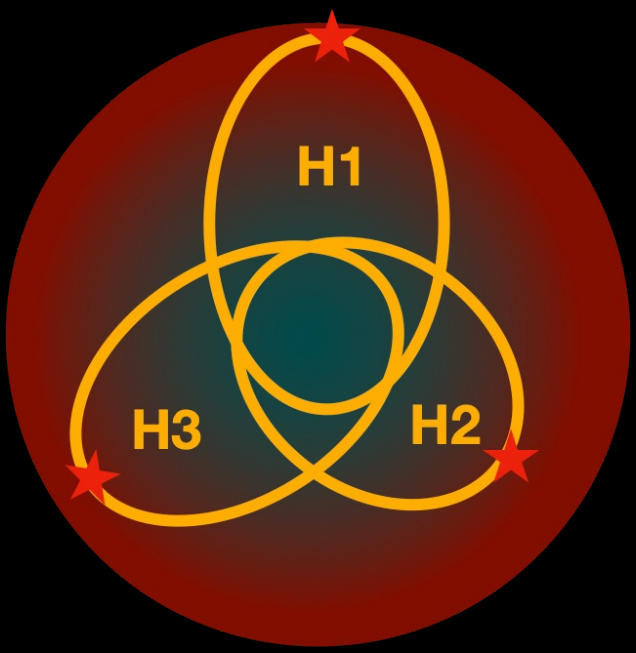
2022

Towards a normative understanding of higher-order brain activity

Severin Berger



Dissertation presented to obtain the **Ph.D degree in Neuroscience**

International Neuroscience Doctoral Programme

Oeiras, November, 2022

INSTITUTO DE TECNOLOGIA QUÍMICA E BIOLÓGICA ANTÓNIO XAVIER | UNIVERSIDADE NOVA DE LISBOA

Towards a normative understanding of higher-order brain activity

Severin Berger

Dissertation presented to obtain the Ph.D degree in Neuroscience

Instituto de Tecnologia Química e Biológica António Xavier | Universidade NOVA de Lisboa

Research work coordinated by: Champalimaud Foundation

Oeiras, November, 2022

Towards a normative understanding of higher-order brain activity

Severin Berger

A Dissertation
Presented to the Faculty
of Universidade Nova de Lisboa
in Candidacy for the Degree
of Doctor of Philosophy

Supervised by: Christian Machens
International Neuroscience Doctoral Programme
Champalimaud Research
Lisbon, Portugal

2022

Acknowledgments

First, I would like to thank Christian, my supervisor, for all the guidance, patience, and support. I admire his clarity of thought, his relaxed attitude, his natural curiosity about science, and his general friendliness. Especially when work was going tough, meetings with him always gave me new energy and hope, which was immensely helpful and I am very grateful for this.

Next, I want to thank the Theoretical neuroscience lab members, all of you made my work experience so much better. Especially, I want to thank Bill and Allan for being good friends and for proof-reading chapters of this thesis. Also Oihane, Adrien, Bertrand, Nuno, and Michael who I all appreciate very much and have always been there to discuss science as well as personal matters.

To my INDP2015 class, thank you Basma, Baylor, Mauricio, Dennis, Patricia, and Rodrigo. You have made Lisbon an immediate home to me and it has been a lot of fun to have you alongside throughout the whole of my Ph.D. work.

I also would like to extend my thanks to my thesis committee, namely Gonzalo de Polavieja and Joe Paton. I highly appreciated your inputs that were original, diverse, and very helpful.

Lastly, I want to thank my family, and as I love them all, I go by age. My father Urs, who has always been supporting and encouraging me to follow my interests. I admire his practical attitude towards life, his thoughtfulness, and his talent for almost everything. My mother Christine, who I missed very much while being away from home. My sister Catherine, who has been a role model all my life. My brother Bastian, who is the best brother possible. Finally, I want to thank my partner Nun, you always make me happy and I am truly blessed to have you. I am looking forward to spending a lot more time with you, because I know that we will never get bored. I also want to thank all my friends, most of them in Switzerland. When I left, you were my family that I took for granted, which, unfortunately, I found out only later. Especially Marc and Felipe, you both deserve a big thank you!

List of Abbreviations

ALM Anterior Lateral Motor Cortex
LDS Linear Dynamical System
M1 Primary Motor Cortex
MDP Markov Decision Process
OMDP Observation Markov Decision Process
PFC Prefrontal Cortex
POMDP Partially Observable Markov Decision Process
RL Reinforcement Learning
RNN Recurrent Neural Network
roSLDS recurrent-only Switching Linear Dynamical System
SLDS Switching Linear Dynamical System
V1 Primary Visual Cortex
WM Working Memory

Título

Em direção a um entendimento normativo da actividade cerebral de ordem superior

Resumo

A capacidade de adaptar o comportamento é uma das principais marcas da inteligência animal e as áreas do cérebro de ordem superior, tais como o córtex pré-frontal, são consideradas como tendo um papel crucial na mesma: estão particularmente bem ligados ao resto do cérebro, estão desproporcionadamente bem desenvolvidos em primatas superiores, e demonstraram estar envolvidos nas funções executivas que se pensa permitirem esta flexibilidade, tais como a tomada de decisões, memória de trabalho, planeamento, e atenção. Com o advento de novas técnicas de medição, estamos a ganhar uma visão cada vez mais completa da actividade cerebral de ordem superior, enquanto os animais executam tarefas comportamentais. Utilizando estes dados, podemos agora tentar acrescentar à compreensão conceptual das funções cerebrais de ordem superior que já temos, elucidando a sua base neural.

Uma abordagem para o fazer é desenvolver modelos computacionais que reproduzam a actividade neural registada. Uma abordagem popular deste tipo é baseada em redes neurais recorrentes que são simplesmente treinadas para executar a mesma tarefa que o animal. Embora reproduzam a actividade notavelmente bem, fazem apenas um uso limitado da compreensão conceptual das funções cerebrais de ordem superior que já temos. No entanto, esta abordagem mostra-nos que a actividade cerebral de ordem superior parece ser largamente determinada pela tarefa, uma vez que os modelos de redes neurais recorrentes, por exemplo, apenas fazem referência limitada à natureza biológica subjacente da área cerebral modelada. Consequentemente, a actividade neural de ordem superior pode ser compreendida em grande medida pelo estudo da representação teórica das tarefas.

Nesta tese, começamos por rever e analisar os registos neurais de várias áreas cerebrais de ordem superior em vários organismos modelo, para obter uma imagem clara dos fenómenos neurais que pretendemos reproduzir. Estudamos então as representações de tarefas e salientamos que diferentes representações servem diferentes propósitos. Isto permite-nos ligar os nossos modelos ao conhecimento conceptual que já temos. Por exemplo, uma representação serve o propósito de planeamento. Em seguida, procuramos as representações de tarefas correspondentes que servem estes

diferentes propósitos, que depois podemos comparar com as representações neurais correspondentes.

Esta abordagem é, contudo, tal como a abordagem das redes neurais recorrentes, mal definida porque cada propósito pode ser cumprido por muitas representações diferentes. Resta-nos assim a escolha de qual a representação que se pode comparar com os dados. Por conseguinte, adoptamos uma abordagem normativa. Especificamente, seguimos a hipótese de codificação eficiente que afirma que os sistemas neurais devem eliminar toda a informação redundante e irrelevante. Esta teoria já foi bem sucedida na explicação de fenómenos neurais antes, particularmente em áreas visuais, e generalizamo-la a áreas cerebrais de ordem superior com a pergunta: Qual é a representação de tarefas mais eficiente para um determinado propósito?

Finalmente, aplicamos a nossa abordagem normativa e representativa a quatro tarefas em dois organismos-modelo: macacos e ratos. Consideramos dois propósitos hipotéticos: Um é o propósito de servir um comportamento baseado em modelos, ou seja, planeamento, e o outro é servir um comportamento habitual, ou seja, tomada de decisões sem planeamento, e formalizamos ambos os propósitos no quadro de uma aprendizagem de reforço parcialmente observável (*partially observable reinforcement learning*). Em macacos, como esperado, verificamos que o córtex pré-frontal parece utilizar uma estratégia baseada no planeamento, enquanto que nos ratos, a actividade numa área motora de ordem superior é bem explicada por uma estratégia habitual mais directa. Nesta tese, apresentamos assim uma abordagem normativa, orientada por hipóteses, que fornece potencial conhecimento sobre a função e o objectivo das representações cerebrais de ordem superior, e abre a possibilidade de fazer o mesmo para outras áreas cerebrais.

Abstract

The ability to flexibly navigate complex environments is a major hallmark of animal intelligence and higher-order brain areas such as the prefrontal cortex are thought to play a crucial role in it: They are particularly well connected to the rest of the brain, they are disproportionately well developed in higher primates, and they have been shown to be involved in the executive functions that are thought to enable this flexibility, such as decision making, working memory, planning, and attention. With the advent of novel recording techniques, we are gaining an increasingly complete view of higher-order brain activity while animals perform behavioural tasks. Using these data, we can now try to add to the conceptual understanding of higher-order brain areas, such as their involvement in working memory and planning, by elucidating the neural basis of their functions.

One approach to do so is to develop computational models that try to reproduce recorded neural activity. A popular such approach is based on recurrent neural networks which are simply trained to perform the same task as the animal. While they can reproduce activity remarkably well, they only make limited use of the conceptual understanding of higher-order brain function that we already have. For example, they often ignore that tasks can be solved in many different ways in the first place. Nevertheless, this approach shows us that higher-order brain activity seems to be largely determined by the task at hand, as the recurrent neural network models for example only make limited reference to the underlying biological nature of the modelled brain area. Accordingly, higher-order neural activity might be understood to quite an extent by simply studying the ways in which tasks can be represented in principle.

In this thesis we first review and analyse neural recordings from various higher-order brain areas in various model organisms in order to get a clear picture of the neural phenomena that we aim to reproduce. We then study task representations and point out that different representations serve different purposes. For example, one task representation might serve the specification of the correct responses in a particular task, while another might serve the specification of the rules of a given task. This allows us to connect our models to the conceptual knowledge of higher-order brain function, such as their involvement in planning, and accordingly, one representation will serve the purpose of planning. We then search for the corresponding task representations that serve these different purposes, which we can then compare to the corresponding neural representations.

This approach is, however, as is the recurrent neural network approach, ill-defined because every purpose can be fulfilled by many different representations. The decision of which representation to compare to the data is thus left up to the user. We therefore take a normative approach. Specifically, we follow the efficient coding hypothesis that states that neural systems should eliminate all redundant and irrelevant information. This theory has been successful in explaining neural phenomena before, particularly in sensory areas, and we generalize it to higher-order brain areas by asking: What is the most efficient task representation for a given purpose?

We finally apply our normative, representational approach to three tasks in two model-organisms, namely monkeys and mice. We consider two hypothetical purposes: One is the purpose of serving a model-based behaviour, i.e. planning, and the other is to serve a habitual behaviour, i.e. decision making without planning, and we formalize both purposes within the framework of partially observable reinforcement learning. We find that prefrontal cortex in monkeys, as expected, seems to use a planning-based strategy, while activity in a higher-order motor area in mice is well explained by a more direct habitual strategy. In this thesis, we thus present a hypothesis-driven, normative approach that provides potential insights into the function and the purpose of higher-order brain representations, and it opens up the possibility to do the same for brain areas beyond.

Author Contributions

This thesis was written by Severin Berger. Detailed contributions are provided at the beginning of each chapter in the "Contributions" section. In general, Severin Berger and Christian Machens developed all concepts and methods. Christian Machens proposed the initial research plan, and Severin Berger implemented all simulations and analyses under the supervision of Christian Machens.

Artwork by Severin Berger.

Financial Support

This work was carried out under the International Neuroscience Doctoral Programme (INDP). Funding was provided by the Portuguese FCT (Fundação para a Ciência e a Tecnologia, FCT Bolsa PD/BD/114279/2016) and the Fundação Champalimaud.

Contents

Chapter 1

Introduction

Our human experience shows us that we are capable of doing incredible things. From performing simple everyday tasks such as navigating a busy street or chatting with a neighbour, to building collaborative ventures like the global economy. We can set our minds to acquiring complex skills, such as playing the piano or mastering the art of cooking, or even spending several years thinking about a narrow topic in order to write a doctoral thesis. We can do all of this thanks to our ability to perceive and remember our surroundings, make plans and goals, and make decisions that we can implement through actions.

Of course, many of these abilities are not unique to humans, but common to many animals, as we all participate in this cycle from perception to action. Arguably, it is this cycle that impelled animals to develop brains, and the brain is in turn crucial to its control (Kandel et al., 2000). Indeed, the areas of the brain may be roughly grouped according to their involvement in this cycle: Sensory areas are mainly concerned with the inputs (perception), motor areas mainly with the outputs (action), and the rest with the processing that links the two.

In recent decades, neuroscientists have made progress in determining more precisely how each of these different areas is involved in generating behaviours such as those described above. Lesion studies, in which a specific area is removed from an animal's brain and then the resulting behavioural deficits are examined, have been a fruitful approach (Damasio and Damasio, 1989; Wiegert et al., 2017; Szczepanski and Knight, 2014). This is even more true when an animal is first trained to perform a task specifically designed to test a particular skill. Conversely, which brain regions are involved in the execution of a task depends on the specifics of the particular task:

What sensory modalities, e.g., vision, hearing, or smell, are stimulated, and what movements, e.g., walking, grasping, or licking, are required? For example, does the animal need to remember certain aspects of the task in order to evaluate different options?

One striking discovery was that the higher-order areas of the frontal cortical lobe — to which we will refer simply as **higher-order brain areas** in this thesis — are crucial when a task requires more complex cognitive abilities, such as planning, working memory, or decision making (Goldman-Rakic, 1987; Funahashi et al., 1993; Passingham, 1985; Szczepanski and Knight, 2014; Guo et al., 2014). The most prominent such area is likely the prefrontal cortex (PFC), which has been studied extensively in monkeys (Fuster, 2015). The PFC is heavily connected to the rest of the brain, making it well positioned to be a major part of the connection between perception and action. The PFC is also particularly well developed in humans, making it likely to be a crucial structure for enabling complex human abilities. Thus, higher-order brain areas appear to be at the heart of the complex cognitive abilities that allow animals to flexibly navigate environments and problems (Miller and Cohen, 2001). Naturally, it has been of great interest to gain a better understanding of how these brain areas work, not only from a neuroscientific perspective, but even from the perspective of artificial intelligence (AI), as today's AI systems, despite enormous recent progress, are often vastly inferior in this regard (Russin et al., 2020; Lake et al., 2017).

Over the years, neuroscientists have made great strides in advancing this understanding — we will review some of this work in the next section 1.1 — in particular by using the approach of developing behavioural tasks, observing the animal's behaviour and neural activity, and possibly even causally manipulating neural activity by for example using optogenetics (Lima and Miesenböck, 2005; Boyden et al., 2005). Many different tasks have been developed, often specifically tailored to test the core cognitive functions, sometimes also called executive functions, of working memory, planning, attention, and decision making (Fuster, 2015). Different sensory modalities have been used, be it visual, auditory, or somatosensory, and also different model organisms, mainly monkeys, but also e.g. rats and mice. The resulting behaviour and neural recordings have been analysed in various ways resulting in concise descriptions of what is happening on a neural level.

However, these concise descriptions are often not straightforward to interpret, especially if one compares them across tasks. We will review them more closely in chapter 2. For example, the neural population activities usually depend on multiple

task variables, such as stimulus, reward, and decision, at the same time, often in an intermingled fashion (Kobak et al., 2016; Rigotti et al., 2013). Their dynamics are sometimes slowly varying (Constantinidis et al., 2018; Funahashi et al., 1989; Romo et al., 1999; Inagaki et al., 2019) and at other times they move fast and in complicated ways (Fujisawa et al., 2008; Inagaki et al., 2019; Harvey et al., 2012; Sreenivasan and D'Esposito, 2019). It is thus challenging to develop models that reproduce these activity features, even though they are the result of simple neuroscience tasks which are vastly less complex than real life environments. At the same time, the above hypotheses about higher-order brain function, such as the involvement in the executive functions, suggest that we should be able to explain some of their observed activities. In the remainder of the introduction, we first give a brief summary of previous work attempting to explain and model higher-order brain activity and function. We then go on to introduce our main contributions.

1.1 The current understanding of higher-order brain areas

1.1.1 Methodology

Tasks, behaviour and neural activity

As introduced above, higher-order brain areas are crucial for flexible task solving. This insight has been demonstrated in recent decades by, among other things, simultaneously recording animals' behaviour and brain activity while they perform behavioural tasks.

There are many different types of tasks, but essentially they are all quite simple. Some involve navigation, e.g. a rat must find a food source in a maze (Tolman, 1948); others require memory, e.g. a monkey must remember the location of a stimulus and then respond to it after a delay (Funahashi et al., 1989); or they test an animal's perceptual abilities, e.g. by asking an animal to report the direction of a noisy stimulus (Shadlen and Newsome, 2001). Although each of these tasks requires slightly different skills, all are fundamentally about decision making and reporting those decisions through behaviour.

The resulting behaviour is thus recorded in various degrees of detail (Machado et al., 2015; Pardo-Vazquez et al., 2019; Wiltschko et al., 2015; Mathis et al., 2018). Crucially, choice behaviour is usually recorded, i.e. how often the animal takes the correct decision, how often the animal takes the opposite decision, and how often it

does something completely unrelated. If trials in a task vary in difficulty, one can also draw a psychometric curve that measures the animal's decision performance as a function of difficulty. Furthermore, all these measurements can be taken over the course of learning, or even before and after learning a second task or a change in task conditions. Analysing such measurements can give insight into how an animal is making decisions, without even looking into the brain (see e.g. Tolman (1948); Krakauer et al. (2017)).

However, if we are interested in how the brain produces these behaviours, we need to look inside the brain. There are several ways to do this, each with different advantages. Essentially, however, the different recording methods trade-off three aspects: How much they damage the brain and affect the animal's abilities, i.e., their invasiveness; how well or directly they record the activity of a particular neuron, i.e., their recording resolution; and how many neurons of the brain they record. On the one hand, there are methods such as functional magnetic resonance imaging that are noninvasive and record the activity of the entire brain simultaneously, but their measurement of neuronal activity is very indirect and the resolution is low. On the other hand, there are methods such as patch-clamping, which record the activity of a neuron directly and with high temporal resolution, but are highly invasive. In between are methods such as extracellular electrophysiological recordings and imaging. These methods can record many neurons simultaneously with reasonable resolution and invasiveness. Since the focus of this thesis is to understand higher-order brain activities, we discuss their analysis in more detail in the next section.

Neural data analysis

In order to establish a direct link between a higher-order brain area's activity and its function, it is crucial to capture the main characteristics of a data set of recorded neural activity. Broadly speaking there are two ways of analysing the activity of recorded neurons: The first is to analyse single neurons separately, one at a time, and the second is to analyse the whole population of recorded neurons together. Many of the classical discoveries about the function of neural areas is based on single neuron analysis. For example, Fuster and Alexander (1971) were the first to show that monkey PFC was active at an elevated rate during the delay period of a delayed response task, hence suggesting a neural substrate of working memory, and this discovery was made by investigating the firing rates of single neurons.

More recently, though, the field has increasingly moved to population analyses where a population of neurons is analysed jointly. This is mainly due to two problems:

First, making sense of the single neuron responses of a large collection of neurons is not straightforward as individual neurons might have quite distinct responses. In this case it will be hard to succinctly summarize and pinpoint the main insights gained from the recordings. Second, neurons do not work in isolation and the correlation between neurons might carry important information. Even if no single neuron is significantly tuned to a certain stimulus, the population as a whole might be.

The main workhorse of analysing population data are linear dimensionality reduction techniques (Cunningham and Yu, 2014; Humphries, 2020). The most commonly used such method is principal component analysis (PCA). Briefly, PCA tries to find activity patterns, or components, that are shared between neurons. The most widely shared pattern is the first principal component (PC), the second most widely shared pattern is the second principal component, and so forth. If there are N neurons in the population, there will be N PCs, yet crucially, often the first few, say $M \ll N$, PCs will describe the bulk of the data. If this is the case, a single neuron's activity can be fully reconstructed — except for some noise — using M PCs, only. PCA thus addresses the first problem outlined above. Similarly, PCA addresses the second problem, as it can pick up a pattern that is weak in single neurons, but if it is shared among many neurons, strong on the population level. Due to these benefits, PCA-type methods have enabled great insights into how populations work (Cunningham and Yu, 2014).

If the population data was recorded during a task, it often makes sense to not only find shared activity patterns overall, but shared activity patterns that are specifically due to a certain task variable (Brendel et al., 2011; Mante et al., 2013; Kobak et al., 2016; Aoi et al., 2020). For example, a task stimulus might excite the population as a whole which will result in a shared activity pattern due to that task stimulus. Similarly, a decision that is formed and then executed might also leave a trace in the population activity. Two methods that pick out such task variable dependencies are targeted dimensionality reduction (Mante et al., 2013; Aoi et al., 2020), and demixed principal component analysis, or dPCA (Brendel et al., 2011; Kobak et al., 2016). These 'targeted' or 'demixed' components often paint a clearer picture of what exactly is happening on a population level during a task.

1.1.2 Theories and models

Theories of behaviour

From a purely behavioural perspective, it has long been debated about how animals solve tasks. One school of thought, known as behaviourism (Skinner, 1938), held that an animal's behaviour can be described as a sequence of stimulus-response associations, known as habits. The other school argued that a decision or action is the result of reasoning that animals do based on a mental model of the world, which is often called goal-directed or model-based behaviour (Tolman, 1948). Nowadays, it is generally accepted that animals are capable of both these behavioural strategies (Dolan and Dayan, 2013; Wassum et al., 2009). From a computational perspective, the two strategies each offer different advantages: While model-based behaviour is flexible and computationally intensive, habitual behaviour is rigid but cheap. As such, the two strategies are not mutually exclusive, and animals are likely to be able to arbitrate between them depending on what better suits their current needs (Dolan and Dayan, 2013; Lengyel and Dayan, 2008; Wassum et al., 2009). Yet whether, at a given time and a given task, an agent is using one strategy or the other is often challenging to disambiguate, even when using tasks specifically designed for that purpose (Akam et al., 2015). One way to address this problem might be to go beyond pure behavioural analysis and also look at neural activity (Akam et al., 2015).

From a neural perspective, mainly two brain areas have been associated with habitual and model-based behaviour (Dolan and Dayan, 2013; Balleine and O'Doherty, 2009). First, the PFC, which is primarily thought to carry a model (Behrens et al., 2018), but in rats another part of it has also been shown to mediate habitual behaviour (Killcross and Coutureau, 2003; Coutureau and Killcross, 2003), and second, the striatum, which is again involved in model-based and habitual control via separate subregions. These roles are usually demonstrated by inactivation and lesion experiments, by simple decoding studies (Behrens et al., 2018), or by indirect evidence of how each strategy affects dopamine responses (Motiwala et al., 2022; Daw et al., 2011), but often a direct link between function and neural representations is missing. As the focus of this thesis is on higher-order brain areas, such as the PFC, we will not address striatal representations any further.

Concepts and theories of higher-order brain function

In this and the next sections, we review models of higher-order brain areas that we broadly categorize according to their focus. Here, we first review concepts and

theories of higher-order brain function, i.e. models that focus on capturing the main principles underlying higher-order brain function, and PFC function in specific. They are less focused on a specific mathematical formalism and they are less aimed at reproducing particular neural representations. In contrast, the models reviewed in the next section are more mathematically explicit, and they are principally concerned with reproducing neural activities.

In the classical textbook of prefrontal cortex (Fuster, 2015), the core function of PFC is summarized as the representation and execution of goal-directed behaviour. The executive functions that Fuster (2015) attributes to PFC — namely planning, decision-making, working memory, and attention — all serve this core function. In goal-directed behaviour, in contrast with habitual behaviour, the goal comes first, and decisions are made and actions are taken to achieve that goal. In order to know which action will achieve the goal, planning is needed, which ultimately includes some internal model about how actions affect the external environment. Lastly, working memory and attentive mechanisms are paramount for implementing the core function. Working memory will keep relevant information online, including the goal, whereas attentive mechanisms, together with working memory, influence other brain regions in a top-down manner in order to achieve the goal.

This last process, which is sometimes also called cognitive control, is the main focus of an influential theory by Miller and Cohen (2001). Their theory, though in essence similar to the one by Fuster (2015), focuses slightly more on mechanism and neural implementation. They argue that the function of PFC is to represent goals and rules as patterns of activity, which in turn control the processing in other parts of the brain by top down signaling. They reason that the brain works in a default setting — think simple habitual stimulus-response relationships — and top-down control adjusts this default setting in order to achieve the goal.

The same set of ideas has been cast in the framework of active inference (Pezzulo, 2012; Friston et al., 2016). There, goals correspond to priors that are encoded in higher-order brain areas, and actions are chosen such that these goals are fulfilled by minimizing variational free energy. Signatures of such goal codes have been found in recordings of PFC neurons of monkeys that were solving a set of tasks with various goals (Stoianov et al., 2016). The authors further showed that these goal codes correspond to a compact categorisation or clustering of task parameters, preserving their relation to goals and rewards. They note that this an efficient way of coding

tasks that also enables transfer learning over a range of task variations. This idea of efficiency will also be a main pillar of our own work, as we will see in the next section.

Clustering and categorisation of sensory inputs itself has also been proposed as a principal function of PFC (Seger and Miller, 2010). Animals arguably act within an environment of infinite complexity, i.e. our world. No two situations that an animal might encounter are identical, and in order to learn anything about the world, and in order to take action that improves an animal's own case, these specific situations have to be grouped. PFC particularly groups them according to their relation to goals and the actions to achieve them, and not their physical properties (Freedman et al., 2001; Stoianov et al., 2016).

Apart from these rather general theories, various models have been proposed that address more specific aspects of PFC function. For example, recently, Wang et al. (2018) put forward a model that casts PFC as a meta-learning system. There, the focus is more on how PFC's function contributes to learning and how it is embedded within the reinforcement learning theory of the brain that is based on dopamine signals from midbrain neurons that signal reward prediction errors (Schultz et al., 1997). Using these dopamine error signals, PFC is picking up regularities in the structure of a series of tasks as well as how they are learnt, thus learning to implement a learning algorithm itself. They show that this assumption reproduces several experimentally observed characteristics associated with PFC. One such characteristic is how behaviour is adapted in a volatile environment (Behrens et al., 2007). If volatility is high, adaptation is fast, yet if volatility is low, so is adaptation. Activities in human anterior cingulate cortex, a subarea of PFC, seem to track this volatility which potentially relies on Bayesian inference (Behrens et al., 2007). Donoso et al. (2014) also propose that such inferences are an important part of PFC function. They propose that PFC constantly infers whether one of a set of alternative behaviours is most suited to achieve a goal, or whether new behaviors have to be learned.

Finally, another line of research focuses specifically on the function of working memory (Baddeley, 2003). For example, classically it was thought that working memory consists of a fixed set of about seven slots (Miller, 1956). More recently, the idea was brought forward that working memory is a resource allocable based on task demands (Ma et al., 2014). According to this account, if a task requires more to be kept in memory, the quality of that memory will decrease. Naturally, as the working memory resource is quite limited, it appears to be of central importance that working

memory represents task information in an appropriate form. This possibly links back to the theories about clustering and categorisation reviewed above.

In summary, while there are various ideas and concepts about the function of PFC, most of them can be understood in the light of goal-directed behaviour. For example, as environments are complex and stochastic, categorisation, probabilistic inference, and working memory are likely crucial to achieve a goal. Goal-directed behaviour relies on an internal model of the environment, that potentially operates on a suitable categorisation of environmental states, that are potentially constructed over time using working memory. We will pick up these ideas to define the models in this thesis.

Models of higher-order brain activity

The general theories introduced above are not aimed at explaining the exact dynamics of population activity during a given task. This is usually left to models with a more mechanistic view. A first set of such models is concerned less with overall function of higher-order brain area, but rather how specific observed activity features are implemented in the first place. For example, how can a dynamical system remain active without any external stimulation in order to reproduce the persistent activity observed in working memory tasks? For this, different types of artificial neural networks or dynamical systems have been proposed. Some rely on slowly passing activity from neurons to neurons in a feedforward manner (Goldman, 2009; Ganguli et al., 2008), and others rely on recurrent activity that is fed back yielding so called attractors (Ben-Yishai et al., 1995; Seung, 1996; Compte et al., 2000).

A second generation of models then used such mechanisms in order to design dynamical systems or neural networks that are capable of solving specific behavioural tasks (Wang, 2002; Machens et al., 2005). Building on this, a third generation then replaced the hand-designing by learning algorithms that directly adjust the connection weights between neurons in order to solve a certain task (Barak et al., 2013; Mante et al., 2013; Sussillo et al., 2015; Song et al., 2016, 2017; Yang et al., 2019; Orhan and Ma, 2019; Sohn et al., 2019; Cueva et al., 2020; Flesch et al., 2022). This latter approach of training artificial recurrent neural networks (RNNs) has turned out to be remarkably apt at reproducing neural activities. Given that RNNs are also fully amenable for analysis, a lot of research has focused on analysing them in attempt to reverse engineer the mechanisms that the actual recorded neural population might use. This makes RNNs the currently main method of modelling and analysing higher-order brain activity. For example, Sohn et al. (2019) showed that RNNs reproduce

signatures of Bayesian inference observed in PFC, which allowed them to propose a mechanism of how this inference is implemented in PFC. Yet while in principle fully amenable to analysis, in practice it is hard to readily explain their behaviour and mechanism. This has spawned interesting research in how to analyse RNNs in terms of dynamical motifs (Sussillo and Barak, 2013; Maheswaranathan et al., 2019; Vyas et al., 2020).

1.2 Our contribution

We position our work at the intersection between the two types of models introduced above: Our principal aim is to reproduce recorded population dynamics during tasks, yet our models are fully determined by the assumed conceptual model or function of the modelled brain area. More specifically, we follow a normative representational framework and ask how should a task be represented given what we assume about the function of the brain area. We now introduce the main three pillars of our work.

1.2.1 Pillar 1: What makes a good representation?

We start by assuming that an observed neural representation is in some form a representation of the task at hand. It is then natural to ask what a good representation of a task would be, or, more generally, what makes a good representation. We start with a simple fact: Whether the representation of something is good depends on what the representation will be used for. Consider for example the representation of numbers. If one is concerned with adding numbers together, then an addition algorithm might work better using the decimal representation of numbers than using the Roman numeral system. On the other hand, if one is concerned with counting up to three, the Roman system might be more straightforward. Therefore, first and foremost, it will be the purpose that determines what a good representation is and what is not. Or, in other words, before we can answer what a good representation is, we have to be clear about its purpose.

Once we have completely established what the purpose is, we can in principle look for the corresponding representation that optimally achieves it. This is usually done by setting up an objective function that measures how well a given representation fulfils the purpose, and then changing said representation in a way that improves the objective function. In neuroscience one can use this approach to build normative models of neural phenomena. If the optimal representation for a given purpose has

some characteristic features that can also be found in the brain, then we have evidence that the brain might actually be concerned with achieving that purpose. One of the more successful stories of theoretical neuroscience is built on this approach: the explanation of receptive field properties of neurons in the primary visual cortex (V1, Olshausen and Field (1996); Bell and Sejnowski (1997); Simoncelli and Olshausen (2001)). If one assumes that the purpose of early visual areas is to faithfully represent natural images as sparsely as possible, the edge-detector-like receptive fields observed in visual cortex of monkeys and cats are to be expected. Similarly, this approach was used to explain the activity of neurons in primary motor cortex (M1) of monkeys (Sussillo et al., 2015). If one assumes that the purpose of late motor areas is to drive muscle activity in a simple way, then the rotational dynamics in M1 are to be expected.

Can we transfer this approach to normatively explain high-order brain activity? What could be the purpose there that defines the representation of a given task? Before introducing an answer to this question in the next section, we first note the complicating fact that for higher-order brain areas we do not have a natural external frame of reference. This is different to the two cases introduced above: For V1, it is sensible to relate its properties to the visual world that animals see, as it is to relate M1 activity to an animal's movements. The space of visual inputs and the space of movement outputs are natural frames of reference for the two areas. Higher-order brain areas are neither purely sensory nor purely motor, but rather they are thought to be at the centre of converting sensory inputs into sensible motor outputs. Therefore, the information, and thus the activities, that we can find there are likely influenced by both sensory and motor information. In this thesis we will propose a theoretical framework to formalize such a frame of reference and define tasks within it.

1.2.2 Pillar 2: The purpose of higher-order brain areas

We assume that a higher-order brain area's purpose is in some sense related to the solution of tasks. Yet tasks can be solved in many different ways, as we saw above. In order to introduce some of these ways we here quickly introduce the three main ingredients making up a task: the type of sensory stimuli that are presented to the animal, the type of responses or actions from the animal, and the task rules that determine how those stimuli and actions lead to rewards.

The first, and arguably the simplest way, is the **habitual** behaviour. Here, an agent has to only remember to take the correct action given any situation. This habitual type of purpose is the one that is usually, albeit implicitly, assumed in RNN models, e.g. (Barak et al., 2013; Mante et al., 2013). In order to achieve this habitual

purpose, an agent, and thus the corresponding representation, needs to remember everything that is predictive of these correct actions, but beyond that, no further information is needed. This makes the habitual behaviour inflexible in that it will be hard to adapt to slight task changes, especially if these changes require information that was not needed to produce the correct actions previously. On the other hand, a representation serving the habitual purpose will potentially be cheap to store.

To enable more flexibility, an agent needs richer representations that capture more information about the environment. Apriori, it is unknown how tasks might change, and accordingly, it is unknown what additional information should be remembered about the environment. One possibility is to not only predict the correct actions but to predict how actions in general lead to rewards. Rewards, by definition, are of interest to an agent and so might be the ability to predict them. This second, reward-predictive, way to solve a task thus requires a representation that keeps all information that is predictive of rewards. As such a representation is rich enough to capture how actions lead to rewards, it will also be rich enough to predict the actions that will maximize these rewards, i.e. the correct actions. Therefore, the representation required by the reward-predictive purpose must be at least as rich as the one required by the habitual purpose.

A third way is to also predict the sensory inputs that are predictive of rewards, so called reward cues. Such a representation will allow to plan for correct behaviour, even without ever having experienced a task. Lastly, an agent can try to predict everything within the task environment, i.e. to build a model of the environment. This **model-based** way will require the richest representation and it will allow for quick relearning of a wide range of changes to the task.

Beforehand, we will not know which of these purposes will apply, but they will all require representations of different complexities that address different fundamental requirements of how tasks may be solved. For example, all these purposes will pose different requirements on what an agent has to remember. At the same time, they will leave open what an agent can forget. As a result, there will be infinitely many representations that can fulfil each of these purposes. For example, the rich model-based representation will be sufficient to fulfil the habitual purpose, also. Thus, even given a task and an assumed purpose, we still do not know which of the resulting representations we should choose in order to compare it to neural data. Besides the contribution of clearly defining different purposes, we will next introduce the rationale

for our proposal of how to find the one purpose-sufficient representation that should be compared to data.

1.2.3 Pillar 3: Efficiency, compression, and making well defined representations

Suppose we want to test whether a particular neural representation recorded during a directional categorization task serves a habitual purpose. In this task, a mark is presented at one of eight locations on a screen, and an agent must decide whether that mark is to the right or left of the screen's center. To specify the correct action, as required by the habitual purpose, a binary representation that specifies whether the mark stimulus is on the left or right is sufficient. However, a second representation specifying the exact position of the mark would also work. Or even a representation that includes an arbitrary variable in the environment, such as the temperature in the room. So which of these different representations should we compare to the neural recordings?

Apriori, all of these representations are valid. However, the latter two representations contain information that is not needed for the habitual purpose, so it is arbitrary which of these two representations we choose. Given the habitual purpose, we can only be sure that the binary information of left versus right has to be represented, anything else is indeterminate. Accordingly, our approach here is to choose the representation that includes all purpose-relevant information, but otherwise excludes irrelevant information. In other words, we are looking for the simplest or least complex representation sufficient for a purpose.

More specifically, we measure the complexity of a representation using the information theoretic concept of entropy (Shannon, 1948). Briefly, entropy is a feature of a random variable and it measures how surprised we are when observing its outcomes. If the outcome is always the same, entropy is minimal, yet if outcomes tend to be different, then entropy is high. In the case of a task representation, the representation of a particular trial is the outcome, and if all trials are represented the same, then the entropy is minimal. In the first example representation of our above task, trials corresponding to marks on the right-hand side will have identical representations, and so will the trials with marks on the left. The resulting overall representation will thus be of low entropy as it is localized around two points of the representation space only. The faithful representation of the marks' locations, on the other hand, would result in a representation that is localized around eight points, and thus corresponds to a higher entropy representation.

In practice, we will show that finding this simplest, or lowest entropy, representation corresponds to compressing the initial task space, the external frame of reference introduced above, as much as possible, while preserving all task-relevant information. This corresponds to the **information bottleneck** principle (Tishby et al., 2000). Additionally, we assume that the task-relevant information must be available as a linear readout from the representation. Linear readouts have proven useful in both analysing (Machens et al., 2010; Mante et al., 2013; Cunningham and Yu, 2014) neural activity as well as in models of neural activity, as for example in the recurrent neural networks discussed above.

We call the result an **efficient task representation** for a given purpose and they provide normative predictions of how task representations should look like given an assumed purpose. These efficient task representations generalize the theory of efficient coding, which states that neural systems shall eliminate all irrelevant or redundant information (Attneave, 1954; Barlow, 1961; Simoncelli and Olshausen, 2001), to higher-order brain areas. We can then finally compare these efficient representations to the neural recordings, and we will do this comparison by applying dPCA on both the neural recordings as well as the normative predictions. If the neural representation shows any feature that goes beyond, for example, the efficient habitual representation, then our habitual assumption is too narrow and we have to reassess and modify it. If an efficient representation matches the recordings well, on the other hand, we have evidence that the brain area might be serving the assumed purpose in an efficient manner.

Besides the justification from a statistical perspective, considering efficient, or compressed, representations has several other advantages. First, the resulting compressed representation will be fully determined by the task as well as the purpose. For example, if the resulting representation has a transient fluctuation in its dynamics that is also seen in the recorded neural representation, then this fluctuation has an explanation in either the requirements of the purpose, or the specifics of the task. This will make it more straightforward to interpret the resulting model representation. Second, from a computational perspective, compressed representations are cheaper to store (MacKay et al., 2003), and they may generalize better over task variations that only concern task irrelevant information.

Our contributions are thus the following: From a conceptual viewpoint, we introduce a novel normative perspective to analyse higher-order brain activity. We define a frame of reference for higher-order brain areas and formalize tasks and purposes within

it. We lastly propose a method of how to find the corresponding efficient representations. We will use this approach to propose novel interpretations and explanations of the neural representations in three example tasks (Romo et al., 1999; Mante et al., 2013; Inagaki et al., 2019). As a result, we provide a novel perspective and explanation of working memory contents and dynamics, and the purposes of higher-order brain areas.

1.3 General outline of the chapters

We start out, in chapter two, by exploring and highlighting commonalities and differences between neural representations recorded by various labs in higher-order brain areas during various tasks. We then present the main results of our work in the third chapter, by introducing the methodology and the results from its application to three data sets. In chapter four we add an earlier, published, version of the work (Berger and Machens, 2020). Author contributions to each chapter are presented at the beginning of each chapter, followed by a brief summary of the chapter's contents. Finally, we discuss the work of the thesis, the main conclusions, and open problems.

Chapter 2

Task representations in higher-order brain areas

2.1 Contributions

All the analyses were done by Severin Berger under the supervision of Christian Machens.

2.2 Summary

In order to explain higher-order brain activity, we must first agree upon what exactly needs explaining. We thus analyse three data sets with recordings from higher-order brain areas. The first two were recorded in monkey PFC during a somatosensory working memory task and a context-dependent perceptual decision making task, respectively (Romo et al., 1999; Mante et al., 2013). The third set of recordings originates from mouse ALM during a delayed licking task (Inagaki et al., 2019). Using a standardized analysis across data sets, we point out common and distinctive features from the three data sets. First, all data sets analysed show relatively low-dimensional activity, as pointed out before. Second, all neural activities show a clear representation of the variables relevant during the task. At the same time, we find that not only task relevant variables are presented. Similarly, and thirdly, we find interesting dynamics that do not obviously follow from the task structure. Lastly, we find that the neural representations seem to be directly influenced by the precise structure of the task.

2.3 Introduction

We here set out to highlight both common and distinctive features of higher-order brain representations in order to clarify what exactly we are trying to explain with the models that we will develop in the following chapters. In general, a model that captures the inner workings of a given brain area should reproduce that area's activity across various tasks. Thus, it is important to also analyse and describe the neural recordings during various tasks in a standardized way.

In order to do so, we here re-analyse published data in a standardized way. We choose three data sets: The first consists of neural recordings from monkey PFC during a somatosensory working memory task (Romo et al., 1999). This data is a classical example of persistent delay activity in monkey PFC and has been the subject to a lot of modelling and analysis work (for example see Machens et al. (2005); Barak et al. (2010, 2013); Song et al. (2016); Machens et al. (2010); Kobak et al. (2016)). The second data set is again from monkey PFC, recorded during a context-dependent decision making task (Mante et al., 2013). Having two data sets from the PFC of monkeys, even though from different sub-areas, lets us investigate and scrutinize how the neural representations change when the task also changes. Lastly, we analyse data from mouse ALM during a delayed licking task (Inagaki et al., 2019). This task was selected in order to compare data across species and across brain areas. The ALM is hypothesized to correspond to the premotor cortex in monkeys, a brain area immediately downstream to the PFC. Furthermore, the data set by Inagaki et al. (2019) is special in that it includes recordings from two variations of the same delayed licking task that only varies in the specifics of the delay period timing. By investigating how this minor change in task structure affects the corresponding neural representation, we can constrain our models more strongly.

The choice of our analysis method is based on two criteria. First, and most importantly, the method should generate a summary of the neural recordings that captures most of the variance in the data. As we apriori can not know what in the neural recordings is important, our analysis should aim to leave as little as possible ignored. Second, the analysis results should be interpretable both in the way they relate to the actual neural recordings and also in how they relate to the task at hand.

The most commonly used method to analyse neural recordings, i.e. PCA, fulfils the first criterion as it is designed to succinctly capture as much variance in the data as possible. It also fulfils the criterion of relating to the neural recordings in an interpretable way, as PCA does this linearly. Yet, it fails to clearly carve out how the neural

representation depends on the task at hand, as it is an unsupervised method. There are a few methods that relax this unsupervised assumption and specialize in carving out those task dependencies, such as 'targeted dimensionality reduction' (Mante et al., 2013; Aoi et al., 2020), and demixed PCA (dPCA, Kobak et al. (2016)). Here we focus on the latter. In dPCA, one can specify the conditions under which the data was recorded, which it will then use to find linear projections that capture as much variance as possible, as in PCA, but also specifically separate the different conditions.

By applying dPCA on the above three tasks, we show that, first, all data sets are relatively low-dimensional. This has been noted before and it seems to be a main feature of higher-order brain representations (see e.g. Humphries (2020) for a recent discussion). Second, all neural representations show a clear dependency on the variables relevant during the task. At the same time, we find that variables are represented that go beyond the apparently relevant variables. Third and last, we highlight interesting dynamical features in all recordings. These dynamics are not readily explained from the structure of the task at hand, yet the dynamics do seem to be immediately affected by that structure.

2.4 Results

2.4.1 Monkey-PFC representation in a somatosensory working memory task

The somatosensory working memory task study by Romo et al. (1999) is a classical demonstration of persistent working memory activity in the lateral PFC of monkeys. In this task, a monkey is asked to discriminate the frequency of two vibrotactile stimuli, F1 and F2, that are presented one after the other and separated by a fixed delay period. Depending on which frequency is higher, the monkey has to report its choice by pushing one of two buttons (Fig.2.1a).

Two monkeys were trained on this task and spikes were recorded from their PFC, resulting in a total of 1354 units. The persistent activity during this task has first been shown using single neuron analysis (Romo et al., 1999; Brody et al., 2003), and we here reviews some of these results (Fig.2.1b). Persistently active neurons, such as the example neuron 1 (Fig.2.1b,top), show clear selectivity to the frequency of the presented first vibrotactile stimulus F1, starting from its presentation and lasting until the end of the trial. While the activity moves faster during stimulus presentation, it is more or less stationary during the delay period. Other neurons, such as example neuron 2 (Fig.2.1b,bottom), are also selective to F1, yet their dynamics are distinctly

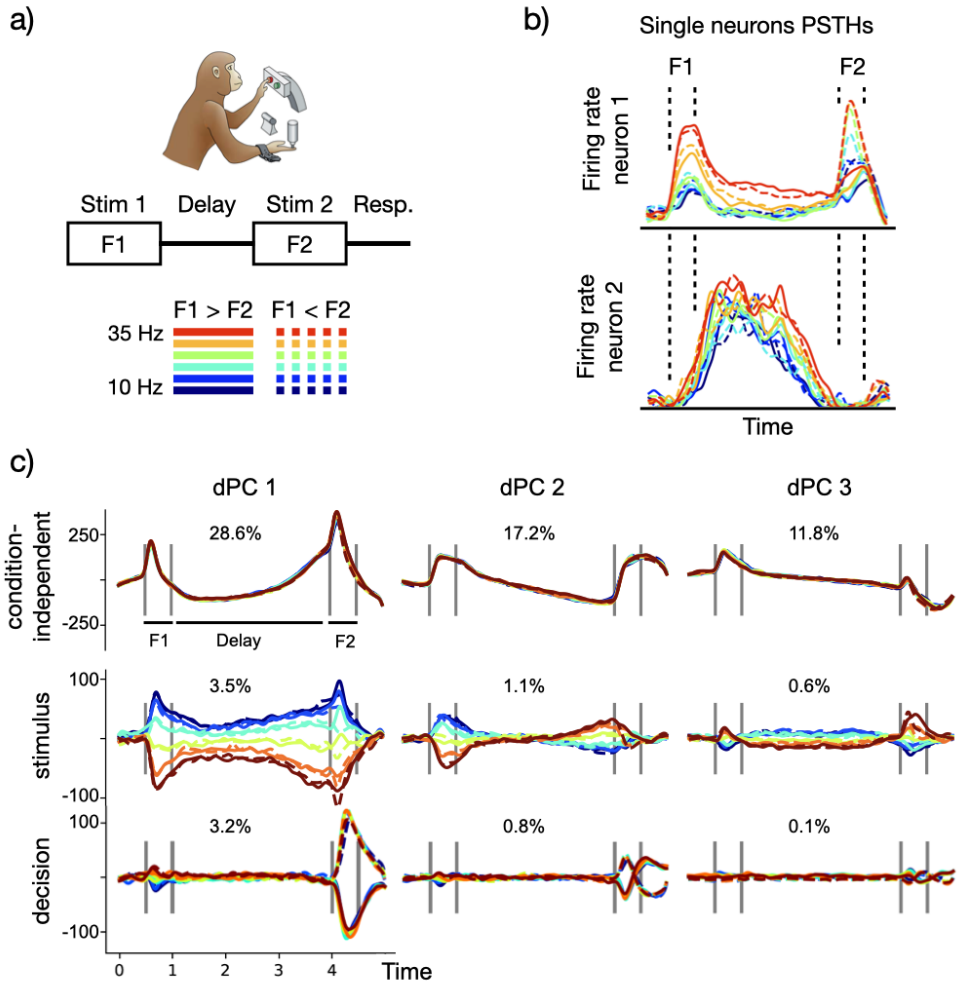


Figure 2.1: PFC representation of somatosensory working memory task in monkeys (Romo et al., 1999). a) In the task, monkeys receive a vibratory stimulus (F1) on their fingertips for half a second, followed by a three second delay period after which a second stimulus (F2) is presented, again for half a second. The monkey has to discern whether F1 or F2 was of greater frequency and respond accordingly by pushing a button. The color code of the trial types used in the rest of this figure is depicted at the bottom. F1 is sampled from six possible frequencies, indicated by the colored lines. The two decisions are indicated by solid and dashed lines, respectively. b) Peri-stimulus time histograms of two example PFC neurons. Stimulus presentation periods are indicated by vertical bars. c) Demixed PCA of the population of the 785 selected units recorded in PFC. The first row corresponds to the first three condition independent components, the middle row to the first three stimulus components, and the last row to the first three decision components. How much variance each component explains is indicated as a percentage. Only correct trials were analysed, and again stimulus presentation periods are indicated by vertical bars. Monkey figure in a) adapted from (Romo and Salinas, 2003)

different. Example neuron 2 is mostly inactive during stimulus presentation, but then becomes transiently active during the delay period. In fact, the population as a whole shows a multitude of firing patterns (Romo et al., 1999; Brody et al., 2003; Machens et al., 2010). Furthermore, neurons are also selective to the animal's decision (Fig.2.1b,top).

To get a concise summary of both the dynamics as well as the dependence of the representation on stimulus and decision, we ran dPCA on the population (Fig.2.1c), following (Kobak et al., 2016) and reproducing the results of Kobak et al. (2016). As a result we get three sets of components: First, we have the condition-independent components which capture the variance in the data that is common to all conditions (Fig.2.1c,top row). Thus, these components are due to neural activity that is not selective to the stimuli presented nor the decisions made by the animal. The first three of them capture more than half of the variance in the data. Second, we have the stimulus components which capture the variance in the data that is due to the different F1 values presented (Fig.2.1c, middle row). These components clearly show that the population as a whole encodes the frequency stimulus. In particular, it does so to a similar extent both during the stimulus presentation period as well as during the delay period, as already indicated by the single neuron analysis. At the same time, the stimulus components explain much less variance than the condition-independent ones. Lastly, we have the decision components which capture the change in neural activity due to the animal choosing one option versus the other (Fig.2.1c, bottom row). Of course, these components only show decision related activity once F2 is presented, as before no decision could have been made. These component explain a similar amount of variance as the stimulus components.

All together, the demixed principal components paint a picture of the neural representation that seems comprehensive, yet also raises some questions. As PFC is considered crucial for working memory and the solution of tasks in general, the encoding of the stimulus information throughout the delay period makes sense, and also the decision component. However, the stimulus components also show that the stimulus information is moving around throughout the trial. These dynamics go beyond what one might expect from a simple stimulus short term memory.

Similarly, the condition-independent components also show non-trivial dynamics. Furthermore, they capture the majority of the variance in the data. As they are not encoding the stimulus nor the decision, the question poses itself: what are they doing? One possibility is that they encode something that is completely unrelated to

the task. This seems unlikely, though, as the condition-independent activity is clearly locked to the timing of the task. Another possibility is that this activity corresponds to some other function, such as for example the animal's attention, that is locked to the task, but uncontrolled by the experimenter. A third possibility is that this condition-independent activity simply keeps track of the passage of time in a trial, so that the animal knows how far in the trial they are. From the recordings alone it is hard to discern between these different possibilities.

2.4.2 Monkey-PFC representation in a context-dependent perceptual decision making task

The study of Mante et al. (2013) was influential in a number of ways. It addressed the question of how flexible context-dependent behaviour is enabled by PFC. They trained monkeys on a task that required them to integrate noisy visual information towards one of two choices (Fig.2.2a). Crucially, which of the two choices is the correct one depended on the context that was provided at the beginning of each trial. In the first context, the motion context, the direction of motion in the visual stimulus was determining which choice was correct. In the second context, the color context, it was the color of the visual stimulus that determined which choice was correct. The animal thus had to flexibly switch between two contexts.

Mante et al. (2013) then recorded neural activity in the PFC, and the frontal eye fields in specific, of two monkeys while they performed this rather complex task. Using a novel type of population level analysis they could paint a clear picture of how the neural population was treating the two types of information: Both motion and color information were available in the PFC population activity, irrespective of the context. They finally proposed an explanation for this surprising finding using an artificial RNN model that was trained on the same task. In particular, they proposed a mechanism for how a particular decision is formed: Both motion and color information enters PFC, yet only one of them is integrated towards a decision, depending on the context.

For standardization purposes, we here re-analyse the PFC activity during the stimulus presentation period using dPCA (see Materials and Methods and Fig.2.2b). We are demixing motion information in both the motion and color context trials, and find motion information in both (Fig.2.2b top row). Similarly, we demix color information in both contexts, and find color information in both (Fig.2.2b middle row). This reproduces Mante et al. (2013)'s findings. Lastly, we project trials from both contexts onto the first two motion axes and find that in both contexts, motion information is represented similarly (Fig.2.2b bottom left). The same is true for color

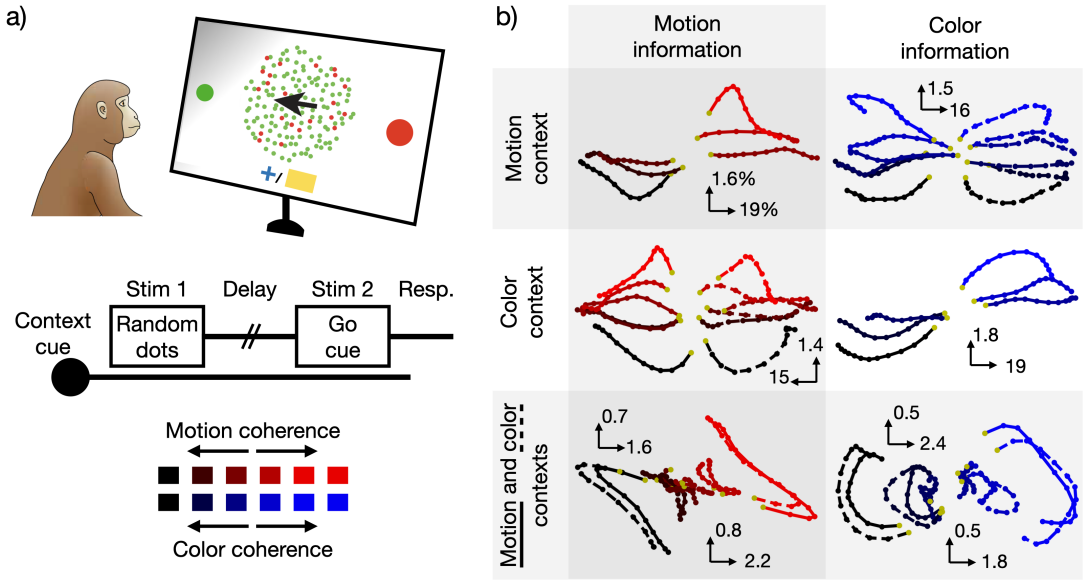


Figure 2.2: PFC representation of context-dependent perceptual decision making task in monkeys (Mante et al., 2013). a) In this task, the monkey has to make a decision based on a random dot stimulus that varies in two ways, the direction (left, right) in which the dots move, and their coloring (red, green). Depending on the context, the monkey has to make a decision either based on motion or on color. Throughout each trial, the context (motion or color) is cued by the shape of the fixation cue (yellow rectangle versus blue cross). After the random dot stimulus and a delay period of random length, the removal of the fixation cue signals the start of the window in which the monkey has to respond by saccading towards one of the targets (green, red). The colors used to code the trial types for the rest of the figure are depicted at the bottom. b) Demixed principal component analyses of PFC recordings. Only data during the stimulus presentation period, i.e. during the random dots, are analysed. In the left column, motion information is demixed, and in the right column, color information. The first two rows correspond to the two contexts. For example, the top left plot shows the first decision component on the horizontal axis, and the first motion component on the vertical axis, from motion context trials plotted against each other. The lines show trial-averages of the six motion coherence categories, and are colored according to the motion coherence. The fractions of explained variance for both components are indicated in the inset. The top right plot shows the first decision component on the horizontal axis, and the first color component on the vertical axis, from the motion context trials. The lines are here colored by the out-of-context color coherence. Solid and dashed lines correspond to the two responses, respectively. The middle row is analogous. The last row depicts the motion and color subspaces. The left plot shows the first two motion components, and the right plot the first two color components. In the last row, the two contexts, motion and color, are coded by solid and dashed lines, respectively. In each panel, horizontal and vertical axes have the same scale. Monkey figure in a) adapted from (Romo and Salinas, 2003)

information (Fig.2.2b bottom right), further reproducing the findings of Mante et al. (2013). In each of the contexts, the first decision component explains around 15% of the variance in the trial-averaged data.

A striking question that these data pose is about the reason why both types of information are present. Is it for a mechanistic reason, as proposed by Mante et al. (2013), or is there a deeper reason why both types of information are present? This type of question points to an important feature that has to be addressed when explaining higher-order brain activity, namely which variables are represented in the first place.

2.4.3 Mouse-ALM representation in a delayed licking task

The study by Inagaki et al. (2019) is uniquely positioned to constrain models of higher-order brain activity. They study the neural responses in ALM of mice in a simple delayed licking task (Fig.2.3a). In this task a mouse is presented, during a sample period, with an auditory instruction stimulus that can take two possible frequencies. After a delay period, a go cue is presented which signals the mouse to report a decision by licking one of two water delivery ports. Which of the two ports is rewarded will depend on the initial instruction cue. Interestingly, they trained mice on two different versions of the task: The first consists of a delay period of fixed length, while the second version consists of a delay period of randomized length. This subtle change in the structure of the task lets us scrutinize the precise dependence of neural representations on such changes. Accordingly, it lets us constrain our models.

Here, we run dPCA on the population recordings from ALM (Fig.2.3b). For both tasks, we see that the first three condition-independent components explain well above half of the variance in the data. Furthermore, the overall profile of these components is quite similar during both tasks: Slow activity up until the go cue, after which activity moves fast. Similarly, the stimulus components also move slow before the go cue, followed by fast dynamics, and again, their profile is quite similar during both tasks.

Apart from all these similarities between the neural representations of the two tasks, there is one striking difference. The dynamics during the delay period are almost stationary during the random delay task, while they are slowly ramping during the fixed delay task. This dynamical switch was already pointed out by the authors on the basis of single neuron responses (Inagaki et al., 2019). Here, we now show that the phenomenon also exists on a population level. It is both visible in the condition-

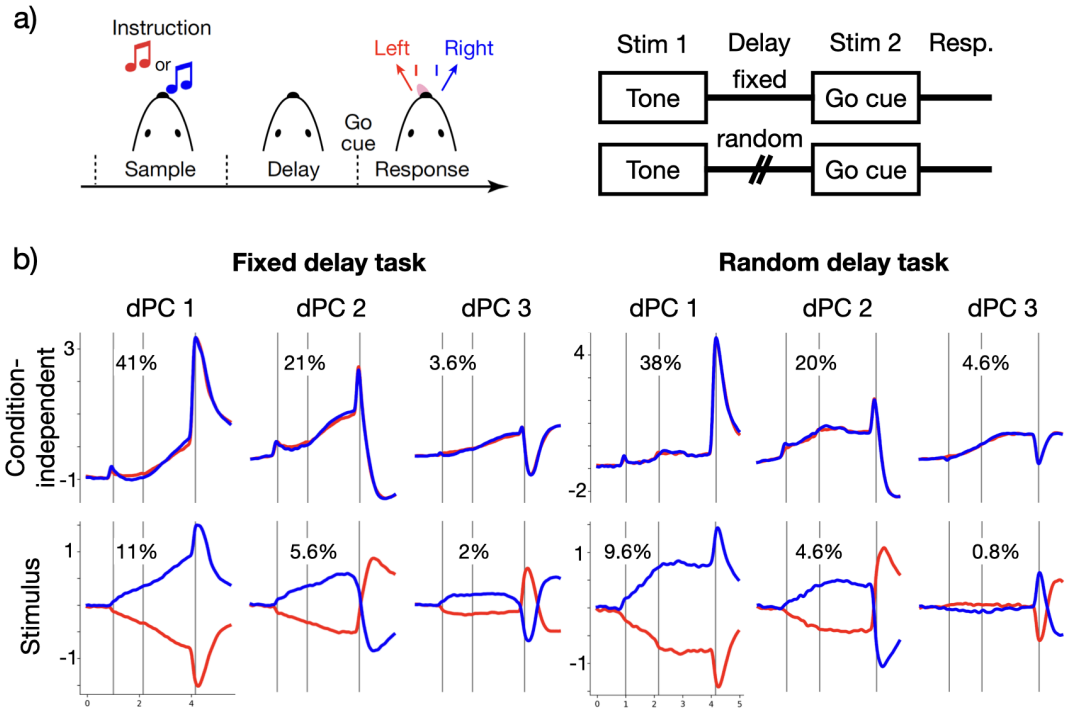


Figure 2.3: ALM representation of two delayed licking tasks in mice Inagaki et al. (2019). a) In this task, a mouse is first presented with one of two possible tones, here color coded by red and blue, during a sample period. After a following delay period, a go cue signals the mouse to respond and report its decision based on the initial tone. There are two versions of the task, depicted on the right. The first has a delay period of fixed length (top), and the second a delay period of random length (bottom). b) Demixed principal component analysis of 755 ALM neurons in the fixed delay task (left) and 1000 ALM neurons in the random delay task (right). Only correct trials are analysed. The top row shows the first three condition-independent components of each task, and the second row the first three stimulus components. As only correct trials are analysed, stimulus and decision coincide. The stimulus components thus include decision information. In the random delay task panel, only trials with a two second delay period are analysed. In each sub-panel, the sample period is indicated by the first two vertical bars and the go cue by the third vertical bar. Left panel of a) from (Inagaki et al., 2019).

independent components, as well as in the stimulus components. For example, consider the first stimulus components. In both tasks, activity is ramping up during the sample period, and while it keeps on ramping up in the following delay period during the fixed delay task, the activity is quite abruptly slowing down in the random delay task.

This dynamical switch based on a slight change in the time structure of the task first shows us that neural representations, at least in ALM, are quite directly influenced by the specifics of the task. Furthermore, it raises the questions of what the ramping dynamics in the fixed delay task mean. In this task, the mice are able to predict when they will be able to respond. This would allow them to anticipate their response in order to then quickly respond once the go cue was presented. Ramping dynamics could thus correspond to preparatory activity. Similarly, the ramping could correspond to a prediction of when the go cue is going to occur.

2.5 Discussion

We have analysed population recordings from three studies, two model organisms and two higher-order brain areas, using dPCA. We saw that all data sets were quite low-dimensional, meaning that only a few dimensions in the high-dimensional neural space actually varied. This might not be surprising, given the low complexity of the tasks studied and a possible constraint on the time scale with which neural systems can vary (see e.g. Gao et al. (2017)). At the same time, there is no reason to believe that neural populations underlie such mechanistic constraints and in principle they should be able to explore all their available dimensions even in the simplest of tasks. This raises the question of why higher-order representations are low-dimensional.

A second overall theme that we observed was that all neural activities showed a clear dependence on the relevant task variables, the reason why we call these activities neural task representations. At the same time, in the task by Mante et al. (2013), we saw the representation of a task variable that was not strictly relevant. Furthermore, how the representation of these task variables, be they relevant or irrelevant, changed over time, i.e. their dynamics, was quite complex, and it is not obvious why we see them.

Such representational dynamics have received quite a lot of attention, especially during delay periods, such as seen in the tasks by Romo et al. (1999) and Inagaki et al. (2019), as this activity might correspond to short-term memory or working memory. When this kind of activity — often called persistent activity — was first observed (Fuster and Alexander, 1971), not much attention was laid on their dynamics. More

recently, researchers have found a variety of different dynamics, from slow, almost stationary dynamics, to faster dynamics, such as ramping. Dynamics where activity is passed from one neuron to the next, so called sequential activity, have also been observed, see e.g. (Fujisawa et al., 2008; Harvey et al., 2012). This has caused a discussion about the nature of working memory (Sreenivasan and D'Esposito, 2019). Mechanistic arguments have dominated this discussion, such as whether working memory is implemented by attractor or by feedforward dynamics (Goldman, 2009; Ganguli et al., 2008). The study by Inagaki et al. (2019) shows, though, that task structure has an immediate impact on the delay dynamics. This raises the question of what the underlying reason for such dynamics is.

Lastly, we saw that condition-independent activity contributed heavily to the overall neural variance. As it is locked to the trial structure, it probably has also a role to play in the way the respective brain areas are functioning. What this role is, is another open question.

2.6 Materials and Methods

All demixed principal component analyses (dPCA, Kobak et al. (2016)) were done with the python (Van Rossum and Drake Jr, 1995) module from (Kobak et al., 2016). The analysis was regularized to prevent overfitting, and the value of the regularization parameter was determined by cross-validation.

2.6.1 Somatosensory working memory task

Extracellular recordings from two monkeys were available under `https://crcns.org/data-sets/pfc/pfc-4/about-pfc-2` (Romo et al., 1999; Brody et al., 2003; Romo et al., 2016), with a total of 1354 units. We excluded all error trials and all units with less than 5 trials per condition from further analysis. This yielded 785 units. For pre-processing, we binned spikes in 10 milliseconds bins and then convolved them with a Gaussian kernel with a standard deviation of 50 milliseconds. We then ran dPCA as described in (Kobak et al., 2016).

2.6.2 Context-dependent perceptual decision making task

We downloaded the data referred to in Mante et al. (2013), that includes PFC activity of two monkeys, A and F, under `https://www.ini.uzh.ch/en/research/groups/mante/data.html`. Data is available from 100ms after stimulus onset until 100ms after dots offset. Here, we analysed data from monkey A only (as in Fig. 2 of

Mante et al. (2013)). This yielded a total of 762 units. We then binned spike trains in 40ms windows. For monkey A, average motion coherences were 0.05, 0.15, and 0.50, in both directions, and average color coherences were 0.06, 0.18, and 0.50, in both directions. We used these averages as categories and every specific trial was assigned to the closest category. All together we thus have 36 stimulus categories. Together with the two contexts, we have 72 categories or trial types. We call this data set D.

Our goal is to demix context, motion, color, and decision information in this data set. Unfortunately, for many of the 72 trial types we do not have any data. For example, for easy trials, e.g. high motion coherence, there will be very few, if any, error trials. As a result, for such a trial type, we won't be able to demix decision from motion coherence. In order to allow dPCA to demix the numerous parameters in this task, we therefore ran a linear regression, following Mante et al. (2013), and used the result to interpolate the missing trial categories. The activity of each unit i in trial k was modelled as a linear function of the trial parameters, i.e. the context $e_k \in \{-1, 1\}$ for motion and color contest respectively, the normalized motion coherence $-1 \leq m_k \leq 1$, the normalized color coherence $-1 \leq c_k \leq 1$, the decision $d_k \in \{-1, 1\}$ and all their products with the context:

$$\begin{aligned} r_i^k(t) = \beta_i^e(t)e_k + \beta_i^m(t)m_k + \beta_i^c(t)c_k + \beta_i^d(t)d_k + \beta_i^{em}(t)e_k e_m \\ + \beta_i^{ec}(t)e_k e_c + \beta_i^{ed}(t)e_k e_d + \beta_i(t) \end{aligned}$$

Here, $\beta_i^j(t)$ corresponds to the linear coefficient, which is a function of time, of unit i and task variable j, and $\beta_i(t)$ is an offset, or condition-independent, term. The coefficients were fit by linear regression, which we then used to reconstruct the unit activities. Furthermore, we used the coefficients to generate, or interpolate, data for the missing trial types. In particular, as this linear regression only models the average response per trial type, we added independent Gaussian noise with standard deviation 0.1 to the trial parameters, and generated the corresponding interpolated rates. By repeating this process ten times, we ended up with ten interpolated trials for every trial type. We call the resulting data set D_{LR}. We then have two data sets, the interpolated data set D_{LR} and the original data set D.

We then ran dPCA on D_{LR}, demixing context, motion coherence, color coherence, decision, and time. The regularization weight was optimized by cross-validation on the interpolated trial data. This analysis gave us the principal axes in neural state space that demix context, motion coherence, color coherence, decision, and time. We then used the original data set D, and projected the trial averages from D, which we

further smoothed by a Gaussian kernel with a standard deviation of 120ms, onto these principal axes. First, we selected all trials from the motion context, sorted them by motion coherence, and projected them onto the first decision axis and the first motion axis (Fig.2.2b top left). The fraction of explained variance indicated in the panel was calculated using the motion context trials only. Second, we sorted the trials by color coherence, i.e. by the irrelevant stimulus, and projected them onto the first decision axis and the first color axis (Fig.2.2b top right). The middle row of figure 2.2b shows the analogous projections for the color context trials. Lastly, we projected both the motion context trials as well as the color context trials onto the first two motion axes (Fig.2.2b bottom left), and the first two color axes (Fig.2.2b bottom right).

2.6.3 Delayed licking task

We downloaded the data referred to in Inagaki et al. (2019) under `https://figshare.com/articles/dataset/Discrete_attractor_dynamics_underlies_persistent_activity_in_the_frontal_cortex/7489253`, that includes silicone probe recordings from both the fixed delay task and the random delay task. We only analysed correct trials without any perturbations. The fixed delay task data was recorded from six mice in a total of 20 sessions. We pooled all recordings which yielded a total of 755 units. In the random delay task recordings were taken from nine mice in a total of 34 sessions, which resulted in 1307 units.

In both tasks, we only kept trials with a delay period of 2 seconds. We then binned spikes in non-overlapping windows of 10 milliseconds, the results of which we then further convolved with a Gaussian with a standard deviation of 50 milliseconds. We then aligned all trials to the sample period onset, and analysed data from one second before the sample period onset up through 1.5 seconds after the go cue for the fixed delay task, and up through 1 second after to go cue for the random delay task.

We then ran dPCA by demixing stimulus information and condition-independent information. For plotting, we further smoothed the data by convolution with a Gaussian with a standard deviation of 100 milliseconds.

Chapter 3

Efficient representations for habitual and model-based behaviours

3.1 Contributions

Conceptualization and method development by Severin Berger and Christian Machens. Simulations were done by Severin Berger.

3.2 Summary

We set out to develop normative models for neural representations in higher-order brain areas. While such neural representations often show a clear dependence on the task at hand — and indeed, models with recurrent neural networks (RNNs) trained for the same task often reproduce the representations well — it is often unclear why they look the way they do. To answer this 'why' question, we first define a clear hypothesis about the purpose of a particular observed representation, and then make a normative prediction about what a neural task representation should look like based on the hypothesised purpose. Specifically, we examine two purposes, the purpose of supporting habitual behaviour and the purpose of supporting model-based behaviour, and we formalise the two in a generalised framework of partially observable reinforcement learning. To make a normative prediction, we must then choose a representation or state space of reinforcement learning that is sufficient to satisfy the purpose, but

there are infinitely many from which to choose. Thus, we invoke the information bottleneck principle to find a representation that is as compact as possible but still purpose-sufficient, and we call such a representation efficient. Efficient habitual and efficient model-based representations can then be compared to neural recordings, and we do this for various tasks and neural recordings in both monkeys and rodents. Not surprisingly, for two datasets of monkey PFC (prefrontal cortex) recordings, we find that the efficient model-based representations match better than the efficient habitual ones. At the same time, however, this is surprising because the RNNs used in previous models generally follow the habitual hypothesis. In the mouse ALM (anterior lateral motor cortex), we find better agreement with the habitual hypothesis.

3.3 Introduction

Higher-order brain areas, such as the PFC, are thought to be crucial for enabling complex animal behaviours (Fuster, 2015). Understanding how these brain areas work is thus of interest not only from a neuroscientific perspective, but from an engineering perspective as well, as today's artificial intelligence systems often lack the versatility and flexibility of animal intelligence (Lake et al., 2017; Russin et al., 2020). Scrutinizing higher-order brain areas has thus a long history, and for example for PFC, over the years, a consensus about their function has crystallised: it is involved in the executive functions of planning, decision making, working memory, and cognitive control (Fuster, 2015). However, the neural basis of these functions is often unclear.

When looking at higher-order brain representations, they often vary in complicated ways across stimuli (Rigotti et al., 2013), tasks (Inagaki et al., 2019), and time. How neural activities vary over time has specifically raised interest in tasks that require to keep a memory of certain task variables. In such working memory (WM) tasks, neurons sometimes fire with slowly varying rates (Fuster and Alexander, 1971; Funahashi et al., 1989; Inagaki et al., 2019), and in other tasks, they fire more dynamically (Romo et al., 1999; Brody et al., 2003), or even in sequence (Fujisawa et al., 2008; Harvey et al., 2012). Indeed, dynamic WM activity is rather the rule than the exception (Sreenivasan and D'Esposito, 2019). Yet why such dynamics are present has remained unclear.

Higher-order brain activities, such as for example during WM tasks, have so far mainly been modelled from a mechanistic, network perspective. Earlier work has focused on how a network of neurons can store memory over time (Goldman, 2009; Ganguli et al., 2008; Compte et al., 2000; Ben-Yishai et al., 1995). A next iteration of

network models adopted a perspective that is more oriented towards the function of higher-order brain areas (Wang, 2002; Machens et al., 2005). Nowadays, mostly RNNs are used that have been trained on the same task as the animal. Such RNN models ask how higher-order brain areas may implement task solutions, and they have often been quite successful at reproducing task representations in higher-order brain areas (Barak et al., 2013; Mante et al., 2013; Song et al., 2016; Inagaki et al., 2019). However, the training of RNNs is usually an ill-posed problem, in the sense that, at least in principle, multiple solutions exist for any particular task (Maheswaranathan et al., 2019). Accordingly, a specific match of a trained RNN to data can be serendipitous, providing only limited insight into the reasons underlying the similarity.

At the same time, none of the RNN solutions may match the neural data in the first place. In such cases, the training targets, and thus the underlying assumption about the modelled neural representation's purpose, may be adapted until a match is found, for example by requiring an RNN to produce its output in a different form or by requiring it to output additional variables (Cueva et al., 2020). However, this raises the question whether a given match of a model to neural data is due to the particular choice of the training targets or rather due to the ill-position of the training objective.

Here we take a normative approach by first stating the purpose of an agent's internal task representation, and thus, the purpose of a modelled neural representation. We distinguish two purposes: First, the purpose of a 'habitual agent' is to take the correct actions. This is the purpose usually assumed in the RNN approach. Second, the purpose of a 'model-based agent' is to predict all ethologically relevant observations, such as task stimuli and rewards. We define these two behavioural strategies within the framework of partially observable reinforcement learning. Each strategy imposes different constraints on the representation of task variables and its dynamics. We then follow the efficient coding hypothesis which states that neural circuits should eliminate all redundant or irrelevant information (Attneave, 1954; Barlow, 1961; Simoncelli and Olshausen, 2001). Specifically, we search, among all representations consistent with the assumed purpose, the one that eliminates all irrelevant and redundant information, and we call such task representations efficient.

We follow the efficient coding hypothesis for several reasons: First, it has already proven useful in explaining the tuning properties of neurons in several sensory areas, namely in the primary visual cortex (Bell and Sejnowski, 1997; Olshausen and Field, 1996; Simoncelli and Olshausen, 2001), in the retina (Atick and Redlich, 1992), and

in the cochlea (Lewicki, 2002). It is therefore attractive to test the efficient coding hypothesis for higher-order brain areas as well. Second, from a modelling perspective, given a task and a purpose, one should always use the simplest explanation, according to the principle of parsimony (MacKay et al., 2003). Third, all features in an efficient representation must either follow from the particular task, or the assumed purpose. This leaves the interpretation of the resulting model representation more straightforward. Lastly, efficiency is beneficial from a computational perspective as well: efficient representations are cheaper to store as they include only relevant information (MacKay et al., 2003), and they may generalize better over task variations that concern irrelevant information.

We showcase our approach on three tasks: A classical WM task in monkeys with PFC recordings (Romo et al., 1999), a context-dependent perceptual decision making task, also in monkeys with PFC recordings (Mante et al., 2013), and a delayed licking task in mice with ALM recordings (Inagaki et al., 2019). Formally, we parameterize the habitual and model-based representations with switching linear dynamical systems (SLDS, Barber (2006); Linderman et al. (2017); Petreska et al. (2011)) regularized by an information bottleneck (Tishby et al., 2000), which squeezes out all the information in the representation that is not needed to achieve the behavioural purpose. We then compare the resulting efficient representations to the recorded neural representations by a population level analysis (Kobak et al., 2016).

For both of the tasks with PFC recordings, we find that the efficient habitual representations are missing key features. In particular, the characteristic population dynamics recorded by Romo et al. (1999) are not reproduced, and neither is the characteristic representation of out-of-context information as observed by Mante et al. (2013). In contrast, the efficient model-based representations reproduce these key features, and further provide a close match on the population level in general. For the task with ALM recordings, we find the opposite: the rich efficient model-based representation shows little similarity, while the efficient habitual representation matches quite well overall, but misses the key feature of how the ALM dynamics change over task variations. As the modelled representational features are directly interpretable in terms of the purpose they serve, we are able to identify an explanation for how the habitual purpose is insufficient. We then make the corresponding adjustment to the habitual purpose and finally find good agreement between the resulting efficient representation and the ALM recordings. Our approach thus provides potential insights into the goals underlying higher-order brain representations.

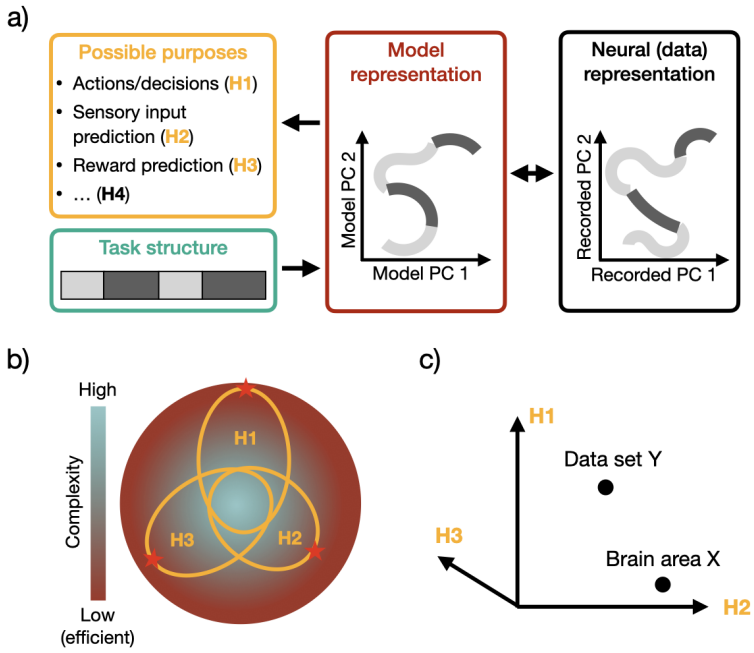


Figure 3.1: A normative framework for modelling neural representations. a) Our goal is to model neural task representations. Our model first depends on the structure of the task at hand, and second, it depends on the purpose that we assume the representation is serving. The purposes correspond to different strategies of solving a task, and they may correspond to predicting the correct actions or decisions (H1), or the sensory inputs (H2), among others. The resulting model representation is compared to the neural representation using population analyses. b) Every purpose may be fulfilled by many representations. The full circle corresponds to the set of all possible representations, and the shading corresponds to the complexity of a representation. Each purpose can be achieved by a subset. We then look for the representation within each subset that is of least complexity, and we call them efficient representations. c) We can find such efficient representations for each purpose, and compare them to the neural representation of a certain brain area or data set (black points).

3.4 Results

3.4.1 A normative framework for modelling neural representations

A neural representation, or any other representation for that matter, is always a representation *of* something as well as a representation *for* something. Thus, the core assumption in this work is that every neural representation serves a purpose. Apriori, we of course can not be sure about what the purpose is, yet if we knew it, we could directly investigate how the recorded neural representation is serving it, which would open up many different avenues of research and therapy. Until then, we have to assume what the purpose could be, in other words, we have to make hypotheses. Given a hypothesized purpose, we can generate the corresponding representation and then gather

evidence for it if the representation compares well to neural data (Fig.3.1a). While this is a straightforward approach, there are two immediate problems in practice.

First, what is a sensible set of hypotheses to test? Preferably, this depends on the prior knowledge we have about the brain area from which the neural recordings originate. Our prior knowledge may tell us about the type of information an area is concerned with. For example, if we are interested in explaining the recordings of a visual area, then our hypotheses will probably have to be concerned with the visual world the animal is experiencing. Similarly, for recordings in motor areas, a good first guess is to consider hypotheses that deal with the movements of an animal. Next, our prior knowledge may inform us about what an area is doing with this information, i.e. its purpose. To continue with the example, if we are interested in an early visual area like the primary visual cortex, a faithful representation of the visual world might be a good guess (Olshausen and Field, 1996; Bell and Sejnowski, 1997; Simoncelli and Olshausen, 2001), whereas a late motor area like the primary motor cortex might directly drive muscle activity (Sussillo et al., 2015). Higher-order brain areas have been implicated in the control of tasks, hence we assume that their purpose is related to the representation of task information in order to facilitate the control of tasks. In particular, the task space will include both sensory as well as motor information, and the purposes will correspond to different strategies of solving tasks (Fig.3.1a).

The second problem is that a given purpose may be fulfilled by many different representations (Fig.3.1b). Because of this, there are various possible model representations of which some might reproduce the data and some others do not. This makes the approach ill-defined. We therefore follow a normative approach and ask how a representation could best serve a purpose in principle. Specifically, we follow the principle of efficient coding which states that a neural representation shall eliminate all redundant and irrelevant information (Attneave, 1954; Barlow, 1961). This corresponds to finding the representation, among the ones sufficient for the purpose, that is the simplest, or the most compressed. We call such representations efficient. Besides rendering the problem well-defined, efficient representations will also be well-interpretable as any representational feature must result from the assumed purpose as applied to the task structure. Furthermore, efficient representations are also beneficial from a biological or computational perspective as they use less resources and should be robust against changes orthogonal to task and purpose.

Finally, the efficient representation of a given task-purpose pair presents a normative prediction. Given a task and a set of neural recordings, one can generate efficient

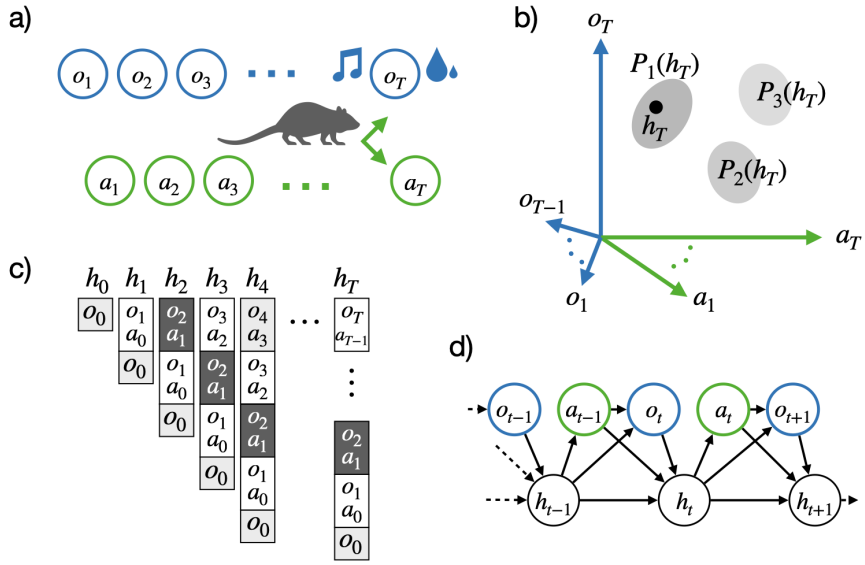


Figure 3.2: The history representation. a) A trial in a task consists of the observations $o_t \in O$ that are presented to the animal, the actions $a_t \in A$ that the animal is required to take, and the temporal ordering of both. Observations may include task relevant stimuli, e.g. a tone, irrelevant noise stimuli, as well as rewards, e.g. a sip of water. b) A trial of length T thus results in a sequence of actions and observations which we call an action-observation history h_T. This history is a point (in black) in a big space that has an axis for the observations and actions at any time step, and we call it the history space. A task can then be defined as a distribution over histories, e.g. P_1, and different tasks, or trial types, correspond to different distributions P_2, P_3, etc. c) The final history h_T is made up of sub-histories. Each of these sub-histories contains all the information that the animal can in principle have about the trial at a given time. As such, they must be sufficient for any purpose, including the prediction of what is coming next in the trial, i.e. actions or observations. d) Histories thus form a valid state space for reinforcement learning with a transition function $P(h'|h, a, o')$, an observation function $P(o'|h, a)$, and a policy $P(a|h)$.

representations for all hypothesized purposes and compare them against the neural recordings (Fig. 3.1c).

3.4.2 The history representation of tasks

Our goal is to find efficient representations of a task for a purpose. We achieve this by first defining a task representation that we know is sufficient for the purpose, and then compressing it. In order to do so, imagine the following: In a particular trial of a task, an animal is confronted with different observations — for example a relevant stimulus for solving the task, a rewarding food pellet, or any other random occurrences — and it has to respond with the required actions. A trial can thus be defined by the resulting sequence of observations $o \in O$ and actions $a \in A$, and we

call such a sequence an action-observation history $h_T = (o_0, a_0, o_1, a_1, \ldots, a_{T-1}, o_T)$ of length T (Fig.3.2a).

As a task is nothing more than a collection of trials, we can define a task as a collection of such histories, or alternatively, as a probability distribution in the history space $P(h_T)$ (Fig.3.2b). The history space is a space that includes an axis for every possible observation at every possible time step. It is therefore a prohibitively large space and it will be hard to define $P(h_T)$ directly. Yet it is also a useful concept, as we can for example define different tasks all in the same space. Our higher-order brain area models are then concerned with this history space.

Histories may further be split into sub-histories, i.e a history h_t for every time step t. A next history h_{t+1} is then simply the current history, plus the action taken and the resulting observation made, i.e. $h_{t+1} = (h_t, a_t, o_{t+1})$ (Fig.3.2c). The sub-histories thus form a Markov chain, given the corresponding action and observation, which is useful to link this task representation to reinforcement learning and Markov decision processes (MDPs, Sutton and Barto (2018)). Specifically, we can define a task as a tuple $\langle H, O, A, P(h'|h, a, o'), P(o'|h, a)\rangle$, which is a MDP equipped with observations, and we call it an observation MDP, or OMDP (Fig.3.2d). H corresponds to the state space of histories. The behaviour in the task is described by a policy $P(a|h)$ which gives the probability of choosing an action a given a history h. The policy that maximizes the collected rewards is the optimal policy, which we assume from here on. The OMDP is a generalization of the partially observable MDP (see Methods).

The OMDP also simplifies the definition of our task distribution $P(h_T)$, making use of its Markov property (see Methods, Eq.3.2). In general though, we could define a task by an OMDP which is not based on history states — any state space allowing for the Markov properties of Fig.3.2c would suffice. Yet, as noted above, the history state contains all information about the task that the agent can know. As such, the history state is a sufficient representation not only for representing a task, but for any purpose that is achievable in principle. To make the history-dependence of our OMDP explicit, we call it a history-OMDP. At the same time, the history is an abstract theoretical object and in practice not realizable, as it for example grows in time without bounds. It is a useful starting point for us, though, as we can now compress it as much as possible while retaining the information that is needed to achieve a certain purpose. One such purpose will be the representation of the task itself, thereby replacing the history-OMDP with an efficient OMDP, as we will see in the next section.

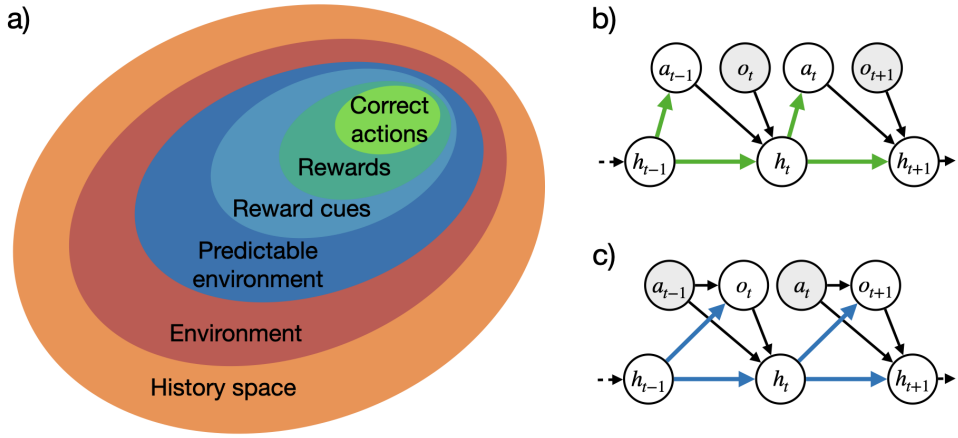


Figure 3.3: A family of behavioural strategies, and the associated purposes. a) The outer orange circle depicts the history space or the set of all histories, and the red circle the tasks or task within it, i.e. the environment. This is equivalent to Fig.3.2b. Note that we are not yet assuming that a task is solved, thus the histories in the environment are the result of any arbitrary actions. Blue circle: Not everything within the environment is predictable (think a random occurrence within the lab such as a person walking in or an earthquake happening). The most a behaving agent can thus do is to use, or model, the predictable subset of the environment, which is the first, most complex, possible behavioural strategy, called model-based. Next, not everything predictable might be relevant to the agent. We follow the reinforcement learning framework and call the variables of interest rewards. Then an agent might only include the prediction of these rewards in its model, as well as any other variables that help predicting future rewards, i.e. reward cues. A simpler strategy still is then to only predict rewards. Finally, the simplest strategy is to fix the actions to the correct actions and only predict these correct actions. This is the habitual strategy. The last strategy has a special standing, as we assume that the correct actions are given. It is the simplest because all the previous strategies require representations rich enough to also predict the correct actions. b) The habitual purpose is to predict the correct actions given a history, i.e. $P(a_t|h_t)$. As histories h_t are not given to an agent, but only observations o_t, the habitual purpose includes — see also Materials and Methods — the transition $P(h_t|h_{t-1}, a_{t-1}, o_t)$. c) The model based purpose is to predict observations given a history and an action, i.e. $P(o_{t+1}|h_t, a_t)$. This again includes the transitions $P(h_t|h_{t-1}, a_{t-1}, o_t)$.

3.4.3 The habitual and model-based purposes

If an animal is able to give the correct response in any task situation, one can say the animal has solved the task. Yet, there are many different strategies of how the animal might come up with the correct responses. The two main such strategies are called habitual and model-based (Dolan and Dayan, 2013; Lengyel and Dayan, 2008; Sutton and Barto, 2018; Tolman, 1948; Skinner, 1938). The former is based on stimulus-response associations, i.e. given any situation in a task, the animal just learns the correct corresponding response. The latter, often also called goal-directed, is based on learning the rules of the task and in particular how an animal's actions will affect the task, and whether they will lead to the goal or not. The animal can then

infer what the correct responses would be, and then simply execute them (Sutton and Barto, 2018; Dolan and Dayan, 2013).

The fundamental difference between the two strategies is how much information the agent learns, or represents, about the task or the environment it is in (Fig.3.3a). While the result is apparently the same — a sequence of correct actions and a solved task — the situation can change if the task is modified. As a brief example, assume a simple devaluation task (Balleine and Dickinson, 1998) where a water-deprived animal has to choose among two options, a water-rewarded option A or a sugar-rewarded option B. The animal would thus have to choose option A. Now, if the animal, e.g. while resting over night, drinks a lot of water, water will be devalued. Using a model-based strategy, the animal can then infer that option A is now undesirable, as it would lead to water, and immediately choose option B instead. This is different to a habitual agent that would first have to unlearn that A is desirable and then learn that B is.

How much information an agent has about the environment thus directly influences how well it can adapt to changes. Besides the habitual and the model-based strategies, a number of intermediate strategies exist that use intermediate levels of information (Fig.3.3a). For example, instead of trying to remember all the inner workings of the environment, an agent might only keep information about how it can achieve the rewards of a particular task (see also caption of Fig.3.3a). Such a representation still allows for inferring the optimal behaviour given the simple changes in the example task above, yet maybe not for more drastic changes.

As keeping more information is also more costly, which strategy to use will depend on how certain an agent can be that the environment or task will not change. In particular at the early stages of exploring an environment, the agent cannot be certain and might choose a rich model-based representation. If the agent knows the task and is sure it doesn't change, it should switch, perhaps gradually, to a habitual behaviour. As such, the different strategies are not mutually exclusive and animals might be able to arbitrate between them based on which better fits their current needs (Dolan and Dayan, 2013; Lengyel and Dayan, 2008; Wassum et al., 2009). Yet whether, at a given time and a given task, an agent is using one strategy or the other is often challenging to disambiguate, even when using tasks that were specifically designed to this end (Akam et al., 2015).

Here we are interested in the neural bases of the two classical strategies. Following our approach, we hence formalize two purposes: the habitual purpose and the model-based purpose. The habitual purpose is to be able to respond with the correct action

given any task situation. Within a history-OMDP, which is a sufficient representation for any purpose, situations are represented by histories and responses correspond to actions. Any representation that is sufficient for the habitual purpose therefore must have the same amount of information about actions as the history-OMDP. More specifically, a full trial h_T is associated with a sequence of actions $\bar{a}_T = (a_0, a_1, \ldots, a_T)$ with probability $P(\bar{a}_T|h_T)$, and a sufficient representation needs to preserve these probabilities (Fig.3.3b).

The model-based purpose, on the other hand, is to be able to predict what is going to happen, i.e. what are the outcomes, in a trial given any situation and any taken action. Again, in the history-OMDP situations correspond to histories and outcomes to observations (which subsume rewards). A full trial h_T is then associated with a sequence of observations $\bar{o}_T = (o_0, o_1, \ldots, o_T)$ with probability $P(\bar{o}_T|h_T)$ and any sufficient representation needs to preserve these probabilities (Fig.3.3c). The model-based representation then has all the information necessary to plan for and represent the correct actions (see Materials and Methods and Sutton and Barto (2018)).

3.4.4 Efficient task representation for a purpose

Given the definitions of tasks and purposes within the history-OMDP framework, we can now find the associated representations. As mentioned initially, each purpose can be fulfilled by many representations. For example, the history representation fulfills all of them, and a model-based representation can also be used for habitual behaviour.

We follow here the efficient coding hypothesis and aim to find the most efficient representation for a given purpose. This means, for any given purpose, the relevant variables must be remembered, and the irrelevant ones must be forgotten. We achieve this by compressing the history space H into a new representation space $\bar{Z}$, while preserving all information about the purpose. Formally, we are looking for an *encoder* $P(\bar{z}|h)$ that maps any history h of length T, sampled from the task distribution $P(h)$, into a sequence of states $\bar{z} = (z_0, z_1, \ldots, z_T)$, as illustrated in Fig.3.4a. At the same time, using the same sample h, we get, as introduced above, a sample from the purpose, which corresponds to a sequence of actions $\bar{a}$ sampled from $P(\bar{a}|h)$ in the habitual case and a sequence of observations $\bar{o}$ sampled from $P(\bar{o}|h)$ in the model-based case. For ease of exposition, we denote here both $\bar{a}$ and $\bar{o}$ by $\bar{y}$, with corresponding distribution

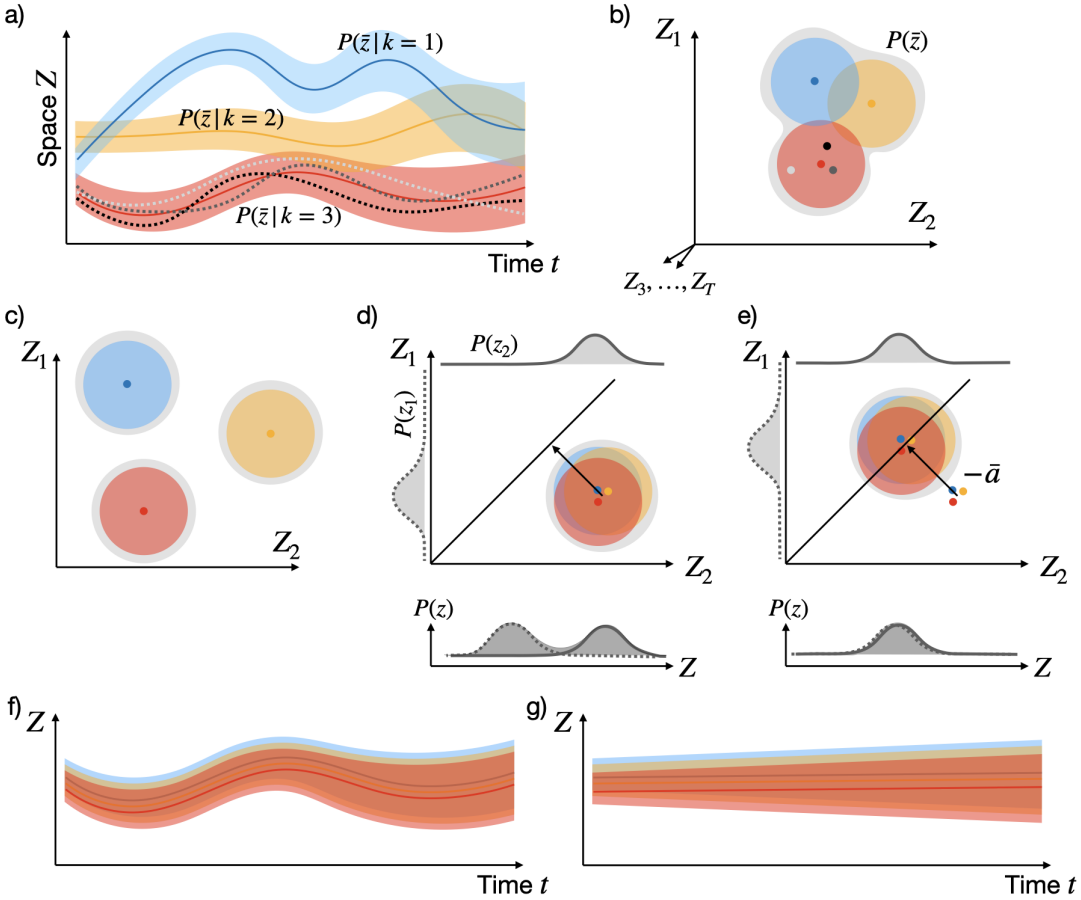


Figure 3.4: Relationships between components determine the entropy of the representation as a whole. a) Any history h, or the corresponding discrete trial type k, is encoded into a distribution or component in space $\bar{Z}$. Here we draw three such components, color-coded in blue, yellow, and red. The representation $P(\bar{z})$ is then a mixture distribution with one component $P(\bar{z}|k)$ per trial type k. For each component, we also draw its mean as a slightly darker line. Given a trial type, different trial instances, or histories, will result in representation trajectories that lie within the shaded region (drawn as dotted lines in different shades of grey for the red trial type). b) We assume that each component $P(\bar{z}|k)$ can be described by a fixed-variance Gaussian. The representation is then a Gaussian mixture model that we can depict more simply: We draw an axis corresponding to space Z for every time step and each component is a Gaussian in this new space, depicted as colored circles. We here only draw the axes Z_1 and Z_2 for the first two time steps. The mean-trajectories of a) are then single red, yellow, and red dots in b), and so are the trial instances in shades of grey. The mixture distribution $P(\bar{z})$ is depicted as a grey shading. c,d) Given fixed-variance components, the entropy of $H(\bar{Z})$ is c) maximal if all components are non-overlapping, and d) minimal if all components are completely overlapping. d-g) Entropy is invariant under addition of constants $\bar{a}$, i.e. $H(\bar{Z} + \bar{a}) = H(\bar{Z})$. Thus, the entropy $H(\bar{Z})$ is independent of its overall mean. I.e. d) and e), where the mean was translated by $\bar{a}$, are equivalent in terms of entropy. Yet, the addition of $\bar{a}$ will affect the entropy of the marginal distribution $P(z)$. We draw in d) and e) the time marginals $P(z_1)$ and $P(z_2)$ in grey on the corresponding axes, and the combination of the two marginals is depicted on the bottom panels. If $P(z_1)$ and $P(z_2)$ are overlapping, as in e), then the entropy of $P(z)$ is minimal. f,g) The corresponding trajectory views of d) and e), respectively.

$P(\bar{y}|h)$. Our objective is then to maximize the following:

$$\max_{P(\bar{z}|h)} I(\bar{Y};\bar{Z}) - \lambda H(\bar{Z}) \tag{3.1}$$

This corresponds to the information bottleneck principle (Tishby et al., 2000; Strouse and Schwab, 2017). The term $I(\bar{Y};\bar{Z})$ is the mutual information between the representation $\bar{z}$ and the purpose $\bar{y}$, and it has to be maximized.[1] At its maximum, we know that our representation has as much information about the purpose as possible, which will ensure purpose-sufficiency. $H(\bar{Z})$ is the entropy of the new representation, and it should be minimal. Entropy is an information theoretic measure of how surprising any particular instance $\bar{z} \sim P(\bar{z})$ of a representation $P(\bar{z}) = E_{P(h)}[P(\bar{z}|h)]$ is. If every history h leads to the same representation $\bar{z}$, then the entropy is minimal. The parameter λ trades off the two terms, and we choose the largest value such that the purpose information $I(\bar{Y};\bar{Z})$ is still at its maximum.

Unfortunately, the objective of Eq.3.1 is difficult to optimize, as both terms are often intractable to compute. For example, to compute the entropy $H(\bar{Z}) = E_{P(z)}[-\log P(\bar{z})]$ of a representation, we need to compute $P(\bar{z})$, which involves an intractable expectation over all histories. Luckily, it is possible to make variational approximations to both terms in Eq.3.1, which allows us to formulate a lower bound on the objective that we can maximize instead (see Materials and Methods and Chalk et al. (2016); Alemi et al. (2016)). Again for the case of the entropy, we can replace the intractable $P(\bar{z})$ by a variational distribution $Q(\bar{z})$, where $E_{P(z)}[-\log Q(\bar{z})] \geq H(\bar{Z})$ yields an upper bound on the true entropy. At the same time, this requires us to choose particular variational distributions, such as $Q(\bar{z})$, that in turn potentially influence the results. Since we are interested in compressed representations, a good choice for the variational approximation to the entropy term is especially important, as inappropriate variational distributions $Q(\bar{z})$ might lead to a strong overestimation of the true entropy. This would leave the representations poorly constrained, and as a result, the representations may contain a lot of irrelevant information. To ensure that our choice of $Q(\bar{z})$ yields an effective, or close, upper bound, we next discuss what kind of distributions $P(\bar{z})$ we can expect in the first place and what it means for such distributions to be of low entropy.

For this, assume we have a finite number of histories, trial types, or even tasks, k where each is approximated by a Gaussian distribution of fixed variance in $\bar{Z}$ space. The distribution of our interest $P(\bar{z})$ is then a Gaussian mixture model (Fig.3.4a,b). In

[1] Here, capital letters denote random variables.

such a case we know what maximizes its entropy and what minimizes it (see Materials and Methods). If all Gaussians are well-separated, it is maximal, which means that each trial results in a very distinct representation (Fig.3.4c). On the other hand, if all Gaussians overlap, the entropy is minimal, which means that all trials result in the same distribution (Fig.3.4d). Furthermore, a similar argument can be made for how the components vary over time. If the sub-components, i.e. components sliced in time, overlap over time, entropy will be further minimized (Fig.3.4d-g). The entropy of a representation will thus be mainly influenced by how the representation changes upon changing the trial, e.g. the specific stimuli in a trial, as well as changing time, e.g. how far in the trial we are. A representation that does neither change over time nor over the different trials has low entropy, and is as simple as possible also from an intuitive perspective. We thus define a variational distribution $Q(\bar{z})$ that captures these component relationships (see Materials and Methods for more details).

Our objective of Eq.3.1 is finally optimized with respect to the encoder $P(\bar{z}|h)$. Hence, we need to define the encoder's parameterization. We do this by using a type of switching linear dynamical system (SLDS, see e.g. Barber (2006); Linderman et al. (2017)). Briefly, the SLDS can be understood as a collection of linear dynamical systems and a corresponding partitioning of its state space Z. As observations are made and actions are taken, the state z is moving through Z according to the linear dynamical system of the current partition. As a result, the encodings $\bar{z}$ can be flexible nonlinear functions of the histories h, yet the dynamics in Z are locally linear (Materials and Methods). As both terms in the objective rely on expectations over the encoder — e.g. the entropy $H(\bar{Z}) \leq E[-\log Q(\bar{z})] = -\int dh\ d\bar{z}\ P(h)P(\bar{z}|h)\log Q(\bar{z})$ — we can simplify the optimization by reparameterizing the encoder, thereby making the expectation independent of it (see Materials and Methods, and Kingma and Welling (2014)). Finally, the optimization is done by gradient ascent.

3.4.5 Applications of the approach

We now analyse the neural recordings of three tasks, each with its own distinct features. The first by Romo et al. (1999) is a working memory task with monkey-PFC recordings that has been subject to a lot of analysis and modeling over the years (for example see Machens et al. (2005); Barak et al. (2010, 2013); Song et al. (2016); Machens et al. (2010); Kobak et al. (2016)). The second by Mante et al. (2013) is a context-dependent perceptual decision making task, also with recordings from the PFC of monkeys. Having two data sets from PFC lets us test the consistency of our results. Lastly, we test data from a different model organism. The task by Inagaki

et al. (2019) is a delayed response task with recordings in mouse anterior later motor cortex (ALM). The ALM, which is a possible homologue of the monkey premotor cortex, is downstream of the PFC and we thus possibly expect a different purpose for this area.

The analysis is done the following way: We generate both the efficient habitual as well as the efficient model-based representations for each task. These efficient representations are normative predictions of neural representations given the assumed purpose. We then analyse both the neural and the efficient representations on the population level using dimensionality reduction techniques (demixed principal component analysis, dPCA, Kobak et al. (2016)). We may encounter three possible scenarios: If both efficient representations match the neural one, then the task is not rich enough to disambiguate between the two hypotheses. We can then conceive of possible task alterations which would disambiguate between the different purposes. If one of the two matches, then we have some evidence that the matching purpose might actually be the purpose of the analysed area, and also that the non-matching one is missing something about the neural activity. Similarly, if none match, then we have to update our hypotheses.

Monkey-PFC in a somatosensory working memory task: Working memory dynamics can be understood as a sequence through a cognitive model.

The somatosensory working memory task by Romo et al. (1999) is a classical study that demonstrated persistent working memory activity in lateral PFC of monkeys. In their task, the authors asked a monkey to discriminate the frequency of two vibrotactile stimuli, F1 and F2, that are presented one after the other and separated by a fixed delay period. In a particular trial, the monkeys had to choose and push one of two buttons, depending on whether the difference between F1 and F2 was positive or negative (Fig.3.5a). Interestingly, many of the recorded units encoded frequency by adjusting their firing rates proportionally to it, even so during the delay period which occurred after the stimulus presentation, a feature expected of a memory trace. Some of these neurons fired this way starting from F1 presentation up until the decision, yet others were inactive during presentation, but became active and F1-encoding during the delay period (see Fig.2.1b). In fact, the recorded population as a whole showed a multitude of firing patterns, suggesting that the working memory activity was more complex than what is probably expected from a simple stimulus short term memory (Brody et al., 2003; Machens et al., 2010). The zoo of firing patterns throughout

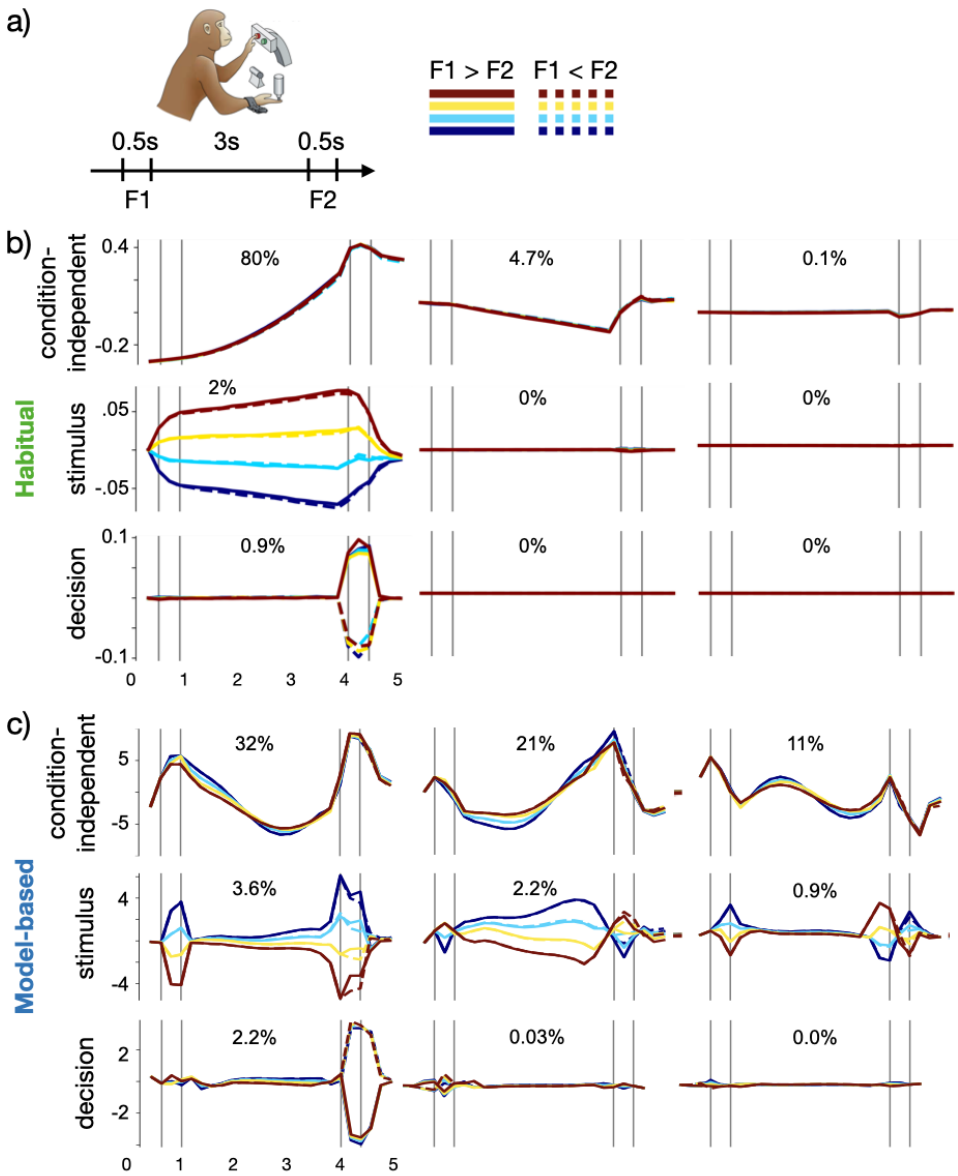


Figure 3.5: Modelling results of the somatosensory working memory task. a) Task illustration (for detailed description see Fig.2.1) and color code used in following panels. b) Demixed PCA of efficient habitual representation. The first row corresponds to the first three condition-independent components, the second row to the first three stimulus components, and the last row to the first three decision components. c) Demixed PCA of efficient model-based representation. The organisation of this panel is the same as panel b). In both panels only correct trials were analysed.

the population can also be succinctly summarized by dPCA (Kobak et al. (2016), Fig.2.1c).

What is this activity then representing, or asked differently, what purpose is this activity serving? To investigate this question, we generated the efficient habitual and efficient model-based representations for this task (see Materials and Methods). For both representations, the optimizations converged towards an optimum (supplementary Figs.3.1-3.4). We then analysed the resulting efficient representations using dPCA (Fig.3.5b,c). In some respects, the efficient habitual and the efficient model-based representations share similarities, and in other respects, they differ. As necessitated by the working memory task, both representations encode F1 stimulus information throughout the trial (Fig.3.5b,c middle rows). While in the habitual representation this F1 memory is used to predict the correct action, in the model-based representation the memory is used to predict rewards or punishments that are contingent on the difference between F1 and F2, as well as on the action taken. Furthermore, both representations have a decision component that separates trials of opposite decision just after all information is available to make a decision, i.e. after presentation of F2 (Fig.3.5b,c bottom rows).

In terms of the dynamics, the two representations look very different, however. This is best seen in the stimulus components (Fig.3.5b,c middle rows). While the efficient habitual representation is almost stationary throughout the delay period, the efficient model-based representation shows different kinds of ramping dynamics. These kind of dynamics are characteristic of the recorded neural representation also (Fig.2.1b). To predict the correct action, a habitual representation only needs to remember the value of F1 in order to compare it to F2. As any delay dynamics are therefore unnecessary, our efficiency criterion compresses them away. This is different to a model-based representation that is required to preserve all information that is predictive of when F1 and F2 are presented. This requirement leads to fast dynamics during both presentation periods, as well as to dynamics during the delay period. The dynamical picture that we can see in the stimulus components is largely reflected in the condition-independent components as well (Fig.3.5b,c top rows). Surprisingly, however, the condition-independent components of the efficient habitual representation are not stationary, but show ramping dynamics, a feature probably explained by the underlying switching dynamics of the SLDS (see supplementary Fig.3.1 for more details).

Compared to the dPCA analysis of the neural representation (Fig.2.1b), the overall picture of the efficient model-based representation matches much better than that of the efficient habitual representation. This suggests an interpretation of the activity in this task in terms of a model-based representation. As we are here looking at a working memory task, this interpretation also taps into the debate about how PFC delay activity encodes working memory information. There are various reports of delay period activity being more stationary, and other more dynamic (Sreenivasan and D'Esposito, 2019). In our view these different dynamics are all a result of the requirements set by the task as well as the purpose of the brain area recorded. Here, we can understand the working memory dynamics as a sequence through a model that the monkey has made of the task. In this interpretation, the delay dynamics indeed correspond to an encoding of time, as proposed by Brody et al. (2003). However, we additionally propose an interpretation for why time is encoded, namely to predict when the observations are occurring. Furthermore, this time-encoding also provides an interpretation for the condition-independent neural activity which is largely captured by the model-based representation as well (compare Fig.2.1b, top row, with Fig.3.5c, top row).

Monkey-PFC in a context-dependent perceptual decision making task: Out-of-context information suggests a model-based representation

The study by Mante et al. (2013) investigated the neural basis of a typical feature of intelligent animal behaviour, namely the ability to flexibly adapt behaviour to a change of context. Given an identical sensory experience animals can change their behaviour based on internal goals or previous instructions. In their task, Mante et al. (2013) trained monkeys to integrate noisy visual information towards one of two possible choices. The visual information consisted of a random dot display whose dots varied in two ways: they moved either to the left or to the right with a certain coherence, and they were randomly colored either in red or in green with a probability termed the color coherence. During a trial, the monkey was instructed whether they had to attend to the motion coherence or to the color coherence, and then had to respond with a saccade to one of two visual targets depending on whether the overall motion was to the left or to the right, or whether more of the dots were colored in red or in green, respectively (Fig.3.6a).

Performing this context-dependent decision making task relies on flexibly adapting behaviour based on a changing context. The animal thus had to keep track of which stimulus-response contingencies, or task rules, had to be followed at a particular trial.

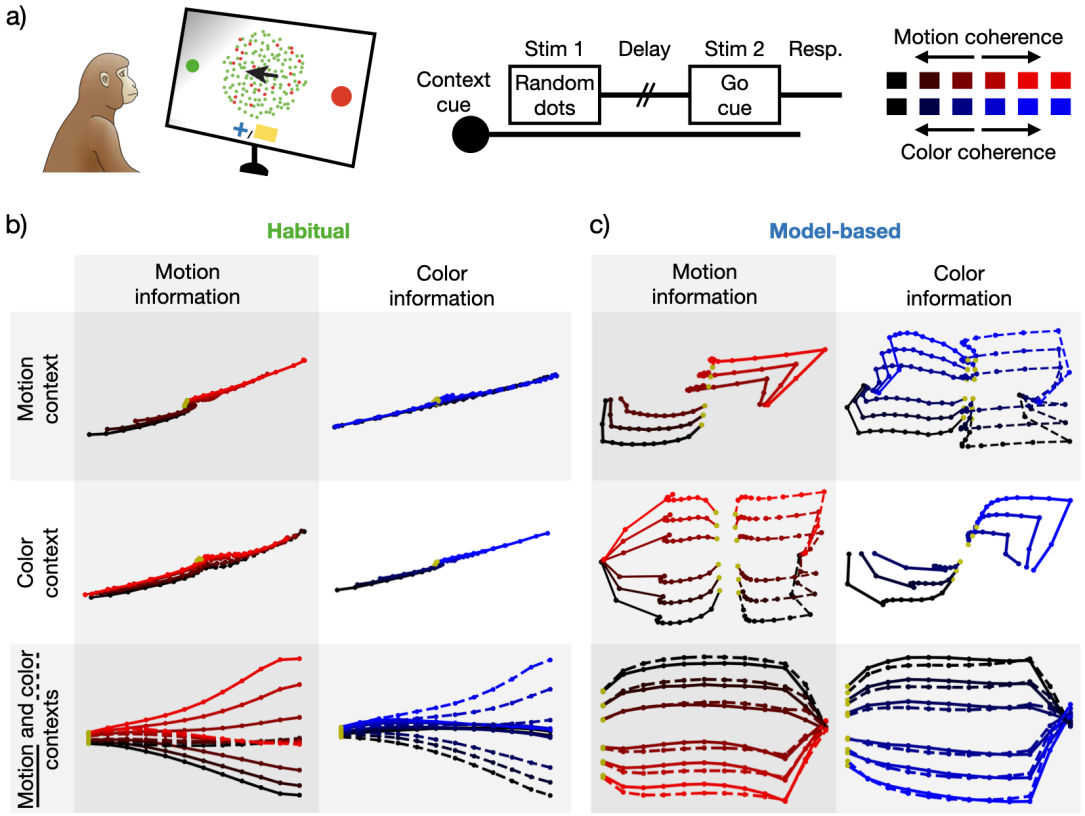


Figure 3.6: Modelling results of the context-dependent perceptual decision making task. a) Task illustration (for detailed description see Fig.2.2) and color code used in following panels. b) Demixed PCA of the efficient habitual representation, and c) demixed PCA of efficient model-based representation. The organisation of these panels is the same as described in Fig.2.2b, except that in the last row we have time on the horizontal, and the first motion component, or the first color component, on the vertical, respectively. We did not plot against the second motion or color components as they did not explain any variance in our models. Again, only correct trials are shown. In each panel, horizontal and vertical axes have the same scale (except for the last row).

Rule representation is a putative function of PFC (Fuster, 2015). Furthermore, the task possibly depends on cognitive control, i.e. the ability to choose a mode of action that goes against the immediate, possibly default, response, a function that is also thought to rely on PFC (Miller and Cohen, 2001). Accordingly, while monkeys were performing the above described task, Mante et al. (2013) recorded activity in their PFC, namely in the frontal eye fields (FEF) that is involved in the control of saccades (Bruce and Goldberg, 1985; Schall, 2002). In order to interpret the resulting PFC activity, as summarized in section 2.4.2 and Fig.2.2b, we again generated the efficient representations of this task, for both the habitual and the model-based purposes (Fig.3.6b,c).

For both representations, the optimizations converged towards an optimum (supplementary Figs.3.5-3.8). Both representations are similar in how they separate trials according to the decision. For example, the top left panel of Fig.3.6b shows how the efficient habitual representation separates trial-averages of positive motion coherences (red shades) from trial-averages corresponding to negative motion coherences (darker shades) on the decision axis, and so does the efficient model-based representation (Fig.3.6c, top left). The same separation can be seen in the rest of the panels of Fig.3.6b,c. However, when looking at relevant information more closely, i.e. motion information in motion context and color information in color context (Fig.3.6b,c, top left and middle right), we can see that the efficient habitual representation barely separates the different trial-averages by the strength of motion/color coherence (i.e. by different shades of red/blue), while the model-based pendant does. As the evidence towards a decision and the relevant coherence coincide, the habitual purpose, which only requires the specification of the correct decision, leads to linear trajectories. Besides integrating evidence towards a decision, the model-based purpose additionally requires to predict when the random dots stimulus is ending, thus explaining the more curved trajectories seen in the top left and middle right panels of Fig.3.6c.

In terms of the 'irrelevant' — or maybe more appropriately out-of-context — information, i.e. motion information in color context and color information in motion context (Fig.3.6b,c, top right and middle left), the two representations again look quite different. As expected, the efficient habitual representation does not keep track of this out-of-context information (note the overlapping trajectories for opposite motion coherences in the color context, and overlapping trajectories for opposite color coherences in the motion context, in Fig.3.6b top right and middle left, respectively). The efficient model-based representation, on the other hand, is representing this information, as it is learning a model of the environment where this out-of-context

information is locked to the trial structure (note that all trajectories in Fig.3.6b top right and middle left are non-overlapping, and thus both types of information are preserved).

Compared to the neural recordings, that we reviewed in section 2.4.2 and Fig.2.2b, the model-based representation provides a better match. Both the data and the model-based representation show out-of-context information and a curved representation of within-context, or relevant, information. Furthermore, motion as well as color information is represented similarly during both contexts, both in the neural recordings as well as in the model-based representation (compare Fig.2.2b, bottom row, and Fig.3.6c, bottom row, where trajectories corresponding to different contexts are overlapping). Our results thus suggest that the reason why we see out-of-context information in the PFC recordings, is that the recorded brain area, i.e. the FEF, is serving a purpose that goes beyond the simple habitual purpose. The model-based purpose is such a purpose, and it reproduces the neural representation during the random dots period quite well.

Furthermore, as the FEF is known to be involved in the control of saccades, one would also expect that the recorded FEF activity actually generates the saccades in the task. The model-based purpose has no notion of action generation, though. However, Mante et al. (2013) proposed a mechanism for how the recorded neural activity is integrating noisy information towards a saccade. Specifically, both within-context as well as out-of-context information is represented in FEF, yet only the within-context information is integrated. As the model-based purpose requires the representation of out-of-context information, our results may suggest why PFC uses the specific mechanism proposed by Mante et al. (2013).

Mouse-ALM in a delayed licking task: Habitual representation with anticipation

Lastly, we consider a study in mice with recordings from ALM (Inagaki et al., 2019). In this study, mice were trained to perform a delayed licking task, where an instruction tone of two possible frequencies was followed by a delay period. The end of this delay period was marked by a go tone after which the mice were allowed to lick one of two ports that contained a water reward based on the initially presented instruction cue (Fig.3.7a). Interestingly, two separate sets of mice were trained on two different versions of the same delayed licking task. The first consisted of a delay period of fixed length and the second of a delay period of randomized length. Thus, this study is uniquely positioned to constrain a model of ALM function. Furthermore,

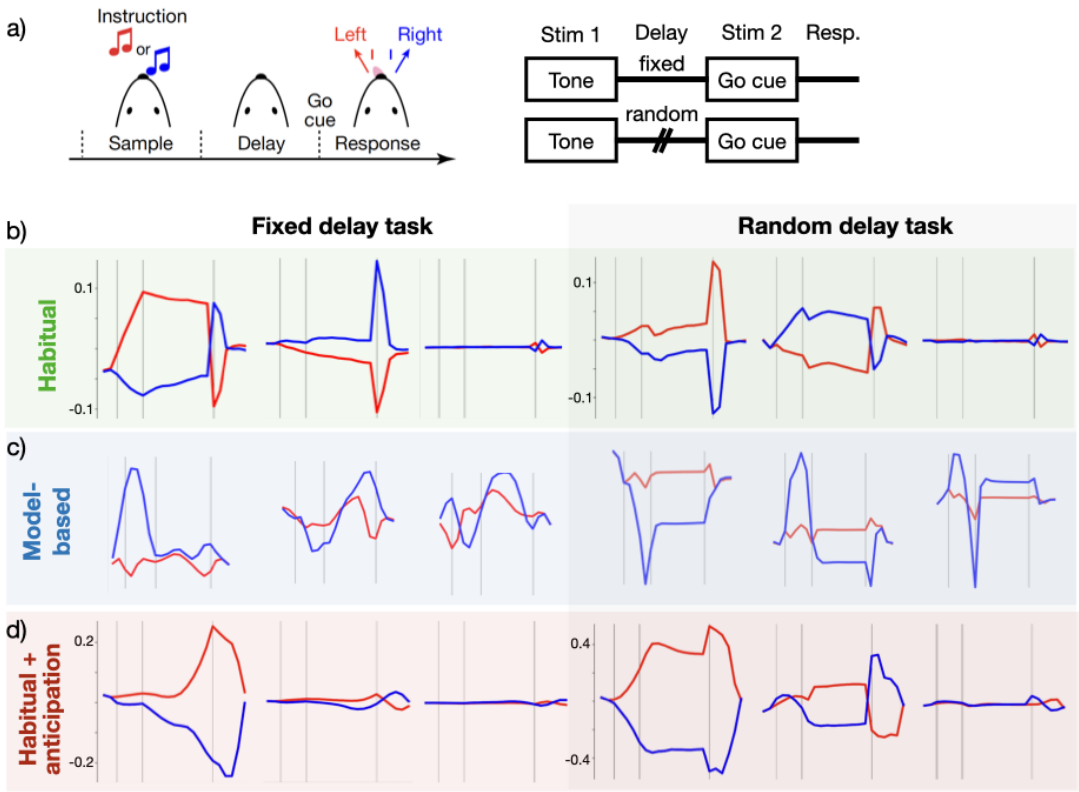


Figure 3.7: Modelling results of the delayed licking task task. a) Task illustration (for detailed description see Fig.2.3) and color code used in following panels. b) Demixed PCA of efficient habitual representation. The first row corresponds to the first three condition-independent components, the second row to the first three stimulus components, and the last row to the first three decision components. c) Demixed PCA of efficient model-based representation. d) Demixed PCA of efficient representation for habitual purpose with go-cue prediction. The organisation of panels c) and d) is the same as panel b).

due to its delayed response structure, this task has a memory component similar to the somatosensory working memory task considered above.

The resulting ALM activities during both tasks were summarized and discussed in section 2.4.3 and Fig.2.3, with the most notable, and almost only, difference between the neural representations being the dynamics during the delay period. While in the fixed delay task ramping population dynamics were observed during the delay period, they were stationary in the random delay task. In order to interpret these findings we again generated the efficient habitual and the efficient model-based representations and analysed them with dPCA (Fig.3.7b,c).

As opposed to the two previous cases, none of the efficient representations match the neural data: The efficient model-based representations seem too rich, with strong

transient dynamics during the sample period in both tasks (Fig.3.7c). The efficient habitual representations resemble the data well on first sight, yet, in the fixed delay task, is missing the ramping component during the delay period (Fig.3.7b). Ramping possibly corresponds to an anticipation of the response period, yet the habitual agent can perform the optimal behaviour without such an anticipation, as the go cue informs the agent when to act. It is conceivable, though, that anticipating when to act is still beneficial. For example, if it takes some fixed amount of time to start an action, such as getting ready to lick the water delivery port, this start can take place before the go cue, but only in the fixed delay task when the go cue is predictable. Alternatively, a lick might follow from a state of 'readiness', such as a state of increased muscle tension. If it is energetically costly to be in and move to such a state, being able to predict when to act, an agent can slowly move to that state and stay there for as little time as possible.

We thus conjectured that a habitual representation with an additional anticipation of the response period might reproduce the data (Fig.3.7d, supplementary Figs.3.9 and 3.10). As in the habitual case, the random delay task is modelled well. As there the time of the go cue is unpredictable anyway, and hence can not be anticipated, the representation is expected to be the same for the habitual purpose and the habitual purpose with anticipation. However, in the fixed delay task we now see ramping dynamics during the delay period, similar to the neural representation. Overall, the efficient representation for the habitual purpose with anticipation reproduces the neural recordings well on both tasks.

Our interpretation of the neural data is thus similar to the one by the authors Inagaki et al. (2019). They model the data as a decision process, where the two decisions, corresponding to a left and a right lick, are instantiated as discrete attractors. The ramping in the fixed delay task is then modeled as a slow drift towards one of the attractors. They hypothesize, though, that the ramping probably is not generated through recurrent dynamics internal to ALM, but rather fed in from a second area. In this work, we are agnostic towards the mechanism, yet we show that the requirement to specify the correct actions, together with the requirement for anticipation, is sufficient to reproduce the data in both tasks. We thus propose a purpose for the ALM, as it was studied by Inagaki et al. (2019), that is consistent with the mechanism proposed by Inagaki et al. (2019).

3.5 Discussion

Higher-order brain areas are thought to be critical for enabling the flexibility and versatility of natural animal intelligence (Fuster and Alexander, 1971). Large population recordings from these areas during various behavioural paradigms give us a view into their involvement in these abilities, as reviewed in chapter 2. The activities can be understood as neural task representations, as all task variables are being represented. Yet what exactly these representations mean is often unclear.

Here, we proposed a normative approach to understand higher-order brain activity. There are two core problems addressed by our approach: First, we point out that every representation serves a different purpose and this purpose has to be clearly stated when modelling the brain area. In this respect, our work differs from previous models of higher-order brain activity, where the purpose is often neither stated explicitly nor compared to alternative hypotheses (e.g. Song et al. (2016); Yang et al. (2019); Barak et al. (2013)). More precisely, as RNNs are usually trained to simply solve a task, the habitual purpose is implicitly assumed. Second, we point out that every purpose may be fulfilled by many different representations. Accordingly, when proposing a purpose for a particular data set, it is apriori unclear which of the many possible purpose-sufficient representations should be compared to neural data.

For the first part, we develop two purposes that correspond to the main ways the field has thought about behaviour (Dolan and Dayan, 2013; Skinner, 1938; Tolman, 1948): Habitual behaviour, in which an agent acts by simple stimulus-response relationships, and model-based, or goal-directed, behaviour, in which an agent acts by planning based on a mental model. We formalized both within the reinforcement learning framework. For the second part, we assumed the efficient coding hypothesis (Attneave, 1954; Barlow, 1961) that selects the one representation among all purpose-sufficient representations that is the most compact or simplest. We called the resulting representations efficient, and such efficient representations provide normative predictions of how a representation should look like given a purpose.

We then applied our approach to three tasks and found that two data sets (Romo et al., 1999; Mante et al., 2013) recorded in two distinct areas of PFC were better explained by the model-based assumption. This is perhaps unsurprising in that PFC is thought to represent the rules of a task (Fuster, 2015; Miller and Cohen, 2001). Nevertheless, in previous models of both tasks (e.g. Barak et al. (2013); Song et al. (2016) and Mante et al. (2013)) the habitual purpose was usually assumed, as these models were trained to produce the correct decision. In the third data set (Inagaki

et al., 2019), we found that the recordings from mouse-ALM are inconsistent with both the habitual and model-based purposes, but consistent with a habitual purpose that additionally anticipates when to respond. On the one hand, this points to a strength of our approach, as we could exclude hypotheses that one might have assumed initially, and simultaneously propose a new hypothesis. On the other hand, the proposed habitual purpose with anticipation does not directly follow from the reinforcement learning framework, but is rather motivated using an additional constraint. In other words, the purpose is a free variable in our approach, and while the habitual and the model-based purposes are well motivated, it is apriori unclear what a good motivation for a purpose might be.

Our work gives a new perspective on some previous studies that have found compressed representations. For example, Sussillo et al. (2015) regularized their RNNs using a very similar loss function to ours, thus implicitly optimizing for compactness and low entropy. In our view, their results of reproducing primary motor cortical activity is thus not surprising under the assumption that this area is serving the purpose of producing muscle activity. Similarly, Schuessler et al. (2020) show that gradient descent optimization induces low dimensional structure in RNNs trained on simple tasks. Thus, by using this training procedure, activities will be low-dimensional as well (Mastrogiuseppe and Ostojic, 2018; Dubreuil et al., 2021), which, given the often good match that RNNs trained this way provide to neural data, potentially provides further evidence that low-dimensional, compressed, representations are actually optimized for by the brain.

Furthermore, our work helps to contextualize discussions of purposeful representations in previous studies. For example, the study by Cueva et al. (2020) also investigated how a different readout, or purpose, affects the resulting model representations, and they point out that the habitual representation is not sufficient to explain the data of (Romo et al., 1999). Furthermore, Song et al. (2017) considered RNNs under an actor-critic objective, comparing both the activites of the actor and the critic networks to neural data. Similarly, Yang et al. (2019) considered RNNs trained to multiple tasks, which can also be understood as imposing a purpose that is richer than the habitual purpose. Lastly, Russo et al. (2018) also considered RNN solution under constraints, namely to prevent entanglement of dynamical trajectories. Flesch et al. (2022) analysed the learning regime of higher-order representations and found rich representations in PFC (data of Mante et al. (2013)), that are highly adapted to the task at hand. Thus, they also investigated the goal or purpose of a higher-order brain representation.

Our work also leaves some questions unanswered. For example, representations in PFC have been shown to preferentially cluster stimuli based on their relevance for informing correct responses, but less by their sensory properties (see e.g. Freedman et al. (2001)). This might go against our model-based proposal where the precise stimulus is being predicted. However, as both tasks with PFC recordings that we studied only used very simple stimuli, there was no conflict between a stimulus' relevance for informing the correct decision and its sensory similarity to other stimuli. For example, in the task by Mante et al. (2013), distances between relevant motion coherences coincide with the information that a particular motion coherence provides about the decision. Nevertheless, while both within-context and out-of-context information need to be predicted in the model-based purpose, only the within-context stimulus is relevant for predicting the reward at the end of the trial. Thus, after the stimulus presentation period, also the model-based purpose should cluster stimuli by their relevance for the decision, and get rid of out-of-context information all altogether. However, we have not investigated this here, as we did not have access to neural recordings beyond the stimulus period for the study by Mante et al. (2013).

Next, here we did not address two important topics, namely mechanism and learning. The SLDS models that we used are in principle very easy to analyse and are thus a good choice to study the underlying mechanism of, for example, how a decision is formed in the habitual representation. In fact, the nonlinear dynamics of RNNs are often partitioned into separate linear regimes to simplify their analysis, effectively yielding SLDSs (Sussillo and Barak, 2013; Vyas et al., 2020; Maheswaranathan et al., 2019). Yet, we left this analysis of mechanism for future work.

Likewise, we have not studied learning. Our habitual and model-based purposes really present two 'extremes', where the habitual one only cares about generating the correct actions and the model-based one attempts to predict everything that is locked to the trial structure. As the efficient habitual representation serves the optimal policy, it clearly does not address how this optimal policy was learned in the first place. As apriori it can not be known what information is relevant for the optimal policy, it is also initially unknown what to compress away. Furthermore, information that is needed for learning might not be needed to represent the optimal policy (see e.g. McCallum (1996)). A model-based representation would thus make sense for learning a new task, as all available information is considered relevant. In fact, it has been suggested that animals transition from model-based to habitual behaviour throughout learning (Dickinson et al., 1983; Dickinson, 1985). However, our model-based representation is not generating any actions. Model-based approaches usually

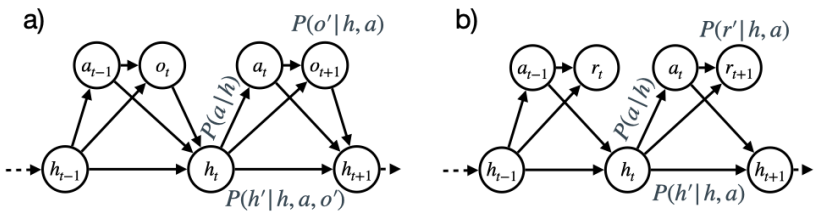


Figure 3.8: Gaphical models of a) the OMDP and b) the MDP, both based on the history state.

use the model to plan for the optimal policy, and it would be interesting to study how planning could be integrated into the model-based purpose.

Lastly, a limitation of the current work is that, even though in theory an efficient representation is only a function of the task and the purpose, our choice of parameterizing the variational distributions as well as the encoder will have an impact. Although the SLDSs we selected for the encoder are quite flexible, they may still place some limitations on the solutions. For example, the condition-independent ramping component in the habitual representation of the somatosensory working memory task is probably an artefact of the SLDS parameterization (Fig.3.5b, supplementary Fig.3.1). For future work, it would thus be valuable to investigate whether our results are reproduced by alternative encoder parameterizations, such as for example by using RNNs.

3.6 Materials and Methods

3.6.1 Definition of a task, a behaviour, and the history state

Our aim is to build normative models of brain activity by finding efficient representations of a given task and purpose. For this, we first need to define what a task is. A task can be completely defined by the observations $o \in O$ that the agent receives, the actions $a \in A$ that the agent is required to take, and the temporal ordering of it all. To formalize, let us assume that any trial that we might observe in such a task, i.e. any sequence, or history, of observations and actions $h_T = (o_0, a_0, o_1, a_1, \ldots, a_{T-1}, o_T)$ of length T, is distributed according to $P(h_T)$. To make the sequential structure of

h_T explicit, we factorize:

$$\begin{aligned} P(h_T) &= P(o_0, a_0, o_1, a_1, \ldots, a_{T-1}, o_T) \\ &= P(o_0)P(a_0|o_0)P(o_1|o_0, a_0)P(a_1|o_0, a_0, o_1)\ldots P(o_T|o_0, a_0, \ldots, a_{T-2}, o_{T-1}) \\ &= P(h_0)P(a_0|h_0)P(o_1|h_0, a_0)P(a_1|h_1)\ldots P(o_T|h_{T-1}, a_{T-1}) \\ &= P(h_0)\prod_{t=0}^{T-1} P(a_t|h_t)P(o_{t+1}|h_t, a_t) \end{aligned}$$

In the last two lines we renamed sub-histories, such as e.g. $h_0 = o_0$ and $h_1 = (o_0, a_0, o_1)$. The relationship between two adjacent sub-histories h and h' can be described by a transition function, here the append-function:

$$P(h'|h, a, o') = \begin{cases} 1 & \text{if } h' = (h, a, o') \\ 0 & \text{otherwise} \end{cases}$$

Using this transition function, we finally have:

$$P(h_T) = P(h_0)\sum_{h_1,\ldots,h_{T-1}}\prod_{t=0}^{T-1} P(a_t|h_t)P(o_{t+1}|h_t, a_t)P(h_{t+1}|h_t, a_t, o_{t+1}) \tag{3.2}$$

This factorization reveals the structure of a Markov-model, more precisely, a Markov decision process equipped with observations (see Fig.3.8a). We call this process an observation Markov decision process, or OMDP, and it is defined by the tuple $\langle H, O, A, P(o'|h, a), P(h'|h, a, o')\rangle$. H is the state space, O is the observation space which includes rewards, A is the action space, $P(o'|h, a)$ is the observation function, and $P(h'|h, a, o')$ is the transition function. The distribution $P(a|h)$ is the agent's behavioural policy. One may recover a standard MDP from an OMDP (Fig.3.8b). A MDP is defined by the tuple $\langle H, R, A, P(r'|h, a), P(h'|h, a)\rangle$, where $R \subset O$ is the space of rewards, $P(r'|h, a) = \sum_{o'\in O\setminus R} P(o'|h, a)$ is the reward function, and $P(h'|h, a) = \sum_{o'} P(o'|h, a)P(h'|h, a, o')$ is the transition function. The OMDP also generalizes the partially observable MDP (POMDP, see section 4.4.1). For ease of exposition we have assumed here that all spaces are discrete, an assumption that we will relax when indicated.

In general, we could define a task by an OMDP which is not based on history states, any state space allowing for the Markov properties of Fig.3.8a would do. Note, though, that the history state contains the entirety of information that the agent can in principle know about a task. As such, the history state must be a sufficient

representation not only for representing a task, but for any purpose that is in principle achievable. To make the history-dependence of our OMDP explicit, we call it a history-OMDP. At the same time, the history is an abstract theoretical object and in practice not realizable, as it for example grows in time without bounds. It is a useful starting point for us, though, as we can now compress it as much as possible while retaining the information that is needed to achieve a certain purpose. One such purpose will be the representation of the task itself, thereby replacing the history-OMDP with an efficient OMDP, as we will see in the next section.

Structured task definition

The simple neuroscience tasks that we are studying in this work are often defined by a finite number of task parameters of which each is varied over a finite number of instances. For example, one such task parameter could be the orientation of a visual stimulus, and in each trial this orientation can either be up-, down-, left-, or rightwards. In such cases we can enumerate the finite number K of trial types k, and, when appropriate, we will use the following structured task description:

$$P(h_T) = \sum_{k=1}^{K} P(h_T|k)P(k)$$

Here, every trial type k produces histories according to $P(h_T|k)$, where experimentally controlled parameters are given. The resulting distribution $P(h_T)$ is then a mixture model, and we assume the probability $P(k)$ to observe a specific trial type k to be uniform.

3.6.2 Definition of a purpose, the habitual and the model-based purpose

Now that we have formalized tasks by history-OMDPs, we next have to define what a purpose is. Formally, for a given trial h_T of length T, a purpose is defined as a sequence of variables $\bar{y}_T = (y_0, y_1, \ldots, y_T)$ that is distributed according to $P(\bar{y}_T|h_T)$.[2] A task-purpose pair is thus defined by $P(h_T)$ — the history-OMDP defining the task — and $P(\bar{y}_T|h_T) = \prod_{t=0}^{T} P(y_t|h_t)$.

In this work we are specifically interested in two purposes that naturally follow from the graphical model in Fig. 3.8a and Eq. 3.2. The first purpose is to represent the animal's behavior or policy, which translates into a sequence of, usually discrete,

[2]We decorate variables denoting sequences with a bar.

actions $\bar{y}_T = \bar{a}_T = (a_0, a_1, \ldots, a_T)$ distributed according to $P(\bar{a}_T|h_T) = \prod_{t=0}^{T} P(a_t|h_t)$ (Fig.3.3b). We call this the habitual purpose and usually assume the optimal policy. An efficient representation of the habitual purpose is reactive in nature and is useful in situations where an optimal policy has already been obtained, i.e. in the over-trained regime. In fact, an efficient habitual representation is neither suited for learning nor for planning. Nevertheless, the habitual purpose is usually assumed when modelling neural activity e.g. in PFC (Barak et al., 2013; Song et al., 2016).

A representation suited for planning and learning needs to contain at least some information about the environment. For planning specifically, a representation needs to be sufficient to predict future observations provided by the environment in response to the actions taken by the agent. We call this second purpose the model-based purpose, which translates into a sequence of, usually continuous, observations $\bar{y}_T = \bar{o}_T = (o_0, o_1, \ldots, o_T)$ that are distributed according to $P(\bar{o}_T|h_T) = P(o_0) \prod_{t=0}^{T-1} P(o_{t+1}|h_t, a_t)$ (Fig.3.3c). On top of a model-based representation, different reinforcement learning algorithms or dynamic programming can be used to plan for an optimal policy. Therefore an efficient model-based representation will always be sufficient to also represent a policy, yet this must not be true the other way around (see e.g. McCallum (1996)). Consequently, for a given task, an efficient model-based representation can not be more compressed than an efficient habitual representation. Indeed, the habitual and the model-based purposes delimit a spectrum of possible purposes, each requiring an efficient representation of different complexity (Fig.3.3a).

3.6.3 Task representation and the memory constraint

Given the definitions of task and purpose, we now formalize the representation of a task. Since our initial task representation based on histories is, as described above, sufficient for any purpose, our approach is to find an encoder $P(\bar{z}|h_T)$, mapping histories h_T to a new representation $\bar{z} = (z_0, z_1, \ldots, z_T)$ with $\bar{z} \in \bar{Z}$ and $z \in Z$. To represent a task thus means to find an encoder $P(\bar{z}|h_T)$, such that the resulting representation is efficient and purpose-sufficient. The optimization procedure ensuring these two constraints will be with respect to the parameters θ of the encoder, and we write $P_\theta(\bar{z}|h)$, where, for simplicity, we have also dropped the time index of h_T.

In principle, our encoder should be as flexible as possible, because the efficient representation of a given purpose might be a complicated function of the histories. Yet, since our goal is to model activities of a specific brain region, we have to constrain the encoder's structure. Specifically, we assume that the modelled neural activities

are the result of a population of neurons that is in some sense autonomous, meaning that its activity is only influenced by itself, and not by a second or third population of which we do not have any recordings from. We call this restrictive assumption on the encoder a memory constraint as it requires our representations to be autonomous in terms of memory.

To formalize this constraint, imagine that we have two consecutive histories h_t and h_{t+1} that are mapped to two corresponding states $\bar{z}_t$ and $\bar{z}_{t+1}$. Then $\bar{z}_{t+1}$ shall not carry more information about h_t than $\bar{z}_t$ does. Formally, we require:

$$I(\bar{Z}_t; H_t) \geq I(\bar{Z}_{t+1}, H_t)$$

This is equivalent to stating that $\bar{z}_{t+1}$ is independent of h_t given $\bar{z}_t$, or $P(\bar{z}_{t+1}|\bar{z}_t, h_t) = P(\bar{z}_{t+1}|\bar{z}_t)$. Since $\bar{z}_{t+1} = (z_{t+1}, \bar{z}_t)$, $P(\bar{z}_{t+1}|\bar{z}_t) = P(z_{t+1}|\bar{z}_t)$. Consequently, z_t will also only depend on $\bar{z}_{t-1}$, and so on, and we have $P(z_{t+1}|\bar{z}_t) = P(z_{t+1}|z_t)$. Therefore, the encoder needs to follow a Markov property:

$$\begin{aligned} P(\bar{z}_{t+1}|h_{t+1}) &= P(\bar{z}_{t+1}|h_t, a_t, o_{t+1}) \\ &= P(\bar{z}_{t+1}|\bar{z}_t, a_t, o_{t+1})P(\bar{z}_t|h_t) \\ &= P(z_{t+1}|z_t, a_t, o_{t+1})P(\bar{z}_t|h_t) \end{aligned}$$

Thus, the encoder $P(z|h)$ factorizes according to $P(z'|z, a, o')$, a transition distribution for our new state space or representation Z. Together with an initial state distribution $P(z_0)$, we have:

$$P_\theta(\bar{z}_t|h_t) = P_\theta(z_0) \prod_{i=1}^{t} P_\theta(z_i|z_{i-1}, a_{i-1}, o_i)$$

The parameters θ of our encoder are thus the parameters of the transition and initial state distributions.

3.6.4 Finding efficient, purpose-sufficient representations

We now describe our encoding objective, i.e. to find a representation such that it is purpose-sufficient but otherwise as compressed as possible. As a result, the encoder should get rid of any information that is irrelevant to the purpose. A natural measure of complexity and thus compression is entropy, which is minimal if the representation consists of a single state. Purpose-sufficiency can be confirmed if the new representation carries as much information about the purpose as the history representation does.

We thus follow a version of the information bottleneck principle (Tishby et al., 2000; Strouse and Schwab, 2017) with objective function $L(\theta)$ which is to be maximized with respect to our encoder $P_\theta(\bar{z}|h)$:

$$L(\theta) = I_\theta(\bar{Y}; \bar{Z}) - \lambda H_\theta(\bar{Z}) \tag{3.3}$$

Here $I(\bar{Y}; \bar{Z})$ corresponds to the mutual information between representation and purpose, and $H(\bar{Z})$ corresponds to the entropy of our representation. The parameter λ determines the trade-off between the the two terms and we are here specifically considering the case where the information about the purpose is at its maximum. Since the maximally achievable value of $I(\bar{Y}; \bar{Z})$ is bound by $I(\bar{Y}; H)$, i.e. the mutual information between the purpose and the history representation, we can confirm that our encoder $P(\bar{z}|h)$ produces a purpose-sufficient representation.

In this work we are considering representation space Z and observation space O to be continuous, while the nature of the space Y will depend on the purpose, as seen in the previous section. While mutual information is in general intractable to compute in continuous spaces, entropy is neither tractable nor well defined. Differential entropy, as entropy of continuous random variables is called, does not have identical properties to the entropy of discrete random variables. Crucially, among others, differential entropy is not bounded from below. This lack of a lower bound can result in pathologies such as the following: Since entropy $H(\bar{Z}|H)$ in the encoder $P(\bar{z}|h)$ can only increase $H(\bar{Z})$, the encoder that minimizes $H(\bar{Z})$ will be deterministic, the reason why the objective in eq.3.3 is sometimes called the deterministic information bottleneck (Strouse and Schwab, 2017). In a continuous space $\bar{Z}$ and with a deterministic encoder, the variation in $\bar{Z}$ can be scaled down arbitrarily, lowering the differential entropy arbitrarily. Therefore, the minimization of $H(\bar{Z})$ with respect to our encoder does not have a well defined minimum.

A simple way of countering this is to make the encoder stochastic by adding independent noise η to our potentially deterministic encoder $P(\bar{z}|h)$. This is a natural modification supported by the fact that neural systems are unlikely to be free of noise in the first place. The noise introduces then a lower bound on $H(\bar{Z})$, making the its minimum well defined. To show this, we write $\bar{z} = \bar{z}' + \bar{\eta}$, where $\bar{z}' \sim P(\bar{z}'|h)$, and then compute the encoder entropy $H(\bar{Z}|H) = H(\bar{Z}' + \bar{\eta}|H) \geq H(\eta)$. Since $H(\bar{Z}) - H(\bar{Z}|H) = I(\bar{Z}; H) \geq 0$, the encoder entropy lower bounds $H(\bar{Z})$ and we have $H(\bar{Z}) \geq H(\bar{Z}|H) \geq H(\bar{\eta})$ (Kirsch et al., 2021). By introducing this lower bound, at the optimum of L, the encoder will have a fixed entropy $H(\bar{Z}|H) = H(\bar{\eta})$. Note that

our objective of eq.3.3 is thus equivalent to the standard information bottleneck with a fixed entropy encoder. The standard information bottleneck as defined by (Tishby et al., 2000), replaces the entropy term in eq.3.3 with another mutual information term:

$$\begin{aligned} L_{\text{IB}} &= I_\theta(\bar{Y};\bar{Z}) - \lambda I_\theta(\bar{Z};H) \\ &= I_\theta(\bar{Y};\bar{Z}) - \lambda(H_\theta(\bar{Z}) - H(\bar{Z}|H)) \end{aligned}$$

Even though we now have a well defined objective, mutual information and entropy are still intractable to compute and we next introduce variational approximations for both (Chalk et al., 2016; Alemi et al., 2016). We discuss the treatment of the two terms $I(\bar{Z},\bar{Y})$ and $H(\bar{Z})$ in turn.

Estimating information about the purpose $I_\theta(Z;Y)$

We first note that we can rewrite $I_\theta(\bar{Z};\bar{Y}) = H(\bar{Y}) - H_\theta(\bar{Y}|\bar{Z})$. $H(\bar{Y})$ is independent of our encoder due to the Markov chain $Y \leftarrow H \rightarrow Z$, i.e. $\bar{y}$ is given by $P(\bar{y}|h)P(h)$. We can therefore ignore it and focus on minimizing $H_\theta(\bar{Y}|\bar{Z})$:

$$H_\theta(\bar{Y}|\bar{Z}) = -\int d\bar{y}\ d\bar{z}\ dh\ P_\theta(\bar{y}|\bar{z})P_\theta(\bar{z}|h)P(h)\log P_\theta(\bar{y}|\bar{z})$$

While $P(\bar{y}|\bar{z})$ could in principle be computed using Bayes' theorem, it is intractable in practice due to expectations over $P(h)$. Instead we use a variational approximation $Q_\theta(\bar{y}|\bar{z})$ to $P(\bar{y}|\bar{z})$. Since the Kullback-Leibler divergence $D_{\text{KL}}[P(\bar{y}|\bar{z})\|Q(\bar{y}|\bar{z})] = \int d\bar{y}\ P(\bar{y}|\bar{z})\log P(\bar{y}|\bar{z})/Q(\bar{y}|\bar{z}) \geq 0$, we have:

$$\int d\bar{y}\ P_\theta(\bar{y}|\bar{z})\log P_\theta(\bar{y}|\bar{z}) \geq \int dy\ P_\theta(\bar{y}|\bar{z})\log Q_\theta(\bar{y}|\bar{z})$$

And we get therefore an upper bound on our objective to be minimized:

$$\begin{aligned} H_\theta(\bar{Y}|\bar{Z}) &= -\int d\bar{y}\ d\bar{z}\ dh\ P_\theta(\bar{y}|\bar{z})P_\theta(\bar{z}|h)P(h)\log P_\theta(\bar{y}|\bar{z}) \\ &\leq -\int d\bar{y}\ d\bar{z}\ dh\ P_\theta(\bar{y}|\bar{z})P_\theta(\bar{z}|h)P(h)\log Q_\theta(\bar{y}|\bar{z}) \\ &= -E_\theta[\log Q_\theta(\bar{y}|\bar{z})] \end{aligned} \tag{3.4}$$

The resulting expectation can be estimated using samples from $P(\bar{y}|h)P(\bar{z}|h)P(h)$, where we again used the Markov condition $Y \leftarrow H \rightarrow Z$ to replace $P(\bar{y}|\bar{z})$ by $P(\bar{y}|h)$, thereby avoiding computing any integral. For the parameterization of the variational distribution $Q_\theta(\bar{y}|\bar{z})$ we assume conditional independence of y_t given z_t, thus $Q_\theta(\bar{y}|\bar{z}) =$

$\prod_{t=0}^{T} Q_\theta(y_t|z_t)$. We then parameterize $Q_\theta(y_t|z_t)$ by a generalized linear model (GLM):

$$Q(y|z) = \mathrm{GLM}(Gz + g)$$

Here, G is a matrix parameterizing the linear predictor and g is an offset term. The form of the GLM will depend on the nature of y: if y is discrete, the GLM models a categorical distribution, and if y is continuous, the GLM models a normal distribution. We deliberately keep $Q(y|z)$ linear and time-independent in order to make the interpretation of the representation in terms of an assumed purpose more transparent. Such linear readouts have also proven very useful in analysing activities of neural populations, which make them a natural choice (Kobak et al., 2016). Furthermore, one can argue that the complexity of a representation is also determined by the complexity of the readout it requires. A time-independent linear readout $Q(y|z)$ is arguably of low complexity.[3]

Estimating the entropy $H_\theta(\bar{Z})$ of a representation

As previously, $H_\theta(\bar{Z}) = \int dh\ P_\theta(\bar{z}|h)P(h)$ involves an intractable integral to compute the expectation over histories which requires us to introduce a variational approximation $Q_\theta(\bar{z})$. Again, since the Kullback-Leibler divergence $D_{\mathrm{KL}}[P(\bar{z})\|Q(\bar{z})] = \int d\bar{z}\ P(\bar{z}) \log P(\bar{z})/Q(\bar{z}) \geq 0$, we have:

$$\begin{aligned} H_\theta(\bar{Z}) &\leq H_\theta(P(\bar{z})\|Q(\bar{z})) \\ &= -\int d\bar{z}\ dh\ P_\theta(\bar{z}|h)P(h) \log Q_\theta(\bar{z}) \\ &= -E_\theta\big[\log Q_\theta(\bar{z})\big] \end{aligned} \tag{3.5}$$

Here $H(P(\bar{z})\|Q(\bar{z}))$ denotes the cross-entropy between $P(\bar{z})$ and $Q(\bar{z})$ and it presents an upper bound on $H(\bar{Z})$. The resulting expectation can again be estimated by sampling from $P(\bar{z}|h)P(h)$, i.e. by sampling from our task and encoder. The bound is tight when $P(\bar{z}) = Q(\bar{z})$ and otherwise we have $H(P(\bar{z})\|Q(\bar{z})) = H(P(\bar{z})) + D_{\mathrm{KL}}[P(\bar{z})\|Q(\bar{z})]$.

Variational approximation

While the parametrization of the variational distribution $Q(y|z)$ was straightforward, this is not the case for $Q(\bar{z})$, as we expect $P(\bar{z})$ to be a potentially complicated

[3]It could also be argued that the encoder has to be of low complexity. However, while we think of the readout as the connection weights to a next area, the encoder can be implemented by the whole sensory system, and is thus potentially less constrained.

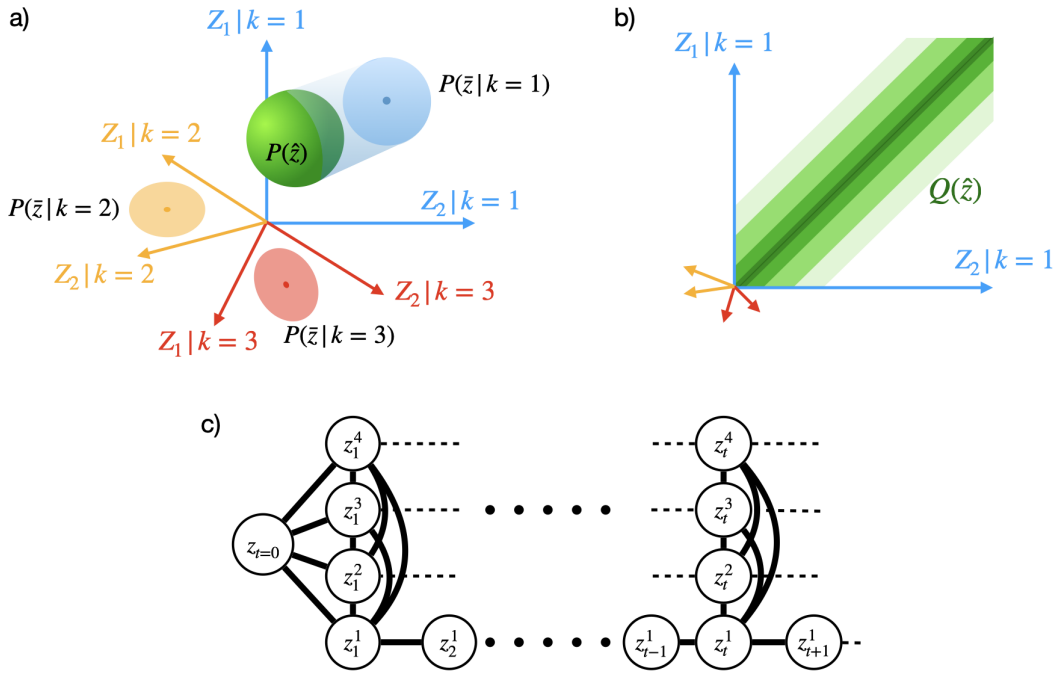


Figure 3.9: Since component relationships are the determining factor of a representation's entropy, our variational distribution should focus on these relationships. In order to do this, instead of directly parameterizing the variational distribution Q in a space $\bar{Z} \times K$, i.e. the space of our mixture model, we parameterize it in the much bigger joint space $\hat{Z} = \bar{Z}^1 \times \bar{Z}^2 \times \cdots \times \bar{Z}^K$. A point $\hat{z} \in \hat{Z}$ will then correspond to K trajectories simultaneously, i.e. $\hat{z} = (\bar{z}^1, \bar{z}^2, \ldots, \bar{z}^K)$. a) We can represent a mixture model with components $P(\bar{z}|k)$ as in e.g. Fig.3.4b) equivalently in $\hat{Z}$: First, every colored component k gets its own set of axes. Second, we define a distribution, drawn as a green ball, $P(\hat{z}) = \prod_{k=1}^{K} P(\bar{z}_k|k)$, where we have made the dependence of $\bar{z}$ on k explicit. We can then simply recover a component's distribution $P(\bar{z}|k)$ as the marginal of $P(\hat{z}) = P(\bar{z}^1, \bar{z}^2, \ldots, \bar{z}^K)$ over all variables but $\bar{z}^k$, visualized as a projection for the blue component. b) The same space as in a). Our variational distribution $Q(\hat{z})$ assigns high probability to the diagonal, representations with highly overlapping components and marginals (as in Fig.3.4d,e)). For visualization reasons, again only 2 dimensions are considered. c) The undirected graphical model used for our variational approximation. Rows correspond to different trial types, columns to different time steps. Connections between rows pull trial types together, while connections between columns prioritize smooth solutions.

multimodal distribution. If we are not able to approximate such a distribution well, i.e. $D_{\mathrm{KL}}[P(\bar{z})\|Q(\bar{z})] \gg 0$, it will be hard to discern between representations and to find the efficient representation. In order to find an effective variational distribution that enforces efficient solutions, we first try to illustrate the kind of $P(\bar{z})$ we are expecting to encounter. For this, we use the structured task description with K trial types that we introduced above:

$$\begin{aligned} P_\theta(\bar{z}) &= \sum_{k=1}^{K} \int dh \; P_\theta(\bar{z}|h)P(h|k)P(k) \\ &= \sum_{k=1}^{K} P_\theta(\bar{z}|k)P(k) \end{aligned}$$

In this formulation, $P(\bar{z})$ takes the form of a mixture distribution with components $P(\bar{z}|k)$, i.e. a component, or representation, for every trial type k (Fig.3.4a,b). The entropy $H(\bar{Z})$ will then primarily be a function of the relationship between the components, but not a function of each component's shape, i.e. each component's mean, itself (Fig.3.4c-g). Specifically, if all components are separated, then the entropy $H(\bar{Z})$ will be at its maximum. Yet if all components are identical, then $H(\bar{Z})$ will be minimal.

To see this, we can set up lower and upper bounds on the true entropy: $H(\bar{Z}|K) \leq H(\bar{Z}) \leq H(K|\bar{Z}) + H(\bar{Z})$. The upper bound is given since $H(K|\bar{Z})$ is a discrete entropy, which is non-negative. The upper bound is attained if $H(K|\bar{Z}) = 0$, i.e. if every sequence $\bar{z}$ is clearly assigned to a component k. This means that components are non-overlapping. The lower bound is given due to the non-negativity of mutual information $I(;)$, $I(\bar{Z}; K) = H(\bar{Z}) - H(\bar{Z}|K) \geq 0$. Thus, $H(\bar{Z}|K)$ is a lower bound that is attained when all components have the same distribution, i.e. $P(\bar{z}) = P(\bar{z}|k)\ \forall k$. Of course, entropy can be further minimized by minimizing the per-component entropy $H(\bar{Z}|k)$, but we assume it to be fixed in this work.

Intuitively speaking, even if the entropy of two mixture distributions is identical due to identical component relationships, their complexity still seems different. Compare for example Fig.3.4 panels f) and g). Their entropy is identical, yet one is varying with time and the other is not. This apparent mismatch is due to the invariance of entropy under the addition of constants $\bar{a}$, i.e. $H(\bar{Z} + \bar{a}) = H(\bar{Z})$. Thus, the entropy $H(\bar{Z})$ of our representation is independent of its mean, and this mean might vary in complicated ways (Fig.3.4d,e). However, the addition of $\bar{a}$ will affect the entropy of the marginal distribution. Assume we are at the optimum where all components are overlapping, as in Fig.3.4d-g. The marginal is formally $P(z) = 1/T \sum_{t=0}^{T} P(z_t)$

with $P(z_t) = 1/K \sum_k \int dz_{-t}\ P(\bar{z}|k)$, where $\int dz_{-t}$ denotes an integral over all $z_{i\neq t}$. If $P(z_1)$ and $P(z_2)$ are overlapping, as in Fig.3.4e, then the entropy of $P(z)$ is minimal. We thus focus on solutions with flat offset $\bar{a}$, where $a_i = a_j\ \forall i, j$. We thus look for representations with overlapping trials and flat or smooth trajectories.

We now use these facts about the entropy of mixture distributions to design a variational distribution that specifically captures these relationships. The variational distribution can then be understood as a prior that assigns high probability to low entropy solutions. In order to do so, we firstly do not work with $P(\bar{z})$ directly, but with the mixture distribution $P(\bar{z}|k)P(k)$, and accordingly we minimize the joint entropy $H(\bar{Z}, K)$. The joint entropy upper bounds the entropy of our interest:

$$H(\bar{Z}, K) = H(\bar{Z}) + H(K|\bar{Z}) \geq H(\bar{Z}) \tag{3.6}$$

The upper bound, as above, is due to the non-negativity of the discrete entropy $H(K|\bar{Z})$. While this is a step away from our original objective, we content ourselves with this upper bound as the joint mixture distribution gives us access to the components of several trial types before they are marginalized out. At the same time, a sample $k, \bar{z}$ from this mixture distribution will only give us a single trajectory from a single trial type, again forbidding us to make any statement about the relationship between the components of several trial types. We thus secondly change the representation of the joint $P(\bar{z}|k)P(k)$ as illustrated and described in Fig.3.9a):

$$P(\hat{z}) = P(\bar{z}^1, \bar{z}^2, \ldots, \bar{z}^K|1, \ldots, K) = \prod_{k=1}^{K} P(\bar{z}^k|k)$$

Here we made the dependence of $\bar{z}^k$ on k explicit using superscipts, and we introduced a new space $\hat{Z} = \bar{Z}^1 \times \bar{Z}^1 \times \cdots \times \bar{Z}^K$. As there is only one combination of indices $1, \ldots, K$, we have omitted the conditioning of $P(\hat{z})$. The space $\hat{Z}$ includes all components jointly, hence a variational distribution in $\hat{Z}$ is able to address their relationships. Note that this change of representation is leaving our components, and thus our mixture, unchanged: As we took the product of all components, and all of them are conditionally independent of each other, we can simply marginalize out the appropriate components to get back a specific $P(\bar{z}^k|k)$:

$$\begin{aligned} P(\bar{z}^k|k) &= \int d\bar{z}^1 \ldots d\bar{z}^{k-1} d\bar{z}^{k+1} \ldots d\bar{z}^K P(\bar{z}^1, \bar{z}^2, \ldots, \bar{z}^K|1, \ldots, K) \\ &= P(\bar{z}^k|1, \ldots, K) \end{aligned}$$

The last equality holds because $\bar{z}^k$ only depends on k. Unsurprisingly, this change of representation does then also not affect the joint entropy of the mixture:

$$\begin{aligned}
H(\hat{Z}) &= \int d\hat{z}\ P(\hat{z}|1,\ldots,K)P(1,\ldots,K)\log P(\hat{z}|1,\ldots,K)P(1,\ldots,K) \\
&= \int d\hat{z}\ P(\hat{z}|1,\ldots,K)P(1,\ldots,K)\log \prod_{k=1}^{K} P(\bar{z}^k|k)P(k) \\
&= \sum_{k=1}^{K}\int d\bar{z}^1\ldots d\bar{z}^K\ P(\bar{z}^1,\bar{z}^2,\ldots,\bar{z}^K|1,\ldots,K)P(1,\ldots,K)\log P(\bar{z}^k|k)P(k) \\
&= \sum_{k=1}^{K}\int d\bar{z}^k\ P(\bar{z}^k|k)P(k)\log P(\bar{z}^k|k)P(k) \\
&= H(\bar{Z},K)
\end{aligned}$$

Here we have used the joint of all indices $P(1,\ldots,K)$ which has one entry only. We can therefore define our variational distribution in $\hat{Z}$, and we define it such that representations with slowly-varying overlapping components are preferred (Fig.3.4c-g, Fig.3.9b). Specifically, the variational distribution factorizes according to the following undirected graphical model (Fig.3.9c):

$$Q_\theta(\hat{z}) = Q_\theta(\bar{z}^1,\bar{z}^2,\ldots,\bar{z}^K|1,\ldots,K) = \frac{1}{c}\prod_{k=1}^{K}\phi_\theta(z_0^k)\prod_{t=1}^{T}\phi(z_t^k,z_{t-1}^k)\prod_{j>k}\phi(z_t^k,z_t^j)$$

Here $\phi(z_0^k) = \phi(z_0) = N(\mu_0,\sigma^2 I)\ \forall k$, $\phi(x,y) = \exp(-\frac{\|y-x\|^2}{\sigma^2})$, and c is the partition sum taking care of the normalization. The mean μ_0 of $\phi(z_0)$ is the only variational parameter. Crucially, factors $\phi(x,y)$ enforce closeness between x and y. On the one hand, factors of the type $\phi(z_t^k,z_{t-1}^k)$ enforce closeness between two consecutive states of a given trial, thus enforcing low marginal entropy (Fig.3.4d-g). Factors of the type $\phi(z_t^k,z_t^j)$, on the other hand, enforce closeness between the representations of two different trial types k and j, thus enforcing low entropy solutions (Fig.3.4). Since such pairwise relations are undirected, the third product only runs over $j > k$ in order to

avoid counting some pairs twice. Using equations 3.5 and 3.6, we then get:

$$
\begin{aligned}
H_\theta(\bar{Z}) &\leq H_\theta(\bar{Z}, K) = H_\theta(\hat{Z}) \\
&\leq -E_\theta[\log Q_\theta(\hat{z})] = -E_\theta[\log (\frac{1}{c}\prod_{k=1}^{K}\phi_\theta(z_0^k)\prod_{t=1}^{T}\phi(z_t^k, z_{t-1}^k)\prod_{j>k}\phi(z_t^k, z_t^j))] \\
&= -E_\theta[\sum_{k=1}^{K}\log\phi_\theta(z_0^k) + \sum_{t=1}^{T}\log\phi(z_t^k, z_{t-1}^k) + \sum_{j>k}\log\phi(z_t^k, z_t^j)] + c' \\
&= \frac{1}{\sigma^2}E_\theta[\sum_{k=1}^{K}\|z_0^k - \mu_0\|^2 + \sum_{t=1}^{T}\|z_t^k - z_{t-1}^k)\|^2 + \sum_{j=k+1}^{K}\|z_t^k - z_t^j\|^2] + c'
\end{aligned}
\tag{3.7}
$$

We have absorbed all constant terms into c'. The expectation can then again be estimated by sampling from $P(\bar{z}|h)P(h|k)P(k)$. We can now see that a sample $Q(\hat{z})$ will have high probability if distances between 'neighbouring' states are low. The lowest entropy solution will thus be a representation that does not change over time nor trial type. This is the same intuition underlying the concept of Fisher information; if a random variable X is invariant to a change about y, then X has low Fisher information about y. The view that our variational distribution is rather a prior than an approximation relates to the idea of default dynamics in control theory, where a deviation from the default dynamics incurs control costs (see e.g. Todorov (2009); Tishby and Polani (2011)).

Finally combining the two bounds in eq.3.4 and eq.3.7 into eq.3.3, we get:

$$
\begin{aligned}
L &\geq \hat{L} \\
&= E_\theta[\log Q_\theta(\bar{y}|\bar{z})] + \lambda E_\theta[\log Q_\theta(\hat{z})] \\
&= E_\theta[\log Q_\theta(\bar{y}|\bar{z})] - \frac{\lambda}{\sigma^2}E_\theta[\sum_{k=1}^{K}\|z_0^k - \mu_0\|^2 + \sum_{t=1}^{T}\|z_t^k - z_{t-1}^k\|^2 + \sum_{j=k+1}^{K}\|z_t^k - z_t^j\|^2]
\end{aligned}
\tag{3.8}
$$

In the last line we dropped the constant c'. The resulting objective is thus simple maximum likelihood estimation under a smoothing constraint. In practice, minimizing the terms of the form $\|z - z'\|^2 = \|z\|^2 + \|z'\|^2 - 2z^\top z'$ will simultaneously minimize the norms of z and z', as well as maximize the overlap between z and z' as per the dot product. This will often lead to the trivial solution of $z = 0\ \forall z$, making it hard for the optimization procedure to pick up purpose information. We thus normalize the terms as in $\|z - z'\|^2/(\|z\|^2 + \|z'\|^2)$ and simultaneously minimize the norms $\|z\|^2$ for z at all

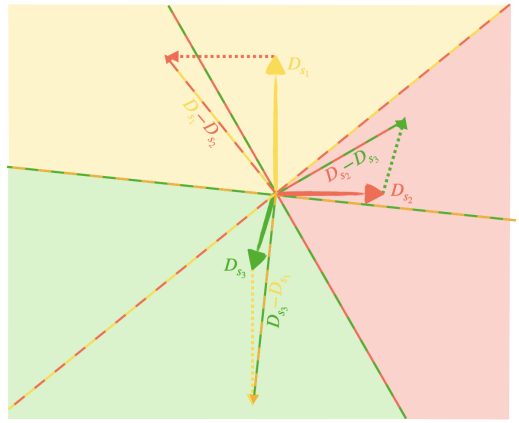


Figure 3.10: The partitioning of Z by the roSLDS. Consider for example the switching variable s_1. For any state z, s_1 is expected to be active if both $D_{s_1}z > D_{s_2}z$ and $D_{s_1}z > D_{s_3}z$ (the yellow area). The center of the partitioning is determined by the offset e, and can be translated by the current observation and action through matrices F and E, respectively.

time steps and for all trial types, to avoid increasing the norm of the representation indefinitely.

The resulting objective is optimized with respect to the parameters θ that are the parameters of our two variational approximations $Q(y|z)$ and $Q(\hat{z})$, as well as the parameters of our encoder $P(z|h)$. We next discuss the parameterization of the encoder.

3.6.5 Parameterization of encoder $P(\bar{z}|h)$ by a switching linear dynamical system

We can now turn to the parameterizaton of our encoder. Due to the memory constraint we have the following factorization:

$$P(\bar{z}_t|h_t) = P(z_0)\prod_{i=1}^{t} P(z_t|z_{t-1}, a_{t-1}, o_{t+1})$$

We thus need to parameterize the initial condition $P(z_0)$ as well as the transition distribution $P(z'|z, a, o')$. The initial condition $P(z_0) = N(\mu_0, \sigma^2 I)$ is parameterized by a Gaussian of fixed variance $\sigma^2 I$ and mean μ_0 that is shared with $\phi(z_0)$ of the variational approximation. For the transition distribution, we add, as discussed above, to enforce a lower bound on the differential entropy, noise η to an otherwise deterministic transition function $f(z, a, o')$. For simplicity, we assume centered Gaussian noise $\eta \sim P(\eta) = N(0, \sigma^2 I)$ with isotropic variance $\sigma^2 I$, where I denotes the identity

matrix. We thus have:

$$P(z'|z,a,o') = N(f(z,a,o'),\sigma^2 I)$$

For f we make a choice based on the trade-off between two considerations: The resulting encoder shall be easy to understand, yet it shall be flexible enough so that it is likely that the efficient representation can actually be encoded. If, for example, f is a linear function, then we have good interpretability, but the resulting representation will likely be heavily influenced by the rigidity of the linear mapping, and hence it is unlikely that we can encode the efficient representation. If, on the other hand, f is a very flexible function, such as a deep neural network, then encoding will be good, but interpretation less straightforward. A choice that nicely balances these two considerations is the switching linear dynamical system (SLDS, Barber (2006); Linderman et al. (2017)). An SLDS uses linear dynamics, but the linear dynamics can change based on where in the space Z we are. Specifically, we have have a set of S different linear dynamical systems indexed by the switching variable s:

$$\begin{aligned} P(z'|z,a,o',s') &= N(f_{s'}(z,a,o'),\sigma^2 I) \\ f_{s'}(z,a,o') &= A_{s'}z + B_{s'}a + C_{s'}o' + b_{s'} \end{aligned}$$

Here, $A_{s'} \in R^{N\times N}$ is the transition matrix of switch s' that maps a current state $z \in R^N$ one step forward. $B_{s'} \in R^{N\times L}$ and $C_{s'} \in R^{N\times M}$ are the input weights of switch s' for actions $a \in R^L$ and observations $o \in R^M$, respectively, and $b_{s'} \in R^N$ is a constant offset. The switching variable s' is determined by the switching distribution which is also a function of z, a, and o':

$$P(s'|z,a,o') = \text{Cat}(\text{softmax}(Dz + Ea + Fo' + e))$$

This corresponds to another GLM with the linear predictor consisting of the matrices $D \in R^{S\times N}$, $E \in R^{S\times L}$, and $F \in R^{S\times M}$, as well as the constant offset vector $e \in R^S$. The switching distribution partitions our state space Z into S linearly separable regions. Given a state z, switch s will, on average, be active if $(D_s - D_{s'})z > 0 \; \forall s' \neq s$, where D_s denotes the s'th row of D (Fig.3.10). Apart from this, the partition may be translated based on the input from actions and observations through weights E and F, respectively.

Note that this is a special version of the SLDS, sometimes called the recurrent-only SLDS (roSLDS, Linderman et al. (2017)). In a standard SLDS, s' does not depend on the current state z through a recurrent connection D, making it less flexible.

At the same time, we drop the standard dependence of s' on the previous switch s, therefore being recurrent-only. While dropping this dependence does render the roSLDS simpler, the reason for dropping it is another: If we allow this dependence, information can flow through a sequence of switches s, s', s' outside of the space Z, thereby making the representation in Z alone potentially insufficient. This would defeat the purpose of our approach.

Finally, we can recover our transition distribution:

$$P(z'|z,a,o') = \sum_{s'=1}^{S} P(z'|z,a,o',s')P(s'|z,a,o')$$

3.6.6 Optimization

We are now ready to maximize our lower bounded objective $\hat{L}_{IB}$ of eq.3.8 with respect to $\theta = (A_1, \ldots, A_S, B_1, \ldots, B_S, C_1, \ldots, C_S, b_1, \ldots, b_S, D, E, F, e, G, g, \mu_0)$, our parameters. All parameters indexed by s parameterize the dynamics of the SLDS, (D, E, F, e) parameterize the switching distribution, G, g parameterize the variational distribution for the purpose, and μ_0 parametrizes the variational distribution for the representation entropy. To present our optimization strategy, we first focus on the purpose-related term of $\hat{L}$:

$$E_\theta\big[\log Q_\theta(\bar{y}|\bar{z})\big] = E\big[\int d\bar{z}\ P_\theta(\bar{z}|h)\log Q_\theta(\bar{y}|\bar{z})\big]$$

As mentioned before, the expectation value can be approximated by sample averages, yet the sampling distribution also depends on θ through the encoder, making it difficult to optimize for the expected log-likelihood. One straightforward way of avoiding this complication is to reparameterize the encoder Kingma and Welling (2014). Instead of sampling from $P_\theta(\bar{z}|h)$ directly, we can sample from a θ-independent distribution $\bar{\epsilon} \sim P(\bar{\epsilon})$ and then plug this sample into an appropriate θ-dependent, differentiable, and deterministic function $\bar{z} = g_\theta(h, \epsilon)$, such that the resulting samples $\bar{z}$ are distributed according to $P_\theta(\bar{z}|h)$. To highlight the dependency of these samples on θ and the histories h, we write $\bar{z}(\theta, h)$. Since $P_\theta(\bar{z}|h)d\bar{z} = P(\bar{\epsilon})d\epsilon$, as required to define a valid density, the expectation above becomes θ-independent:

$$E\big[\int d\bar{z}\ P_\theta(\bar{z}|h)\log Q_\theta(\bar{y}|\bar{z})\big] = E\big[\int d\bar{\epsilon}\ P(\bar{\epsilon})\log Q_\theta(\bar{y}|\bar{z}(\theta,h))\big]$$

In our case, the roSLDS parameterizing the encoder consists of a Gaussian transition distribution and a categorical switching distribution, and both have to be reparameterized. For the Gaussian $P(z'|z,a,o',s') = N(f^\theta_{s'}(z,a,o'),\sigma^2 I)$ as defined above, we can naturally sample the noise from a zero-mean Gaussian $\eta \sim N(0,\sigma^2 I)$ and then add this sample to the determinisitic mean function to get $g_\theta(z,a,o',s',\eta) = f^\theta_{s'}(z,a,o') + \eta$. While reparameterizing a Categorical distribution is less straightforward, it can also be, approximately, done using the so called Gumbel-softmax trick (Maddison et al., 2016; Jang et al., 2016).

To finally calculate the expectation, we can take M samples h^m from our task, the corresponding samples $\bar{y}^m$ from our purpose, as well as a sample $\bar{z}^m$ from our reparameterized encoder:

$$\begin{aligned} E_\theta\big[\log Q_\theta(\bar{y}|\bar{z})\big] &\approx \frac{1}{M}\sum_{m=1}^{M} \log Q_\theta(\bar{y}^m|\bar{z}^m(\theta,h^m)) \\ &= \frac{1}{M}\sum_{m=1}^{M}\sum_{t=0}^{T} \log Q_\theta(y_t^m|z_t^m(\theta,h^m)) \end{aligned}$$

We can then compute the gradient of this sample average with respect to the parameters θ, in order to maximize our objective by gradient ascent:

$$\frac{d}{d\theta}\frac{1}{M}\sum_{m=1}^{M}\sum_{t=0}^{T} \log Q_\theta(y_t^m|z_t^m(\theta,h^m)) = \frac{1}{M}\sum_{m=1}^{M}\sum_{t=0}^{T} \frac{d}{d\theta}\log Q_\theta(y_t^m|z_t^m(\theta,h^m))$$

The term $\log Q_\theta(y_t^m|z_t^m(\theta)$ will tell us how well our model is currently able to explain y_t^m. The corresponding error can then be minimized in two ways: Firstly, by changing the parameters G of the variational distribution, and secondly, by changing z_t^m, i.e. the encoder, itself. How changing the encoder will affect the error is determined by $dz_t^m(\theta,h^m)/d\theta$, and since $z_t^m(\theta,h^m)$ is a deterministic function of z_{t-1}^m and of s_t^m, and z_{t-1}^m, s_t^m are deterministic functions of previous variables likewise, the derivative has to be computed all the way back through the SLDS. In order to compute those gradients, we use the auto-gradient function of the pytorch library (Paszke et al., 2019). Gradient steps are taken using the Adam optimizer (Kingma and Ba, 2015).

The second term of the objective $\hat{L}$ can be optimized equivalently.

3.6.7 Task modeling

In general, we tried to stick as closely as possible to the definitions of the tasks as found in the respective publications (Romo et al., 1999; Brody et al., 2003; Mante

et al., 2013; Inagaki et al., 2019). Special care was taken to respect the time structure of the tasks, as we knew that minor timing changes can have a big impact on the resulting neural representations (Inagaki et al., 2019).

All parameters θ are initialized by sampling from a normal distribution with a standard deviation of 0.1. The hyperparameters of the model are the SLDS's state dimension N, its transition noise σ, the number of switching variables S, and the standard deviation of the noise σ_{obs} added to the observations.

For the habitual models, we set $B_s = 0$ for all switches, i.e the actions produced by the model could not directly influence the SLDS's future dynamics. We did so in order to simplify optimization, as we did not expect such a dependence to be necessary to implement the optimal policy for any of the tasks.

The variational distribution $Q(y|z)$ for the purpose are different for the habitual and the model-based cases. In the habitual case, $Q(y|z) = Q(a|z)$ is a categorical GLM with parameters $G \in R^{L\times N}$ and $g \in R^L$, with L being the number of actions. In the model-based case, $Q(y|z) = Q(o'|z)$ is a normal GLM, or simply a linear model, with parameters $G \in R^{M\times N}$ and $g \in R^M$, with M being the dimensionality of the observations.

Somatosensory working memory task

For the habitual model, we chose the hyperparameters $N = 20$, $\sigma = 0.05$, and $S = 3$. The habitual purpose is to produce the correct action sequences given a sequence of observations. The number of actions L was set to 3, for the actions 'hold', 'push left button', and 'push right button'. The observation dimensionality M was set to 2, where both dimensions receive frequency information F1 and F2, but inversely tuned. This inverse tuning is inspired by the actual frequency tuning of the secondary somatosensory cortex $S2$ in this task (Salinas et al., 2000; Romo et al., 2002; Machens et al., 2005), an earlier stage of somatosensory processing. Specifically, for a given frequency $f_{\min} \geq f \geq f_{\max}$, the positively scaled observation is equal to $0.4 + f - f_{\min}$, and the negatively scaled observation is equal to $0.4 + f_{\max} - f$, following the implementation of (Song et al., 2016).

A trial consisted of $T = 25$ time steps, and F1 was presented at time steps 1, 2, and 3, and F2 was presented at time steps 19, 20, and 21. The (F1,F2) pairs presented were (2,1), (2,3), (4,3), (4,5), (6,5), (6,7), (8,7), and (8,9), all positively and negatively scaled as described above, with additional normal observation noise of standard deviation $\sigma_{\text{obs}} = 0.1$. The optimal policy was to hold until the first F2

presentation, then to produce the correct action during the F2 presentation period, after which any action was equally probable.

For the model-based model, we chose the hyperparameters $N = 30$, $\sigma = 0.05$, and $S = 4$. The model-based purpose is to predict the observations (that include rewards) given a sequence of actions. The number of actions $L = 3$ was unchanged. The dimensionality of the observations was set to $M = 3$, where the first dimension carried the positively tuned frequencies, and the second dimension the negatively tuned frequencies, identical to the habitual case above. The third dimension carried the rewards or punishments.

A trial again consisted of $T = 25$ time steps, and F1 and F2 presentations were identical to the habitual case. In the model-based case actions are not an output of the SLDS model, but rather provided as an input. We provide both correct and incorrect action sequences, in order for the model to learn about the reward contingencies of the task. As the number of all possible action sequences grows exponentially with the number of time steps T, we only presented action sequences that were taking the correct action 'hold' up until the onset of F2 presentation, after which either the correct or the incorrect 'decision' actions 'left' or 'right' were presented. Resulting rewards (10) or punishments (-10) were to be predicted during the time of decision.

Context-dependent decision making task

In this task, a trial consisted of $T = 25$ time steps. In the habitual case, observation dimensionality was set to $M = 4$. The first two dimensions carried the context cue, with $[1, 0]$ for the motion context and $[0, 1]$ for the colour context. The context cue was presented from trial start up until the start of the decision period. The stimuli were presented during time steps 2 to 14, in dimension 3 for motion and 4 for color. Each stimulus was presented as Gaussian noise $N(d, \sigma_{\text{obs}})$, where d corresponds to the coherence of either motion or color, and σ_{obs} to the stimulus noise. The coherence d was selected from 6 possible values, 3 negative, and 3 positive. The magnitude of d was chosen such that the signal to noise ratio for the most difficult conditions, i.e. the ones closest to 0, was 2, for the medium conditions was 4, and for the easiest conditions was 6. For example for the most difficult positive condition, as the stimulus presentation consisted of 12 time steps, we have $d = 2\sigma_{\text{obs}}/\sqrt{12} \approx 0.0577$. Each relevant coherence was paired with a randomly chosen irrelevant coherence. The stimulus presentation was followed by a delay period of a random length, ranging from 3 to 7 time steps. The end of the delay period and the start of the decision period was

signaled by removing the context cue, after which the decision had to be made for two time steps.

The number of actions was again $L = 3$, for 'hold', 'left saccade', and 'right saccade'. The optimal policy was to hold from trial start until the decision period, after which the correct saccade action was to be taken depending on stimulus and context. After the decision period all actions were equally probable. For the habitual model we chose the hyperparameters $N = 20$, $\sigma = 0.01$, and $S = 3$.

For the model-based case, the trial structure was identical, except for an additional observation dimension for the rewards, thus $M = 5$. The rewards were delivered at decision time. As in the somatosensory working memory task, we presented both correct and incorrect action sequences. For correct trials the reward was 10, and for incorrect trials the punishment was -10. We increased the signal to noise ratio of the motion and color stimuli by five times, compared to the habitual case, as otherwise the model-based model had trouble picking up this information. We chose the hyperparameters $N = 20$, $\sigma = 0.01$, and $S = 4$.

To generate the dPCA plots of both agents, we had to generate error trials in order to demix motion and color information from decision information. For example, in the habitual case, since we represent for the optimal policy, we only had very few error trials. Similarly, in the model-based case, we only had error trials because we provided incorrect action sequences from the exterior. To generate error trials, we increased σ_{obs} by four times in the habitual case, and by twenty times in the model-based case.

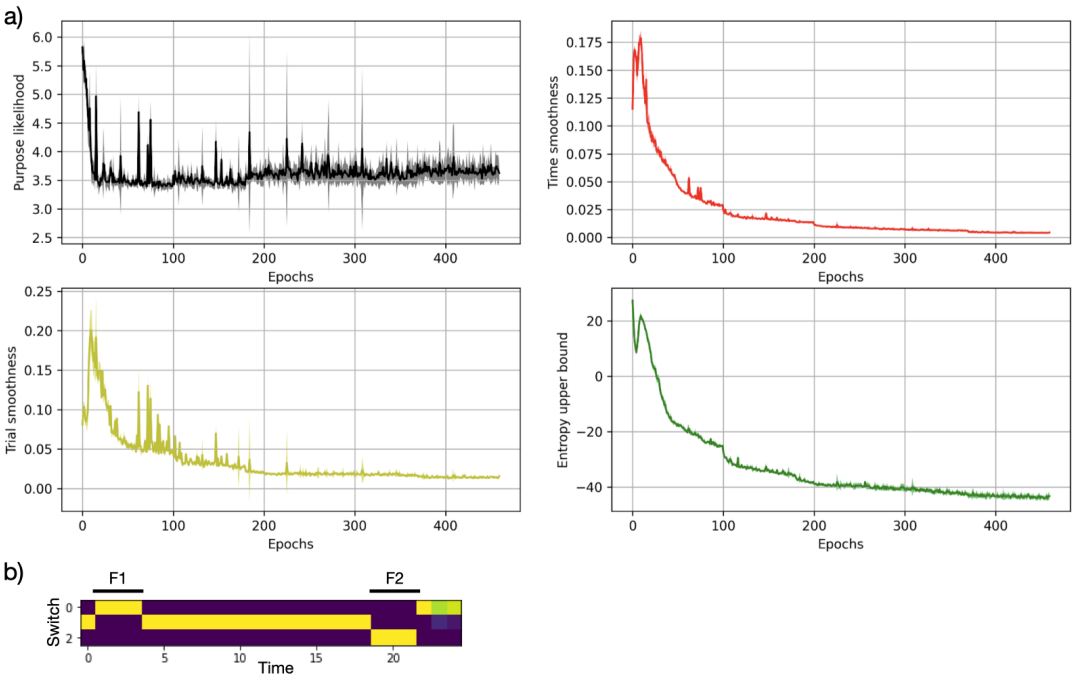
Delayed licking tasks

Both the fixed and the random delay tasks consisted of $T = 23$ time steps. In the habitual case, the observation dimensionality was $M = 1$, a single dimension for the instruction stimulus, presented between time steps 2 and 6, as well as the go-cue stimulus. In the fixed delay task, the go-cue was presented at time step 17. In the random delay task, the time of the go-cue did not occur before time step 8, and otherwise selected from an exponential distribution with rate 0.2. The values of the two instruction tones were 3 and 12, and the value of the go cue was 6. There were $L = 3$ actions, 'hold', 'lick left', and 'lick right'. The optimal policy was to hold, and then to respond after the go cue. For the habitual model, we chose the hyperparameters $N = 20$, $\sigma = 0.05$, and $S = 3$ in both the fixed delay task and the random delay task.

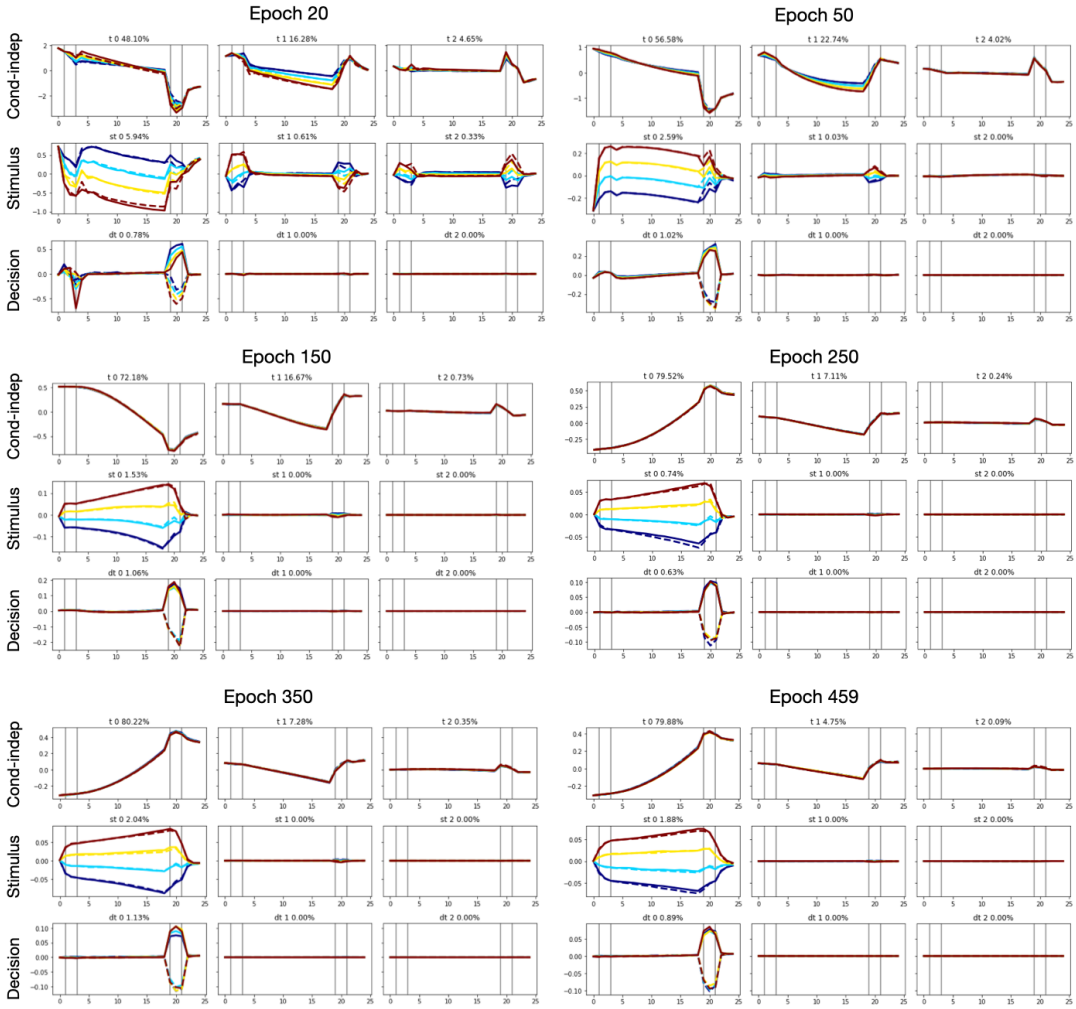
In the model-based case, the observation dimension was increased by one, i.e. $M = 2$, with the reward (10) and punishment (-5) being delivered in the second dimension during the response period. Again, we presented correct as well as incorrect action sequences. We chose the hyperparameters $N = 20$, $\sigma = 0.05$, and $S = 4$ for both tasks.

Lastly, the habitual purpose with anticipation is identical to the habitual purpose, only the go-cue is omitted in the fixed delay task.

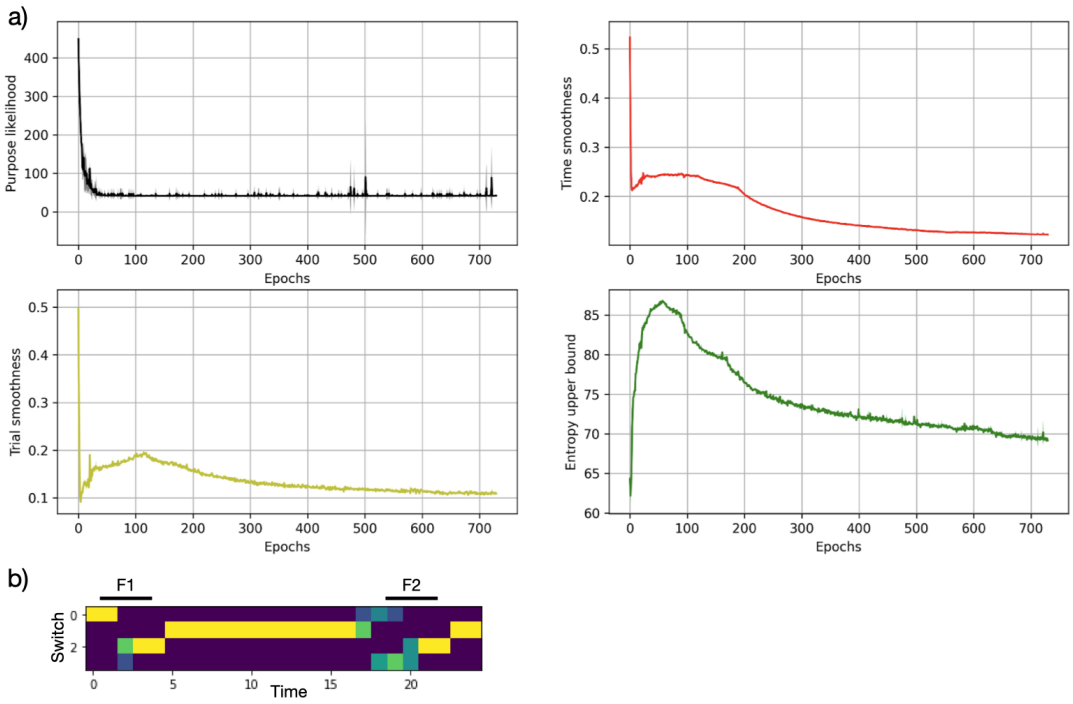
3.7 Supplementary figures



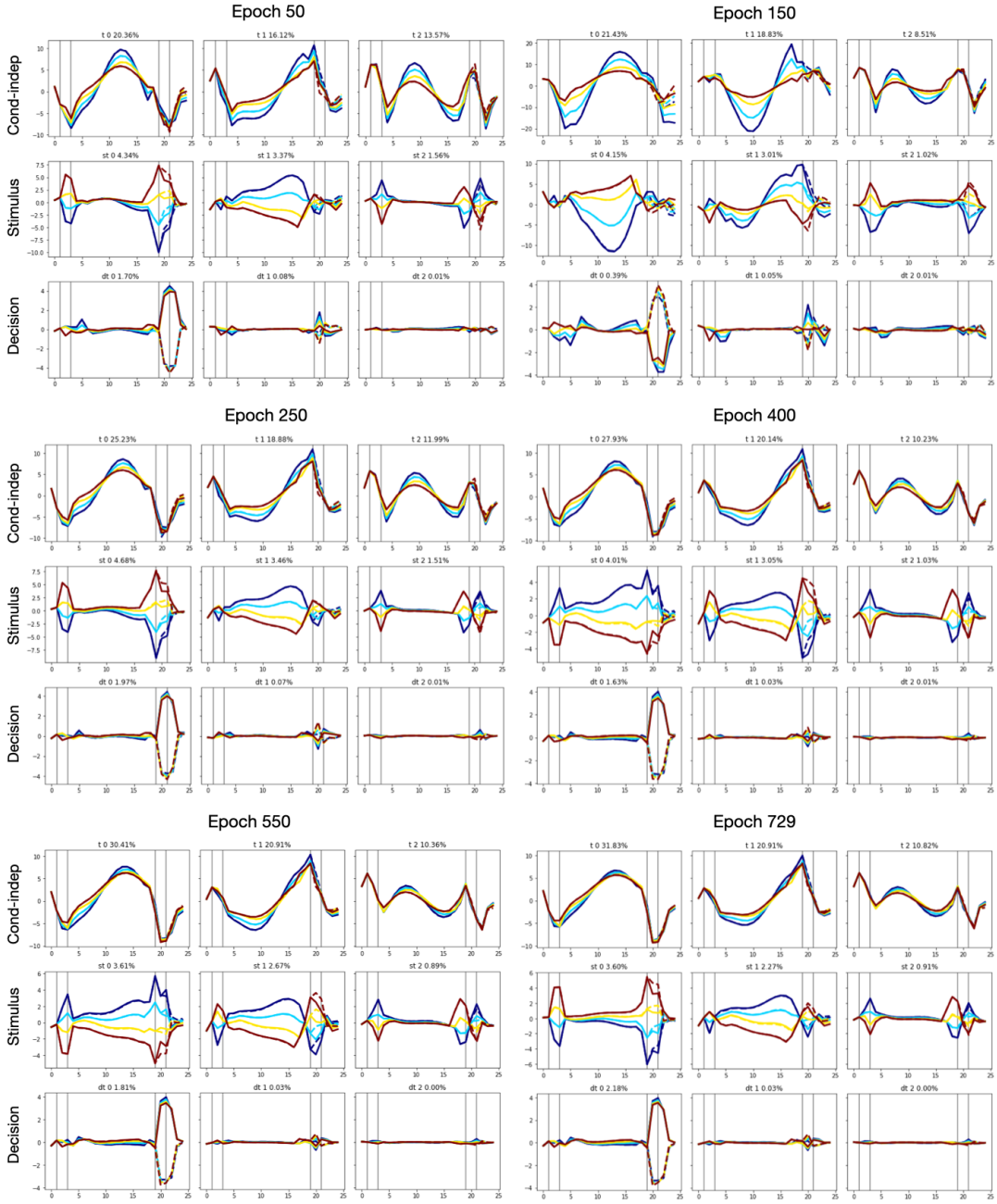
Supplementary Figure 3.1: Optimization progress and switching behaviour of the **efficient habitual representation of the somatosensory working memory task**. a) The four panels show four different measures as they develop over the optimization epochs. The likelihood of the purpose (top left) corresponds here to the likelihood of the optimal actions. As discussed in the Materials and Methods, our choice of the variational approximation $Q(\bar{z})$ of the distribution over representation trajectories $P(\bar{z})$ enforces two types of smoothness, namely smoothness over time (top right) and smoothness over trial types (bottom left). Specifically, given z_i and z_j, where i, j correspond to two adjacent time steps in the same trial type (top right) or to two trial types at the same time step (bottom left), the two panels show $1 - \cos(z_i, z_j)$ averaged over time steps and trial types, and time steps and pairs of trial types, respectively. We choose the cosine-based measure to get a smoothness measure independent of the norm of the representation of a given trial type. The bottom right plot shows an upper bound on the entropy of the marginal representation $P(z) = 1/T \sum_{t=0}^{T} P(z_t)$. Specifically, the vertical axis shows the sum of the log-eigenvalues of the covariance matrix in Z-space, which is proportional to the differential entropy of a Gaussian variational approximation. b) The switching variables selected over a trial. In this case, we have $S = 3$ switching variables (on the vertical axis), and a trial consists of 25 time steps. The plot shows an average over all possible trial types. The times of F1 and F2 presentation are indicated by black bars at the top of the plot. Different switching variables are active during the presentation of F1 and F2. As the same switching variable is active before the presentation of both F1 and F2, this difference can only result from a different continuous state Z. This might explain why we see the ramping in the first condition independent components in Fig.3.5b and supplentary Fig.3.2.



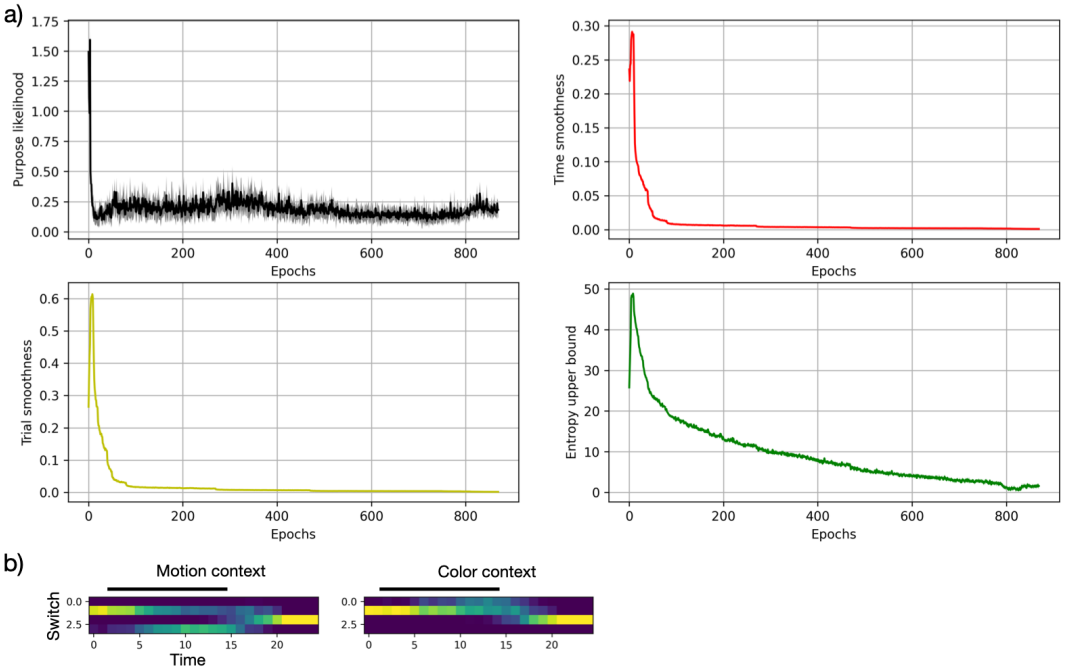
Supplementary Figure 3.2: Development of the **efficient habitual representation of the somatosensory working memory task** over the course of optimization. Each panel shows a dPCA analysis as in Fig.3.5b, but for an intermediate result at a given optimization epoch. The epoch number is indicated at the top of each panel, and the corresponding optimization progress can be found in supplementary Fig.3.1.



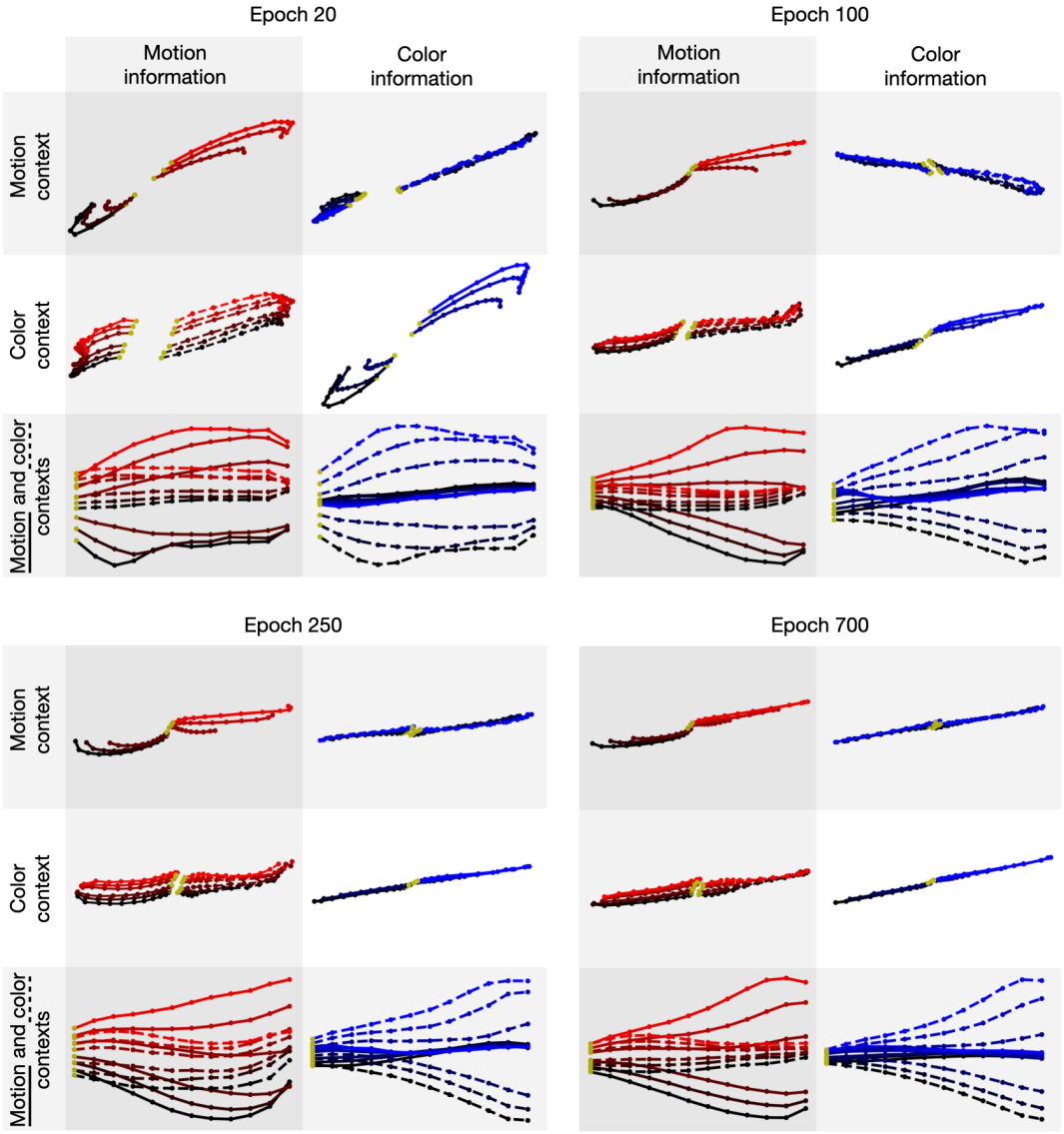
Supplementary Figure 3.3: Optimization progress and switching behaviour of the **efficient model-based representation of the somatosensory working memory task**. a,b) are equivalent to supplementary Fig.3.1.



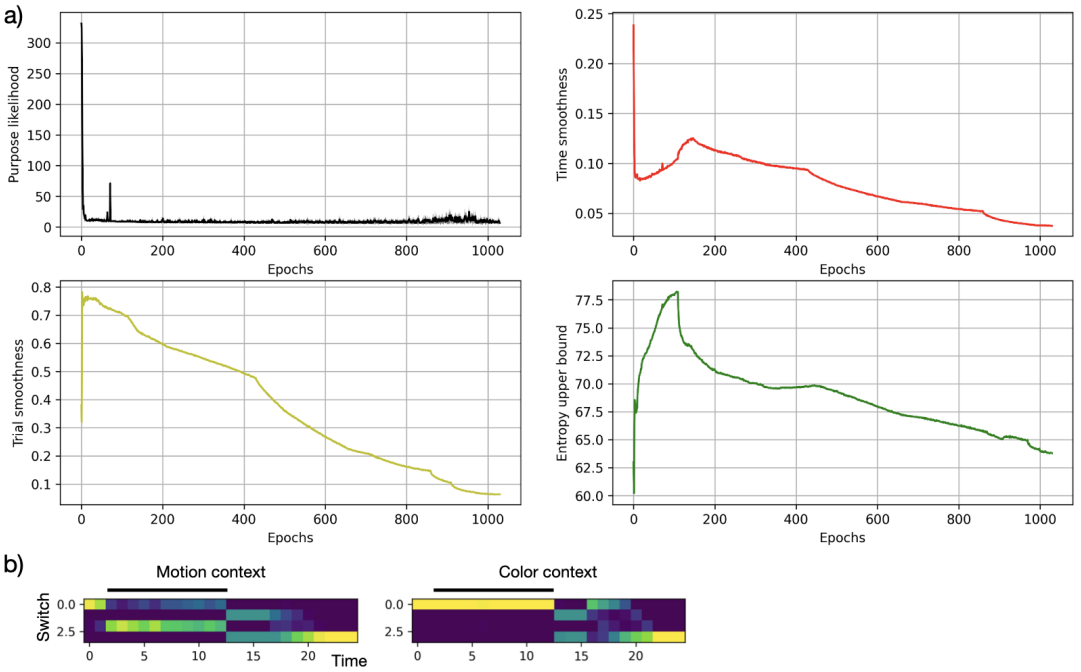
Supplementary Figure 3.4: Development of the **efficient model-based representation of the somatosensory working memory task** over the course of optimization. Panels are equivalent to supplementary Fig.3.2.



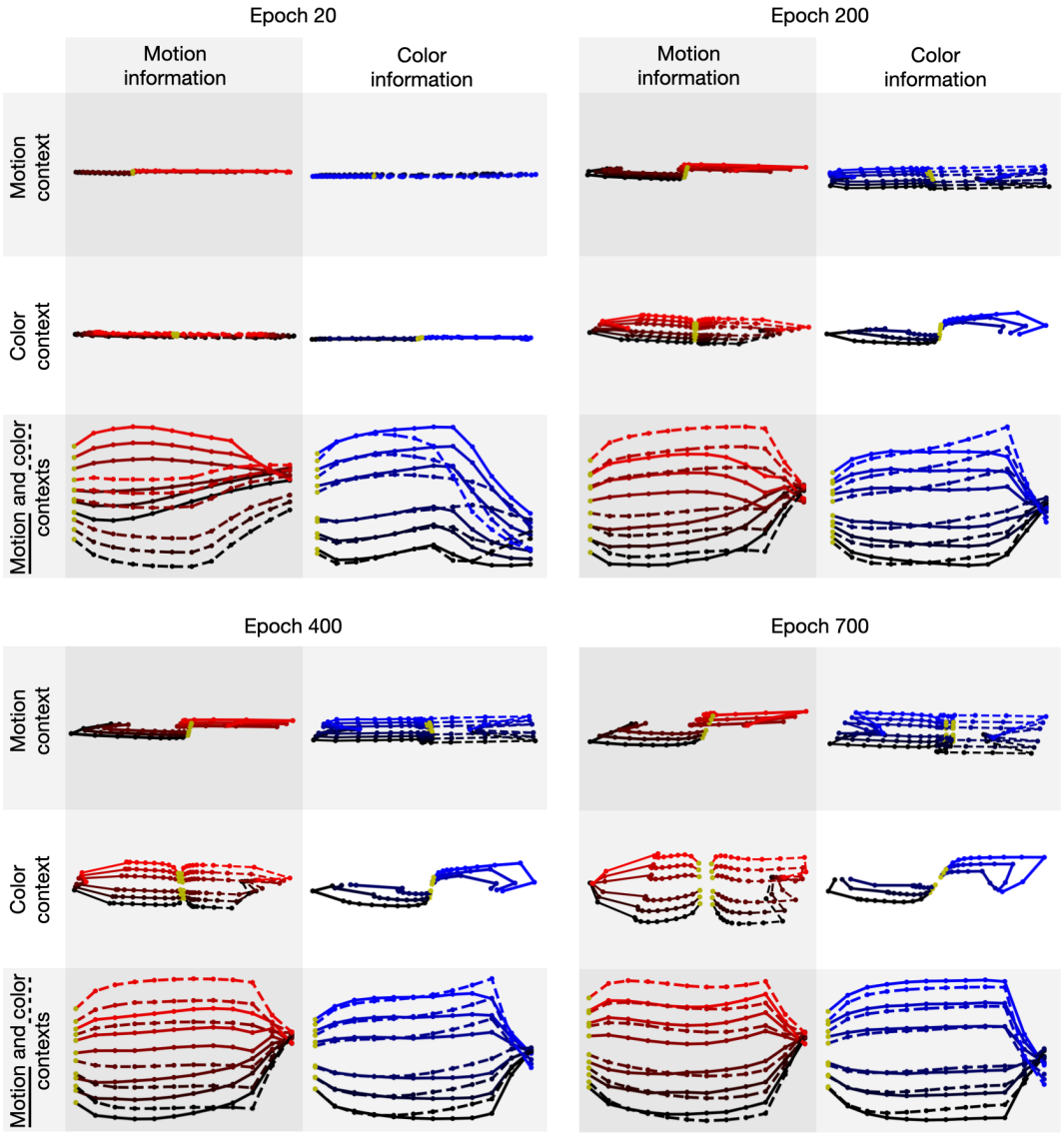
Supplementary Figure 3.5: Optimization progress and switching behaviour of the **efficient habitual representation of the context-dependent perceptual decision making task**. a,b) are equivalent to supplementary Fig.3.1. In b), the black bars on top of each panel denote the random dots period. The left and right panels correspond to the switching behaviour in the motion context and the color context, respectively, averaged over all trial types.



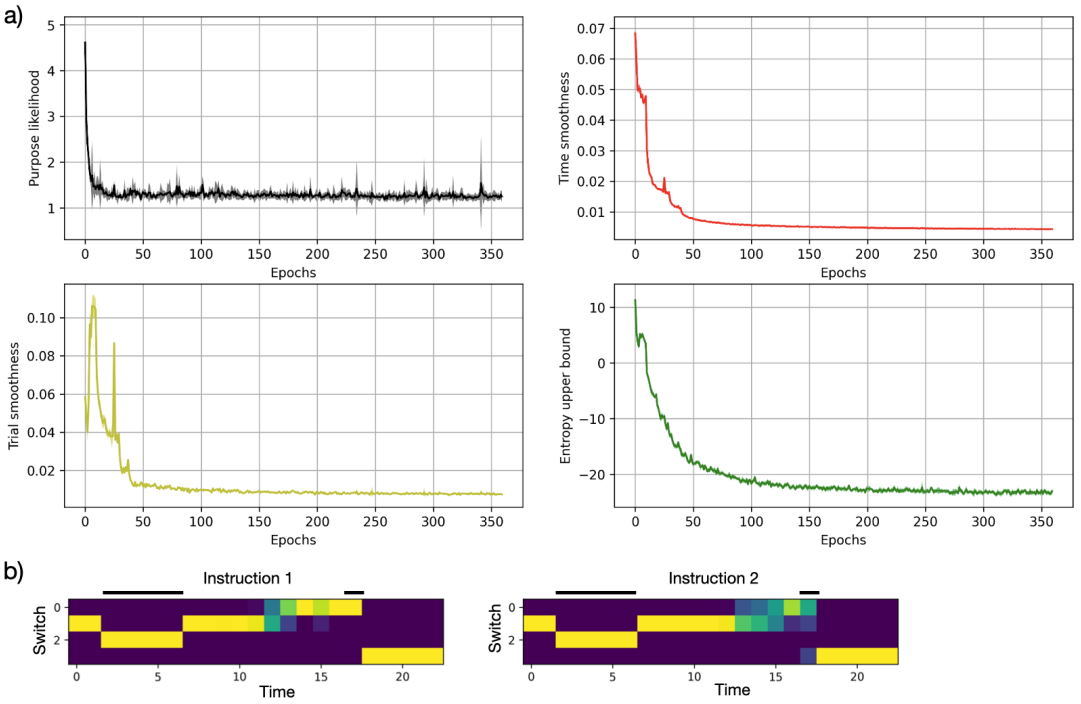
Supplementary Figure 3.6: Development of the **efficient habitual representation of the context-dependent perceptual decision making task** over the course of optimization. Each panels shows the dPCA of an intermediate result during optimization. The optimization epoch is indicated on top of each panel. Panels are organized as described in Fig.3.6b.



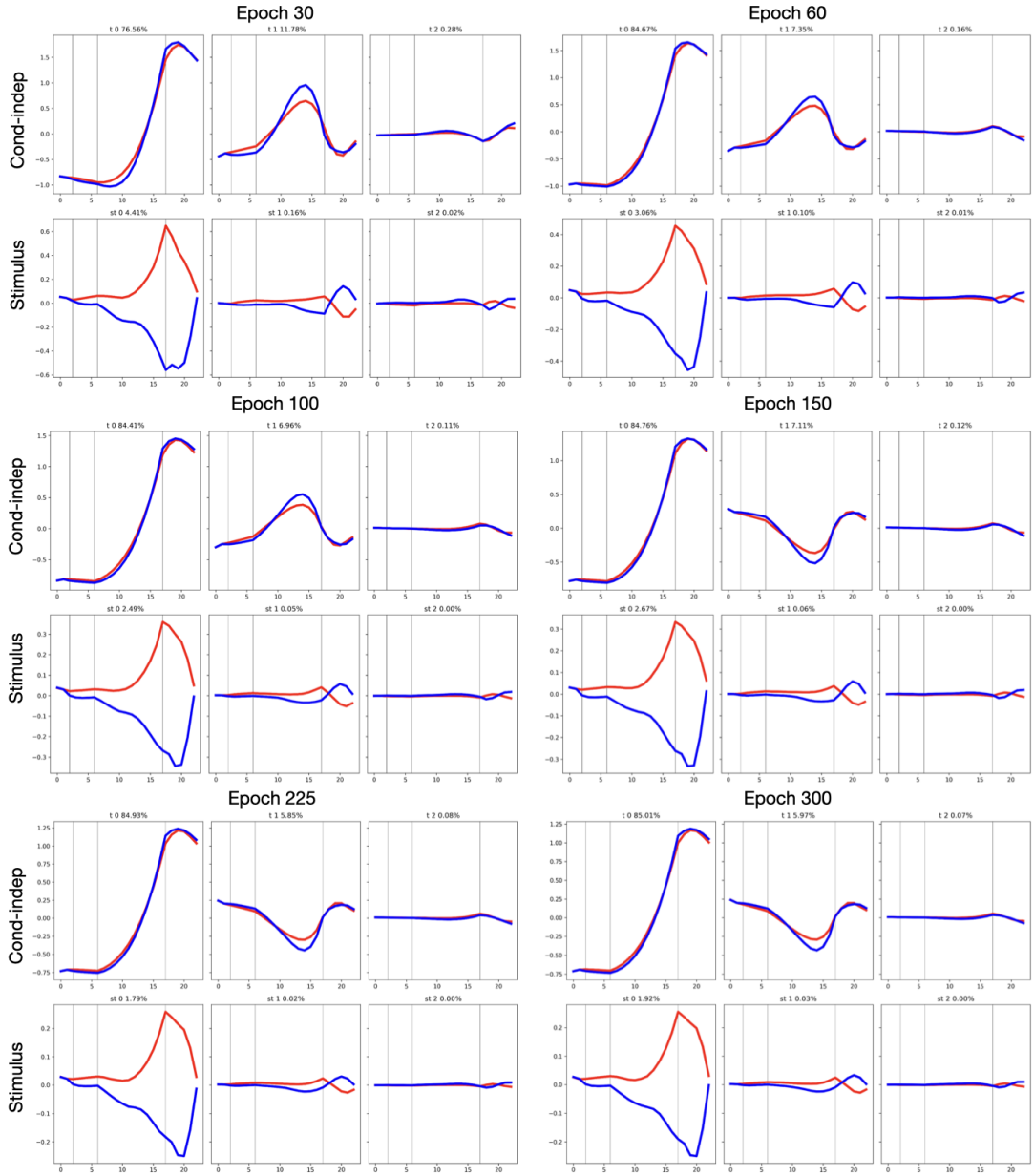
Supplementary Figure 3.7: Optimization progress and switching behaviour of the **efficient model-based representation of the context-dependent perceptual decision making task**. a,b) are equivalent to supplementary Fig.3.1. In b), the black bars on top of each panel denote the random dots period. The left and right panels correspond to the switching behaviour in the motion context and the color context, respectively, averaged over all trial types.



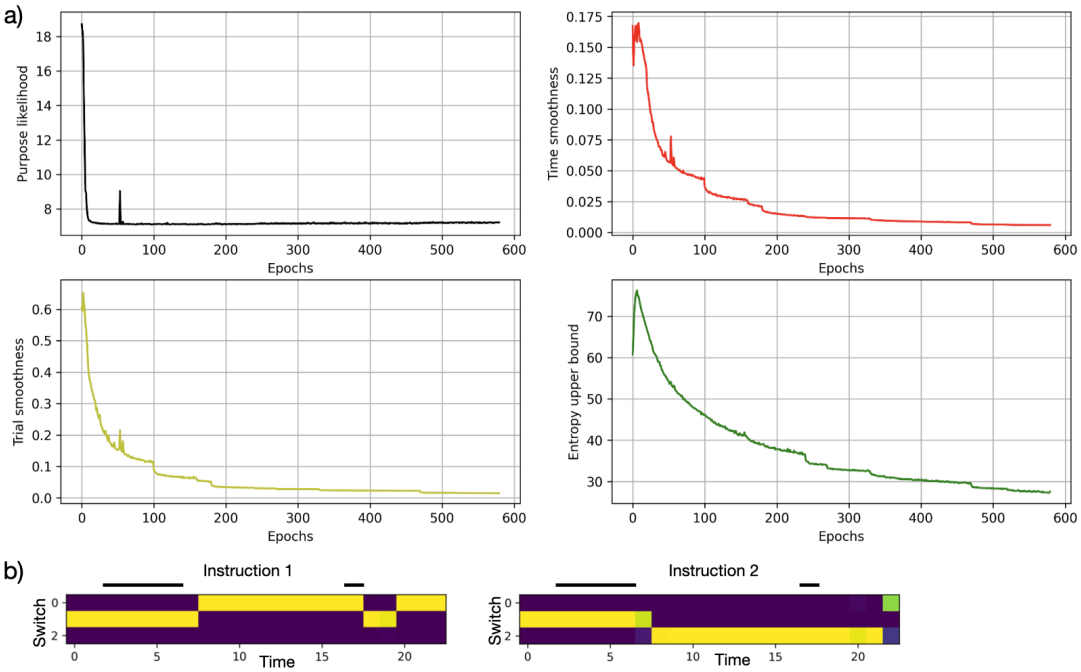
Supplementary Figure 3.8: Development of the **efficient model-based representation of the context-dependent perceptual decision making task** over the course of optimization. Each panels shows the dPCA of an intermediate result during optimization. The optimization epoch is indicated on top of each panel. Panels are organized as described in Fig.3.6b.



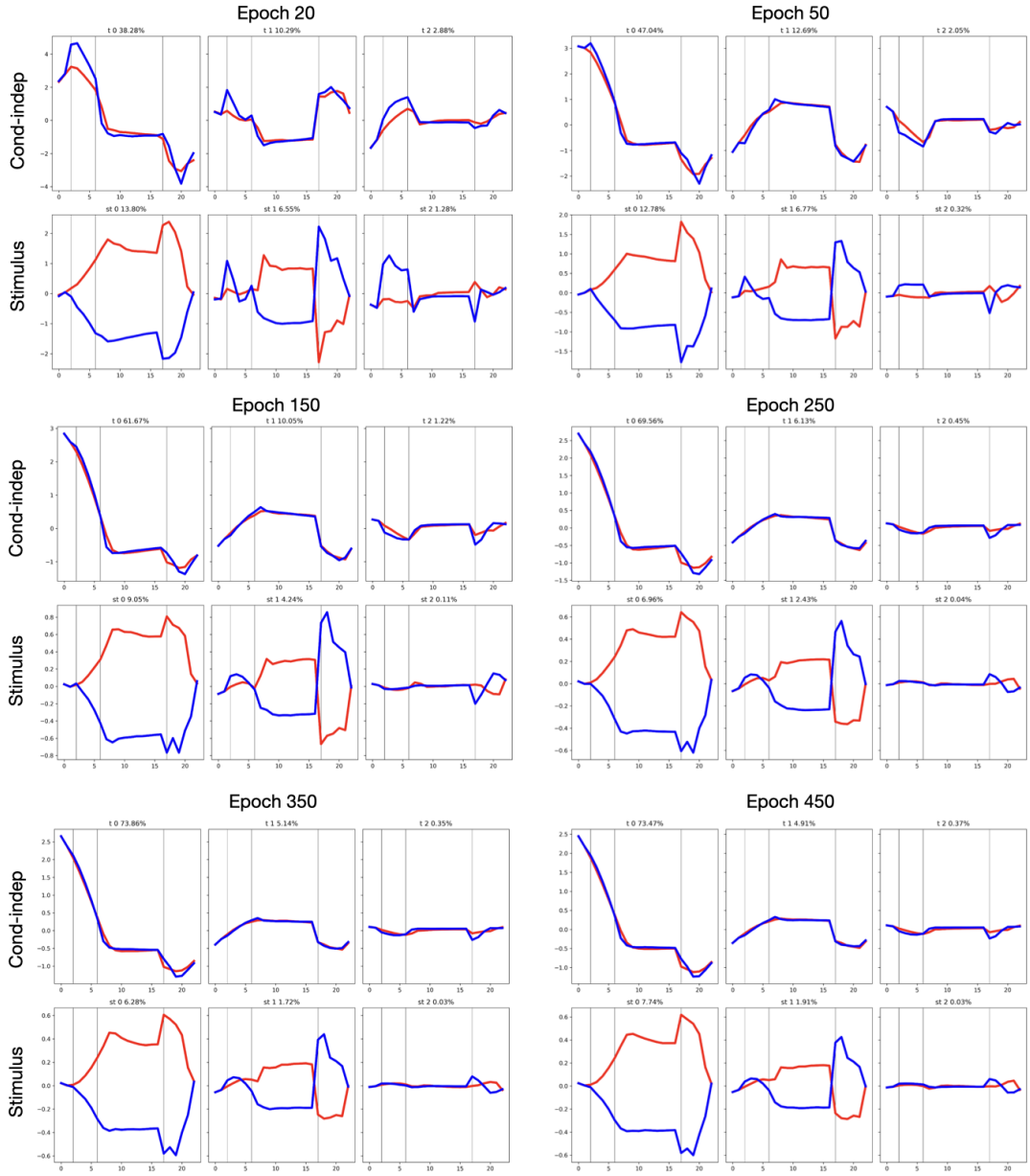
Supplementary Figure 3.9: Optimization progress and switching behaviour of the **efficient anticipatory-habitual representation of the fixed delayed licking task**. a,b) are equivalent to supplementary Fig.3.1. In b), the black bars on top of each panel denote the sample period and the time of the go cue of the trials analysed in Fig.3.7 and supplementary Fig.3.10. The left and right panels correspond to the switching behaviour for the two instruction cues.



Supplementary Figure 3.10: Development of the **efficient anticipatory-habitual representation of the fixed delayed licking task** over the course of optimization. Each panels shows the dPCA of an intermediate result during optimization. The optimization epoch is indicated on top of each panel. In each panel, the top row shows the first three condition-independent components, and the bottom row shows the first three stimulus components. Otherwise, panels are organized as described in Fig.3.7.



Supplementary Figure 3.11: Optimization progress and switching behaviour of the **efficient anticipatory-habitual representation of the randomly delayed licking task**. a,b) are equivalent to supplementary Fig.3.1. In b), the black bars on top of each panel denote the sample period and the time of the go cue of the trials analysed in Fig.3.7 and supplementary Fig.3.12. The left and right panels correspond to the switching behaviour for the two instruction cues.



Supplementary Figure 3.12: Development of the **efficient anticipatory-habitual representation of the randomly delayed licking task** over the course of optimization. Each panels shows the dPCA of an intermediate result during optimization. The optimization epoch is indicated on top of each panel. In each panel, the top row shows the first three condition-independent components, and the bottom row shows the first three stimulus components. Otherwise, panels are organized as described in Fig.3.7.

Chapter 4

Compact task representations as a normative model for higher-order brain activity

This chapter presents an older, published, version of the work presented in chapter 3. Here, we use non-parametric, or discrete, models instead of the switching linear dynamical systems of the third chapter.

4.1 Contributions

Conceptualization and method development by Severin Berger and Christian Machens. Simulations were done by Severin Berger.

The work presented in this chapter was peer-reviewed and published at the thirty-fourth Conference on Neural Information Processing Systems (NeurIPS) in 2020.

4.2 Summary

Higher-order brain areas such as the frontal cortices are considered essential for the flexible solution of tasks. However, the precise computational role of these areas is still debated. Indeed, even for the simplest of tasks, we cannot really explain how the measured brain activity, which evolves over time in complicated ways, relates to the task structure. Here, we follow a normative approach, based on integrating the principle of efficient coding with the framework of Markov decision processes (MDP). More specifically, we focus on MDPs whose state is based on action-observation histories, and we show how to compress the state space such that unnecessary redundancy is eliminated, while task-relevant information is preserved. We show that the efficiency of a state space representation depends on the (long-term) behavioural goal of the agent, and we distinguish between model-based and habitual agents. We apply our approach to simple tasks that require short-term memory, and we show that the efficient state space representations reproduce the key dynamical features of recorded neural activity in frontal areas (such as ramping, sequentiality, persistence). If we additionally assume that neural systems are subject to cost-accuracy tradeoffs, we find a surprising match to neural data on a population level.

4.3 Introduction

Arguably one of the most striking differences between biological and artificial agents is the ease with which the former navigate and control complex environments (Lake et al., 2017). Core functions enabling such behaviours, including working memory and planning, are typically attributed to higher-order brain areas such as the prefrontal cortex (PFC) (Miller and Cohen, 2001; Fuster, 2015), and exactly these functions are thought to be lacking in today's machine learning systems (Russin et al., 2020). Yet, it remains unclear how higher-order brain areas generate these complex behaviours, or even the simple behaviours that are often studied experimentally in rodents and primates. Specifically, both behavioural strategies and neural activities depend in complex ways on the task at hand, and these dependencies have so far evaded a satisfactory or intuitive explanation (Sreenivasan and D'Esposito, 2019). For example, in tasks that require animals to remember some information, neurons are sometimes persistently active (Constantinidis et al., 2018; Funahashi et al., 1989; Romo et al., 1999; Inagaki et al., 2019), while at other times they are sequentially active (Fujisawa et al., 2008; Harvey et al., 2012). Indeed, subtle changes in the timing of a task can lead to a sudden shift from one to the other (Inagaki et al., 2019), but the causes behind these activity shifts have remained unclear.

Currently, these activity patterns are mostly studied from a mechanistic, network perspective. For instance, sequential activity has sometimes been identified with feedforward dynamics, and persistent activity with recurrent or attractor dynamics (Ganguli et al., 2008; Goldman, 2009). More generally, task-related neural activity has been modeled by training recurrent neural networks (RNNs) to perform the same task as an animal (Barak et al., 2013; Inagaki et al., 2019; Mante et al., 2013; Orhan and Ma, 2019; Song et al., 2016; Yang et al., 2019). Surprisingly, RNNs can mimic recorded neurons quite well, if the task is phrased the right way, and if learning is properly regularized to avoid overfitting (Sussillo et al., 2015). However, RNNs are generally difficult to interpret and analyse, although some progress has been made in this direction (Sussillo and Barak, 2013). More importantly, training a RNN does not clarify why a particular solution is a good solution, or, indeed, if it is a good solution at all.

Here we take a step back and first define what determines a good solution. Our goal is to develop a normative approach to explain higher-order brain activities. Our starting point is the efficient coding hypothesis, which states that neural circuits should eliminate all redundant or irrelevant information (Attneave, 1954; Barlow, 1961; Simoncelli and Olshausen, 2001). We then merge the concept of an efficient representation with the formalism of reinforcement learning (RL) and Markov Decision Processes (MDPs). As most realistic tasks are only partially observable, we first endow the underlying MDPs with a notion of observations. Instead of assuming hidden causes for these observations, as in the popular partially-observable MDPs (Kaelbling et al., 1998), we simply assume that agents can accumulate large observation and action histories. As a result, states in our MDPs are not hidden, but the state space is huge and includes (short-term) memories. We then use the size of the state space as a proxy for efficiency, and we show how to eliminate redundancy and compress the state space, while preserving the behavioural goal of the agent. Some of the mathematical theory underlying the compression of dynamical systems has been developed before in other context (Bertsekas, 1995; Wolpert et al., 2015), but its application to behavioral tasks and neural data is new.

We obtain two key results. First, we illustrate that model-based agents, which may seek to adjust their policy flexibly depending on context, require a different compression strategy from habitual agents, which are already set on a given policy. Second, we generate efficient representations for two standard behavioral paradigms (Romo et al., 1999; Inagaki et al., 2019), and we show that the transition from se-

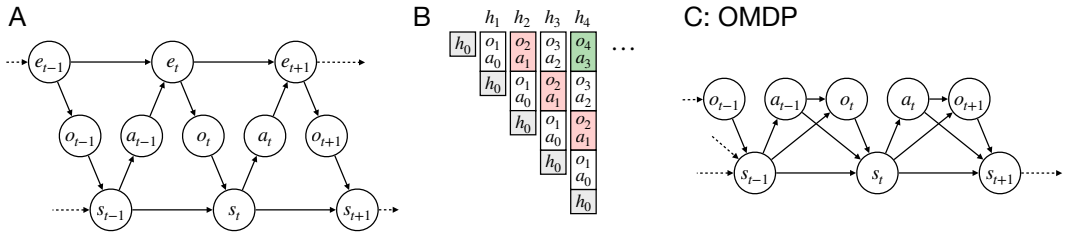


Figure 4.1: A: The agent-environment loop. The environment, e, emits observations, o, which include rewards. Based on its internal (belief) state, s, the agent chooses actions, a, that affect the environment. B: The most information the agent can have about the environment is to remember all past observations and actions, i.e., the history, h. C: Dependency graph for the OMDP.

quential to persistent activity depends on the temporal basis needed to represent the task, as well as the behavioral goal (model-based versus habitual) of the agent.

4.4 Results

4.4.1 From task structure to representation

A task is defined by a set of observations, a set of required actions, and their respective timing. Each trial of a task is a specific sequence (or trajectory) through the observation-action space. Any task representation is a function of these sequences, and the specific function may be defined by a RNN, or by a normative principle as in this study, that may then be compared with the trials' corresponding neural trajectories. Throughout this study, we follow the reinforcement learning (RL) framework and assume that the agent's control problem is to maximize future rewards.

Control under partial observability: Observation Markov decision processes

RL theory was extensively developed on the basis of Markov decision processes (MDP, (Sutton and Barto, 2018)). In MDPs agents move through states, $s \in S$, and perform actions, $a \in A$. Given such a state and action, the probability of reaching the next state, $s' \in S$, and collecting the reward, $r' \in R$, is specified by the environmental dynamics, $P(s', r'|s, a)$. An MDP is therefore defined by the tuple $\langle S, A, R, P(s', r'|s, a)\rangle$. Usually, a discount factor γ is included, but since we are dealing with episodic problems only, we set $\gamma = 1$ for the remainder of this article. The MDP state is fully observable, meaning that the observations made by the agent at each time point fully specify the state.

More realistic tasks are partially observable, so that the agent cannot access all task-relevant information through its current sensory inputs, see Fig. 4.1A. A popular extension for such RL problems are the partially observable MDPs (POMDPs, (Kaelbling et al., 1998)), which distinguish between the underlying environmental states, $e \in E$, and the agent's observations thereof, $o \in O$. Here, agents move through environmental states with probabilities $P(e'|e, a)$. In turn, they make observations $o' \in O$ (which include rewards) with probability $P(o'|e', a)$. A POMDP is therefore fully specified by the tuple $\langle E, A, O, P(e'|e, a), P(o'|e', a)\rangle$.

At each time point t, the environmental state e_t is hidden to the agent. Consequently, the agent needs to infer this state using its action-observation history $h_t = (o_t, a_{t-1}, o_{t-1}, \ldots, o_1, a_0)$, see Fig. 4.1B. This inference process can be summarized in the agent's belief state, $s_t \in S$, where $S = \{x \in \mathbb{R}_{\geq 0}^{|E|} | \sum_i x_i = 1\}$ is an $|E|$-dimensional simplex. The elements of this belief state are given by $s_t(e) = P(e|h_t)$. Upon taking an action a_t and making an observation o_{t+1}, the agent can update its belief state through Bayesian inference:

$$s_{t+1}(e') = P(e'|o_{t+1}, a_t, h_t) = \frac{P(o_{t+1}|e', a_t) \sum_{e \in E} P(e'|e, a_t) s_t(e)}{P(o_{t+1}|h_t, a_t)} \tag{4.1}$$

Here the denominator, $P(o_{t+1}|h_t, a_t) = \sum_{e'} P(o_{t+1}|e', a_t) \sum_e P(e'|e, a_t) s_t(e)$, is the observation-generating distribution given a belief state. Formally, the state update function can be summarized by the distribution $P(s'|s, a, o')$ that equals one if Eq. (4.1) returns s' given s, a, o', and zero otherwise. On the level of beliefs, we therefore recover a MDP, called the belief MDP, defined by the tuple $\langle S, A, R, P(s', r'|s, a)\rangle$ with $P(s', r'|s, a) = \sum_{o' \in O \setminus R} P(s'|s, a, o') P(o'|s, a)$ and $O \setminus R$ denoting the set of observations excluding rewards (Kaelbling et al., 1998).

Belief MDPs are generally hard to work with, since the belief states live on a (generally high-dimensional) simplex. Since the belief states are simply functions of the action-observation history, h_t, however, we could also simply use the histories themselves as states, $s_t = h_t$. To generalize this idea, we therefore define an alternative MDP directly on the level of $P(s'|s, a, o')$ and $P(o'|s, a)$ and call it observation MDP (OMDP, Fig. 4.1 C, given by the tuple $\langle S, A, O, P(s'|s, a, o'), P(o'|s, a)\rangle$). Importantly, S may simply be chosen a discrete set.

If we choose histories as states, then the transition function $P(h'|h, a, o')$ becomes simply the append function, i.e., $h' = (h, o, a)$ (Fig. 4.1B,C), and only the observation function $P(o'|h, a)$ has to be specified. We will call this specific OMDP a history-

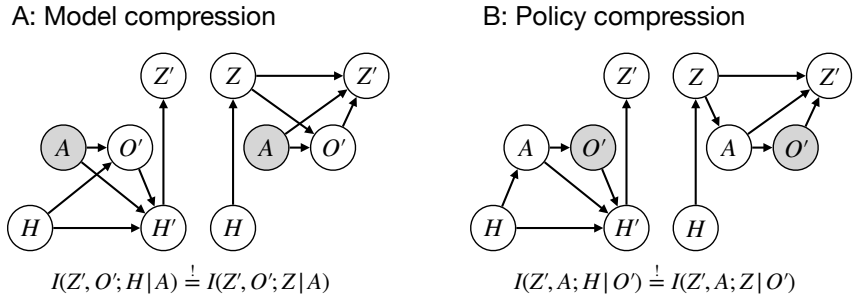


Figure 4.2: A: Dependency graphs for a history-OMDP (left) and a model-compressed OMDP (right). Conditioned variables are shaded. B: Same as A, but for policy compression.

OMDP for the remainder. Obviously, the history-OMDP will not be the most compact choice in general, since the set of histories grows exponentially with time, but it contains all task-relevant information, and therefore allows us to ask the question of how to compress the history space to get rid of the task-irrelevant bits.

State space compression

Our central goal is to find the most compact state space, Z, for a given task. For simplicity, we assume that the task's history-OMDP with state space H is already given. As states are given by action-observation histories, $h \in H$, we first attempt to directly find the compression function $P(z|h)$ that maps histories to compressed states, $z \in Z$, such that $|Z| < |H|$. We also define a decompression function, $P(h|z)$, by inverting $P(z|h)$ using an uninformative prior on h.

The compression map will depend on the specification of the type of information in H that needs to be preserved in Z. In the following, we will distinguish two types of compression, *model compression*, which finds an efficient representation for model-based agents, which have not converged on a fixed policy, and *policy compression*, which finds an efficient representation for habitual agents, which have already inferred the optimal policy.

State space compression for model-based agents

The model-based agent needs a compression that preserves all information in H about future observations (Bialek et al., 2001). In principle, the information will include observations (and their history) that may be irrelevant for a given task, but could become relevant in the future. (Once an agent has attained certainty about what is relevant or irrelevant for a given task, it should choose the more powerful compression for habitual agents, see next section). The history-OMDP's information

about future observations is contained in both the observation function, $P(o'|h,a)$, and the transition function, $P(h'|h,a,o')$, and the compressed agent, with functions $P(o'|z,a)$ and $P(z'|z,a,o')$, needs to preserve information about both (Fig. 4.2A). Similar compressions of world-models have been studied before, see e.g. (Bertsekas, 1995; Poupart and Boutilier, 2003), and we here build on these results.

Let us first consider preserving observation information when we compress the state space representation with a map $P(z|h)$. To do so, we simply require that, given any action $a \in A$, the mutual information between observations, O', and either the full or compressed state space representation, H or Z, remains the same, so that $I(O';H|A) = I(O';Z|A)$. Accordingly, whether we compute the next observation probability through $P(o'|h,a)$, or whether we first compress into z, and then compute the observation probability from there, using $P(o'|z,a) = \sum_h P(o'|h,a)P(h|z)$, should be the same.

Next we need to ensure that the compression also preserves our knowledge about state transitions. Assume we start in h, predict o' as described above, transition to h', and then compress h' into z'. Ideally, we would obtain the same result if we start in z, decompress into h, transition to h', and then compress back into z'. In terms of information, we thus obtain the condition $I(Z',O';H|A) = I(Z',O';Z|A)$. Given this constraint, we find the maximally compressive map $P(z|h)$ by minimizing the information $I(Z;H)$ between Z and H using the information bottleneck method (Tishby et al., 2000; Friedman et al., 2013):

$$\min_{P(z|h)} I(Z;H) \quad \text{subject to} \quad I(Z',O';H|A) = I(Z',O';Z|A) \tag{4.2}$$

State space compression for habitual agents

For the habitual agent, we assume that an optimal policy, $P(a|h)$, has been obtained, and we aim to find the most compact representation of this policy. The agent thus no longer needs to predict observations, but actions. A compressed representation for a habitual agent therefore requires the transition function $P(z'|z,a,o')$ and the policy $P(a|z)$. Following the logic of the model-based agent above, we therefore need to preserve transition and action information (Fig. 4.2B), yielding the condition $I(Z',A;H|O') = I(Z',A;Z|O')$.

In practice, this condition requires the mutual information conditioned on observations, yet many state-observation combinations are never provided by the environment or the experimenter. An alternative and equivalent approach, which we follow here,

is to preserve one-step information about actions and transitions by preserving future action sequences given future observation sequences. A trial k of length T is defined by the observation sequence $\{o\}_k = (o_1^k, o_2^k, \ldots, o_T^k)$ and the corresponding optimal action sequence $\{a\}_k = (a_0^k, a_1^k, \ldots, a_T^k)$. Given the history-OMDP and the policy $P(a|h)$ we can compute the likelihood of an action sequence given an observation sequence:

$$P_H(\{a\}_k|\{o\}_k) = \sum_{\{h\}} P(h_0)P(a_0^k|h_0) \prod_{i=0}^{T-1} P(h_{i+1}|h_i, a_i^k, o_{i+1}^k)P(a_{i+1}^k|h_{i+1}) \tag{4.3}$$

We now try to find the smallest state space representation, Z, with transition probabilities $P(z'|z, a, o')$ and policy $P(a|z)$, such that the action sequence likelihoods are preserved:

$$P_H(\{a\}_k|\{o\}_k) = P_Z(\{a\}_k|\{o\}_k) \quad \forall k \tag{4.4}$$

Here $P_Z(\{a\}_k|\{o\}_k)$ is the action sequence likelihood given the compressed representation, computed analogously to $P_H(\{a\}_k|\{o\}_k)$ in Eq. 4.3. Importantly, k only runs over observed trials, thereby ignoring observation sequences that never occur. We use a non-parametric setting and optimize the model parameters using expectation maximization. As many state-observation combinations and thus entries of $P(z'|z, a, o')$ are encountered in none of the trials and to prevent overfitting, we put a Dirichlet prior on transitions preferring self-recurrence (see e.g. (Montañez et al., 2015)). Furthermore, we find the smallest state space Z by brute-force. Specifically, we initialize the model with different $|Z|$, optimize the model parameters, and then take the smallest model that fulfils the likelihood condition 4.4.

Towards a more biologically realistic setting: Linear Gaussian OMDP parametrization

So far we have discussed the discrete or non-parametric treatment of tasks using discrete OMDPs. As we will show below, the non-parametric case can already give us several conceptual insights on task representations. However, to become more realistic and deal with real-valued neural activities, continuous observation spaces, and the noisiness of the brain, we need to look at possible parametrizations. Here we discuss a linear parameterization that allows us to intuitively interpret the model and make several connections to neural properties and network dynamical regimes. Furthermore, by introducing representation noise we can describe trade-offs between accuracy and complexity of representations, given a limited capacity. This automatically compresses the state space for efficiency reasons, as we will show below.

We only consider the habitual agent here, for brevity, but a model-based agent with a full OMDP model can be modelled analogously. In the non-parametric case the model parameters were parameters of categorical distributions. Assuming an N_z-dimensional state vector $z \in \mathbb{R}^{N_z}$, we here parametrize the model with normal distributions:

$$\begin{aligned} P(z'|z,a,o') &= \mathcal{N}(Az + B_a a + B_o o', \sigma_t^2 I) \\ P(a|z) &= \mathcal{N}(Cz, \sigma_r^2 I). \end{aligned} \tag{4.5}$$

Here, $A \in \mathbb{R}^{N_z \times N_z}$ is the transition matrix, $B_a \in \mathbb{R}^{N_z \times N_a}$ and $B_o \in \mathbb{R}^{N_z \times N_o}$ are the input weights of past actions and observations, respectively, $C \in \mathbb{R}^{N_a \times N_z}$ are the weights of the readout (here the policy), and σ_t and σ_r are scalar standard deviations of the isotropic transition and readout noise, respectively. Our system therefore corresponds to a linear dynamical system (LDS) for the state z. We will set the readout noise to zero for the remainder as we are only interested in how transition noise accumulates over time, modelling memory decay over time. Since there is a degeneracy in the scaling of the parameters A, B_a, B_o, C, and the transition noise, σ_t (see e.g. (Roweis and Ghahramani, 1999)) which allows the system to get rid of noise trivially, we constrain the state values from above and below so that $0 \leq \ \mu(z(i)) \leq z_{\max}$ for all $i = 1 \dots N_z$.

Given this limited capacity, both task-relevant and task-irrelevant information have to compete for resources. Accordingly, policy-irrelevant information will be ignored in favor of an accurate representation of relevant information, thus leading to compressed representations. We discuss this intuition in more detail in the Supplementary Material, and we exemplify in the simulations, below. Finally, we optimize the LDS by maximizing the likelihood of the target policy with respect to parameters A, B_a, B_o, C, analogous to the non-parametric policy compression case before.

4.4.2 Compressed state space representations and neural activities

Non-parametric policy compression for a delayed licking task

We will first apply our non-parametric policy compression on a delayed directional licking task in mice (Guo et al., 2014; Inagaki et al., 2019). In this task, mice have to decide whether a tone is of low or high frequency, and then report their decision, after a delay, by licking one of two water delivery ports. We model two versions of this task, one with a fixed delay period (fixed delay task, FDT, Fig. 4.3A-E) and one with a randomized delay period (random delay task, RDT, Fig. 4.3F-H). Neurons recorded

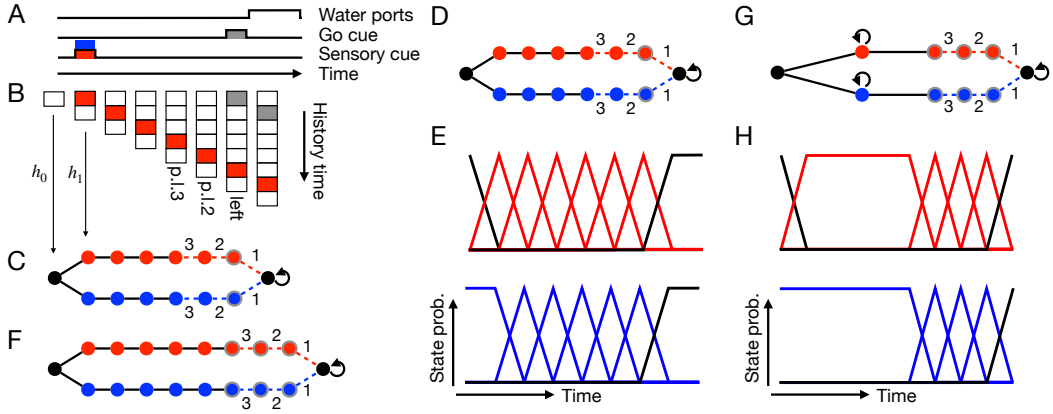


Figure 4.3: A-E: Fixed delay task (FDT). F-H: Randomized delay task (RDT). A: Task structure. Each trial starts with a tone (red or blue) that indicates the reward location. Rewards are available after a go cue (grey) that arrives either after a fixed (FDT) or randomized (RDT) delay. B: Example trial (red tone) and sequence of corresponding history states (columns, only sensory and go cues are shown). Underneath each history state, the corresponding optimal action is indicated (blank means action wait, p.l stand for preparatory left, and so on). C: History task graph with optimal policy. Nodes are history states (post-go cue states have a grey rim), edges are actions. Dashed red edges correspond to (preparatory) left actions, and dashed blue edges to (preparatory) right actions. D: Task graph for FDT after compression of the optimal policy. E: State probabilities for the two trial types. F: Same as in C, but for the RDT. G: Task graph for RDT after compression. H: as in E, for a given delay length.

in the ALM (anterior lateral motor cortex) show a striking distinction between the tasks: while activity changes during the delay period in the FDT, it remains at a steady level in the RDT (Inagaki et al., 2019).

A key difference between the two tasks is that the timing of the go cue is unpredictable in the RDT, but predictable in the FDT. A predictable go cue allows the animal to prepare its action, which we will model by introducing a sequence of preparatory actions (e.g. open mouth, stick tongue out, or internal preparations) before the actual left or right licking action (Fig. 4.3B). Furthermore, we assume that the agent takes decisions as fast as possible, in order to maximize its reward consumption. In turn, the resulting optimal policy for the FDT initiates the action sequence before the go cue (Fig. 4.3C) while in the RDT the sequence is initiated after the go cue (Fig. 4.3F). These differences are reflected in the resulting compressed state space representations shown in Fig. 4.3D and G, respectively.

In the FDT, the task representation keeps precise track of time during the delay period (Fig. 4.3D). Each time point effectively becomes its own state, and the model sequences through them. If we identify each state with the activation of an individual

neuron (or, more realistically, of a population mode), then neural activities turn on and off as in a delay line (Fig. 4.3E). This task representation thereby allows the agent to take the preparatory actions before the onset of the go-cue. We note that recorded neural activities are generally slower (they 'ramp' up or down) than the fast delay line proposed here. Such 'ramping' provides a less precise (and thereby 'cheaper') encoding of time which may be sufficient for this task as the gain for precise timing is only minor (faster access to reward). Here we only consider compressed representations that preserve future returns, and do not consider possible tradeoffs between the future returns and the compressed representations. These idealized representations require a fast delay line.

In contrast, the compressed state representation of the RDT combines all delay states and thereby discards timing information (Fig. 4.3G). In turn, the (compressed) state does not change during the delay (Fig. 4.3H). This representation is sufficient to represent the optimal RDT policy.

Non-parametric and linear compression for a somatosensory working memory task

Next, we study model and policy compression in a (somatosensory) working memory task in monkeys (Romo et al., 1999), see Fig. 4.4A. In this task, each trial consists of two vibratory stimuli with frequencies f_1 and f_2 that are presented to a monkey's fingertip with a 3sec delay. To get a reward, the monkey has to indicate which of the two frequencies was higher. Neural activities in the prefrontal cortex recorded during the task show characteristic, temporally varying persistent activity during the delay period, as observed for many other working memory tasks (Kobak et al., 2016), see also Fig. 4.5A,C.

The history-OMDP of this task is shown in Fig. 4.4B. When compressing the history space using the method for the model-based agents, we find that all states during the delay period remain uncompressed, as they are predictive of the f_2 observation. After f_2 is observed, history states with the same action-reward contingencies are combined in the compressed representation, yielding only two states ($f_1 > f_2$ and $f_1 < f_2$), which effectively correspond to the subject's decision (Fig. 4.4C). If we again identify each state with the activation of a neural population mode, we find a component corresponding to the decision, as observed in the data (Kobak et al., 2016), but also a precise encoding of time during the delay period which does not reflect recorded activity (Fig. 4.4D).

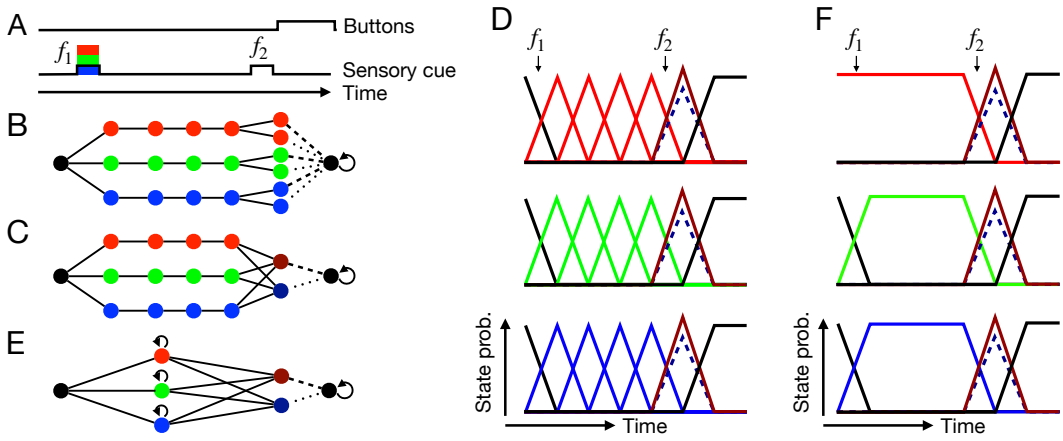


Figure 4.4: A: Task structure. We only model three f_1 frequencies for simplicity, coded red, green and blue. B: Task graph based on history states for the optimal policy (constructed as in Fig. 4.3). C: Task graph for model compression. States requiring the left and right actions are combined into a single dark red and dark blue state, respectively. D: State probabilities over time after model compression for all six trial types. Rows correspond to different f_1 values. E: Task graph for policy compression. Delay states after the f_1 presentation are combined. F same as D, but for policy compression.

Animals well-trained on tasks may be assumed to behave habitually. Indeed, when we seek to only preserve policy information, and when we assume that the animal is not preparing any actions during the delay period, we find that we can compress the state space even further (Fig. 4.4E,F). All delay states corresponding to different f_1 frequencies are merged, so that any timing information is lost. When looking at the state representation over time, we find persistent activity (Fig. 4.4F), just as in the RDT above (Fig. 4.3G). The persistent state dynamics here contrast with the sequential state dynamics of the FDT above (Fig. 4.3E). While in both tasks the delay is fixed, in the directional licking task a decision is stored while here a stimulus is stored and (under the assumption that no action needs to be prepared) timing during the delay is irrelevant.

While the non-parametric treatment yields several conceptual insights, it does not allow for a direct comparison with data. For instance, the delay line activity of the model-based agents crucially depends on the time step of the simulation, and assumes a completely noise-free evolution of the internal representations. To move closer to realistic agents, we finally model the somatosensory working memory task using the parametric LDS approach, which also includes noise. A trial is structured as in Fig. 4.4A, but with $\{f_1, f_2\} \in \mathbb{R}$ being continuous scalars. Given the rigidity of the linear parametrization we make a couple of simplifying assumptions: First, we only maximize the accuracy in the actual decision (left or right) and ignore previous actions

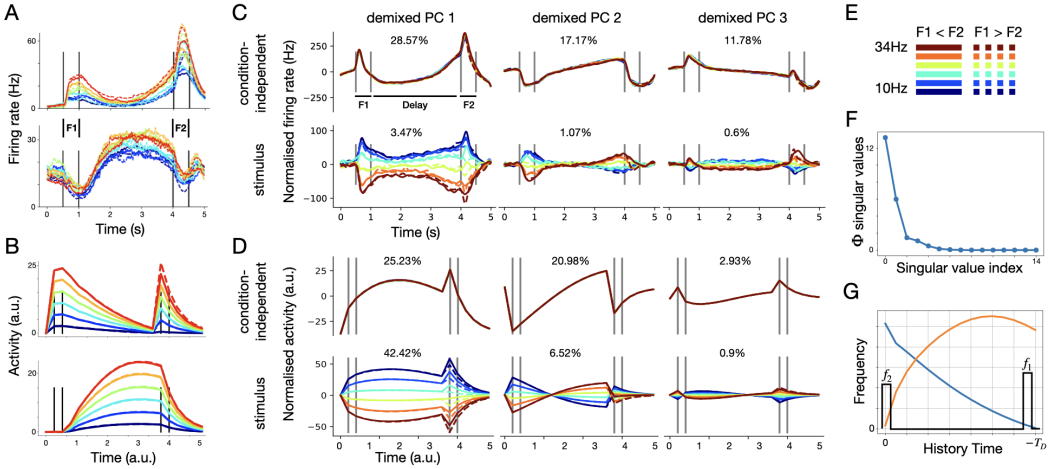


Figure 4.5: A: Peristimulus time histograms of two PFC neurons. Lines follow legend shown in E. B: Two matching model neurons (i.e., two state dimensions of z). C,D: Population level comparison using demixed principal component analysis (Kobak et al., 2016). We demixed condition-independent variance and stimulus dependent variance. C: First three condition-independent components (first row) and stimulus components (second row) of PFC neurons. D: The corresponding components of the model. Fraction of explained variance is indicated on top of each component. As components may be non-orthogonal, they do not have to add up to 100%. F: Singular values of Φ. G: The history space with one example history, h_{T_D}, drawn in black, underlaid by two vectors in Φ's row space. Specifically, since the readout weights c compute the $f_1 - f_2$ difference from the state $\mu(z_{T_D}) = \Phi h_{T_D}$, and since $z_{T_D} \geq 0$, some entries of c are positive (c_+), some negative (c_-). The blue and orange lines correspond to $-c_-^\top \Phi$ and $c_+^\top \Phi$, respectively. All neural data was processed as described in (Kobak et al., 2016).

altogether. The transition function then also becomes action independent, i.e., we set $B_a = 0$. Second, we approximate the (nonlinear) decision function, $d = \text{sign}(f_1 - f_2)$, with a linear function, $y = f_1 - f_2$.

The accuracy of the representation is thus fully defined by the readout distribution, $P(y|h_{T_D}) = N(\mu_y, \sigma_y^2)$, at decision time T_D, right after f_2 is observed. The mean, $\mu_y = c^\top \mu(z_{T_D})$, and variance, σ_y^2, of this readout are functions of the mean and variance of the final state z_{T_D}, which can be computed by unrolling the LDS. Specifically, $c^\top \in \mathbb{R}^{N_z}$ is the readout vector, and the final state mean, $\mu(z_{T_D})$, is computed by $\mu(z_{T_D}) = \Phi h_{T_D}$, with $\Phi = \begin{bmatrix} B_o & AB_o & \dots & A^{T_D-1}B_o \end{bmatrix} \in \mathbb{R}^{N_z \times T_D}$ being the linear map from histories to compressed states, analogous to $P(z|h)$ in the non-parametric case above. The compressed state space can thus be understood as a linear subspace in the space of all histories, defined by Φ. We finally find this subspace by maximizing the likelihood of $y = f_1 - f_2$ given h_{T_D} with respect to A, B_o, c as described in section 4.4.1. Simulation details, parameter values and code are provided in the Supplementary Material.

The resulting state representation dynamics resemble brain activity well on a single neuron level (Fig. 4.5A,B) as well as on a population level (Fig. 4.5C,D). Furthermore, the state dynamics are low-dimensional, a sign of the successful compression (Fig. 4.5D). Indeed, when looking at the linear map Φ directly, we see it dominated by two dimensions (Fig. 4.5F). Specifically, these two dimensions divide the history space in two bins, one for recent observations, i.e. f_2, and one for observations in the past, i.e. f_1 (Fig. 4.5F). Timing information of the stimuli is thus compressed away, similar to the non-parametric case (Fig. 4.4E,F).

4.5 Discussion

In this article we have proposed a new, normative framework for modeling and understanding higher-order brain activity. Based on the principle that neural activity reflects a maximally compact representation of the task at hand, we have reproduced dynamical features of higher-order brain areas for two example tasks involving short-term memory, and we have explained how those features follow from the normative principle.

The key principle underlying this work— representational efficiency—has been proposed before in various context. For instance, the efficient coding hypothesis has held that redundant information in sensory inputs should be eliminated (Simoncelli and Olshausen, 2001). Information-theoretical considerations have led to the proposal that the brain should only keep information about past events that is relevant for maximizing future returns (Bialek et al., 2007), which naturally suggests some combination of efficient coding and reinforcement learning (Botvinick et al., 2015). Moreover, indirect evidence for efficient task-representations has been found in the activity of dopaminergic neurons (Motiwala et al., 2022). We were also inspired by considerations of efficient representation, or coarse-graining, of dynamical systems (Wolpert et al., 2015).

On a technical level, our work extends previous studies that have considered RL under costs. While we have focused on representational costs, previous work has studied RL under control costs (Tishby and Polani, 2011; Todorov, 2009). Our approach also extends previous work on representation learning for RL (Bertsekas, 1995; Poupart and Boutilier, 2003; Lesort et al., 2018; Mahadevan, 2009; Singh et al., 2005). While (Mahadevan, 2009; Singh et al., 2005) consider simultaneous learning of representation and control, we have not considered the problem of learning. In theory, an agent could first learn a history-OMDP model, from there solve, or plan,

for the optimal policy using dynamic programming, and then compress the policy. This learning strategy of going from a model-based strategy to a model-free strategy has been conceived before (Lengyel and Dayan, 2008; Chrisman, 1992; McCallum, 1996). In practice, starting with the full and detailed history representation will often prove infeasible, and one would therefore assume that agents also have to go the opposite way: starting with a coarse representation that is then expanded (McCallum, 1996). Consequently, there are many paths conceivable on how to get to a compact representation, each of which might have different advantages. We therefore consider learning a separate, and presumably more difficult problem, and leave it for future work.

Finally, we think that the OMDP framework and especially parametrizations thereof might be a fruitful avenue for partially observable RL research. POMDPs have been shown computationally untractable and new ways of dealing with partial observability are considered to be needed (see e.g. (Sutton and Barto, 2018), chapter 17.3). Furthermore, RNN systems used for RL, such as in (Wang et al., 2018), are effectively using OMDP parametrization.

4.6 Materials and Methods

4.6.1 History representation under a noisy, constrained linear dynamical system

Here we give a short analysis of how different linear dynamical systems (LDS) result in different tradeoffs of representing relevant versus irrelevant parts of a history, given transition noise. We will use the same notation as in the main text.

Let us look at a single scalar readout y of an arbitrary LDS, which, at time t, is given by:

$$y_t \sim N(c^\top \mu(z_t), c^\top \sigma^2(z_t) c) \tag{4.6}$$

Here, c correspond to the readout weights. The mean, $\mu(z_t)$, and the covariance, $\sigma^2(z_t)$, of the state variable, z_t, are given by (under the assumption of certain initial state conditions $\mu(z_0) = 0$ and $\sigma^2(z_0) = 0$):

$$\begin{aligned} \mu(z_t) &= \Phi h_t \\ \sigma^2(z_t) &= \sigma_t^2 \sum_{n=0}^{t-1} A^n (A^n)^\top \end{aligned} \tag{4.7}$$

with

$$\Phi = \begin{bmatrix} B_o & AB_o & \dots & A^{t-1}B_o \end{bmatrix} \tag{4.8}$$

The dynamics and input weights are denoted by A and B_o, respectively. As noted in the main text, there is a degeneracy between the scaling of the transition noise σ_t and the weights of the LDS. We chose to constrain this degeneracy by bounding state values from below and above. Here we want to first describe our rationale behind this choice of constraint. Second, we describe how an LDS under this constraint must trade-off between relevant and irrelevant information.

Constraining the LDS to remove scaling degeneracy

As seen in the equations above, the uncertainty in the readout is determined by two parameters, the readout vector c and the dynamics A. The input weights, B_o, only feature in the mean, it is thus easy to see that the readout noise can be eradicated by decreasing $\|c\|$ arbitrarily and increasing $\|B_o\|$ accordingly, while leaving the readout mean unchanged. This leads to a first degeneracy, independent of the dynamics A, that quenches the noise trivially. A second degeneracy may arise through transient amplifications by non-normal dynamics. An example would be a purely feedforward dynamics that amplifies the input.[1]

Common to both degeneracies is that they amplify the input such that the additive transition noise, σ_t, is relatively scaled down. Hence, both degeneracies may be removed by constraining the scale, or some norm, of the state z. We chose to bound each dimension separately for the following reasons:

Each state dimension z_i can be interpreted as a component in the representation of a history. Specifically, $\mu(z_t) = \Phi h_t$ and thus an estimate $\hat{h}_t$ of the history can be obtained through $\hat{h}_t = \Phi^\dagger \mu(z_t)$, where $\Phi^\dagger$ denotes the Moore-Penrose inverse of Φ. Each column $\Phi_i^\dagger$ thus defines the direction in history space that component z_i is representing. Since we have N_z such components we can represent N_z, potentially overlapping, directions in the history space. By bounding the activity of each component, or state dimension, separately, as opposed to bounding the L_1 or the L_2 norms of the state vector, we effectively assume all components to be independent, i.e. if one component is more active it does not imply that another component has to be less active. For example, if only one history direction is task-relevant, then all N_z compo-

[1] We note that dynamics amplifying the input will in general also amplify the noise. Without making any structural assumptions on A, such as assuming normal dynamics, it is hard to treat this intricate relationship analytically. We thus content ourselves with noting that dynamics that trivially quench the noise exist, and thus must be constrained.

nents can code for the same direction, reducing the uncertainty in the representation of that direction because the readout may average N_z representations. If there are N_z independent relevant directions in the history space, every component has to code for a separate direction, thus effectively increasing the noise. This trade-off, that follows from the limited capacity of the constrained LDS, is discussed in more detail in the following section.

Representational tradeoffs under limited capacity

As described above, our LDS has a limited capacity defined by the number of components N_z and the range of activity $[0, z_{\max}]$. Furthermore, both degeneracies will scale up the system state to effectively scale down the relative size of the transition noise σ_t, thereby pushing the system's dynamic range to have $z_{\max}$ as an upper bound. Here, we first describe how such a system has to focus on relevant information only in order to increase the precision in the representation of this relevant information (Fig. 4.6A). Second, we illustrate how a system may also increase precision by ignoring timing information, if irrelevant (Fig. 4.6B).

Let us consider a single state dimension $\mu(z_i) = \phi_i^\top h$, where $\phi_i^\top$ is the i'th row of Φ and h is a history. Due to the upper bound $z_{\max}$ we have:

$$\begin{aligned} z_{\max} \geq \mu(z_i) &= \phi_i^\top h \\ &= \cos\alpha \|\phi_i\| \|h\| \end{aligned} \tag{4.9}$$

As the length of ϕ_i will scale any signal in h, $\|\phi_i\|$ effectively determines the signal to noise ratio. More specifically, each component can either represent the whole history faithfully ($\alpha = 0$) with low precision $\|\phi_i\| = z_{\max}/\|h\|$, or only parts of the history $\alpha > 0$ with higher precision $\|\phi_i\| > z_{\max}/\|h\|$. (For simplicity we assumed here that h is the history hitting $z_{\max}$, i.e. $z_{\max} = \phi_i^\top h$). Figure 4.6A illustrates this trade-off for a single represented history.

The same trade-off applies to the representation of two histories adjacent in time that both contain relevant information. Specifically, we consider $h_1 = \begin{bmatrix} h & 0 \end{bmatrix}^\top$ and $h_2 = \begin{bmatrix} 0 & h \end{bmatrix}^\top$, and assume that only the magnitude h is task-relevant, but not at what time h was observed. Representations ignoring timing information then increase precision (Fig. 4.6B). Ignoring timing information generally leads to a smoothing of the representation over time.

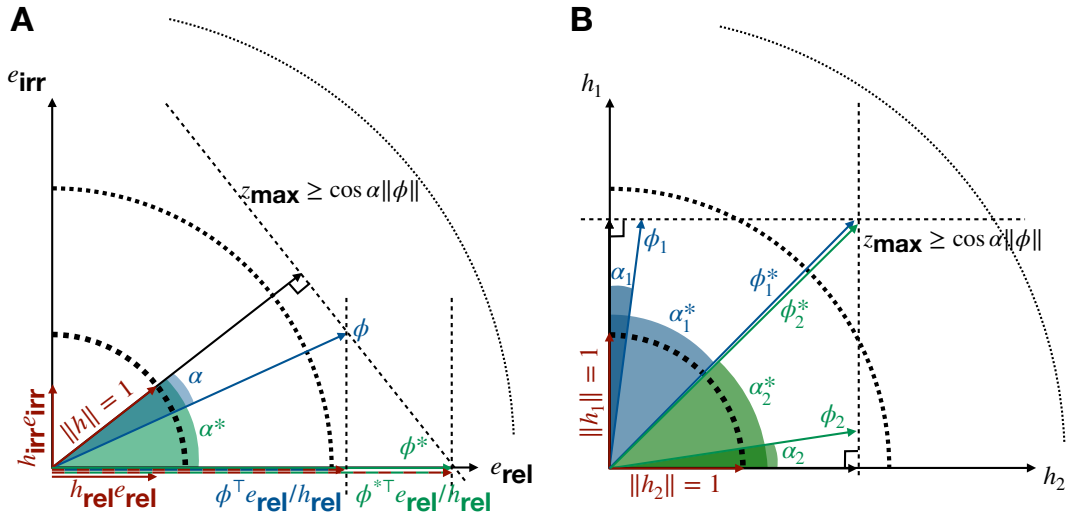


Figure 4.6: Trade-offs in a noisy, constrained LDS. A: We consider a history $h = h_{\text{rel}}e_{\text{rel}} + h_{\text{irr}}e_{\text{irr}}$ consisting of a relevant direction e_{rel} and an irrelevant direction e_{irr}, scaled by h_{rel} and h_{irr}, respectively. All three vectors are assumed to have unit length. We consider two choices of represented directions, ϕ (blue) and ϕ^* (green). Their respective angles to the history are α and α^*. The lengths $\|\phi\|$ and $\|\phi^*\|$ are constrained by their projection onto the history h, specifically by the upper bound $z_{\max}$ (dashed titled line). As a longer ϕ in general signifies lower noise, dashed half circles are depicted akin to noise isoclines. Direction ϕ^* leads to a longer projection onto the relevant direction (red-green dashed arrow) than ϕ (red-blue dashed arrow), thus leading to a higher precision representation of the relevant information. The increased precision is thus enabled by discarding irrelevant information. B: Representation of two histories adjacent in time, specifically $h_1 = \begin{bmatrix} h & 0 \end{bmatrix}^\top$ and $h_2 = \begin{bmatrix} 0 & h \end{bmatrix}^\top$. Both histories have unit length. We assume only the magnitude of h is relevant, but not when in the history h was observed. We consider two sets of two representation directions, ϕ_1, ϕ_2 and ϕ_1^*, ϕ_2^*. Angles and constraints are depicted as in A. The first set, ϕ_1, ϕ_2, codes for both directions almost orthogonally (small α_1, small α_2). Each history is thus represented faithfully. In the second set, ϕ_1^* and ϕ_2^* have reduced overlap with h_1 and h_2, respectively, and thus no longer represent the two histories faithfully. Specifically, ϕ_1^* and ϕ_2^* give up timing information, but both still represent the task-relevant magnitude h. In fact, they represent h at both time steps, the readout can therefore average the two representations to reduce readout noise.

As a note of caution we emphasize here that the illustrations in Figure 4.6 and the explanations given here aim at describing the main effects governing the representation of histories, but they do not represent a rigorous analysis. For example, even though we treated rows ϕ_i as independent, the matrix Φ is not any arbitrary matrix, but a matrix following the specific structure in equation 4.8. Furthermore, this structure depends on the dynamics A which also influences the noise accumulation, and, as noted above, we ignore this interaction. Nevertheless, we found the intuitions presented here quite useful and accurate, as for example seen in figure 4.5 of the main text.

4.6.2 Description of the LDS model of the somatosensory working memory task

Here we describe in more detail the LDS model of the somatosensory working memory task Romo et al. (1999) introduced in section 4.4.1 and figure 4.5.

Model setup and parameter choices

In our model, a trial consists of $T_D = 15$ time steps. The first frequency, f_1, is presented on the second time step, the second frequency, f_2, on the last time step. Thus all entries of the history h_{T_D} at decision time T_D will be zero, except for $h_{T_D,1} = f_2$ and $h_{T_D,14} = f_1$. Since $h_{T_D} \in \mathbb{R}^{15}$, we set $N_z = 15$ in order to give the system the capacity to represent the history space faithfully. The upper bound $z_{\max}$ will be reached by the history $h_{T_D}^{\max}$ of maximum length. Assuming $\|h_{T_D}^{\max}\| = 1$, we set $z_{\max} = 1$. Lastly, we set the transition noise variance $\sigma_t^2 = 0.01$.

Optimization

As described in the main text, we maximize the log-likelihood of the $f_1 - f_2$ difference y, given the history h_{T_D}, subject to the state values being within the lower (0) and upper ($z_{\max}$) bounds. The likelihood, for a trial i with frequencies f_1^i and f_2^i, is given by:

$$P(y_i|h_{T_D}^i) = N(\mu_{y^i}, \sigma_{y^i}^2) \tag{4.10}$$

Using equations 4.6 and 4.7, we have $\mu_{y^i} = c^\top \mu(z_{T_D}^i) = c^\top \Phi h_{T_D}^i$, and $\sigma_{y^i}^2 = c^\top \sigma^2(z_{T_D}^i)c = \sigma_t^2 c^\top \sum_{n=0}^{t-1} A^n (A^n)^\top c$. We thus see that the readout variance is trial independent and we write σ_y^2.

The log-likelihood $L(y_i|h^i_{T_D})$ is then given by:

$$L(y_i|h^i_{T_D}) = -k - \log(\sigma_y) - \frac{(y_i - c^\top \Phi h^i_{T_D})^2}{2\sigma_y^2} \tag{4.11}$$

Here k is a constant. To speed up optimization, we approximate the trial-dependent term $(y_i - c^\top \Phi h^i_{T_D})^2$ by first writing $f_i = \begin{bmatrix} f_1^i & f_2^i \end{bmatrix}^\top$, and then $y_i = \begin{bmatrix} 1 & -1 \end{bmatrix} f_i$ and $h^i_{T_D} = M f_i$, where $M \in \mathbb{R}^{T_D \times 2}$ is the appropriate matrix mapping f_1^i and f_2^i to the history $h^i_{T_D}$. We then have $(y_i - c^\top \Phi h^i_{T_D})^2 = (\delta^\top f_i)^2$, with $\delta^\top = \begin{bmatrix} 1 & -1 \end{bmatrix} - c^\top \Phi M$. The log-likelihood is thus maximal for every trial if $\|\delta\|^2 = 0$, i.e. when the composition of the readout c with the map Φ performs the $f_1 - f_2$ operation. We thus approximate $(\delta^\top f_i)^2$ by $\|\delta\|^2$, thereby making the log-likelihood trial independent. Finally, we optimize using gradient ascent.

Generating model sequences to compare to neural data

After optimizing the LDS as described above, we want to compare the resulting state trajectories to the corresponding neural trajectories.

To generate such state trajectories, we run our optimized LDS providing both positively and negatively scaled f_1 and f_2 frequencies, as is common among other models of this task, e.g. Song et al. (2016); Machens et al. (2005), and as motivated by the frequency coding in the secondary somatosensory cortex S2 Romo et al. (2002), the structure providing the frequency information to the PFC. Specifically, after generating a state sequence for trial i with frequencies f_1^i and f_2^i, we generate a second sequence with frequencies f^i_{1-} and f^i_{2-}, where $f^i_- = a + (b - f^i)$. The scalar b is set to the maximum of all frequencies presented, and $a > 0$. Each trial's modelled state sequence thus has $2N_z$ dimensions.

Next we modify the state trajectories after f_2 is presented. Neural trajectories move back to baseline after the f_2 presentation and the decision, thereby discarding for example frequency information. This makes sense from our efficiency perspective, but is not within the scope of our LDS model. In order to not let the principal dimensions of our model data be affected by this post-f_2 activity, we artificially let the state decay after f_2 presentation.

Both steps described here, as well as all the steps described in section 4.6.2, are implemented in the provided code.

Chapter 5

General Discussion

It is believed that higher-order brain regions play a significant role in enabling the flexibility and versatility of natural animal intelligence (Fuster, 2015). In chapter 2, we looked at the role that population recordings from these regions play in these skills across a range of behavioral paradigms. We concluded that the activities serve as task representations after observing that they represent all task variables. However, it is often unclear what these representations actually mean or, to be more precise, why certain task variables are represented in the manner that we find.

A large set of studies has aimed at explaining these neural activities from a mechanistic perspective. While this work of course is very important, it does not answer why we see these representations or what overarching function the representation is achieving. The currently most popular strain of modelling work of higher-order brain activity uses RNNs that are optimized to perform a specific function. While these methods in principle consider function as the main determinant of the resulting model, they often suffer from two short-comings that we address in this work. First, studies using the RNN-approach often do not explicitly state the assumed function or purpose of the modelled activities, nor is it tested against a well defined set of alternative purposes. Second, for a given assumed purpose, a myriad of RNN solutions exist. In other words, any purpose can be fulfilled by many representations.

Here, we proposed a normative approach that asks what the underlying purpose of a neural representation might be. We first clearly defined what a purpose is and formalized it within the reinforcement learning framework. Specifically, we formalized two purposes, the purpose of serving a habitual behavior, and the purpose of serving a model-based behaviour. We then, for each purpose, searched for the corre-

sponding task representation that is as compact as possible, which we called efficient representation.

Looking for efficiency is justified by its previous success in visual areas, but also by its several benefits: First it makes the representation associated with a purpose unique, making a match to neural data not a matter of serendipity. Second, efficiency is beneficial from a computational perspective, as having efficient or compact representations saves storage, energy, and they also generalize across certain task variations. Third, efficient representations are easier to interpret as all representational features are either determined by the assumed purpose or by the task structure.

We applied our approach to three tasks in both monkeys and mice, and with recordings from PFC and ALM. In a first task, the somatosensory working memory task (Romo et al., 1999), we could explain the observed PFC activities in terms of a model-based purpose, which matches with previous conceptual models of PFC. Similarly, in the second task, a context-dependent perceptual decision making task (Mante et al., 2013), we also found better agreement with the model-based hypothesis to explain the PFC data. In the last task, the delayed licking task in mice (Inagaki et al., 2019), we had to extend our set of purposes in order to explain the recordings from ALM.

We think our normative approach has the potential to push forward research in higher-order brain areas due to it its hypothesis-driven nature. Once a purpose is agreeing with a particular task and a neural data set, it can be used to look for possible task alterations or new tasks that are specifically designed to test the current hypothesis or to generate new hypotheses. This iterative approach is arguably what underlies the scientific method.

The purposes we studied here directly correspond to two strategies of behaviour, i.e. the habitual behaviour and the model-based or goal-directed behaviour. It is thought that animals, such as the mice and monkeys of the studies discussed here, are capable of using both these strategies (Dolan and Dayan, 2013). However, whether at any given time an animal is using one strategy or the other, is difficult to determine (Akam et al., 2015). So far, the attempts to do so rely on the analysis of behaviour only, yet it might be helpful to use neural data as well. For this, a clear connection between a behavioural strategy and its neural basis would be needed, and our approach potentially contributes to this endeavor.

At the same time, we do not claim here any causal relationship between the modelled purpose of a neural representation and the behaviour that an animal is following. Therefore, while our hypotheses are based on behavioural strategies, we do not use any behavioural data to constrain our models. To establish a causal relationship between neural activity and behaviour is a main goal of neuroscience, though. Thus, in a future extension of our approach, studying whether the behaviour predicted by a purpose agrees with behavioural recordings, and how the behaviour relates to the neural representation, might give evidence for such a causal relationship.

Two related topics have also not been addressed in this thesis, namely how neural activities mechanistically implement a putative purpose and how a neural representation arises over the course of learning. In principle, our SLDS models are especially well-suited for analysis of their mechanism, as their state space is linearly divided into a set of partitions, where each partition has its own linear dynamics. However, proposing an underlying mechanism opens up further questions. For example, apriori, it is unclear whether a particular efficient representation has a unique mechanism associated to it. Apart from this, in the spirit of the work of this thesis, how the complexity of a mechanism relates to the complexity of a representation would be interesting to study.

Similarly, the problem of learning is also beyond the scope of this thesis. Over the course of learning, probably different behavioural strategies are involved (see e.g. Lengyel and Dayan (2008)). From a normative perspective, it is thus unclear how, for example, a habitual representation should develop over learning, or how the different behavioural strategies, and the different brain areas that are associated to each strategy, interact to learn a new behaviour.

5.1 Future directions

There are several possible avenues for furthering the work presented in this thesis. A first extension would be to formalize purposes beyond the habitual and the model-based purposes, especially the ones presented in Fig.3.3. All of these purposes are reasonable hypotheses for how a task may be represented, as they all correspond to a valuable representation within the reinforcement learning framework. For example, the reward-predictive representation is suited for an agent that wants to estimate the value, i.e. the expected cumulative amount of reward, of a given situation, by using methods such as temporal difference learning (Sutton and Barto, 2018). As signatures of reward prediction errors, which underlie the temporal difference algorithm, have

been found in the firing patterns of midbrain neurons (Schultz et al., 1997), one might also expect to find reward-predictive representations.

Apart from studying more purposes, it might be fruitful as well to characterise the tradeoff between a representation's efficiency and its purpose-sufficiency. In other words, while we considered lossless compression of a task in this work, it would be interesting to test lossy compression as well, such as e.g. in Motiwala et al. (2022). For example, a habitual behaviour that is suboptimal in terms of the rewards it incurs, might be a lot simpler to represent (Ma and Hermundstad, 2022). Actor-critic objectives (Sutton and Barto, 2018) could be used to directly trade off the value incurred by a policy — or a habitual behavior — and the compactness of the required representation.

Furthermore, here we considered representations of one task at a time, yet nothing prevents our approach to treat a family of tasks simultaneously. What happens if tasks have related structure and how does this differently affect e.g. habitual and model-based representations (see e.g. Whittington et al. (2020); Yang et al. (2019))?

Beyond these extensions of the current work, our approach can also be improved in a number of ways. First, it would benefit from a more rigorous comparison between normative predictions and neural data, for example through a quantitative comparison. This would also open up the possibility for statistical inference over our hypothesized purposes, by using model comparison or as e.g. in Młynarski et al. (2021).

The practicality of the approach could be improved by finding a more straightforward way of obtaining efficient representations, given a task and a purpose. So far we take the dynamical systems approach using switching linear dynamical systems, in order to fulfil the memory constraint (see section 3.6). These approaches suffer from vanishing and exploding gradients, though, which makes optimization brittle and slow. Another possibility is to not enforce the memory constraint through structural constraints, i.e. through recurrent dynamics, but to define more direct, feed-forward, encoders, and then enforce the memory constraint through the objective.

Finally, while in principle an efficient representation is only a function of the task and the purpose, our choice of parameterization of the variational distributions as well as the encoder will have an effect. The SLDSs that we chose are quite flexible, however, they might still constrain the solutions in certain ways (see e.g. caption of supplementary Fig.3.1). In this thesis, we have already seen that an inappropriate

choice of encoder can lead to misleading results. Fig.4.5d suggests that an efficient habitual representation reproduces the neural representation quite well, yet this seems only due to the fact that we have used a rigid linear dynamical system that is able to only move around information gradually. For future work, it would thus be valuable to investigate whether alternative encoder parameterizations, such as for example a RNN, yield the same results.

References

Akam, T., Costa, R., and Dayan, P. (2015). Simple plans or sophisticated habits? state, transition and learning interactions in the two-step task. *PLoS Computational Biology*, 11:e1004648.

Alemi, A. A., Fischer, I., Dillon, J. V., and Murphy, K. (2016). Deep variational information bottleneck. *arXiv*, pages 1–19.

Aoi, M. C., Mante, V., and Pillow, J. W. (2020). Prefrontal cortex exhibits multi-dimensional dynamic encoding during decision-making. *Nature Neuroscience*, 23:1410–1420.

Atick, J. J. and Redlich, A. N. (1992). What does the retina know about natural scenes? *Neural Computation*, 210:196–210.

Attneave, F. (1954). Some informational aspects of visual perception. *Psychological review*, 61(3):183–193.

Baddeley, A. (2003). Working memory: Looking back and looking forward. *Nature Reviews Neuroscience*, 4:829–839.

Balleine, B. W. and Dickinson, A. (1998). Goal-directed instrumental action: contingency and incentive learning and their cortical substrates. *Neuropharmacology*, 37:407–419.

Balleine, B. W. and O'Doherty, J. P. (2009). Human and rodent homologies in action control: Corticostriatal determinants of goal-directed and habitual action. *Neuropsychopharmacology 2010 35:1*, 35:48–69.

Barak, O., Sussillo, D., Romo, R., Tsodyks, M., and Abbott, L. F. (2013). From fixed points to chaos: Three models of delayed discrimination. *Progress in Neurobiology*, 103:214–222.

Barak, O., Tsodyks, M., and Romo, R. (2010). Neuronal population coding of parametric working memory. *Journal of Neuroscience*, 30:9424–9430.

Barber, D. (2006). Expectation correction for smoothed inference in switching linear dynamical systems. *Journal of Machine Learning Research*, 7:2515–2540.

Barlow, H. B. (1961). Possible Principles Underlying the Transformations of Sensory Messages. In Rosenblith, W., editor, *Sensory Communication*, pages 217–234. The MIT Press.

Behrens, T. E., Muller, T. H., Whittington, J. C., Mark, S., Baram, A. B., Stachenfeld, K. L., and Kurth-Nelson, Z. (2018). What is a cognitive map? organizing knowledge for flexible behavior. *Neuron*, 100:490–509.

Behrens, T. E., Woolrich, M. W., Walton, M. E., and Rushworth, M. F. (2007). Learning the value of information in an uncertain world. *Nature Neuroscience*, 10:1214–1221.

Bell, A. J. and Sejnowski, T. J. (1997). The 'independent components' of natural scenes are edge filters. *Vision Research*, 37:3327–3338.

Ben-Yishai, R., Bar-Or, R. L., and Sompolinsky, H. (1995). Theory of orientation tuning in visual cortex. *Proceedings of the National Academy of Sciences*, 92:3844–3848.

Berger, S. and Machens, C. K. (2020). Compact task representations as a normative model for higher-order brain activity. In *Advances in Neural Information Processing Systems*, volume 2020-Decem.

Bertsekas, D. P. (1995). *Dynamic programming and optimal control*, volume 1. Athena Scientific.

Bialek, W., de Ruyter van Steveninck, R. R., and Tishby, N. (2007). Efficient representation as a design principle for neural coding and computation.

Bialek, W., Nemenman, I., and Tishby, N. (2001). Predictability, Complexity, and Learning. *Neural Computation*, 13(1949):2409–2463.

Botvinick, M., Weinstein, A., Solway, A., and Barto, A. (2015). Reinforcement learning, efficient coding, and the statistics of natural tasks. *Current Opinion in Behavioral Sciences*, 5:71–77.

Boyden, E. S., Zhang, F., Bamberg, E., Nagel, G., and Deisseroth, K. (2005). Millisecond-timescale, genetically targeted optical control of neural activity. *Nature neuroscience*, 8(9):1263–1268.

Brendel, W., Romo, R., and Machens, C. K. (2011). Demixed principal component analysis. In Shawe-Taylor, J., Zemel, R., Bartlett, P., Pereira, F., and Weinberger, K., editors, *Advances in Neural Information Processing Systems*, volume 24. Curran Associates, Inc.

Brody, C. D., Hernández, A., Zainos, A., and Romo, R. (2003). Timing and neural encoding of somatosensory parametric working memory in macaque prefrontal cortex. *Cerebral Cortex*, pages 1196–1207.

Bruce, C. J. and Goldberg, M. E. (1985). Primate frontal eye fields. i. single neurons discharging before saccades. *Journal of Neurophysiology*, 53:603–635.

Chalk, M., Marre, O., and Tkacik, G. (2016). Relevant sparse codes with variational information bottleneck. *arXiv*, pages 1–9.

Chrisman, L. (1992). Reinforcement learning with perceptual aliasing: The perceptual distinctions approach. In *AAAI*, volume 1992, pages 183–188. Citeseer.

Compte, A., Brunel, N., Goldman-Rakic, P. S., and Wang, X.-J. (2000). Synaptic mechanisms and network dynamics underlying spatial working memory in a cortical network model. *Cerebral Cortex*, 10:910–923.

Constantinidis, C., Funahashi, S., Lee, D., Murray, J. D., Qi, X. L., Wang, M., and Arnsten, A. F. (2018). Persistent spiking activity underlies working memory. *Journal of Neuroscience*, 38(32):7020–7028.

Coutureau, E. and Killcross, S. (2003). Inactivation of the infralimbic prefrontal cortex reinstates goal-directed responding in overtrained rats. *Behavioural Brain Research*, 146:167–174.

Cueva, C. J., Saez, A., Marcos, E., Genovesio, A., Jazayeri, M., Romo, R., Salzman, C. D., Shadlen, M. N., and Fusi, S. (2020). Low-dimensional dynamics for working memory and time encoding. *Proceedings of the National Academy of Sciences of the United States of America*, 117:23021–23032.

Cunningham, J. P. and Yu, B. M. (2014). Dimensionality reduction for large-scale neural recordings. *Nature Neuroscience*, 17:1500–1509.

Damasio, H. and Damasio, A. (1989). *Lesion Analysis in Neuropsychology*. Oxford University Press.

Daw, N. D., Gershman, S. J., Seymour, B., Dayan, P., and Dolan, R. J. (2011). Model-based influences on humans' choices and striatal prediction errors. *Neuron*, 69:1204–1215.

Dickinson, A., Nicholas, D. J., and Adams, C. D. (1983). The effect of the instrumental training contingency on susceptibility to reinforcer devaluation. *The Quarterly Journal of Experimental Psychology Section B*, 35:35–51.

Dickinson, A. D. (1985). Actions and habits: the development of behavioural autonomy. *Philosophical Transactions of the Royal Society of London. B, Biological Sciences*, 308:67–78.

Dolan, R. J. and Dayan, P. (2013). Goals and habits in the brain. *Neuron*, 80:312–325.

Donoso, M., Collins, A., and Koechlin, E. (2014). Foundations of human reasoning in the prefrontal cortex. *Science*, 344.

Dubreuil, A., Valente, A., Beiran, M., Mastrogiuseppe, F., and Ostojic, S. (2021). The role of population structure in computations through neural dynamics. *bioRxiv*, page

2020.07.03.185942.

Flesch, T., Juechems, K., Dumbalska, T., Saxe, A., and Summerfield, C. (2022). Orthogonal representations for robust context-dependent task performance in brains and neural networks. *Neuron*, 110:1258–1270.e11.

Freedman, D. J., Riesenhuber, M., Poggio, T., and Miller, E. K. (2001). Categorical representation of visual stimuli in the primate prefrontal cortex. *Science*, 291:312–316.

Friedman, N., Mosenzon, O., Slonim, N., and Tishby, N. (2013). Multivariate information bottleneck. *arXiv preprint arXiv:1301.2270*.

Friston, K. J., Fitzgerald, T., Rigoli, F., Schwartenbeck, P., and Pezzulo, G. (2016). Active inference: A process theory. *Neural computation*, 1872:1–49.

Fujisawa, S., Amarasingham, A., Harrison, M. T., and Buzsáki, G. (2008). Behavior-dependent short-term assembly dynamics in the medial prefrontal cortex. *Nature Neuroscience*, 11(7):823–833.

Funahashi, S., Bruce, C. J., and Goldman-Rakic, P. S. (1989). Mnemonic coding of visual space in the monkey's dorsolateral prefrontal cortex. *Journal of Neurophysiology*, 61(2):331–349.

Funahashi, S., Bruce, C. J., and Goldman-Rakic, P. S. (1993). Dorsolateral prefrontal lesions and oculomotor delayed-response performance: evidence for mnemonic "scotomas". *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 13:1479–1497.

Fuster, J. (2015). *The prefrontal cortex*. Academic Press.

Fuster, J. M. and Alexander, G. E. (1971). Neuron activity related to short-term memory. *Science*, 173:652–654.

Ganguli, S., Huh, D., and Sompolinsky, H. (2008). Memory traces in dynamical systems. *Proceedings of the National Academy of Sciences of the United States of America*, 105(48):18970–18975.

Gao, P., Trautmann, E., Yu, B. M., Santhanam, G., Ryu, S., Shenoy, K., and Ganguli, S. (2017). A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, page 214262.

Goldman, M. S. (2009). Memory without Feedback in a Neural Network. *Neuron*, 61(4):621–634.

Goldman-Rakic, P. S. (1987). Circuitry of primate prefrontal cortex and regulation of behavior by representational memory. *Comprehensive Physiology*, pages 373–417.

Guo, Z. V., Li, N., Huber, D., Ophir, E., Gutnisky, D., Ting, J. T., Feng, G., and Svoboda, K. (2014). Flow of cortical activity underlying a tactile decision in mice. *Neuron*, 81(1):179–194.

Harvey, C. D., Coen, P., and Tank, D. W. (2012). Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature*, 484(7392):62–68.

Humphries, M. D. (2020). Strong and weak principles of neural dimension reduction. *Neurons, Behavior, Data analysis, and Theory*, 5.

Inagaki, H. K., Fontolan, L., Romani, S., and Svoboda, K. (2019). Discrete attractor dynamics underlies persistent activity in the frontal cortex. *Nature*, 566(7743):212–217.

Jang, E., Gu, S., and Poole, B. (2016). Categorical reparameterization with gumbel-softmax. *arXiv preprint arXiv:1611.01144*.

Kaelbling, L. P., Littman, M. L., and Cassandra, A. R. (1998). Planning and acting in partially observable stochastic domains. *Artificial Intelligence*, 101(1-2):99–134.

Kandel, E. R., Schwartz, J. H., Jessell, T. M., Siegelbaum, S., Hudspeth, A. J., Mack, S., et al. (2000). *Principles of neural science*, volume 4. McGraw-hill New York.

Killcross, S. and Coutureau, E. (2003). Coordination of actions and habits in the medial prefrontal cortex of rats. *Cerebral Cortex*, 13:400–408.

Kingma, D. P. and Ba, J. L. (2015). Adam: A method for stochastic optimization. In *3rd International Conference on Learning Representations, ICLR 2015 - Conference Track Proceedings*.

Kingma, D. P. and Welling, M. (2014). Auto-encoding variational bayes. In *2nd International Conference on Learning Representations, ICLR 2014 - Conference Track Proceedings*. International Conference on Learning Representations, ICLR.

Kirsch, A., Lyle, C., and Gal, Y. (2021). Unpacking information bottlenecks: Unifying information-theoretic objectives in deep learning. *arxiv*.

Kobak, D., Brendel, W., Constantinidis, C., Feierstein, C. E., Kepecs, A., Mainen, Z. F., Qi, X. L., Romo, R., Uchida, N., and Machens, C. K. (2016). Demixed principal component analysis of neural population data. *eLife*, 5(APRIL2016):1–36.

Krakauer, J. W., Ghazanfar, A. A., Gomez-Marin, A., MacIver, M. A., and Poeppel, D. (2017). Neuroscience needs behavior: Correcting a reductionist bias. *Neuron*, 93:480–490.

Lake, B. M., Ullman, T. D., Tenenbaum, J. B., and Gershman, S. J. (2017). Building machines that learn and think like people. *Behavioral and Brain Sciences*, 40.

Lengyel, M. and Dayan, P. (2008). Hippocampal contributions to control the third way. *Advances in Neural Information Processing Systems*, pages 1–8.

Lesort, T., Díaz-Rodríguez, N., Goudou, J. F., and Filliat, D. (2018). State representation learning for control: An overview. *Neural Networks*, 108:379–392.

Lewicki, M. S. (2002). Efficient coding of natural sounds. *Nature Neuroscience*, 5:356–363.

Lima, S. Q. and Miesenböck, G. (2005). Remote control of behavior through genetically targeted photostimulation of neurons. *Cell*, 121(1):141–152.

Linderman, S. W., Johnson, M. J., Miller, A. C., Adams, R. P., Blei, D. M., and Paninski, L. (2017). Bayesian learning and inference in recurrent switching linear dynamical systems. In *Proceedings of the 20th International Conference on Artificial Intelligence and Statistics, AISTATS 2017*, pages 914–922. PMLR.

Ma, T. and Hermundstad, A. M. (2022). A vast space of compact strategies for highly efficient decisions. *bioRxiv*, page 2022.08.10.503471.

Ma, W. J., Husain, M., and Bays, P. M. (2014). Changing concepts of working memory. *Nature Neuroscience*, 17:347–356.

Machado, A. S., Darmohray, D. M., Fayad, J., Marques, H. G., and Carey, M. R. (2015). A quantitative framework for whole-body coordination reveals specific deficits in freely walking ataxic mice. *eLife*, 4.

Machens, C. K., Romo, R., and Brody, C. D. (2005). Flexible control of mutual inhibition: a neural model of two-interval discrimination. *Science*, 307(5712):1121–1124.

Machens, C. K., Romo, R., and Brody, C. D. (2010). Functional, but not anatomical, separation of "what" and "when" in prefrontal cortex. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 30:350–60.

MacKay, D., Kay, D., and Press, C. U. (2003). *Information Theory, Inference and Learning Algorithms*. Cambridge University Press.

Maddison, C. J., Mnih, A., and Teh, Y. W. (2016). The concrete distribution: A continuous relaxation of discrete random variables. *arXiv preprint arXiv:1611.00712*.

Mahadevan, S. (2009). Learning Representation and Control in Markov Decision Processes: New Frontiers. *Foundations and Trends in Machine Learning*, 1(4):403–565.

Maheswaranathan, N., Williams, A. H., Golub, M. D., Ganguli, S., and Sussillo, D. (2019). Universality and individuality in neural dynamics across large populations of recurrent networks. *Advances in neural information processing systems*, 2019:15629–15641.

Mante, V., Sussillo, D., Shenoy, K. V., and Newsome, W. T. (2013). Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84.

Mastrogiuseppe, F. and Ostojic, S. (2018). Linking connectivity, dynamics, and computations in low-rank recurrent neural networks. *Neuron*, 99:609–623.e29.

Mathis, A., Mamidanna, P., Cury, K. M., Abe, T., Murthy, V. N., Mathis, M. W., and Bethge, M. (2018). Deeplabcut: markerless pose estimation of user-defined body parts with deep learning. *Nature Neuroscience 2018 21:9*, 21:1281–1289.

McCallum, A. K. (1996). *Reinforcement Learning with Selective Perception and Hidden State*. PhD thesis, University of Rochester.

Miller, E. K. and Cohen, J. D. (2001). An integrative theory of prefrontal cortex function. *Annual Review of Neuroscience*, 24:167–202.

Miller, G. A. (1956). The magical number seven, plus or minus two: Some limits on our capacity for processing information. *Psychological review*, 63(2):81.

Montañez, G. D., Amizadeh, S., and Laptev, N. (2015). Inertial hidden Markov models: Modeling change in multivariate time series. In *Proceedings of the National Conference on Artificial Intelligence*, volume 3, pages 1819–1825.

Motiwala, A., Soares, S., Atallah, B. V., Paton, J. J., and Machens, C. K. (2022). Efficient coding of cognitive variables underlies dopamine response and choice behavior. *Nature Neuroscience*, 25:738–748.

Młynarski, W., Hledík, M., Sokolowski, T. R., and Tkačik, G. (2021). Statistical analysis and optimality of neural systems. *Neuron*, 109:1227–1241.e5.

Olshausen, B. A. and Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609.

Orhan, A. E. and Ma, W. J. (2019). A diverse range of factors affect the nature of neural representations underlying short-term memory. *Nature Neuroscience*, 22(2):275–283.

Pardo-Vazquez, J. L., de Saa, J. R. C., Valente, M., Damião, I., Costa, T., Vicente, M. I., Mendonça, A. G., Mainen, Z. F., and Renart, A. (2019). The mechanistic foundation of weber's law. *Nature Neuroscience*.

Passingham, R. E. (1985). Memory of monkeys (macaca mulatta) with lesions in prefrontal cortex. *Behavioral Neuroscience*, 99:3–21.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., DeVito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. (2019). Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc.

Petreska, B., Yu, B. M., Cunningham, J. P., Santhanam, G., Ryu, S. I., Shenoy, K. V., and Sahani, M. (2011). Dynamical segmentation of single trials from population neural data. *Advances in Neural Information Processing Systems 24: 25th Annual Conference on Neural Information Processing Systems 2011, NIPS 2011*, pages 1–9.

Pezzulo, G. (2012). An active inference view of cognitive control. *Frontiers in Psychology*, 3.

Poupart, P. and Boutilier, C. (2003). Value-directed compression of POMDPs. In *Advances in Neural Information Processing Systems*.

Rigotti, M., Barak, O., Warden, M. R., Wang, X.-J., Daw, N. D., Miller, E. K., and Fusi, S. (2013). The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497:1–6.

Romo, R., Brody, C. D., Hernandez, A., and Lemus, L. (1999). Neuronal correlates of parametric working memory in the prefrontal cortex. *Nature*, 399(6735):470–473.

Romo, R., Brody, C. D., Hernández, A., and Lemus, L. (2016). Single-neuron spike train recordings from macaque prefrontal cortex during a somatosensory working memory task. *CRCNS.org*.

Romo, R., Hernández, A., Zainos, A., Lemus, L., and Brody, C. D. (2002). Neuronal correlates of decision-making in secondary somatosensory cortex. *Nature neuroscience*, 5(11):1217–1225.

Romo, R. and Salinas, E. (2003). Flutter discrimination: Neural codes, perception, memory and decision making. *Nature Reviews Neuroscience*, 4:203–218.

Roweis, S. and Ghahramani, Z. (1999). A Unifying Review of Linear Gaussian Models. *Neural Computation*, 11:305–345.

Russin, J., O'reilly, R. C., and Bengio, Y. (2020). Deep learning needs a prefrontal cortex. In *ICLR 2020 Workshop on Bridging AI and Cognitive Science*.

Russo, A. A., Bittner, S. R., Perkins, S. M., Seely, J. S., London, B. M., Lara, A. H., Miri, A., Marshall, N. J., Kohn, A., Jessell, T. M., Abbott, L. F., Cunningham, J. P., and Churchland, M. M. (2018). Motor cortex embeds muscle-like commands in an untangled population response. *Neuron*, 97:953–966.e8.

Salinas, E., Hernández, A., Zainos, A., and Romo, R. (2000). Periodicity and firing rate as candidate neural codes for the frequency of vibrotactile stimuli. *Journal of Neuroscience*, 20:5503–5515.

Schall, J. D. (2002). The neural selection and control of saccades by the frontal eye field. *Philosophical Transactions of the Royal Society B: Biological Sciences*, 357:1073.

Schuessler, F., Mastrogiuseppe, F., Dubreuil, A., Ostojic, S., and Barak, O. (2020). The interplay between randomness and structure during learning in rnns. *arXiv*, 33.

Schultz, W., Dayan, P., and Montague, P. R. (1997). A neural substrate of prediction and reward. *Science*, 275:1593–1599.

Seger, C. A. and Miller, E. K. (2010). Category learning in the brain. *Annual Review of Neuroscience*, 33:203–219.

Seung, H. S. (1996). How the brain keeps the eyes still. *Proceedings of the National Academy of Sciences of the United States of America*, 93:13339–13344.

Shadlen, M. N. and Newsome, W. T. (2001). Neural basis of a perceptual decision in the parietal cortex (area lip) of the rhesus monkey. *Journal of Neurophysiology*, 86:1916–1936.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423.

Simoncelli, E. P. and Olshausen, B. A. (2001). Natural image statistics and neural representation. *Annual Reviews of Neuroscience*, 24:1193–1216.

Singh, S. P., Jaakkola, T., and Jordan, M. I. (2005). Reinforcement Learning with Soft State Aggregation. *Nips*.

Skinner, B. (1938). *The Behavior of Organisms: An Experimental Analysis*. Appleton-Century.

Sohn, H., Narain, D., Meirhaeghe, N., and Jazayeri, M. (2019). Bayesian computation through cortical latent dynamics. *Neuron*, 103:934–947.e5.

Song, H. F., Yang, G. R., and Wang, X. J. (2016). Training Excitatory-Inhibitory Recurrent Neural Networks for Cognitive Tasks: A Simple and Flexible Framework. *PLoS Computational Biology*, 12(2):1–30.

Song, H. F., Yang, G. R., and Wang, X. J. (2017). Reward-based training of recurrent neural networks for cognitive and value-based tasks. *eLife*, 6:1–24.

Sreenivasan, K. K. and D'Esposito, M. (2019). The what, where and how of delay activity. *Nature Reviews Neuroscience*, page 1.

Stoianov, I., Genovesio, A., and Pezzulo, G. (2016). Prefrontal goal codes emerge as latent states in probabilistic value learning. *Journal of Cognitive Neuroscience*, 28:140–157.

Strouse, D. and Schwab, D. J. (2017). The deterministic information bottleneck. *Neural computation*, 29(6):1611–1630.

Sussillo, D. and Barak, O. (2013). Opening the black box: Low-dimensional dynamics in high-dimensional recurrent neural networks. *Neural Computation*, 25(3):626–649.

Sussillo, D., Churchland, M. M., Kaufman, M. T., and Shenoy, K. V. (2015). A neural network that finds a naturalistic solution for the production of muscle activity. *Nature Neuroscience*, 18(7):1025–1033.

Sutton, R. S. and Barto, A. G. (2018). *Reinforcement learning: An introduction*. The MIT Press.

Szczepanski, S. M. and Knight, R. T. (2014). Insights into human behavior from lesions to the prefrontal cortex. *Neuron*, 83:1002–1018.

Tishby, N., Pereira, F. C., and Bialek, W. (2000). The information bottleneck method.

Tishby, N. and Polani, D. (2011). Information Theory of Decisions and Actions. *Perception-Action Cycle: Models, Architecture and Hardware*, pages 601–636.

Todorov, E. (2009). Efficient computation of optimal actions. *Proceedings of the National Academy of Sciences of the United States of America*, 106(28):11478–83.

Tolman, E. C. (1948). Cognitive maps in rats and man. *Psychological Review*, 55:189–208.

Van Rossum, G. and Drake Jr, F. L. (1995). *Python reference manual*. Centrum voor Wiskunde en Informatica Amsterdam.

Vyas, S., Golub, M. D., Sussillo, D., and Shenoy, K. V. (2020). Computation through neural population dynamics. *Annual Review of Neuroscience*, 43:249–275.

Wang, J. X., Kurth-Nelson, Z., Kumaran, D., Tirumala, D., Soyer, H., Leibo, J. Z., Hassabis, D., and Botvinick, M. (2018). Prefrontal cortex as a meta-reinforcement learning system. *Nature Neuroscience*, 21(6):860–868.

Wang, X.-J. (2002). Probabilistic decision making by slow reverberation in cortical circuits. *Neuron*, 36:955–968.

Wassum, K., Cely, I., Maidment, N., and Balleine, B. (2009). Disruption of endogenous opioid activity during instrumental learning enhances habit acquisition. *Neuroscience*, 163(3):770–780.

Whittington, J. C., Muller, T. H., Mark, S., Chen, G., Barry, C., Burgess, N., and Behrens, T. E. (2020). The tolman-eichenbaum machine: Unifying space and relational memory through generalization in the hippocampal formation. *Cell*, 183:1249–1263.e23.

Wiegert, J. S., Mahn, M., Prigge, M., Printz, Y., and Yizhar, O. (2017). Silencing neurons: Tools, applications, and experimental constraints. *Neuron*, 95:504–529.

Wiltschko, A. B., Johnson, M. J., Iurilli, G., Peterson, R. E., Katon, J. M., Pashkovski, S. L., Abraira, V. E., Adams, R. P., and Datta, S. R. (2015). Mapping sub-second structure in mouse behavior. *Neuron*, 88:1121–1135.

Wolpert, D. H., Libby, E., Grochow, J., and DeDeo, S. (2015). Optimal high-level descriptions of dynamical systems. *arXiv*, pages 1–33.

Yang, G. R., Joglekar, M. R., Song, H. F., Newsome, W. T., and Wang, X.-J. (2019). Task representations in neural networks trained to perform many cognitive tasks. *Nature Neuroscience*, 22(2):297–306.

itqb nova

ITQB NOVA, Avenida da República, 2780-157 Oeiras, Portugal | Tel +351 214 469 100 | itqb.unl.pt