 - 3 - 319 - 95001 - 3` (visited on 09/11/2023).

[283]   J.S. Yedidia, W.T. Freeman, and Y. Weiss. "Constructing Free-Energy Approximations and Generalized Belief Propagation Algorithms". In: *IEEE Transactions on Information Theory* 51.7 (07/2005), pp. 2282–2312. ISSN: 0018-9448. DOI: `10 . 1109 / TIT . 2005 . 850085`. URL: `http : / / ieeexplore . ieee . org / document / 1459044 /` (visited on 12/23/2022).

[284]   Ernst Zermelo. "Über Eine Anwendung Der Mengenlehre Auf Die Theorie Des Schachspiels". In: *Proceedings of the Fifth International Congress of Mathematicians*. Vol. 2. Cambridge University Press, 1913, pp. 501–504.

[285]   Jie Zhou et al. "Graph Neural Networks: A Review of Methods and Applications". 12/20/2018. arXiv: `1812 . 08434v4 [cs.LG]`.