THE UNIVERSITY
of EDINBURGH

Statistical modelling of neuronal population activity: from data analysis to network function

Martino Sorbaro Sindaci

Doctor of Philosophy

Institute for Adaptive and Neural Computation
School of Informatics
University of Edinburgh

2018

Abstract

The term *statistical modelling* refers to a number of abstract models designed to reproduce and understand the statistical properties of the activity of neuronal networks at the population level. Large-scale recordings by multielectrode arrays (MEAs) have now made possible to scale their use to larger groups of neurons. The initial step in this work focused on improving the data analysis pipeline that leads from the experimental protocol used in dense MEA recordings to a clean dataset of sorted spike times, to be used in model training. In collaboration with experimentalists, I contributed to developing a fast and scalable algorithm for spike sorting, which is based on action potential shapes and on the estimated location for the spike. Using the resulting datasets, I investigated the use of restricted Boltzmann machines in the analysis of neural data, finding that they can be used as a tool in the detection of neural ensembles or low-dimensional activity subspaces. I further studied the physical properties of RBMs fitted to neural activity, finding they exhibit signatures of criticality, as observed before in similar models. I discussed possible connections between this phenomenon and the "dynamical" criticality often observed in neuronal networks that exhibit emergent behaviour. Finally, I applied what I found about the structure of the parameter space in statistical models to the discovery of a learning rule that helps long-term storage of previously learned memories in Hopfield networks during sequential learning tasks. Overall, this work aimed to contribute to the computational tools used for analysing and modelling large neuronal populations, on different levels: starting from raw experimental recordings and gradually proceeding towards theoretical aspects.

Lay Summary

In the brain, computation happens thanks to the concerted activity of a large number of neurons, which exchange electrical signals. To study their coordinated behaviour, recording these signals, for example using arrays of electrodes, is essential. When dealing with these recordings, it is often hard to distinguish the "voices" of individual neurons from all the others and from the background. In the first chapter of this thesis, I developed a computational method that addresses this problem in a fast way for modern, large, electrode arrays. Once the data was obtained in this way, it was used to understand how the neural population encodes and processes information. I adopted a "statistical" approach, whereby the neurons and their interactions are characterised by studying the probability of each possible pattern of activity. I found that the model I used can correctly reproduce the neurons' activity and can be used to find which groups of cells work together and have similar responses. As was previously observed in other cases, these models always seem to lay near the transition point between order and disorder, and I showed that this behaviour occurs regardless of the characteristics of the underlying neural activity. This finding contributes to an ongoing debate over how neural activity encodes information. In the last chapter, I considered a model of the same class, traditionally used as a paradigm for memory, and studied an approach to learning which helps avoiding overwriting previously-stored information. Overall, this thesis discusses tools for analysing and modelling large populations of neurons, from experimental recordings to theoretical results, applying computational and machine learning techniques to problems in neuroscience.

Declaration

I declare that this thesis was composed by myself and the work contained herein is my own, except where explicitly stated otherwise in the text; and that this work has not been submitted for any degree or professional qualification outwith the EuroSPIN doctoral programme, except as specified. Where articles and manuscripts were included, my contributions were clearly stated.

(*Martino Sorbaro Sindaci*)

Table of Contents

Chapter 0

Introduction

> I used to think that the brain was the most wonderful organ in my body. Then I realized who was telling me this.
>
> Emo Philips

An oft-heard quote is that *the brain is the most complex object in the universe*. To argue for or against such a claim, we would first need to precisely define what we mean by *complex*. A human brain contains an estimated 86 billion neurons, connected to one another by some 10^{14} synapses, arranged according to a sophisticated wiring scheme (called by some the "connectome" [Sporns et al., 2005]). The inner workings of neurons and synapses are themselves a very large area of ongoing research, both exhibiting a large variety of behaviour and functions. The brain can be studied, and *is* actively studied, from the level of single molecules to the level of the whole nervous system. So, the brain is complex in the sense that it is large, diverse, interconnected. It is also responsible for functions we don't fully understand. Studying it requires a diverse set of experimental tools that have developed over more than a century and cover molecular biology, cell biology, systems biology, electrophysiology, neurology, and even chemistry and biophysics.

But the brain is complex also in another sense, the "complex systems" sense: it exhibits emergent properties. Much of the behaviour it displays is very hard to describe in reductionist terms, merely linking it to the properties of individual neurons and synapses. Even if neurons were extremely simple systems (and they aren't), their collective behaviour could still be fascinatingly complex. Making

sense of this form of complexity required the development of another, new, set of tools, parallel to experimental research: mathematical modelling, information theory, nonlinear dynamics, signal processing, computational science.

Approaches to neuroscience Historically, scientists approached the brain (and the nervous system in general) in two ways that required different tools and different understanding: one focused on the anatomy and structure of neural systems, and another stemmed from the discovery of electrical activity in nerves.

The former culminated with the discoveries of Santiago Ramón y Cajal in the late XIX century, who first exposed the anatomy of neurons [Ramón y Cajal, 1904]. This breakthrough was primarily a methodological one: we owe the discovery of neurons to Camillo Golgi's staining technique, that allowed Ramón y Cajal to observe them under his microscope and make his beautiful drawings. Golgi and Ramón y Cajal, scientific rivals, went on to share the 1906 Nobel Prize in medicine or physiology.

An initial understanding that electrical signal played a role in the nervous system, however, had previously been developed starting with Luigi Galvani. His experiments on "animal electricity", showed, already in the late XVIII century, that muscles move when electrically stimulated through the nerves. Research about the transmission of electrical impulses within the nervous system continued for two centuries, and culminated with the discovery of the mechanisms behind action potentials and the electrical properties of the membrane by A.L. Hodgkin and A.F. Huxley in the post-world war II period [Hodgkin and Huxley, 1952]. Their discipline is now called electrophysiology (figure 1).

These two aspects, respectively related to the structure and the activity, are still part of today's neuroscience. But the field has gone much farther, thanks to the development of new tools: some of them are direct descendants of these early methods of experimental investigation, but some others arose in the framework of entirely new perspectives on the brain — such as an increased interest in computation and information, which made neuroscience, in the last few decades, an excitingly multidisciplinary science.

As for all new sciences that emerge, the discovery of new tools has driven the advancement of neuroscience. But "tools" should be interpreted in the broadest sense, not only including experimental techniques, but also the application of novel theoretical ideas. This introductory chapter will continue with a very

general introduction to these tools, and progressively focus on the ideas and breakthroughs that are necessary to motivate and understand the contents of this thesis. At the end of this introduction, I will present the line of thought that has led to the work presented in the other chapters, and that links them to each other.

0.1 Electrophysiology

Electrophysiology is the branch of science that focuses on the electrical properties of neural systems. At the scale of neurons, the techniques involved can be roughly categorised as *intracellular* or *extracellular* recordings. In the former case, an electrode is inserted in the cell in order to keep the potential difference between its inside and outside constant, and measure the currents through the membrane (voltage clamp), or to inject arbitrary currents and measure the corresponding membrane potentials (current clamp). The invention of patch clamp in the 1970s and its subsequent development allowed for significant improvement of intracellular recordings, including a much increased survival of the cell during the experiment and single-channel recording [Neher and Sakmann, 1976].

Extracellular recordings, conversely, consist in placing electrodes in the immediate vicinity of the neurons one wants to measure, the advantage being that this leaves the cells intact, is easier to perform, and can easily be applied to many neurons at the same time. The advent of multi-electrode arrays (MEA) allowed extracellular recording many neurons simultaneously, opening up the possibility of studying neural dynamics and neural coding at a new level. MEAs are planar chips over which a number of micro-electrodes is placed, all recording extracellular potential at all times. They can be used for *in vivo*, *ex vivo* or *in vitro* recordings. The first MEAs were developed in 1972 and, independently, in 1977 [Thomas et al., 1972, Gross et al., 1977]: they started with around 30 electrodes, each around 10 µm in diameter, placed approximately 100 µm apart from each other. For the first time, not only the individual properties of many neurons could be studied, but also their dynamical interaction, and the emergent properties of the neural code in small sensory networks. The number of electrodes in a current state-of-the-art MEA can vary significantly, but can reach the thousands. The spacing between electrodes, on the other hand, decreased, improving the spatial resolution. As a consequence, the modern MEA is sufficiently advanced

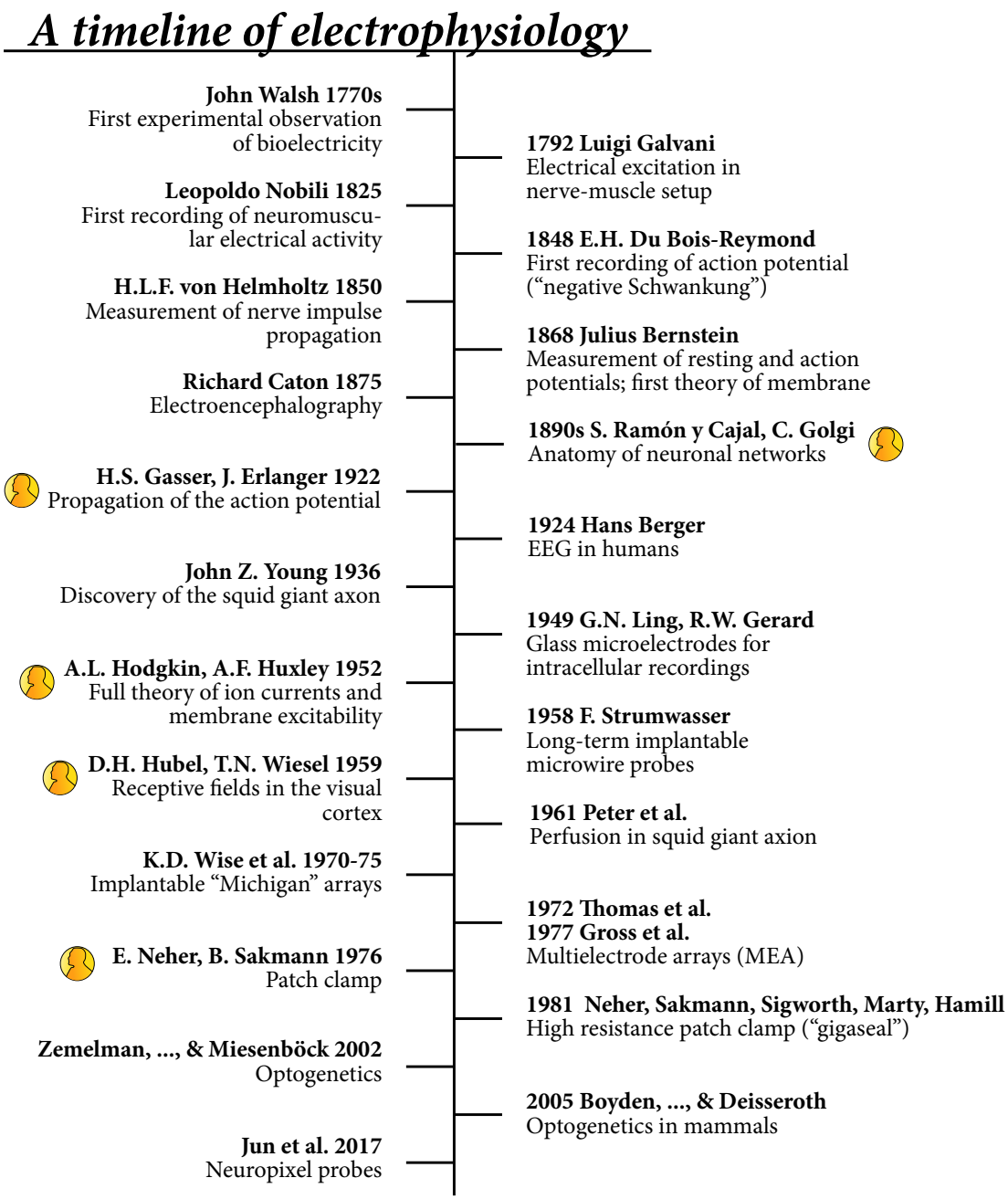


Figure 1: A timeline (by no means complete) of electrophysiological methods and discoveries. In-depth sources can be consulted for the history of patch clamping [Verkhratsky and Parpura, 2014], MEAs [Pine, 2006], and implants [Cheung, 2007].

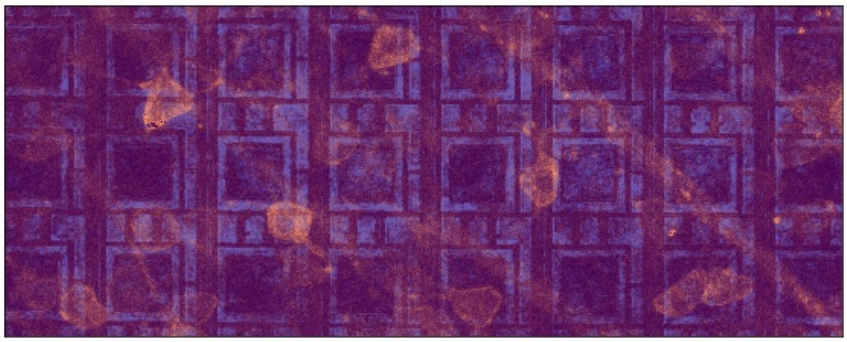


Figure 2: Mouse retinal ganglion cells on one of the BioCam chips used for the collection of the data used in this thesis, photographed by two-photon microscopy. Somas and axons are clearly visible. The image is a false-colours composition of a microphotograph of the chip itself (purple) and one of the retina placed on it (orange). For reference, the spacing between electrodes is 42 µm.

to allow simultaneous recording of a majority of the neurons in a certain systems, such as the retina of a young mouse, or a culture of thousands of neurons [Hilgen et al., 2017a].

0.1.1 Spike sorting

The improvement in spatial resolution and signal quality that MEAs have seen in the last four decades was, of course, the result of a large amount of research in microelectronics [Pine, 2006]. With larger and larger arrays, significant challenges started to arise on the data analysis side too. Modern software methods analyse the raw voltage trace recorded by each electrode and, as a first step, detect the presence of action potentials, their timings, and their amplitudes and electrical shapes. However, depending on the size of the electrodes, the conductance and capacitance of the extracellular medium, and the properties of the array, each electrode is likely to record the action potentials of multiple neighbouring neurons. When the research question is about the firing patterns of individual neurons, identifying which spikes belong to which neurons may be a critical step. This operation is called *spike sorting*. A general presentation of the challenges involved can be found in [Rey et al., 2015]. Traditional spike sorting typically relies on the assumption that spikes emitted by a given neuron share a similar

"signature": their amplitude, duration of depolarisation, intensity of hyperpolarisation are similar. Under this assumption, the detected spike shapes can be sorted by clustering in the space of spike waveforms, often preceded by dimensionality reduction. [Bestel et al., 2012] includes a list of such methods, and more complex ones, which are applicable to single-electrode extracellular recordings or, electrode by electrode, to MEAs where the inter-electrode spacing is large.

In modern dense arrays, however, the electrodes are so closely spaced that multiple ones may record from the same neuron, leading to duplicates in the results or spurious correlations in the signals. Detecting spikes and sorting them separately for each electrode's voltage trace is, therefore, not a viable option. The methods to overcome this problem are still an active area of research, and the last few years have seen several laboratories experimenting with a variety of solutions. Table 1 lists the ones I am aware of at the time of writing this thesis. [Hennig et al., 2018] reviews the problem in more detail, and offers insight into perspectives in the near future.

Other than dealing with the problem of source separation at high densities, the algorithms belonging to this new generation of spike sorting methods have a common interest in automation and scaling (essential when working with the large amounts of data produced by modern probes). There are two main approaches: one involves going through the whole dataset and learning "templates", the signatures a neuron leaves on the electrode traces when it spikes, then looking in the data again, matching each event to a template [see citations in table 1]. This can be slower, since it involves multiple passes over the recording, but deals very well with spikes that overlap in time. A second approach assumes simultaneous events in neighbouring channels are due to the same spike, and tries to estimate a location for its source, then clusters both according to location and to spike waveform. This can be done in a very fast way, as shown by chapter 1 of this thesis. Before presenting the work done with these recordings, I will introduce our reference system, the retina, in section 0.3.

0.2 Computational approaches

The birth of the computer started a revolution in neuroscience, on different levels. Theoretically speaking, the work of Turing and others on the foundations of computer science was the beginning of a great reflection on the concept of

Name and reference	Method	Notes
Kilosort [Pachitariu et al., 2016] github.com/cortex-lab/KiloSort	TM	MATLAB based; semi-automated final curation.
YASS [Lee et al., 2017] yass.readthedocs.io	TM	Neural network-based detection; outlier triaging; template matching; clustering.
Herding Spikes [Hilgen et al., 2017b] github.com/mhhennig/HS2	SL+D	Fast and scalable; tested on multiple array geometries
MountainSort [Chung et al., 2017] github.com/flatironinstitute/mountainsort	D	Fully automatic; scalable; graphical user interface; unique clustering method
JRCLUST [Jun et al., 2017a] jrclust.org	SL+D	Probe drift correction.
SpyKING CIRCUS [Yger et al., 2018] spyking-circus.rtfd.org	TM	Tested on many datasets; robust to overlapping spikes; graphical user interface.

Table 1: Summary of the most recent spike sorting methods developed for large, dense arrays. For a summary of older algorithms (mostly for smaller, sparser arrays) see [Bestel et al., 2012]. TM = Template Matching; SL = Spike Localisation; D = Density-based clustering. Table reproduced from [Hennig et al., 2018].

computation, and sparked the obvious interest in the question of whether the brain is analogous to a computer, and *in what sense* it is; a debate that continues to this day. More practically, computers enabled neuroscience research in two related ways: the analysis of large amounts of experimental data, and the design of models of different levels of abstraction. "Modelling" can in turn mean several things: a quest for algorithms that imitate certain *functions* performed by neural networks, with only a loose connection to their implementation; detailed *reconstruction and simulation* of biological processes; mimicking of *statistical* properties of biological systems, regardless of their actual structure. David Marr described three levels of interest in this discipline, which can work with

different tools and different aims: the computational level (what problems is the system solving? why?), the algorithmic level (what calculations are performed in order to solve the problem?), and the implementation level (how are these calculations performed by biological circuits?) [Marr and Poggio, 1976].

0.2.1 Dynamical modelling of neurons and networks

When, in 1952, Hodgkin and Huxley published their detailed account of membrane potentials and trans-membrane currents in the squid giant axon, they also presented the first full mathematical model of the generation of action potentials in neurons: a set of differential equations that gave the first quantitative description of the opening and closing of ion channels, the membrane potential, and the firing of the action potential [Hodgkin and Huxley, 1952]. The work they inspired developed in two directions: one that studied and simulated, in more and more detail, single neurons and their interactions in small groups [De Schutter, 2000], and one that sacrificed neuron-level accuracy and focused on the emergent behaviour of larger populations of neurons [Sejnowski et al., 1988].

Computational neuroscience quickly became a large field of research that would be quite overambitious to summarise here. It covers all three of Marr's levels and, most notably, it does not disregard the biological implementation. Today, it is an increasingly popular field, boosted by ambitious endeavours such as the European Union's Human Brain Project [Amunts et al., 2016] and the BRAIN initiative in the USA [Markoff and Gorman, 2013]. Computational techniques are now employed from the molecular level up to the whole brain level.

Learning is, in my opinion, the point of interest that has attracted more focus in recent years, and the one that offers the most important unsolved problems — at multiple scales, from synapses to systems [Roelfsema and Holtmaat, 2018]. It is also the concept that can be seen overarching both neuroscience and artificial intelligence, but the one where they currently disagree the most, which makes research in this area particularly interesting [Marblestone et al., 2016].

0.2.2 Artificial neural networks

In parallel with neuroscientists interested in the mathematical modelling of biological processes, other researchers also tried to investigate *intelligence* independently of its biological substrate. Warren McCulloch and Walter Pitts were the

first to introduce an abstract, high-level model of artificial neuron, with only its computational properties in mind. The McCulloch-Pitts neuron was a simple unit that performed a weighted sum of its binary inputs, and thresholded the results to return a binary output [McCulloch and Pitts, 1943]. This line of research slowly proceeded until the 60s, with Frank Rosenblatt's Perceptron [Rosenblatt, 1958], but later saw a period of inactivity, due to its limitations. Research in what was by then called artificial intelligence started again in the 1980s, but had a boom and gained widespread popularity both in theory and applications only in the 2010s, when the availability of much larger computational power enabled the advent of deep learning [LeCun et al., 2015].

Although the motivation behind these models was initially to understand human intelligence, the relationship between this particular field and neuroscience has been comparatively sparse. Artificial intelligence shares the first of Marr's levels with neuroscience, but works independently in terms of the other two, developing its own algorithms and its own implementation. Moreover, much research in this field has been devoted to technological applications. Recently, however, a few similarities have been observed between information processing in deep artificial networks and in brain networks, even if the two remain clearly different, for example in how they learn [Barrett et al., 2019, Hassabis et al., 2017]. Although computational neuroscience and machine learning have largely pursued different aims, the last few years have seen an interest in finding contact points, with some early development of deep networks in the direction of (relatively) better biological realism [Whittington and Bogacz, 2019]. This may, in the near future, open a new and exciting interdisciplinary field [Marblestone et al., 2016]. This thesis itself employs various techniques that were developed in a more general machine learning context, as I will discuss in the conclusions.

0.2.3 Statistical modelling of neural activity

What I call "statistical modelling" is a third approach, where the model does not try to provide a detailed simulation of a neuronal system, nor it tries to perform a function in a way independent of the actual implementation. Conversely, it tries to reproduce a given set of properties of a network at the *algorithmic* level. It is called *statistical* because it seeks to attribute a probability value to the state of the network at a given time, conditional, if relevant, to the stimulus or the

behaviour. Importantly, I'm talking about a data-driven approach, which relies on observing the actual dynamics of a neural network, reproducing it in order to understand it.

Here is a simple example: suppose we formulate the hypothesis that a population of neurons implements a pure rate coding: in other words, they encode information through their firing rates only. If this is the case, and the firing rates change on time scales much larger than the typical inter-spike interval, a single-neuron spike train should look like a Poisson process with a stimulus-determined, and slowly varying, probability of activation [Dayan and Abbott, 2001]. This is an experimentally verifiable prediction of our hypothesis. If it turns out to be incorrect, we may conclude that the information is encoded in a different way, or that there are other biological reasons why the precise spike timings are not as random as we thought.

In general, the aim of statistical modelling is typically to understand some properties of the neural code. For example, do neurons encode information separately, or is the synchrony between neurons also a source of information [Nirenberg and Latham, 2003]? For sensory areas, one may build a model that, subject to a given stimulus, produces the same spike trains as the correspondent biological network, thereby imitating its encoding properties. By judging whether a given model is sufficient to satisfactorily reproduce a network's computation, one can then ask questions such as, is interaction between its neurons essential for the encoding? are these interactions simply due to common input? do the neurons act as a linear filter of the input? how different are these filters? how important is a neuron's own past spiking history?

An early example of an abstract model of neural responses that follows this paradigm is the so called linear-nonlinear-Poisson model (LNP). Given a time-dependent stimulus, the model applies a linear filter, and the result is passed through a nonlinear function; the output is then used as the rate of a Poisson process. The linear filter is directly related to the spike-triggered stimulus average (STA), and is therefore an interpretable quantity — the average stimulus that causes the neuron to spike. In other versions of these "cascade models", the model may account for additional factors, such as the internal state of the neuron itself, and the dynamics of the rest of the network [Berry and Meister, 1998, Chichilnisky, 2001, Keat et al., 2001].

A further generalisation of this idea can be achieved incorporating a Gener-

alised Linear Model (GLM) into this approach. GLMs, introduced in the 1970s [Nelder and Wedderburn, 1972, McCullagh, 1983] are an extension of linear regression to non-Gaussian outcomes, such as when the target variables are confined to a restricted range. In neuroscience, GLMs found an early application in the prediction of a rat's position from the firing rates of hippocampal cells [Frank et al., 2000], but are particularly popular for the prediction and characterisation of sensory responses, in a way first developed by L. Paninski [Paninski, 2004]. Comparing the fitness of various models, which include or don't include certain interaction terms, allows us to understand the relative importance of these terms in determining the probability of firing [Truccolo et al., 2005]. With this method, [Pillow et al., 2008] found that models including correlation terms perform 20% better than independent coding when applied to retinal ganglion cells encoding visual stimuli, even accounting for the presence of stimulus-driven correlations. In their instance of the model, in addition to the spatio-temporal linear filter applied to the stimulus in LNP, the neuron's own spiking history is convolved with a linear filter and added to the input. When modelling multiple neurons simultaneously, an interaction term is also added, again in the form of the filtered spike trains, and other arbitrary terms can be considered, based on what one is interested in modelling. Then, analogously to the LNP case, the input goes through a nonlinear response function, before giving the probability of firing. GLMs were found to provide an accurate but inexpensive model of the neurons' function.

Maximum entropy models For a general approach to statistical modelling, [Martignon et al., 2000, Schneidman et al., 2003] introduced the usage of maximum entropy models. The idea is simply to decide what relevant properties of the activity distribution we want to model, and to derive a form for a probability distribution that exactly fits them, while leaving all the other aspects *as undetermined as possible* — hence the term "maximum entropy", which refers to a method that goes back to E. T. Jaynes's ideas in statistical physics [Jaynes, 1957]. This is usually done on the system described in terms of binary variables $\sigma_1(t), ..., \sigma_N(t)$: here, $\sigma_i(t)$ takes the value 1 if the i-th neuron fires between time t and $t + \delta t$, and 0 otherwise (see figure 1 of the paper included in chapter 3). The bin size δt is chosen so that the activity of the network at time t has negligible correlations with that at time $t + \delta t$. Then, $\boldsymbol{\sigma}(t)$ can be interpreted as the "codeword" that the N neurons are transmitting at time t. To understand

the encoding scheme, we are interested in modelling the probability distribution $P(\boldsymbol{\sigma})$, possibly conditional on the stimulus or on previous activity. In general, maximum entropy distributions over this system take the form

$$P(\boldsymbol{\sigma}) = \frac{1}{Z} \exp\left(\sum_k a_k f_k(\boldsymbol{\sigma})\right),$$

where Z is included for normalisation, and the functions f_k are the properties of the distributions that we want to constrain. If, for example, we want the model to exactly reproduce the values of $\langle\sigma_i\rangle$ and $\langle\sigma_i\sigma_j\rangle$ (equivalent to firing rates and pairwise correlations), we get the pairwise maximum entropy (PME) model

$$P(\boldsymbol{\sigma}) = \frac{1}{Z} \exp\left(\sum_{i=1}^{N} a_i\sigma_i + \sum_{j\neq i} J_{ij}\sigma_i\sigma_j\right),$$

in which $\boldsymbol{a}$ and the matrix J are parameters to be determined.

The PME was the model adopted by two seminal 2006 papers [Schneidman et al., 2006, Shlens et al., 2006], which introduced this approach in computational neuroscience. Schneidman et al. observed that an independent model (in which only firing rates are considered) fails to account for the activity distribution, whereas a PME model explains 90% of the information in the data. They conclude that observed higher-order interactions are simply explained by pairwise interactions in cultured cortical neurons and in retinal cells. Similarly, Shlens et al. worked on retinal ganglion cells, but limited to *local* interactions (correlations between neighbouring cells). They also found these are largely sufficient to explain the activity of retinal ganglion cells, and concluded that complex retinal circuitry is not needed to explain multi-neuron synchrony: local pairwise interactions are sufficient to describe the network's activity even if synchronised firing extends to non-adjacent cell pairs and to multi-cell ensembles.

Schneidman's article also discussed the possibility of extending the model's result from a few neurons to larger systems. By extrapolating their measurements of the system's entropy to large N, the authors argued that sets of cells should show the physical characteristics of a "frozen" system starting at around $N \approx 200$. This effectively turns the neural network into a Hopfield-like system, which has a small number of possible states and strong error-correcting properties. However, this result was later contested [Roudi et al., 2009] when both theoretical considerations and simulations showed that extrapolation is not a valid way of inferring

large-N behaviour. This criticism does not invalidate extrapolation in Shlens et al.'s local model, nor it affects other conclusions drawn by using PME fits.

Later research about maximum entropy model extended them to different sets of constrained measures. Fitting higher cumulants, like $\langle \sigma_i \sigma_j \sigma_k \rangle$, is of course a possibility, but a computationally expensive one. Moreover, estimating, to a statistically significant level, the values of $O(N^p)$ cumulants requires vast amounts of training data for $p \geq 3$. Another choice was to keep firing rates and pairwise correlation, but also constraining the probability distribution for a measure of global network activity, $P(K)$, where $K(t) = \sum_{i=1}^{N} \sigma_i(t)$. This model, called K-pairwise, results in better fits compared to the pure PME [Tkačik et al., 2014, Mora et al., 2015].

One of the problems with general maximum entropy models, especially beyond pairwise, is the computational complexity of the fitting algorithms. More lightweight approaches were subsequently developed, such as the "dichotomised Gaussian" model [Macke et al., 2009] and the "population tracking" model, which fits $P(K)$ and the conditional probabilities $P(\sigma_i|K)$ [O'Donnell et al., 2016]. Very recently, building on maximum entropy literature, but adopting a different approach, Generative Adversarial Networks (GANs) have been used to accurately reproduce and interpret retinal spike trains [Molano-Mazon et al., 2018]. It's worth mentioning that efforts have been made to model temporal dependencies as well. Typically, this means adding further units to the model, which represent the same neurons, but at different times [Marre et al., 2009, Nasser et al., 2013, Gardella et al., 2018].

In computer science, the PME is equivalent to the fully visible version of the so-called Boltzmann machine (FVBM) [Ackley et al., 1985]: subsequent research used the more general restricted (or semi-restricted) Boltzmann machines (RBMs), which include hidden units as latent variables, and showed it can learn the spiking distribution of cortex recordings [Köster et al., 2014, Spicher, 2014]. Learning in RBMs is still resource consuming, but scales better than pure Boltzmann learning, and fitting several hundreds of units is feasible. However, this model does not have a justification in terms of the maximum entropy principle. It is important, then, to understand what it can teach us about neural activity, other than knowing that it fits the data better than a PME. Chapter 2 of this thesis will present some work done in this direction.

Finally, it is worth noting that the PME probability distribution is equivalent,

in physics, to that of an Ising model with arbitrary couplings: a spin glass. Because of this equivalence to a classic example for phase transitions, later research focused on the critical properties of PME and similar models. Criticality, in this "statistical" sense, was observed on multiple occasions, by studying the scaling properties of the models' specific heats [Tkačik et al., 2015, Mora et al., 2015]. Notably, a notion of criticality had already been discussed in computational neuroscience prior to this observation [Beggs and Plenz, 2003, Beggs and Timme, 2012]. This other concept of criticality, which I will call "dynamical", links the dynamics of a neural system to that of other complex systems, such as forest fire and sandpile models, through avalanche dynamics. In systems exhibiting emergent behaviour, a single event can cause a downstream chain of events, thanks to a feedback mechanism. The statistics, in space and time, of such "avalanches" offer insight on the physical properties of the system.

A discussion of statistical criticality, its extension to RBMs, and its relation to dynamical criticality, will be the main topic of chapter 3.

The retina is perhaps the system that was most commonly studied using this kind of paradigm, and is the system on which I started my work. I will describe its structure and function in section 0.3. Section 0.1 already gave an overview of the experimental methods which are essential for obtaining the datasets used in fitting statistical models. Later, chapter 1 will discuss a novel algorithm useful in that data analysis pipeline.

0.3 Vision and the retina

Like the hair cells in the inner ear, the retina is a fundamental entry point of sensory information, and the brain's only access to visual stimuli. Researching into its structure and function is not only of essential medical importance for the preservation of a healthy vision, but is also extremely interesting both for computational and experimental neuroscience. The advantage of studying the retina is that it is a relatively simple system, in that information flows essentially in one direction: from photoreceptors to the optic nerve. Therefore, it is a self-contained system, for which the stimulus can be controlled from outside by simply shining light. While feedback input from the rest of the brain does exist, it seems to have little influence on the activity of retinal neurons, and its function is not

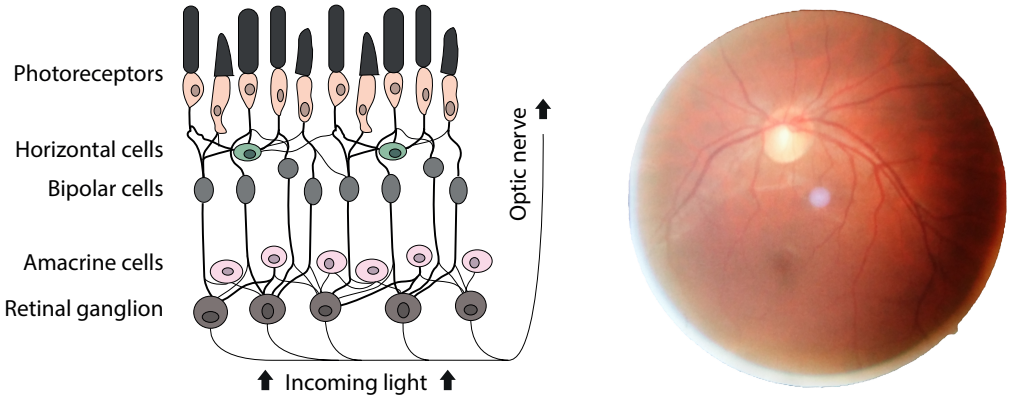


Figure 3: (a) Schematic representation of the neuronal layers in a patch of retina. (b) Photograph of the back of my own eye, where the retina lies. The dark red lines are blood vessels. The darker spot is the macula, which includes the fovea, the most sensitive and central part of the retina, responsible for colour vision. The bright white spot is the optic disc, the blind spot where the optic nerve crosses the other layers.

yet entirely understood [Repérant et al., 2006]. Thus, the retina is a model system that can teach us a lot about how neural systems encode information about the outside world into their spiking patterns. Given these reasons, it is no wonder the retina is one of the best studied parts of the vertebrate nervous system.

Developing a basic understanding of the retina's structure and function was an essential prerequisite to my PhD project. I will here summarise some essential concepts about its anatomy and computational properties, which seems necessary in order to put chapter 1 in a clearer context.

Structure The anatomy of the mammalian retina is shown in figure 3a. The incoming light is deflected by the rest of the eye and focused on the retina, where it passes through all layers of nervous tissue, and finally hits the photoreceptors located at its far end. Photoreceptors are of two main types: cones and rods. The former are concentrated in the central part of the retina, and are able to differentiate colour, being sensitive to three different ranges of light frequencies. Rods, conversely, do not see colour, are concentrated in the outer parts of the retina (and therefore used for peripheral vision), and are very sensitive even in very low lighting conditions, but are easily saturated by intense light. Rods and cones spontaneously fire when there is no incoming light, and are conversely inhibited by incoming light in their respective ranges of sensitivity [Masland,

2012].

In the *outer plexiform layer*, photoreceptors connect to the next two types of retinal neurons: horizontal cells and bipolar cells. Horizontal cells provide feedback to the photoreceptor cells themselves, acting laterally, i.e. connecting larger patches of neighbouring neurons. The feedback is useful to regulate the activity of the photoreceptors in cases where bright and dim areas coexist, and allow for an early mechanism of detection of local contrast. Bipolar cells carry the photoreceptors' signal to the inner layers. They connect selectively either to rods, "blue" cones, or any kind of cone; they can specialise in detecting a decrease or an increase in light intensity, and can have a sustained or a transient response to such changes, for a total of about twelve types of bipolar cells, which are anatomically and functionally distinguishable [Wässle et al., 2009].

Bipolar cells feed onto amacrine cells and retinal ganglion cells (RGCs). Amacrine cells have dendritic trees spanning relatively small areas, but axonal processes diffused on large areas, even half of the whole retina [Masland, 2012]. Like horizontal cells, they enable computations on a larger spatial scale, such as the detection of motion or centre-surround and foreground-background profiles. RGCs are the recipients of all the information encoded and computed by the retina. Their axons form the optic nerve, which connects the eye to the centre of the encephalon. The retinal ganglion is the best studied of the retinal layers, because all the information sent by the eye to the brain is encoded in its firing patterns.

Coding in the retinal ganglion The complex machinery described in the previous section serves as the first level of processing of visual information, from single photons hitting the retina to the first meaningful features of the image. It is also essential in order to optimise information for transmission through the optic nerve, reducing information loss but accounting for energy requirements and anatomical constraints.

The easiest way of studying retinal coding is subjecting the retina to a range of different stimuli, and measuring the response of retinal ganglion cells. Since the late 1980s, it has been possible to use MEAs to study *ex vivo* retinas, which can be kept alive while placed on the MEA chip, for a few hours, and sometimes for up to a few days [Maccione et al., 2014]. A projector is then used to shine appropriate light patterns on the preparation, so that the electrodes measure how the retinal ganglion encodes each of them. The first obvious observation, which

was already made in the 1930s [Hartline, 1938, Lettvin et al., 1959], is that, like bipolar cells, certain RGCs, the so-called "off" cells, mainly respond when there is a diminution of brightness in incoming light, while others, the "on" cells, fire when the stimulus luminosity increases. On/off cells, which fire in both cases, have also been observed. The response can be transient, if it decays shortly after the change in stimulus, or sustained, if it peaks at the stimulus presentation but continues for a longer time. The fact that retinal responses depend on changes in the stimulus, and not on its absolute lightness value, is the very first step of visual information processing by the brain; it is also an example of optimisation, since this means no energy needs to be spent whenever the stimulus is static, which is a very common situation. Studies of visual encoding from the point of view of information optimisation have been a popular subject of research for several decades [Laughlin, 1981, Barlow, 2001].

The activity of RGCs, however, encodes much more than just local changes in lightness. Gollisch and Meister's 2010 review, which is appropriately titled "Eye smarter than scientists believed", lists a number of complex functions that are already implemented in the retina, among which: 1) capability of amplifying the signal-to-noise ratio of photoreceptors, thus enabling the perception of very dim light flashes; 2) motion sensitivity: discrimination of the movement of textures, even with constant overall lightness, of the movement of the foreground, of approaching versus lateral movement; suppression of saccade-related motion, anticipation of motion by extrapolation; 3) latency coding (using the time of first spike) for edge detection; 4) adaptation: to very bright or very dim conditions, to high or low contrasts, to patterns, to the expected value of an intermittent stimulus [Gollisch and Meister, 2010]. Later discoveries have increased the number of RGC types in the mouse retina even further, by looking at both functional properties (based on what stimuli they respond to) and anatomical properties [Baden et al., 2016] of mouse RGCs. The authors of this study find a minimum of 32 types, and suggest retinal encoding resembles that of modern neural networks used in computer vision.

Clearly, then, the retina is not only a "sensor", but already performs an essential first part of the computations needed to elaborate the visual scene into meaningful elements. The axons of the retinal ganglion cells constitute the optic nerve, which carries visual information towards the brain. There, the rest of the visual pathway takes visual processing much further.

The higher visual system Except for olfactory information, all sensory systems project onto the thalamus, and from there to the brain's cortex; the visual pathway is no exception. One pathway leads from the retina towards the superior colliculus, which controls motor tasks essential for vision, such as saccades [Klier et al., 2001]. The main pathway projects to the lateral geniculate nucleus, a part of the thalamus. From the LGN, the optical pathway continues to the back of the brain, where the primary visual cortex (V1) is located. Neurons in V1 are tuned for location, spatial frequency, direction of motion, colour, and local contrast [Carandini, 2012]. After V1, it is believed that further areas of the visual cortex are specialised either in the identification of the content of the visual scene, or on spatial information and large-scale motion [Ungerleider and Pessoa, 2008].

Each of the main visual areas contains a complete representation of the visual scene [Miller, 2018]. However, the visual information is decomposed in different ways: the higher we go into the visual cortex, the more complex the selectivity of single cells to stimuli become: from simple Gabor filters, followed by simple patterns such as moving gratings, simple geometric shapes, and patterns of increasing complexity [Desimone et al., 1984, Ungerleider and Pessoa, 2008]. Neurons in the medial temporal cortex are selective to entire objects, complete of conceptual labels, independently of their orientation and location, and even of whether they are represented by images or written names [Quiroga et al., 2005, Quiroga, 2012]. In turn, the size of the receptive fields of each neuron becomes larger and larger, i.e. the location of the original stimulus in the visual field is less and less important. Finally, attention mechanisms become increasingly relevant in higher areas [Posner and Gilbert, 1999].

0.4 Outline of the project

This thesis contains selected parts of the work that I completed in the course of four years of doctoral programme, shared between the University of Edinburgh and the Royal Institute of Technology in Stockholm. After this introduction, it develops over four main chapters, which are set along a thread from the most related to experiments to the most theoretical, and a discussion. Although the chapters cover a variety of different topics, they were developed along a common line of thought, that links each of them to the next.

Starting from the extensive literature on statistical models, and in particular

following the footsteps of [Köster et al., 2014, Spicher, 2014], the main idea of this project was to explore the use of restricted Boltzmann machines for the analysis of neural activity in the mouse retina, especially regarding their interpretability. Multielectrode array recordings were provided by collaborators at the University of Newcastle and at the Italian Institute of Technology in Genoa. However, having a good understanding of these recordings, of the experimental protocols, and especially of the data processing pipeline, seemed important before proceeding to their analysis. When this project started, there was also no established method to deal with the large amount of data recorded by the dense, 64x64-electrode MEA used by our collaborators. For these reasons,the first part of this project was dedicated to working on spike sorting for dense MEAs. The first chapter, which consists of an article published on *Cell Reports* in 2017, is the result of this work, a joint effort between experimentalists and computational scientists. It is one of the various solutions that emerged in the last few of years to the problem of spike sorting for dense multielectrode arrays.

After this phase was completed, I started working on my initial research question, exploring the use of RBMs. In chapter 2, I will first summarise the few publications that have already attempted similar results, before or during the development of my project. Then, the main part of the chapter will study the role of hidden units in reproducing neural activity, and will ask whether RBMs can be used for factor analysis in that context. The problem of the low dimensionality of neural activity — i.e. of how only a small subspace of the space of all possible dynamical states of a neural system is actually in use — has recently become a popular research area [Gallego et al., 2017]. The study of "neural ensembles" — groups of neurons that activate together performing a similar function — is a related problem, in the sense that they both can be expressed as finding components in neural data. Principal component analysis and independent component analysis have been used for this purpose, even in work published this year [See et al., 2018], but are limited by their assumptions.

In the course of my RBM work, I became interested in the issue of criticality in neural systems and in statistical models. I verified that a similar concept can be defined for RBMs, and that similarly to what happens with fully visible Boltzmann machines, RBMs fitted to neural data always seem to be poised near the critical point. Chapter 3 is the result of my reflection on this phenomenon. It was written in the form of a book chapter, intended to review the phenomenon

of criticality in statistical models, and connect it with the "dynamical" criticality of emergent phenomena in recurrent networks.

The concept of "sloppiness" is another notion that arose in systems science [Gutenkunst et al., 2007, Machta et al., 2013] and has been only recently applied to neural modelling [Panas et al., 2015]. It is the observation that most models used in systems science and related fields have parameter spaces with an interesting geometry: many of their dimensions (the "sloppy" ones) are of little or no importance for the overall fitness of the model. Similarly to the content of the previous chapter, this one is also about the structure of the models' parameter spaces, and a connection between sloppiness and criticality cannot be ruled out. Starting from these ideas, we wondered if knowledge of this geometry of the parameter space can be used to design more efficient learning rules, that would allow longer retaining of information, on the lines of what was discovered by [Kirkpatrick et al., 2017] for artificial neural networks. Chapter 4 discusses this idea, applying it to a Hopfield network — itself analogous to a zero-temperature Boltzmann machine, used since the early 1980s as an archetype of memory-capable artificial neural networks [Little, 1974, Hopfield, 1982]. The results are included as a journal article, which has been submitted for review.

Although every chapter discusses its own conclusions, the last chapter of this thesis will once again draw the connections between the previous ones and frame them in the context of the future perspectives of this general field. For convenience, I summarise the main findings below:

- As a result of our work on spike sorting, we found that combining the signals from neighbouring electrodes to estimate the location of the spike source, and then using this information for sorting, is a feasible approach, provided the spike waveforms are taken into consideration at the same time. The result is a fast sorting algorithm that was initially developed for the BioCam multielectrode array, but was later tested on other dense arrays.

- I used the data from retinal recordings, sorted in this way for the subsequent work on restricted Boltzmann machines. I found that RBM hidden units are capable of rudimentary decoding of the stimulus, and, notably, can couple selectively to groups of neurons with similar responses, effectively classifying them according to their co-activation patterns.

- Designing a simple network model where binary neurons are driven accord-

ing to the activation of a small number of latent variables, I analogously found that the RBM hidden units are capable of retrieving said latent variables and, partly, their couplings to the neurons. This can be applied to the identification of neuronal assemblies or to analyse neural activity by means of dimensionality reduction.

- Following an investigation of specific heats in RBMs, I noticed that their divergence, which is considered a signature of "statistical" criticality, happens despite wide differences in the properties of the fitted datasets. In particular, there is no evidence of connection between model criticality and the "dynamical" criticality defined in the sense of avalanche statistics.

- The retina is an example of system that exhibits Zipf's law (which is connected to the aforementioned notion of statistical criticality). I verified that this holds true for different stimulus statistics, and even despite pharmacological intervention. I also found that the Zipf relation can hold even in absence of correlations, if a suitable firing rate distribution is enforced.

- A definition of Fisher information can be written for Hopfield networks, even if they are not stochastic. Interestingly, the value of Fisher curvature corresponding to each synaptic weight can be approximately retrieved from the weight itself. As a consequence, one can write a learning rule that avoids changing weights which are essential for storing previous memories, and this rule is entirely local. Learning through this rule significantly reduces catastrophic forgetting compared to a naive Hopfield rule.

Chapter 1

Unsupervised spike sorting for large-scale, high-density multielectrode arrays

This chapter consists of a published journal article. The paper is available as Hilgen, G., Sorbaro, M., Pirmoradian, S., Muthmann, J. O., Kepiro, I. E., Ullo, S., ... and Hennig, M. H. (2017). Unsupervised spike sorting for large-scale, high-density multielectrode arrays. Cell reports, 18(10), 2521-2532. [Hilgen et al., 2017b] *The first two authors were reported as contributing equally (on the experimental side and computational side respectively) and are listed in alphabetical order. As required, the following introduction motivates the work, delineates my contributions, and illustrates work done after publication.*

Processing experimental data is an essential step between the laboratory experiment and the analysis that follows, whose results are compared with a model or theory. In neuroscience, we may be interested in the properties of single neurons or of larger neuronal networks, up to scales comparable to the size of a brain. When working on the global dynamics of a network, or studying how a whole neural population encodes sensory information, it is often necessary to know about the behaviour of each neuron in a large group. Electrophysiological recordings consequently evolved from single-neuron intracellular recordings to large-scale extracellular ones.

Multi-electrode arrays (MEAs) are now the most common tool for electrophysiology. They consist of a chip on which a number of microelectrodes lies,

all recording simultaneously. The problem with extracellular recordings is that the trace recorded by an electrode does not directly correspond to the membrane potential of a unique cell. With the recent advent of dense MEAs, in particular, sorting spikes into individual units requires solving a complex source separation problem, as explained in the Introduction. This chapter proposes a solution to this issue, which exploits both the spatial signature left by a spike over multiple electrodes, and its waveform in time.

Before this method, no automated spike sorting method was available for the 3Brain BioCam used by our collaborators, who resorted to channel-by-channel sorting through a commercial software (PlexonSorter), followed by manual removal of duplicate units — an unreliable and time-consuming step on any 4096-channel dataset. Other methods (see section 0.1.1) were in early stages of development, or too computationally demanding for large-scale datasets or long recordings. Therefore, the results presented here were essential in order to provide a clean dataset for the analysis performed in the following chapters.

My contributions A good acquaintance with the methods of experimental data collection and analysis seemed a necessary step for a more informed work in theoretical analysis. I, therefore, started my project by looking at this early stage of data analysis. I was provided with recordings from neuronal cultures and mice retinas, taken with the 4096-electrode BioCam from 3Brain GMBH [Berdondini et al., 2009]. Software for the detection of spikes, starting from the raw traces of each electrode was already available thanks to previous work [Muthmann et al., 2015]. This first step also exploited the density of the array, not only eliminating duplicate spikes, but also inferring the location of the 'centre of mass' of the electrical activity for each event. Thus, my work consisted of analysing a dataset comprising, for every spike, a timestamp, an estimated location, and the 'spike shape', i.e. the snippet of electrical recording around the event that contains the waveform of the action potential.

Starting from this dataset, I tested the possibility of sorting spikes into units based on the location features obtained as explained above. This step involved the testing of a few different clustering algorithms, the issue being that the number of clusters is not known, and no assumptions on the shapes should be made: the method should therefore be non-parametric wherever possible. I settled for the Mean Shift algorithm [Comaniciu and Meer, 2002], initially using the version

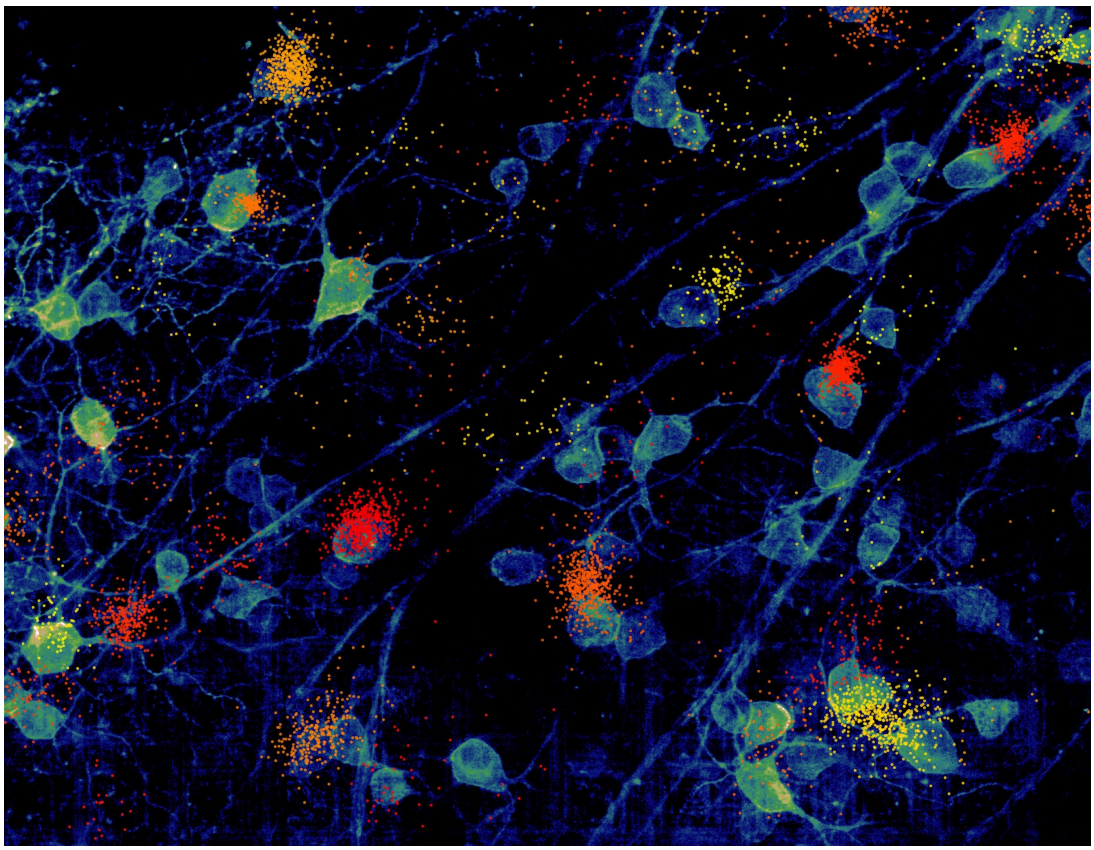


Figure 1.1: A section of mouse retina, imaged by two-photon microscopy, with retinal ganglion cells visible in false colours (green-blue-black). Superimposed to the photograph, a scatter plot of neural events detected by a dense micro-electrode array. Estimated spike locations tend to cluster near the somas of retinal ganglion cells. The colour of each cluster is selected at random, in order to distinguish it from the others.

provided in the scikit-learn machine learning library for Python [Pedregosa et al., 2011], and later developing an improved implementation, that could run in parallel on multiple cores. This version was later submitted and is now part of that open source library.

Among the vast literature on data clustering, the choice of Mean Shift was mostly justified by its scalability. While it can be slower than the well known k-means algorithm, it does not require prior knowledge of the number of clusters, but only an estimate of the cluster radius. A popular clustering method in the previous literature, KlustaKwik [2000, not described in a published manuscript] clustered data by fitting Gaussian mixture models (GMM) with the expectation-maximisation algorithm. That method, however, was applied to small tetrodes;

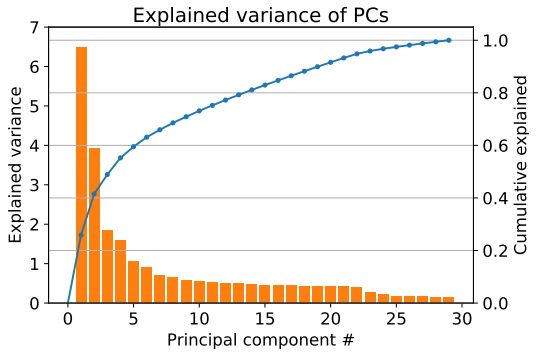


Figure 1.2: Explained variance (orange bars) and cumulative explained variance ratio (blue line) for the principal components of the spike shapes recorded by a BioCam MEA at 7 kHz sampling. The shape cutout size is 29.

my attempts at using GMMs for large MEAs failed, since the method appears not suitable for large cluster numbers. A later version of KlustaKwik was adapted to large tetrodes in recent years [Rossant et al., 2016]. Notably, while I empirically found Mean Shift to be satisfactory, our spike sorting method could be adapted to rely on other clustering algorithms. After the article was published, successful attempts have been made using DBSCAN [Ester et al., 1996] and HDBSCAN [McInnes et al., 2017], which also satisfy the same requirements of being scalable and non-parametric, and can also be run on multiple cores.

Since using only the estimated spike location didn't seem satisfactory, I started devising a way of including waveform features in the same algorithm. I had the idea of concatenating location and PCA-extracted waveform features (weighed by a dimensional constant) into a 4-dimensional feature space, and applied Mean Shift clustering to this space. The result seemed much more reliable, and this ended up being the method upon which the entire paper is based. Applying dimensionality reduction before clustering is usual in unsupervised pattern recognition, and has essentially two reasons: it allows for plotting of features in two or three dimensions, so that manual inspection of the dataset is possible; and it guarantees a better accuracy and a faster performance of the clustering algorithm. The choice of the number of principal components (PCs) to include was based on a trade-off between the spike shape variability explained by each PC and the increased computational complexity of the clustering step due to additional dimensions. On a 7 kHz BioCam dataset, such as the one that was used for all preliminary experiments, the first two PCs explain more than 40% of the variance (figure 1.2). While this does mean that relevant features of the waveform may be disregarded, adding a further component would contribute only less than another 10% of the variance explained, while increasing the number of dimensions of the clustering space from 4 to 5. It is important to note that other users may be

less concerned about speed, and may prefer a more precise evaluation of shapes; other datasets may also show different variability. Hence, the number of PCs is left as a parameter in our code, and can be tuned according to other needs.

I studied ways of classifying, post-clustering, which units may correspond to a neuron, and which were likely to be collections of noisy events. I first experimented with filtering out units with low average amplitudes or small numbers of events, but this involved a risk of missing neurons firing at low rates. An alternative approach I designed consisted of training a linear classifier in a supervised way, on a dataset that used events coming from areas outside the retina (i.e. from the naked chip) as examples of low-quality events, and high-amplitude spikes as examples of good spikes. This work led to the results in Figure S1 of the paper.

This approach was useful in order to overcome problems due to the choice of detection threshold. If the threshold is set to a high value, many weaker spikes will be missed, and a bias against certain neurons is introduced; conversely, if the threshold is too low, many false positives will be detected. For this reason, post-detection triaging of spikes was particularly helpful in identifying noisy events. This step is less useful for higher recording frequencies, and was used on 7 kHz datasets. Other sorting methods, designed for different chips, may not need such a provision. Previous algorithms, involving manual curation of clusters after sorting, also did not require it, since clusters of non-spike events could be evaluated and discarded by hand.

I contributed to earlier versions of the alignment between localised neural activity and microscope images of the corresponding neurons, and made a figure featuring this alignment, which was submitted as a cover for the issue of Cell Reports where the article was published. This figure was chosen for the best scientific image award at a small Swedish neuroscience conference, the StratNeuro retreat 2017, and is reproduced in Figure 1.1.

In order to offer evidence, beyond what can be noticed with the naked eye, that clusters of spikes tend to be localised near the somas of retinal ganglion cells, and therefore the localisation process works correctly, I studied the distribution of distances between cluster centres and soma locations (hand annotated by experimentalists). I then showed this was incompatible with a random distribution. The result was reported in Figure 4D of the article.

While I was working on algorithm development, and afterwards, including after publication, I wrote and maintained the Python class used for feature selec-

tion, clustering, filtering of spikes and units, plotting, etc. This class constitutes the main backend to our sorting method, and was released as the Herding Spikes library, which raised the interest of a number of experimental labs. After the publication of the paper, most of the codebase was rewritten from scratch to improve usability, to join detection, localisation and clustering together in a single repository, and to extend the code to make it usable on datasets from different MEA makes. My role in this final phase was mostly in curating the user-side aspect and coordinating efforts. This second version has also been released for free use.[1]

I participated with comments and discussion to all subsequent phases of the work, and I contributed to writing the paper.

[1] github.com/mhhennig/HS2

Cell Reports
Resource

Unsupervised Spike Sorting for Large-Scale, High-Density Multielectrode Arrays

Gerrit Hilgen,[1,10] Martino Sorbaro,[2,3,10] Sahar Pirmoradian,[2] Jens-Oliver Muthmann,[2,4,5] Ibolya Edit Kepiro,[6,7,11] Simona Ullo,[8] Cesar Juarez Ramirez,[2] Albert Puente Encinas,[2] Alessandro Maccione,[9] Luca Berdondini,[9] Vittorio Murino,[8] Diego Sona,[8] Francesca Cella Zanacchi,[6] Evelyne Sernagor,[1] and Matthias Helge Hennig[2,12,*]

[1]Institute of Neuroscience, Newcastle University, Newcastle NE2 4HH, UK
[2]Institute for Adaptive and Neural Computation, School of Informatics, University of Edinburgh, Edinburgh EH8 9AB, UK
[3]Department of Computational Biology, School of Computer Science and Communication, Royal Institute of Technology, Stockholm 100 44, Sweden
[4]Manipal University, Manipal 576104, India
[5]National Centre for Biological Sciences, Tata Institute of Fundamental Research, Bangalore 560065, India
[6]Nanophysics (NAPH), Istituto Italiano di Tecnologia, Genova 16163, Italy
[7]Faculty of Science, Engineering and Computing, Kingston University, Kingston KT1 2EE, UK
[8]Pattern Analysis and Computer Vision (PAVIS), Istituto Italiano di Tecnologia, Genova 16163, Italy
[9]Neuroscience and Brain Technologies (NBT), Istituto Italiano di Tecnologia, Genova 16163, Italy
[10]Co-first author
[11]Present address: National Physical Laboratory (NPL), Teddington TW11 0LW, UK
[12]Lead Contact
*Correspondence: m.hennig@ed.ac.uk
http://dx.doi.org/10.1016/j.celrep.2017.02.038

SUMMARY

We present a method for automated spike sorting for recordings with high-density, large-scale multielectrode arrays. Exploiting the dense sampling of single neurons by multiple electrodes, an efficient, low-dimensional representation of detected spikes consisting of estimated spatial spike locations and dominant spike shape features is exploited for fast and reliable clustering into single units. Millions of events can be sorted in minutes, and the method is parallelized and scales better than quadratically with the number of detected spikes. Performance is demonstrated using recordings with a 4,096-channel array and validated using anatomical imaging, optogenetic stimulation, and model-based quality control. A comparison with semi-automated, shape-based spike sorting exposes significant limitations of conventional methods. Our approach demonstrates that it is feasible to reliably isolate the activity of up to thousands of neurons and that dense, multi-channel probes substantially aid reliable spike sorting.

INTRODUCTION

Large-scale, dense probes and arrays and planar multielectrode arrays (MEAs) enable extracellular recordings of thousands of neurons simultaneously (Ballini et al., 2014; Berdondini et al., 2005; Eversmann et al., 2003; Frey et al., 2010; Hutzler et al., 2006; Maccione et al., 2014; Müller et al., 2015; Obien et al., 2015). Exploiting such data requires the reliable isolation of extracellularly recorded spikes generated by single neurons (spike sorting), a computationally costly task that is difficult to scale up to large numbers of recording channels (Rey et al., 2015). For conventional devices with up to tens of recording channels, a typical workflow consists of initial event detection, followed by semi-automated clustering based on spike waveform differences, followed by manual inspection and refinement. If the recording channels are sufficiently well separated, then there is no or little overlap between their signals, and spike sorting can be performed by clustering a low-dimensional representation of spike shapes (Harris et al., 2000; Lewicki, 1998; Quiroga et al., 2004).

This approach is inappropriate for dense, large-scale recordings. First, on dense MEAs, spike sorting becomes a complex assignment problem because not only multiple neurons contribute to the compound signal recorded on distinct channels, but individual spikes are also recorded by several neighboring channels simultaneously (Prentice et al., 2011; Rossant et al., 2016). Events are thus described by multiple waveforms and their locations, with an exponential number of potential assignments that can only be tackled using approximate algorithms. Second, the size of the datasets makes extensive manual intervention impractical; hence, as much of the process as possible, including quality control, should be automated.

Much of the variability in spike shapes is due to measuring them at different positions relative to the neuron. In conventional recordings, relatively small signals are measured using large electrodes averaging currents originating from different parts of the neuron. High-density MEAs with small electrodes detect primarily strong currents at the axon initial segment (AIS). The mechanism for generating action potentials is thus represented with a higher weight in the measured signals, leading to less variability in measured spike shapes. Existing solutions, demonstrated on data from hundreds of channels, are either template-matching methods (Marre et al., 2012; Prentice et al.,

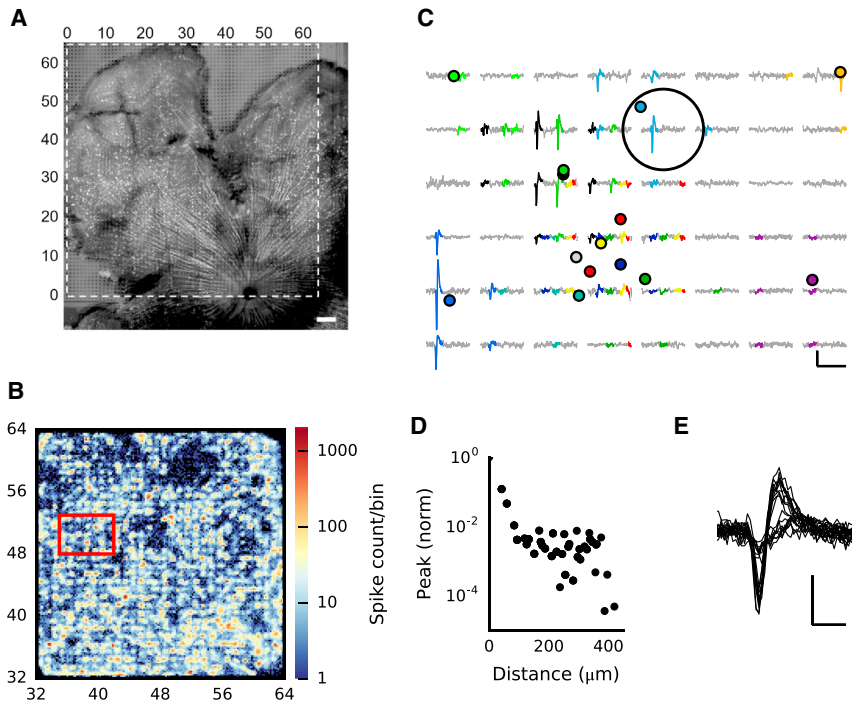


Figure 1. Spatial Event Localization Reveals Isolated Spike Clusters

(A) Confocal image of a Thy1-ChR2-YFP retina expressing yellow fluorescent protein under the Thy1 promoter, placed on the array for recording. Electrodes can be seen as small squares in areas not covered by the retina. The active area of the array is indicated by dashed lines. Scale bar, 200 μm.

(B) Activity map of a quarter of the array after spatial event localization. Spike counts are shown as a density plot, spatially binned with 8.4 μm resolution. Spikes cluster in distinct groups in space, presumably originating from individual neurons.

(C) Several detected events (rectangle in B), shown at their estimated locations (colored circles), and the corresponding episodes in the raw data (colored traces). Scale bars, 5 ms and 200 μV.

(D) Average peak signal decay for detected events as a function of distance. On average, a significant signal is detectable in an area of 100μm around the spike peak location. This plot is based on signal peaks at the spike time ± 2 recording frames, so signals beyond 200 μm reflect primarily noise.

(E) Twenty randomly selected spike shapes for events localized within the area marked by the large circle in (C), indicating the presence of signals from at least two different neurons at this location. Scale bars, 5 ms and 200 μV.

(B–E) The same dataset acquired at 24 kHz on 32 × 32 channels (A shows a different retina).

2011), or the removal of uninformative spike features to make fitting of a mixture model computationally feasible (Rossant et al., 2016).

Here we present a very fast and fully automated method for spike sorting. Dense sampling enabled us to obtain a rough estimate of a source location for each detected event (Muthmann et al., 2015), yielding dense, spatially separated clusters originating from single neurons, as demonstrated using optogenetic stimulation and confocal imaging. Average waveforms are obtained for each event, with noise reduced by signal interpolation. Shape features extracted from this waveform are then combined with spatial locations so that the clustering problem is reduced to finding local density peaks in few dimensions.

We demonstrate this method using light responses in the mouse retina and spontaneous activity in cell cultures recorded with a 4,096-channel MEA. A direct comparison with conventional spike sorting also exposes severe and hard to detect limitations of the latter. A parallelized implementation of this method that is capable of sorting millions of spikes within a few minutes on a fast workstation, as well as a tool for data visualization, can be downloaded at https://github.com/martinosorb/herding-spikes.

RESULTS

Spatial Spike Localization

Figure 1A illustrates a retinal whole-mount placed on the MEA. Spikes are detected using a threshold-based method that exploits dense sampling to improve detection performance and assigns each spike an estimated location based on the barycenter of the spatial signal profile (Muthmann et al., 2015). This procedure yields spatio-temporal event maps, where each event is identified by a time stamp, two spatial coordinates, and a single interpolated waveform. The resulting spatial activity maps provide a higher spatial resolution for spike locations than given by the electrode pitch (Figure 1B). Spikes were found in dense clusters surrounded by areas of low event density. The relationship between recorded signals and spike locations is illustrated in Figure 1C, where estimated spike locations are shown together with corresponding raw data segments from nearby electrodes. The examples show how spike locations relate to the spatial decay of the voltage peaks and that the decay was sufficiently wide to estimate their peak locations (Figure 1D; Pettersen and Einevoll; 2008; Lindén et al., 2011; Mechler et al., 2011). Thus, on dense MEAs, event locations provide a compact summary of the spatial activity footprint for each spike. Inspecting waveforms, however, reveals the presence of multiple units in small areas (Figure 1E), demonstrating that clustering spatial locations alone is insufficient for reliable single-unit isolation (Prentice et al., 2011).

Combined Spatial and Shape-Based Clustering

Next, spikes are clustered using a combination of their estimated locations and dominant waveform features, extracted via principal-component analysis (PCA), which provide a complementary, compact description of the events. The location estimate is an effective way of summarizing the spatial footprint each spike leaves on the array, whereas waveforms enable the separation of spatially overlapping sources, and they remove ambiguities at spatial cluster boundaries.

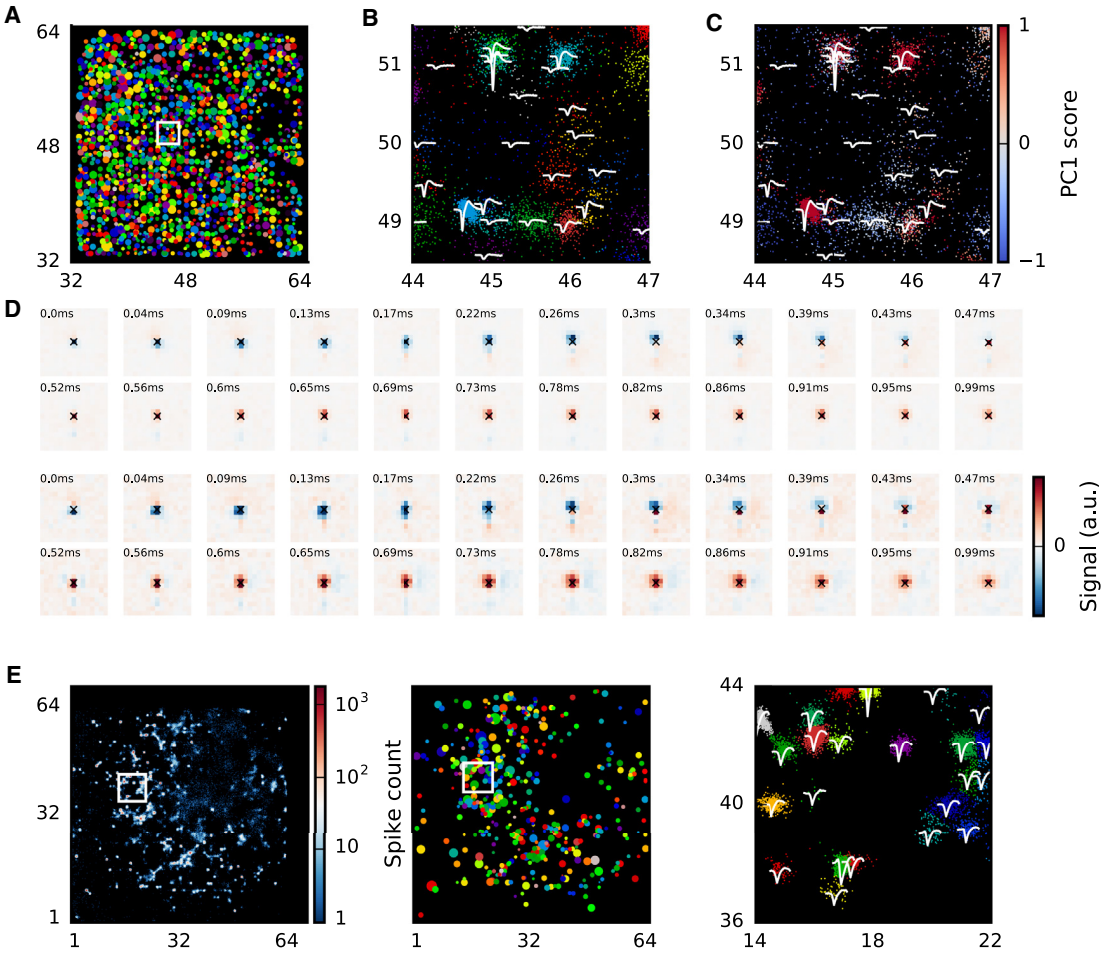


Figure 2. Illustration of Clustered Spike Data

(A) Overview of all single units obtained by clustering a retinal dataset (the same as in Figure 1, acquired at 24 kHz), shown as circles at their estimated locations in array coordinates. Circled areas are proportional to firing rates.

(B) Magnified view of a group of units (the area in the white rectangle in A), showing a subset of spikes at their estimated locations (dots, colored by unit membership; the same colors as in A) and the average waveform associated with each unit.

(C) As (B) but with spike colors encoding the magnitude of the spike waveform projection along the first principal component (PC1 score). Higher scores represent bi-phasic waveforms and low scores weak deflections without a clear bi-phasic shape.

(D) Electrical images for two units. Negative signals relative to baseline are colored in blue and positive signals in red. The cross indicates the centroid of the spike locations. Each square represents one electrode; 15 × 15 (0.63 mm × 0.63 mm) electrodes are shown. Axonal propagation can be seen, moving downward toward the optic disk.

(E) Clustered recording from a hippocampal culture. Shown are raw spike counts (left), all units obtained during the clustering step (center), and a magnified view of a small area of the MEA showing individual spikes and average unit waveforms (right). This recording was acquired with 4,096 channels at a 7-kHz sampling rate, and a waveform classifier was used to remove noise prior to clustering (Figure S1).

The mean shift algorithm was used for clustering, with the number of clusters automatically determined and controlled by a single scale parameter (Comaniciu and Meer, 2002). Clusters are formed by moving spikes along density gradients and augmented by local differences in spike waveforms. Including the first two principal components was sufficient to successfully isolate single units, reducing the high dimensional assignment problem to four-dimensional clustering, which can be performed in minutes for millions of events. In addition to the scale parameter, this method also requires a mixing coefficient for the shape information.

Figures 2A–2C show the result of clustering waveforms acquired at 24 kHz from 1,024 channels, yielding 440,000 spikes separated into 1,600 units. Cluster sizes ranged from tens of spikes to several thousands, corresponding to firing rates ranging from 0.1 to 30 Hz. In a magnified view, Figure 2B shows that units may indeed spatially overlap but are well separated by their waveform features. Overall, units with clearly bi-phasic and large-amplitude waveforms tend to form the more spatially coherent clusters, whereas smaller events are spatially more spread out.

Units with small waveforms originate from neurons with weak signals detected because of low thresholding during the

detection step to avoid false negatives. The first principal component (PC) projection (PC1) for the events is a good indicator of their biphasic character, and, using the convention that positive values always coincide with more biphasic waveforms, this measure may be used to (de)select units for subsequent analysis (Figure 2C). A more precise method, used for all recordings performed at lower sampling rates (<10 kHz), is to train a classifier to pre-select valid spikes prior to clustering based on salient waveform features (Figure S1). This method reliably removes noise because the classifier is well adjusted to the specific recording conditions. Importantly, however, this step is not required for sampling rates of more than 10 kHz.

As a first assessment of the clustered units, we generated electrical images for individual units (Figure 2D). These images provide a spatio-temporal representation of the raw signal recorded around the time of spiking and is generated as a spike-triggered average of the signal on each electrode. Of 406 inspected units with at least 100 spikes, all but one had an estimated location within 40 μm of the electrode that contained the peak signal (median distance, 9.7 μm), indicating that units are indeed well aligned with their spatio-temporal electrical footprint. Furthermore, the recordings were of sufficient detail to isolate axonal propagation (Figure 2D), characterized by a separate, weak positive peak followed by a negative peak traveling downward (toward the optic disk). Because these events peak within less than 100 μs of the main signal, they are not detected as separate events but, instead, introduce a small bias on the location estimates during spike localization.

We also tested our method on activity recorded from cultured hippocampal neurons. Figure 2E illustrates that isolation of single units is also feasible for these preparations, although here the spike localization was less precise than in the retina. We attribute this to a larger effective conductivity in the space above the electrodes, resulting in smaller signal amplitudes, which, in turn, increases the influence of noise on localization (Ness et al., 2015). Such conductivity is likely much lower for the 200- to 300-μm-thick retina, leading to larger and more precisely localizable signals. Ness et al. (2015) show that even small MEA-tissue gaps strongly reduce the signal amplitudes, a likely explanation for the clear, sharp boundaries between areas with and without recorded spikes. Nevertheless, spikes in cultures were typically spatially well clustered, and waveform differences had sufficient detail to allow separation of overlapping units (Figure 2E, right).

Waveform Features Are Essential for Reliable Clustering

To assess the importance of waveform features for sorting and the role of the mixing coefficient α, we compared the correlations between all waveforms within each unit with cross-correlations of waveforms between this unit and its closest neighbor or all nearby spikes within a radius of 42 μm (electrode pitch; Figures 3A–3C). A well sorted unit is expected to have high within-correlations and smaller cross-correlations. Figure 3A shows an example where spatial clustering was sufficient to isolate a unit. Correlations after clustering spatial locations alone ($\alpha = 0$) are very similar to those obtained when waveforms are added ($\alpha = 0.3$), with few spikes re-assigned based on their waveform features. In contrast, Figures 3B and 3C illustrate examples with two clearly distinct units with spatial overlap that could only be separated by waveform features. Increasing α increases self-correlation, with lower cross-correlations for nearby events with sufficiently distinct waveforms in other units (Figure 3B). However, some high cross-correlations can remain for similar but spatially well separated units (Figure 3C).

To quantify the separability of these distributions, we computed the area under the receiver operating characteristic (ROC) curves (AUC), constructed from the distributions of self-correlations and correlations with events in the nearest unit (Figure 3E) or all neighboring events (Figure 3F). The AUC was calculated as the integral of the area spanned by the probability of finding a self-correlation above a sliding threshold, as a function of the probability of finding a cross-correlation above this threshold (true positives versus false positives), so that a value of 1 corresponds to perfectly separated distributions, whereas 0 indicates full overlap.

The median AUC for all units increases with α before plateauing at values about $\alpha \approx 0.4$ (Figure 3D), indicating that the combined features overall improved separation into single units. The AUC distributions show that this effect is substantial (Figures 3E and 3F). Although spatial clustering alone only yielded three (of 788 units with more than 100 spikes) units with AUC > 0.9 compared with events from its closest neighbor, this increases to 130 (of 956 units with more than 100 spikes) for $\alpha = 0.32$. This number rapidly increases when $\alpha \approx 0.25$ and plateaus for larger values, indicating that the precise choice of this parameter is not critical. It is important to note that, although high AUC values indicate well isolated units based on waveform features alone, units with a small AUC should not be rejected because they may still be spatially well isolated.

In summary, waveform features help both to refine existing units found by spatial clustering and to separate spatially overlapping units. Event locations and waveforms provide an effective complementary approach of summarizing the key features of the spatio-temporal footprint left by spikes on the array.

Validation with Optogenetics and Anatomical Imaging

To test whether the detected units indeed correspond to single neurons, we used Thy1-ChR2-YFP retinas (see Experimental Procedures) expressing Channelrhodopsin-2 (ChR2), a light-gated cation channel, under the Thy1 promoter in about half of all retinal ganglion cells (RGCs) (Raymond et al., 2008). This allowed us to stimulate spiking exclusively in a subset of visually identifiable RGCs to clearly establish correlates between single spike-sorted units and individual RGCs.

We first compared the photoreceptor-driven activity recorded during normal light stimulation (irradiance 4 $\mu W/cm^2$, full field flashes at 0.5 Hz) with recordings obtained when these light responses were blocked with 20 μM 6,7-dinitroquinoxaline-2,3-dione (DNQX) and L-AP4, and ChR2-mediated spikes evoked at maximum irradiance (0.87 mW/cm^2; Figure 4A). The activity maps show that only a subset of all RGCs responded to optogenetic stimulation (Figure 4A, top and center). We found 375 units in that dataset with a firing rate of at least 0.5 Hz during photoreceptor-driven light stimulation but only 254 units during direct

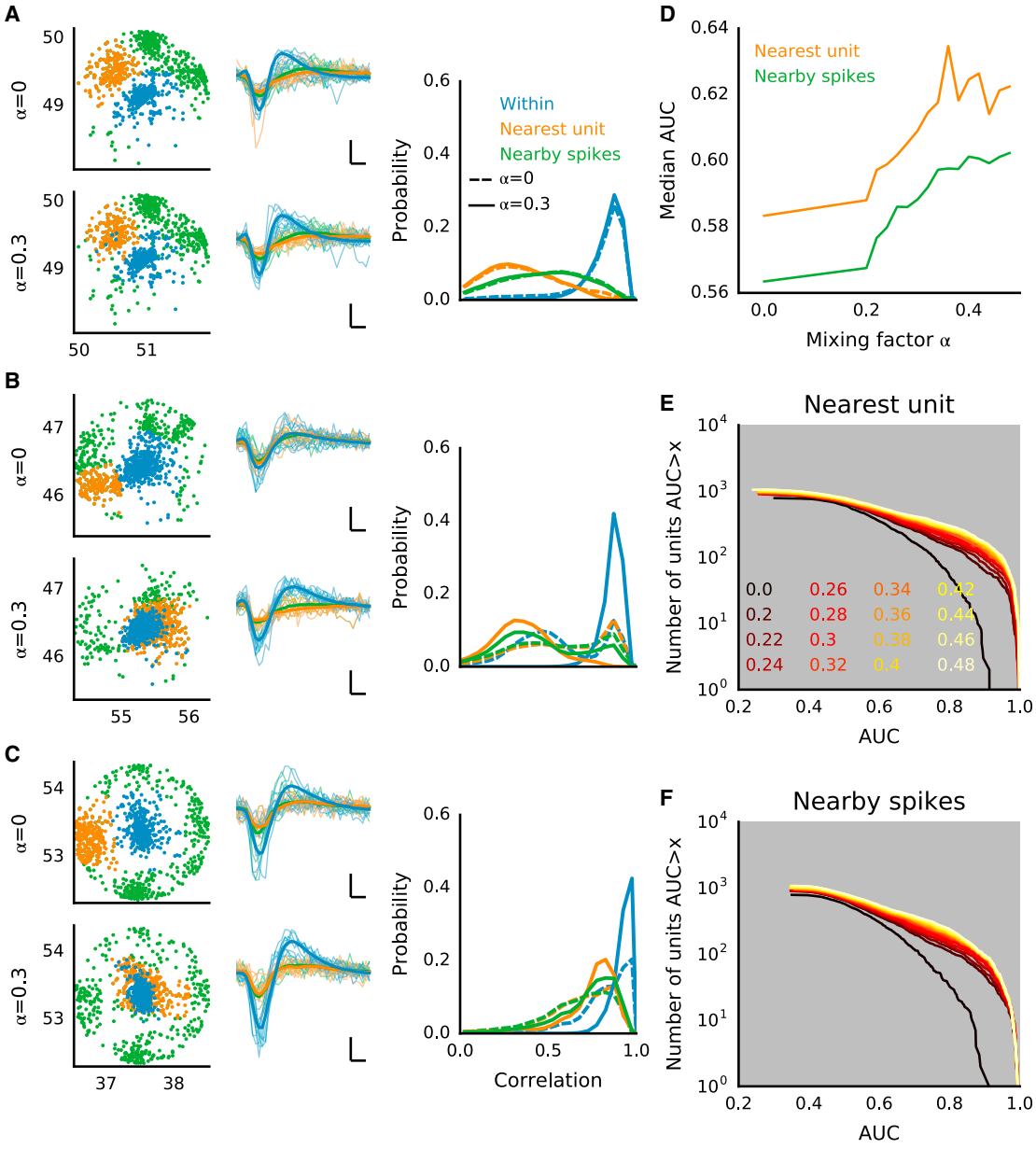


Figure 3. Waveform Correlations Demonstrate Improved Clustering for Combined Event Locations and Waveform Features

All data are from the same experiment as in Figure 1, acquired at 24 kHz.

(A) An example comparing the same unit obtained using spatial clustering alone (with mixing coefficient $\alpha = 0$) with clustering based on combined event locations and waveform features ($\alpha = 0.3$). Shown are event locations (left), example (thin lines) and average (thick lines) waveforms (center; scale bars, 0.2 ms and 100 μV), and normalized distributions of waveform correlations (right; dashed lines, $\alpha = 0$; solid lines, $\alpha = 0.3$). The selected unit is colored in blue (within), the nearest unit in orange, and the remaining events within a radius of 42μm of the target unit location in green (nearby spikes; these also include the spikes of the nearest unit). In this example, spatial clustering is sufficient to isolate the blue unit.

(B and C) Same as (A), but illustrating two units that spatially overlap with their neighbors.

(D) Median AUC for all units, quantifying the overlap between the normalized distributions of waveform correlations for each unit as a function of the mixing coefficient α. The comparison was either done with the spatially closest unit (orange) or with all neighboring spikes (green).

(E) Full distributions of AUC values obtained from comparison with the nearest unit and for different values of α.

(F) Same as (E), but taking all nearby spikes into account.

stimulation of ChR2-expressing RGCs. In addition, 77 units were significantly less active during light stimulation than during ChR2 stimulation, presumably reflecting neurons unresponsive to photoreceptor activation but nevertheless expressing ChR2. The responsiveness of each unit to ChR2 activation was assessed by determining the correlation of an individual unit's

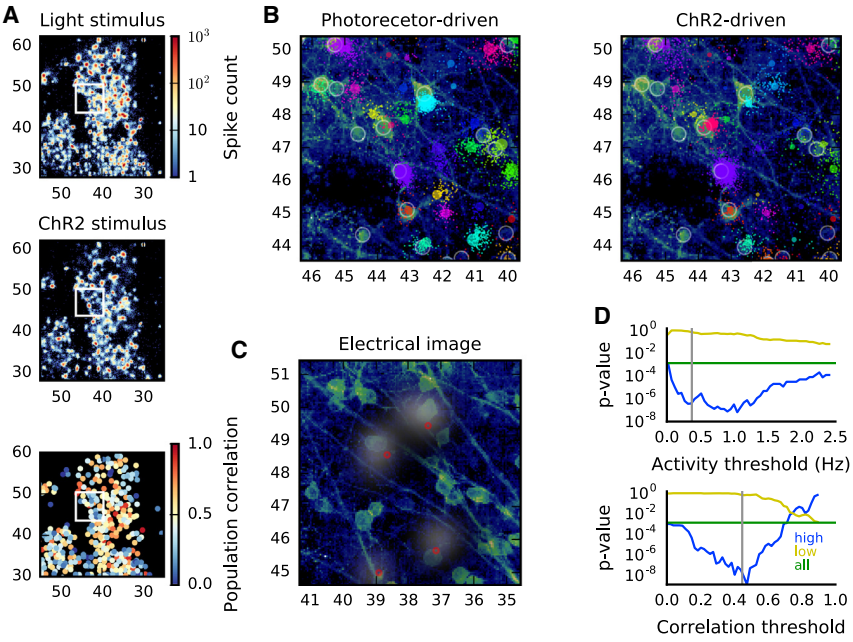


Figure 4. Comparison of Optogenetically Evoked Spikes with Anatomical Imaging

(A) Activity maps obtained during photoreceptor stimulation (top) and ChR2-expressing RGC stimulation under blockade of the glutamatergic pathway from photoreceptors to RGCs (center). The bottom graph shows the correlation of the activity of each unit with the overall ChR2-driven population activity, which quantifies the responsiveness to optogenetic stimulation.

(B) Alignment of neural activity with a confocal image. Individual spikes are shown as dots, colored according to unit membership (note that only a subset of all recorded spikes is shown for clarity). Annotated somata are highlighted by circles and the unit's centroids as colored circles with areas proportional to the spike rate.

(C) A different imaged area with superimposed electrical images of four selected units. Cluster centroids are indicated by red circles.

(D) The distribution of spatial distances between each unit and its closest soma is significantly different from randomness. The one-tailed Kolmogorov-Smirnov test shows incompatibility with the distribution obtained by assuming that somata and units are unrelated ($p = 0.001$, green line). When the units are separated into two sets according to activity level (top) and population correlation (bottom), the effect is strongest for highly active/highly correlated units (blue), whereas weakly active/correlated units are randomly distributed (yellow). The gray line indicates the threshold value for which the two sets have the same number of units. The data in these graphs summarize an imaged area of 0.78 mm^2.

activity with the overall population activity (Figure 4A, bottom). Almost all highly active units during ChR2 stimulation also showed higher correlation, with some exhibiting uncorrelated activity, which we attribute to intrinsic spontaneous activity that could not be blocked. Of all detected units, about 40% had a correlation larger than 40%, close to the expected fraction of Thy1-expressing RGCs.

Next we co-localized the activity with confocal micrographs of labeled neurons (Figure 4B). We analyzed an area of 0.78 mm^2, where 195 somata were manually annotated, and 211 units were detected. An example of the alignment of activity and anatomical image is shown in Figure 4B for activity obtained during photoreceptor stimulation (left) and ChR2 activation (right). All units with significant activity during ChR2 stimulation were closely co-localized with a labeled soma. Similarly, there is a tight co-localization between the neurons and electrical images generated from the raw traces (Figure 4C).

To verify whether labeled somata and localized units where significantly close to each other, we computed the distance to the closest soma for every unit. If units and somata were randomly distributed, the probability of a distance r would be $2\pi n r e^{-\pi n r^2}$, where n is the density of somata (Chandrasekhar, 1943). We compared the distribution of 198 distances to this null model using a one-tailed Kolmogorov-Smirnov test (Figure 4D), confirming that the distances are significantly smaller than predicted by the random model. To account for the effect of spontaneous activity, we applied the test after separating the units into two groups according to their activity level or population correlation, varying the threshold that separates the two sets. The locations of the less active and less correlated units are compatible with a random distribution, whereas the more active and better correlated units are significantly closer to their anatomical counterparts.

Model-Based Validation and Quality Control

As pointed out above, detection was performed with a low threshold to minimize false negatives. Hence some units are expected to contain ambiguities the clustering algorithm cannot fully resolve. For instance, the localization error is typically larger for spikes with small amplitudes (Muthmann et al., 2015); hence, it may not be possible to spatially cluster these events reliably.

To assess the cluster assignments' quality and automatically reject poorly separated units, we followed an approach proposed by Hill et al. (2011). Under the assumption that spike locations and waveform features can be described by a multivariate normal distribution, a comparison of the clusters assignments with those predicted by a Gaussian mixture model provides an estimate of the classification performance. Each unit was investigated in turn, including all of its immediate neighbors, by fitting a six-dimensional Gaussian mixture model with the number of components equal to the number of units (Experimental Procedures). We included four PCA dimensions to ensure that the model best exploits all available waveform features while ensuring reliable convergence. To evaluate the relevance of spatial locations and waveform features for clustering, the model was also fit to each of these features separately.

The model comparison produces a confusion matrix with the estimated number of false positives and negatives for each unit, which is then summarized into a single measure (F-score). Two typical outcomes of this procedure are illustrated in Figures 5A and 5B for relatively crowded areas on the array. Figure 5A shows a unit with a distinct waveform (blue) and four neighbors

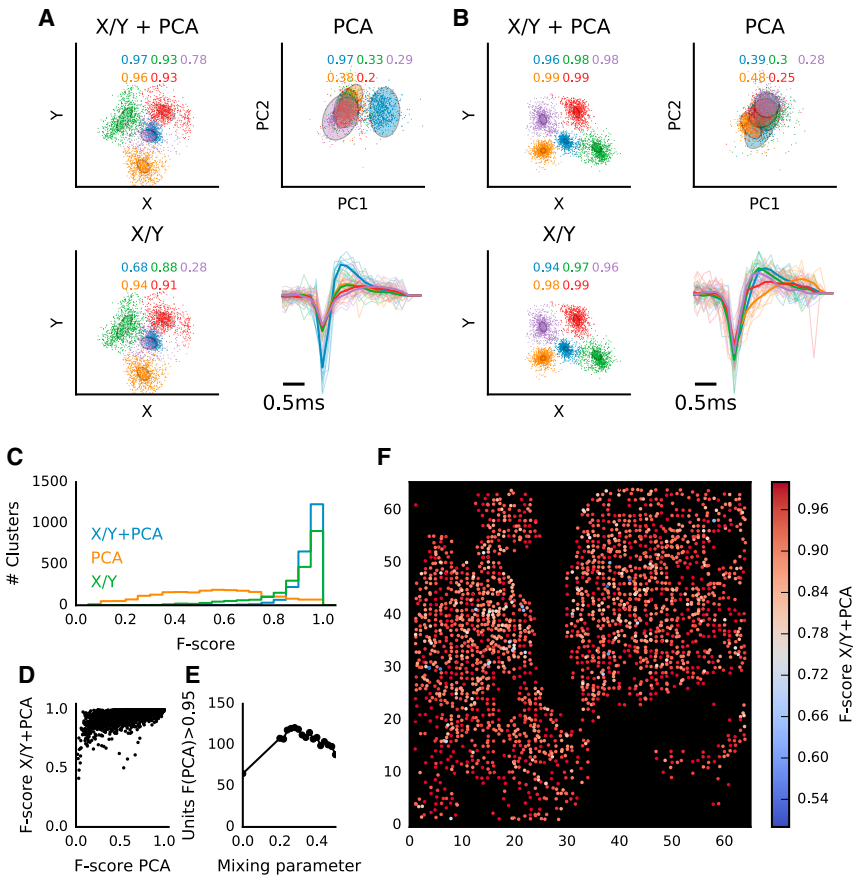


Figure 5. Quantitative Assessment of Sorting Quality with Gaussian Mixture Models

(A) A Gaussian mixture model (GMM) fit to a group of neighboring units. All units within a radius of 42 μm around the unit colored blue were included in the model. The model was then fit to combined spike locations and waveforms (X/Y+PCA), waveforms alone (PCA), or locations alone (X/Y). Spikes are colored to indicate the original cluster assignments. The numbers in each panel are the F-scores for each unit, indicating the average number of false positives and negatives between the two assignments. Examples of spike waveforms and the unit average (thick line) are shown using the same color scheme. In this example, the unit colored in blue is well separated both spatially and by waveform features.

(B) Same as (A), but illustrating a group of units with very similar waveforms, which can only be separated using spike locations.

(C) Histogram of F-scores of all units in one recording, computed as in (A) and (B).

(D) Relationship between F-scores evaluated from waveforms alone and the combined features.

(E) Number of units with an F-score > 0.95, evaluated from waveforms alone for different values of the shape mixing parameter α. The best overlap is obtained for α = 0.28, the value used in the other examples in this paper.

(F) Spatial distribution of F-scores for all units.

within one electrode radius. The blue unit was already well isolated based on waveform features alone (PCA, F-score = 0.97) but not when only spike locations were considered (X/Y, F-score = 0.68). Combining locations and waveforms did not yield further improvement, although it helped to isolate its neighbors based on their spike locations. Figure 5B shows five spatially well separated units with smaller and very similar waveforms. Waveform-based clustering alone gave poor results, but adding spike locations improved it considerably.

Figures 5C–5F summarizes the analysis performed on a 7.6-million spikes dataset. Each of 2,234 units with a spike rate of at least 0.3 Hz took, in turn, the role of the blue unit in Figure 5A, and all units within a radius of 42μm were combined into mixture models. When location and waveform features were used for quality control, 55% of the units (1,230) had an F-score > 0.95 and 15% (334 units) an F-score > 0.99 (Figure 5C; X/Y+PCA). These fractions decreased only slightly when locations were used on their own but substantially for waveforms (PCA) alone. Comparing F-scores for waveforms or combined features shows that adding locations improves fits in most cases, but poor scores for waveforms also result in lower combined scores (Figure 5D). An inspection of the waveform scores for different α values shows an optimum for α = 0.28 (Figure 5E). A spatial overview of these results showed that units with low F-scores are primarily found in crowded areas (Figure 5F).

Functional Assessment of Single-Unit Activity

We recorded RGC responses to full field flashes, allowing us to evaluate whether individual sorted units exhibit the typical On, Off, or On-Off light responses. Figures 6A and 6B show spike locations, spike waveforms, raster plots, and peri-stimulus time histograms (PSTHs) of all units in a small retinal patch, demonstrating excellent separation into fast and slow On, Off, and On-Off responses. Importantly, immediately adjacent neurons generally exhibit different responses, as expected from the mosaic functional organization of RGCs.

The fact that the majority of these units, even those with very small waveforms, exhibit reliable light responses demonstrates that the signal variance is mainly due to physiological causes rather than electrical noise (Muthmann et al., 2015). Units with well defined waveforms are typically also well separated in their PCA projections, whereas small waveforms are mainly clustered based on spatial locations (compare units 1–3 with units 5–7 in Figure 6C). The cluster F-scores (shown above the waveforms in Figure 6B) are lower for units with small waveforms; hence, further analysis for well isolated cells can rely on this measure.

Comparison with Conventional Spike Sorting

Conventional spike sorting relies on differences in spike waveforms. To evaluate how our approach scores in comparison with such methods, we compared our method with the outcome of manually curated spike sorting done on each MEA channel separately. Conventional spike sorting was performed using

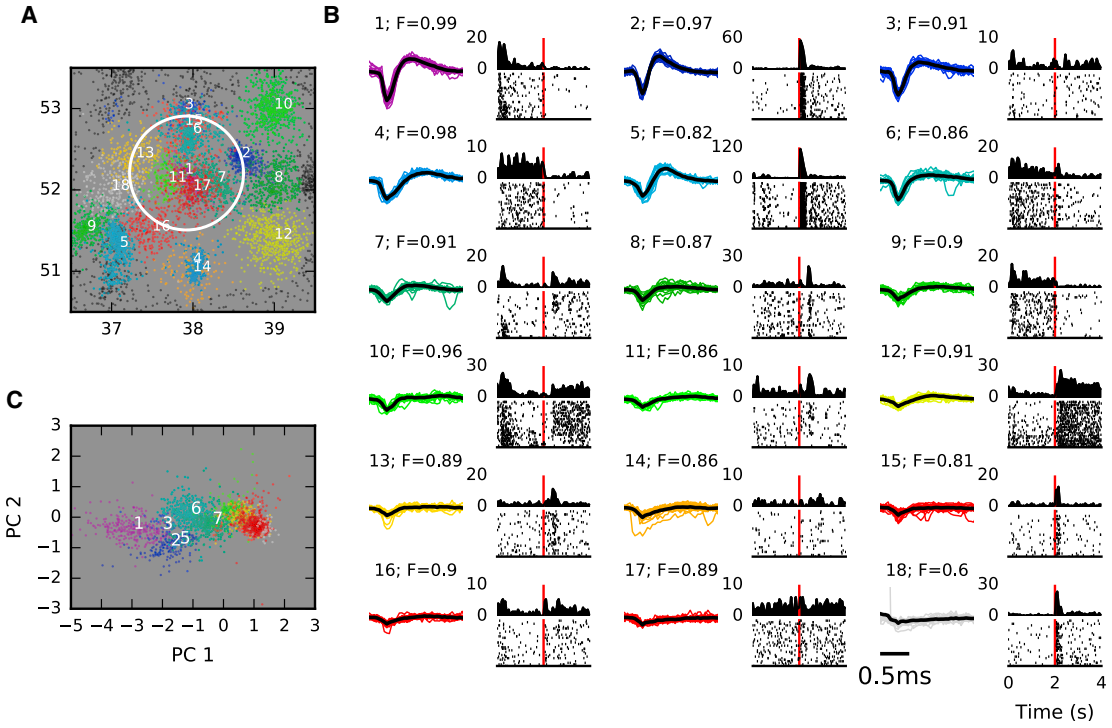


Figure 6. Functional Characterization of Spike-Sorted RGCs

(A) Spatial locations of individual spikes within a small area on the MEA. Only a subset of spikes are shown for clarity. This area contained 18 units, and unit membership is indicated by color. Spikes of units centered outside of the visible area are shown as black dots. Coordinates are in units of electrode distance (42μm).

(B) Overview of the units highlighted in (A) using the same color scheme. Each panel shows example waveforms, the average spike waveform (black line), and the raster and PSTH for full field stimulation (2 s bright, 2 s dark; red lines indicate stimulus offset time). The unit number and cluster F-score are given above the spike waveforms.

(C) Spikes in the circled area in (A), with identical color coding, shown in the space of waveform principal components (PCA space).

Shown are the same data as in Figure 1 with an acquisition rate of 24 kHz.

T-distribution expectation-maximization (E-M) clustering (Shoham et al., 2003) followed by manual inspection and correction (Plexon Offline Sorter).

The data used for this comparison were recorded at 24 kHz with 1,024 electrodes (Figure 1) and included 538 clusters with at least 200 spikes each. For each cluster, we located the most similar sorted unit using spike count cross-correlation following binning (each unit is typically found on multiple electrodes) and obtained the number of spikes in the sorted unit that were not part of the cluster (false negatives) and the number of spikes in the cluster not present in the sorted unit (false positives). As for the mixture model above, we then computed precision, recall, and the F-score for each cluster (Experimental Procedures).

Figure 7A illustrate two common cases we encountered. The first example shows an almost identical assignment for both methods that we found in 96 clusters (18%), with an F-score larger than 0.95 (Figure 7B). Such pairs had very few false positives and negatives (e.g., the pair in Figure 7A, top, had nine false negatives and no false positives of 1,818 spikes).

For many of the remaining clusters, the F-score was dominated by a sizable fraction of false negatives, spikes in the sorted unit that were not included in the corresponding cluster (units with low recall in Figure 7C). An inspection of the spatial locations of these events showed that false negatives were often located far away from the cluster centroid and visually appeared to be part of another unit (Figure 7A, center and bottom, orange events). Figure 7A, center and bottom, illustrates the consequences of erroneous assignment by conventional spike sorting, changing On cells into On-Off cells by merging spikes from other nearby Off cells.

We found that the inclusion of distant spikes happened frequently, with an average distance of false negatives from the cluster centroid typically around 30 μm (Figure 7D). This suggested that they were wrongly included in a sorted unit based on waveform similarities. To see whether these failures are associated with specific waveform features, we compared the F-scores with the average projections of the waveforms along their first principal component (Figure 7E). The PC1 projection provides an indicator of signal quality for each unit (Figure 2B), and, indeed, lower F-scores were observed almost exclusively for low-scoring units. Hence, we conclude that conventional spike sorting only allows reliably isolation of units with strong, very prominent waveform features, whereas smaller, less distinct

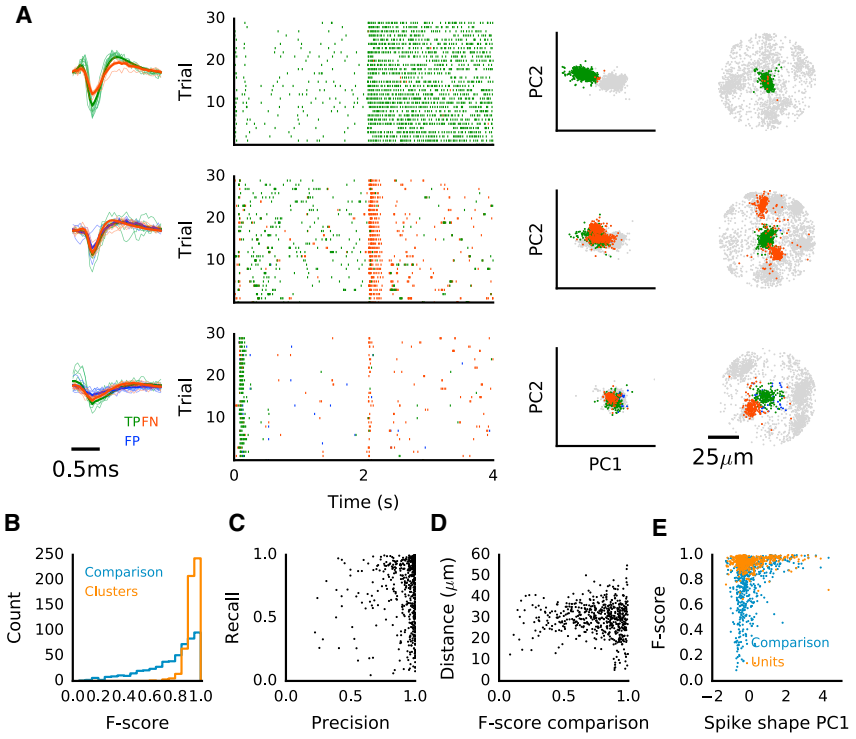


Figure 7. Failure of Conventional Spike Sorting in Isolating Single Units

(A) Examples of three units clustered with our method compared with corresponding units obtained from conventional, spike-shape based sorting. Raster plots show responses to full field flashes (left; 2 s bright, 2 s dark), principal component projections of all spikes found in the area within a radius of 78 μm around the cluster center (center), and all spikes plotted at their locations (right). Spikes colored green were found in both units, those in orange only in the sorted unit, and those in blue only in the clustered unit.

(B) Histograms of F-scores for the comparison (blue) and for mixture model fits for the sorted units (orange).

(C) Precision and recall for the comparison, illustrating that low F-scores are primarily due to spikes missing in the clustered unit (orange events in A).

(D) Average distance of spikes not included in the clustered unit, measured from the cluster centroid.

(E) Comparison of F-scores with the average projection of the waveforms along the first principal component, shown for the comparison of sorting method (blue) and for the mixture model fits of clustered units (orange).

Shown are the same data as in Figure 1.

waveforms cannot be separated reliably on the exclusive basis of their shape.

DISCUSSION

Spike sorting is a critical step in the analysis of extracellular electrophysiological recordings. An erroneous assignment of spikes can have severe consequences for the interpretation of neural activity, which has motivated the development of joint models of spike waveforms and neural activity to avoid spurious or biased correlation estimates (Ventura and Gerkin, 2012). In high-density recordings, increasingly used both for in vitro and in vivo studies, assigning spikes to single units becomes exponentially complex as a function of the number of events; hence, it requires approximate solutions. Moreover, the sheer size of the data prevents detailed manual inspection and quality control.

Here we solve this task by creating an efficient, low-dimensional data representation, based on spatial spike locations and the most prominent waveform features, that can be clustered efficiently. We found that clustering in four dimensions, with two dimensions representing waveform features, was sufficient to achieve high performance, which we attribute to the fact that the signals reliably measured with a dense MEA mainly originate from strong currents at the AIS of each neuron, with limited variability between neurons. This enables estimating their spatial origin but limits variability to support shape-based spike sorting. Comparison of optogenetically evoked spikes with anatomical images indicates that detected spikes typically cluster near the AIS and that localization alone is sufficiently precise to reliably isolate some neurons even without using additional waveform features.

Out method could be used with arrays and probes where an event location estimate can be reliably obtained. The dimensionality of the clustering step can then be adjusted to exploit higher waveform variability. The complexity of the clustering algorithm scales quadratic with the number of spikes, and the highly optimized version used here has a better performance when prominent spatial clustering is present. We developed a parallelized implementation that allows sorting of millions of spikes in minutes (ten million spikes take about 8 min on a 12-core 2.6-GHz Xeon workstation). Together with a method for quality control, this makes it possible to perform parameter sweeps to identify the optimal parameters of the clustering algorithm. Clustering is followed by an automated assessment of clustering quality, allowing the automated rejection of poorly isolated units and manual inspection of borderline cases. We also provide a visualization tool where further annotation can be performed.

The complete workflow consists of event detection, spatial localization, clustering, quality control, and, finally, optional manual inspection. The former two currently constitute the main bottleneck. Detection takes about four times real time and scales linearly with recording duration. The complexity of the spike localization scales linearly with the number of detected events and runs roughly in real time for recordings with normal spike rates. For both methods, parallelized implementations are under development.

Our work with high-density recordings has revealed significant limitations of purely shape-based spike sorting for MEA recordings. It is virtually impossible to evaluate how many units are

represented in a single electrode signal. If the electrode is positioned close to a neuronal cluster, one or two units with strong signals usually have sufficiently distinct waveforms to be separable. However, comparison with spike locations showed that weaker signals arising from more remote cells are generally not distinguishable based on shape alone. We frequently found cases where spikes of neurons with entirely different physiological signatures were mixed by shape-based sorting, a problem that cannot be avoided even by careful manual inspection. Our method, on the other hand, handles such situations much better because spatial location estimates are sufficiently precise to disambiguate borderline cases. Thus, a main factor affecting sorting performance is the noise and bias in spatial localization, both depending on signal quality (Muthmann et al., 2015).

A different strategy, outlined by Marre et al. (2012), is to estimate spatio-temporal templates that are then used to identify spikes from each neuron (Dragas et al., 2015). This shifts the computational burden from spatial interpolation and source localization in our method to the deconvolution of spikes from raw data. We found that adding shape criteria at the detection stage could lead to false negatives, suggesting that templates can only be reliably estimated for neurons with sufficiently high firing rates. A third method, recently developed by Rossant et al. (2016) for high-density in vivo probes, reduces complexity by masking irrelevant parts of the data based on geometric constraints before fitting a mixture model and clustering the data. This avoids an early discarding of potentially useful information, which our method does by using signal interpolation and Marre et al. (2012) did by creating templates. On the other hand, although potentially more precise, this method is computationally more demanding and, hence, more suitable for data from hundreds of channels.

EXPERIMENTAL PROCEDURES

Electrophysiology

Experimental procedures were approved and carried out in accordance with the guidance provided by the United Kingdom Home Office, Animals (Scientific Procedures) Act 1986 (Retinal Recordings), by the institutional Istituto Italiano di Tecnologia (IIT) Ethic Committee, and by the Italian Ministry of Health and Animal Care (Authorization ID 227, Prot. 4127 March 25, 2008) (neural cultures; Panas et al., 2015).

Experiments on the retina were performed on adult wild-type mice (C57BL/6, aged post-natal days [P] 27–39) or on B6.Cg-Tg(Thy1-COP4/EYFP)9Gfng/J mice (Thy1-ChR2-YFP; The Jackson Laboratory; RRID:IMSR_JAX:007615) aged P69–96. Recordings from the RGC layer were performed using the BioCam4096 platform with active pixel sensor (APS) MEA chips (type BioChip 4096S, 3Brain), providing 4,096 square microelectrodes (21μm × 21μm) on an active area of 2.67 mm × 2.67 mm, aligned in a square grid with 42 μm spacing. The platform records at a sampling rate of about 7 kHz/electrode when measuring from the full 64 × 64 MEA, but sampling increases to 24 kHz when recording from 1,024 electrodes. Raw data were visualized and recorded with the 3Brain proprietary BrainWave software. Activity was recorded at 12-bit resolution per electrode, low pass-filtered at 5 kHz with the on-chip filter, and high pass-filtered by setting the digital high pass filter of the platform at 0.1 Hz.

Mice were killed by cervical dislocation and enucleated prior to retinal isolation. The isolated retina was placed, RGC layer facing down, onto the MEA (for details, see Maccione et al., 2014). The retina was continuously perfused with artificial cerebrospinal fluid (maintained at 32°C) containing the following: 118 mM NaCl, 25 mM $NaHCO_3$, 1 mM NaH_2PO_4, 3 mM KCl, 1 mM $MgCl_2$, 2 mM $CaCl_2$, and 10 mM glucose, equilibrated with 95% O_2 and 5% CO_2. All preparations were performed under dim red light, and the room was maintained in darkness throughout the experiment.

Visual and Optogenetic Stimulation

We used a custom-made projection system to deliver visual stimuli to the retina (for details, see Portelli et al., 2016). Photoreceptor-driven responses were acquired at a maximum irradiance of 4μW/cm^2 (neutral density (ND) filter 4.5), low enough to avoid eliciting ChR2-driven responses in the ChR2 retinas. ChR2-driven responses were elicited using the broad RGB spectrum of the projector with a maximum irradiance of 0.87 mW/cm^2 (ND 2.2) following blockade of photoreceptor-driven responses by increasing $[MgCl_2]_{out}$ to 2.5 mM and by decreasing $[CaCl_2]_{out}$ to 0.5 mM (to reduce synaptic transmission) and in the presence of 20μm DNQX and 20μm L-AP4 (Tocris Bioscience) to block glutamatergic neurotransmission in the photoreceptor-bipolar cell-RGC pathway. Responses to repetitive (30×) full field stimuli (0.5 Hz) were analyzed as shown in Figures 6 and 7.

Spike Detection, Localization, and Selection

The procedures for spike detection and localization are described in detail elsewhere (Muthmann et al., 2015). Weighted interpolated signals were generated using two spatial templates to capture both spikes originating close to or between electrodes. The running baseline and noise estimate were computed as signal percentiles, and putative spikes were detected as threshold crossings. This procedure ensures detection of temporally overlapping spikes as long as they leave a distinct spatial footprint. Next, source locations were estimated for each event by considering the spatial signal spread over neighboring electrodes. The signals were baseline-subtracted and inverted, and then the median signal was subtracted to minimize bias because of noise. The signal was clipped to positive values, and the center of mass was determined. To filter out noise and poorly detected neurons in recordings at 7 kHz, we developed an automated post hoc event rejection. To this end, noise events were sampled from areas on the MEA where no activity was recorded, such as at incisions or uncovered areas (identifiable by low spike counts). Up to 1,000 of such events and up to 1,000 events with large amplitudes were used to train a support vector machine with radial basis functions. This model was then used to classify events as true spikes or noise (Figure S1).

Spike Clustering

Data points were clustered using an implementation of the mean shift algorithm (Comaniciu and Meer, 2002), available in the scikit-learn open source machine learning library (Pedregosa et al., 2011). Importantly, this algorithm does not require prior knowledge of the desired number of clusters. It depends on a single parameter, the bandwidth h, which determines the expected cluster size, which, in turn, can be estimated from a typical spatial cluster size in an activity plot (Figure 1B) and was here set to 12.6 μm (the average width of clusters). The clustering process was run on a four-dimensional space consisting of two dimensions, indicating the location of each event on the chip, x and y, and two dimensions representing the first two principal components of the event's waveform. The latter were multiplied by an additional dimensional constant α that tuned the relative importance of the waveform components compared with the spatial coordinates. To parallelize this algorithm, we exploited the fact that all points follow a local density gradient until they converge to a local maximum, the center of a cluster. Because every data point does so independently of the others, this process is run in parallel, which improved performance roughly proportionally to the number of available central processing units (CPUs). The relevant code has been merged into the scikit-learn Python library.

Quality Metric

Following Hill et al. (2011), we fitted a multivariate Gaussian mixture model to a set of N clusters and then estimated their overlap using posterior probabilities to obtain the probability of incorrect assignments under the assumption of a Gaussian cluster shape. The model is fit in six dimensions, with the two spatial coordinates and the projections of the spike waveform along the first four principal components. For each cluster, we assume that only spikes in nearby clusters interfere with the sorting. Therefore, all clusters or spikes within a radius of 42 μm (electrode pitch) are included in the model. To obtain meaningful fits for sets of clusters with very disparate number of spikes, a Gaussian is fit

to each cluster individually before combining them into a mixture model. The assignment quality is evaluated as follows. Let the probability of spike s in cluster c be $P(C=c \mid S=s)$: the estimated fraction of spikes in cluster k that could belong to cluster i is given by $f^p(k,i) = (1/N_k)\sum_{s\in k} P(C=i \mid S=s)$; by generalizing to all other clusters, we obtained the number of false positives in k:

$$f_k^p = \sum_{i\neq k} f^p(k;i) = \sum_{i\neq k}\sum_{s\in k} P(C=i \mid S=s).$$

Correspondingly, the number of false negatives, the fraction of spikes in cluster c that was expected to be assigned to other (i.e., wrong) clusters, was obtained as $f_k^n = \sum_{i\neq k}\sum_{s\in i} P(C=k \mid S=s)$.

The probabilities $P(C=c, S=s)$ were given by mixture model. To obtain a single quality measure, we compute (P_k) and recall (R_k):

$$P_k = \frac{n_k - f_k^p}{n_k}$$
$$R_k = \frac{n_k - f_k^p}{n_k - f_k^p + f_k^n}.$$

The harmonic mean of these quantifies yields the F-score:

$$F_k = 2\frac{P_k R_k}{P_k + R_k}.$$

Confocal Imaging and Image Analysis

To achieve a precise alignment of RGCs with recording electrodes, the retina had to be imaged on a chip with photoreceptors facing upward. The retina was fixed with 4% paraformaldehyde (in 0.1 M PBS and 200 mM sucrose) on the MEA chip for 1 hr after recording. We have determined that tissue shrinkage, which may interfere with activity alignment, is negligible for this protocol. The retina was rinsed several times with 0.1 M PBS, embedded with Vectashield (Vector Laboratories), and sealed with a coverslip (Menzel Glaeser). Imaging was performed with a Leica SP5 confocal upright microscope supplied with a 25×/0.95 numerical aperture (NA) working distance (WD) 2.5 mm water immersion objective for optimal signal collection focusing on areas encompassing 8 × 8 electrodes (300 × 300μm field of view). In each field, images (2,048 × 2,048 pixels) were acquired in z stacks in tissue thickness of 60–100 μm (optical slicing yielding 30–50 image planes). A lateral resolution of 200 nm per pixel, just above the diffraction limit, and optical slicing of 550 nm provided an adequate trade-off between sufficient image details and acquisition time, minimizing the risk of photo damage. For image restoration, the Richardson-Lucy method (Lucy, 1974; Richardson, 1972) was used. In addition to the fluorescence signals in specific fields, large-field images, including images of the MEA, were acquired to enable co-localization of images with RGC spiking activity.

In one Thy1 YFP-ChR2 retina, RGC somata were manually annotated in selected subfields where activity was recorded, and the confocal images of the RGC layer were spatially aligned with the estimated locations of detected events. To this end, the active area of one electrode was determined, and the remaining electrode locations were computed, generating a regular grid using 42 μm electrode spacing. The images and soma locations were then transformed into array coordinates, and spike locations were overlaid with the retinal image.

SUPPLEMENTAL INFORMATION

Supplemental Information includes Supplemental Experimental Procedures and one figure and can be found with this article online at http://dx.doi.org/10.1016/j.celrep.2017.02.038.

AUTHOR CONTRIBUTIONS

Conceptualization, G.H., M.S., E.S., and M.H.H.; Methodology, G.H, M.S., J.O.M., S.P., I.E.K., S.U., E.S., and M.H.H.; Software, M.S., S.P., J.O.M., C.J.R., A.P.E., and M.H.H.; Formal Analysis, M.S., S.U., and M.H.H.; Investigation, G.H., M.S., S.P., I.E.K., and S.U.; Resources, A.M. and L.B.; Data Curation, G.H., A.M., L.B., and E.S.; Writing – Original Draft, M.H.H. with input from co-authors; Supervision, L.B., V.M., D.S., F.C.Z, E.S., and M.H.H.; Funding Acquisition, L.B., V.M., D.S., F.C.Z., E.S., and M.H.H.

ACKNOWLEDGMENTS

We thank Fernando Rozenblit, Vidhyasankar Krishnamoorthy, and Amos Storkey for valuable input. This work was supported by the 7th Framework Program for Research of the European Commission (grant agreement 600847: RENVISION, project of the Future and Emerging Technologies [FET] program Neuro-bio-inspired Systems FET-Proactive Initiative) and the Wellcome Trust (grant number 096975/Z/11/Z). M.S. was supported by the EuroSPIN Erasmus Mundus Program and the EPSRC Doctoral Training Centre in Neuroinformatics (EP/F500385/1 and BB/F529254/1). J.O.M. was supported by the EuroSPIN Erasmus Mundus Program and NCBS/TIFR.

Received: June 20, 2016
Revised: November 21, 2016
Accepted: February 13, 2017
Published: March 7, 2017

REFERENCES

Ballini, M., Muller, J., Livi, P., Chen, Y., Frey, U., Stettler, A., Shadmani, A., Viswam, V., Jones, I.L., Jackel, D., et al. (2014). A 1024-channel CMOS microelectrode array with 26,400 electrodes for recording and stimulation of electrogenic cells in vitro. IEEE J. Solid-State Circuits *49*, 2705–2719.

Berdondini, L., van der Wal, P.D., Guenat, O., de Rooij, N.F., Koudelka-Hep, M., Seitz, P., Kaufmann, R., Metzler, P., Blanc, N., and Rohr, S. (2005). High-density electrode array for imaging in vitro electrophysiological activity. Biosens. Bioelectron. *21*, 167–174.

Chandrasekhar, S. (1943). Stochastic problems in physics and astronomy. Rev. Mod. Phys. *15*, 1.

Comaniciu, D., and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. IEEE Trans. Pattern Anal. Mach. Intell. *24*, 603–619.

Dragas, J., Jackel, D., Hierlemann, A., and Franke, F. (2015). Complexity Optimisation and High-Throughput Low-Latency Hardware Implementation of a Multi-Electrode Spike-Sorting Algorithm. IEEE Trans. Neural Syst. Rehabil. Eng. *23*, 149–158.

Eversmann, B., Jenkner, M., Hofmann, F., Paulus, C., Brederlow, R., Holzapfl, B., Fromherz, P., Merz, M., Brenner, M., Schreiter, M., et al. (2003). A 128 128 CMOS Biosensor Array for Extracellular Recording of Neural Activity. IEEE J. Solid-State Circuits *38*, 2306–2317.

Frey, U., Sedivy, J., Heer, F., Pedron, R., Ballini, M., Mueller, J., Bakkum, D., Hafizovic, S., Faraci, F.D., Greve, F., et al. (2010). Switch-matrix-based high-density microelectrode array in CMOS technology. IEEE J. Solid-State Circuits *45*, 467–482.

Harris, K.D., Henze, D.A., Csicsvari, J., Hirase, H., and Buzsáki, G. (2000). Accuracy of tetrode spike separation as determined by simultaneous intracellular and extracellular measurements. J. Neurophysiol. *84*, 401–414.

Hill, D.N., Mehta, S.B., and Kleinfeld, D. (2011). Quality metrics to accompany spike sorting of extracellular signals. J. Neurosci. *31*, 8699–8705.

Hutzler, M., Lambacher, A., Eversmann, B., Jenkner, M., Thewes, R., and Fromherz, P. (2006). High-resolution multitransistor array recording of electrical field potentials in cultured brain slices. J. Neurophysiol. *96*, 1638–1645.

Lewicki, M.S. (1998). A review of methods for spike sorting: the detection and classification of neural action potentials. Network *9*, R53–R78.

Lindén, H., Tetzlaff, T., Potjans, T.C., Pettersen, K.H., Grün, S., Diesmann, M., and Einevoll, G.T. (2011). Modeling the spatial reach of the LFP. Neuron *72*, 859–872.

Lucy, L.B. (1974). An iterative technique for the rectification of observed distributions. Astron. J. *79*, 745.

Maccione, A., Hennig, M.H., Gandolfo, M., Muthmann, O., van Coppenhagen, J., Eglen, S.J., Berdondini, L., and Sernagor, E. (2014). Following the ontogeny of retinal waves: pan-retinal recordings of population dynamics in the neonatal mouse. J. Physiol. *592*, 1545–1563.

Marre, O., Amodei, D., Deshmukh, N., Sadeghi, K., Soo, F., Holy, T.E., and Berry, M.J., 2nd. (2012). Mapping a complete neural population in the retina. J. Neurosci. *32*, 14859–14873.

Mechler, F., Victor, J.D., Ohiorhenuan, I., Schmid, A.M., and Hu, Q. (2011). Three-dimensional localization of neurons in cortical tetrode recordings. J. Neurophysiol. *106*, 828–848.

Müller, J., Ballini, M., Livi, P., Chen, Y., Radivojevic, M., Shadmani, A., Viswam, V., Jones, I.L., Fiscella, M., Diggelmann, R., et al. (2015). High-resolution CMOS MEA platform to study neurons at subcellular, cellular, and network levels. Lab Chip *15*, 2767–2780.

Muthmann, J.-O., Amin, H., Sernagor, E., Maccione, A., Panas, D., Berdondini, L., Bhalla, U.S., and Hennig, M.H. (2015). Spike Detection for Large Neural Populations Using High Density Multielectrode Arrays. Front. Neuroinform. *9*, 28.

Ness, T.V., Chintaluri, C., Potworowski, J., Łęski, S., Głąbska, H., Wójcik, D.K., and Einevoll, G.T. (2015). Modelling and Analysis of Electrical Potentials Recorded in Microelectrode Arrays (MEAs). Neuroinformatics *13*, 403–426.

Obien, M.E.J., Deligkaris, K., Bullmann, T., Bakkum, D.J., and Frey, U. (2015). Revealing neuronal function through microelectrode array recordings. Front. Neurosci. *8*, 423.

Panas, D., Amin, H., Maccione, A., Muthmann, O., van Rossum, M., Berdondini, L., and Hennig, M.H. (2015). Sloppiness in spontaneously active neuronal networks. J. Neurosci. *35*, 8480–8492.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., et al. (2011). Scikit-learn: Machine learning in Python. J. Mach. Learn. Res. *12*, 2825–2830.

Pettersen, K.H., and Einevoll, G.T. (2008). Amplitude variability and extracellular low-pass filtering of neuronal spikes. Biophys. J. *94*, 784–802.

Portelli, G., Barrett, J.M., Hilgen, G., Masquelier, T., Maccione, A., Di Marco, S., Berdondini, L., Kornprobst, P., and Sernagor, E. (2016). Rank order coding: A retinal information decoding strategy revealed by large-scale multielectrode array retinal recordings. eNeuro *3*.

Prentice, J.S., Homann, J., Simmons, K.D., Tkačik, G., Balasubramanian, V., and Nelson, P.C. (2011). Fast, scalable, Bayesian spike identification for multi-electrode arrays. PLoS ONE *6*, e19884.

Quiroga, R.Q., Nadasdy, Z., and Ben-Shaul, Y. (2004). Unsupervised spike detection and sorting with wavelets and superparamagnetic clustering. Neural Comput. *16*, 1661–1687.

Raymond, I.D., Vila, A., Huynh, U.-C.N., and Brecha, N.C. (2008). Cyan fluorescent protein expression in ganglion and amacrine cells in a thy1-CFP transgenic mouse retina. Mol. Vis. *14*, 1559–1574.

Rey, H.G., Pedreira, C., and Quian Quiroga, R. (2015). Past, present and future of spike sorting techniques. Brain Res. Bull. *119* (*Pt B*), 106–117.

Richardson, W.H. (1972). Bayesian-based iterative method of image restoration. J. Opt. Soc. Am. *62*, 55–59.

Rossant, C., Kadir, S.N., Goodman, D.F.M., Schulman, J., Hunter, M.L.D., Saleem, A.B., Grosmark, A., Belluscio, M., Denfield, G.H., Ecker, A.S., et al. (2016). Spike sorting for large, dense electrode arrays. Nat. Neurosci. *19*, 634–641.

Shoham, S., Fellows, M.R., and Normann, R.A. (2003). Robust, automatic spike sorting using mixtures of multivariate t-distributions. J. Neurosci. Methods *127*, 111–122.

Ventura, V., and Gerkin, R.C. (2012). Accurately estimating neuronal correlation requires a new spike-sorting paradigm. Proc. Natl. Acad. Sci. USA *109*, 7230–7235.

Figure S1 and description of methods for shape-based event filtering (Related to Figure 2)

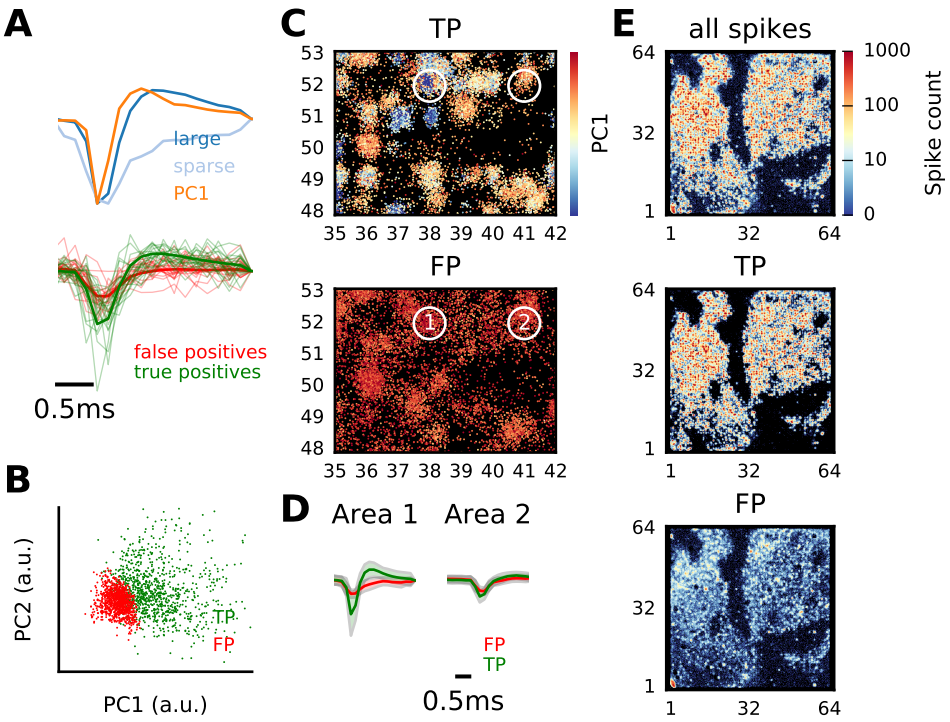
Spike detection was generally performed with a low threshold to reduce false negatives. In particular in recordings with low sampling rates (7 kHz), this could yield events that were clearly not spikes, but threshold crossings due to noise. For instance, a small number of events are usually detected in areas of the chip not covered by tissue (Figure S1E, top). This could be easily avoided by increasing the detection threshold, or by introducing additional shape criteria to remove such events during the detection phase. We however found that this typically leads to an unreasonable increase in the fraction of false negatives (Muthmann et al., 2015), and it is generally difficult to determine appropriate detection parameters a priori to avoid false positives consistently. Therefore, a method for post hoc event selection was developed, which also helped to compensate for variations between preparations.

To distinguish between true and false positives, examples of events with either high amplitudes or from areas with very low spike density were randomly chosen from the data and used to train a radial basis function Support Vector Machine (SVM) classifier. High amplitude events showed the typical biphasic spike waveform, which resembled the first principal component (PC) estimated from all events, while low density events lacked the repolarization (Figure S1A). This confirmed the premise that regions with very low spike density contained almost exclusively noise. A classifier trained on these examples separated events roughly, but not exactly along projections along the first PC (Figure S1B).

The comparison in Figure S1C showed that events classified as true positives were typically part of localized spatial density peaks, while false positives were more homogeneously distributed with low density (Figure S1C). True positives show clear, biphasic waveforms (area 1 in Figure S1C,D). Yet the separation would still remain ambiguous for small events with amplitudes closer to the noise level, where events classified as false positives still showed spatial clustering (area 2 in Figure S1C,D), suggesting the presence of poorly detected current sources as well as other events related to neural activity such as strong synaptic currents (Muthmann et al., 2015). As shown in Figure S1E for a recording from 64x64 channels at 7kHz, most of the spatial structure was retained in the map of spikes classified as true positives, while events in areas where no spikes were expected (e.g. optic disk, incisions) were correctly removed. On the other hand, the map of false positives showed weak spatial clustering in areas with high activity. This indicated that the spike record of some neurons with weak signals was most likely incomplete, and a further selection of events had to be performed after spike sorting.

References

Muthmann, J.-O., Amin, H., Sernagor, E., Maccione, A., Panas, D., Berdondini, L., Bhalla, U. S., & Hennig, M. H. (2015). Spike Detection for Large Neural Populations Using High Density Multielectrode Arrays. Frontiers in Neuroinformatics, 9, 1–21.



Supplemental Figure S1. Related to Figure 2. Filtering of detected spikes through spike shape classification.
(A) Average waveforms of events sampled from areas with low event density ("sparse") and with high amplitude ("large") samples, which were used to train the classifier (top). Example waveforms of events classified as true and false positives are shown below.
(B) Projections along the first two principal components (PCs) of waveforms classified as true and false positives (TP, green, and FP, red, respectively).
(C) Events classified as true (top) and false positives (bottom), at their estimated locations. Color indicates the projection along the first PC.
(D) Average waveforms of all TP and FP in the two circled areas in panel C.
(E) Spatial event density maps for a complete recording. Shown are all spikes (top), true (middle) and false positives (bottom). Data in this figure are from the same retina as in Figure 1 of the main text, but recorded at 7 kHz.

Chapter 2

Restricted Boltzmann Machines as models of neural activity

> [T]he power of retinal computation should be judged by how much raw information it filters out for discard, not by the amount it preserves.
>
> [Gollisch and Meister, 2010]

Boltzmann machines (BM) are a neural network model composed of interacting units with the particular property of being *binary* and *symmetrically interacting*. They are named after the Austrian physicist Ludwig Boltzmann (1844-1906), because of the close analogy between these networks and the physical systems Boltzmann studied while developing a new branch of physics, statistical mechanics. More specifically, Boltzmann machines are stochastic systems that obey the Boltzmann probability distribution.

BMs were introduced as computational models in the 1980s [Ackley et al., 1985], but find widespread application to the present day, particularly in the form of Restricted Boltzmann Machines (RBMs), which are defined by an additional constraint: their units are organised in layers, with no connections within a layer, and all-to-all connections between a layer and the next. The simplest model of RBM, which I will focus on in this chapter, is formed of two layers only, one of which is defined as the *visible* layer, while the other is called *hidden*. The objective of RBM training is to learn a probability distribution over a set of binary variables.

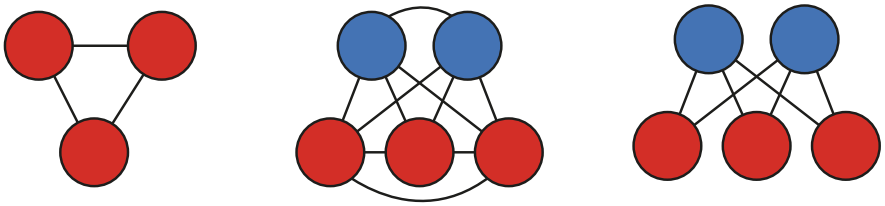


Figure 2.1: Schematics of Boltzmann machines. Left: fully visible Boltzmann machine (FVBM). Centre: the most general case of pairwise Boltzmann machine. Right: restricted Boltzmann machine (RBM). Visible units (in red) are the ones that reproduce the distribution given in the data.

The units in the visible layer, $\mathbf{v}$, are the ones that will reproduce this distribution, and those in the hidden layer, $\mathbf{h}$, act as a link between the former. We can then say that the RBM is a generative model that mimics the distribution of the data.

2.1 Restricted Boltzmann machines

The Boltzmann machine is, in general, a *log-linear* (or *energy-based*) model — that is, the probability distribution of its states can be written as

$$P(\mathbf{v}, \mathbf{h}) = \frac{1}{Z} e^{-E(\mathbf{v},\mathbf{h})}. \tag{2.1}$$

Since, for an RBM, the energy function depends solely on pairwise interactions between the hidden and the visible layer, and, linearly, on the values of the hidden and visible units, it is defined in the form

$$E(\mathbf{v}, \mathbf{h}) = -\mathbf{a}^\intercal \mathbf{v} - \mathbf{b}^\intercal \mathbf{h} - \mathbf{v}^\intercal J \mathbf{h} \tag{2.2}$$

where $\mathbf{v}$ is the vector of visible units' activations, $\mathbf{h}$ are the hidden units' activations, and $\mathbf{a}$, $\mathbf{b}$ and J are parameters corresponding to visible biases, hidden biases, and interactions, respectively. In equation (2.1), Z is a normalisation constant, dependent on the parameters.

2.1.1 Sampling and fitting

The value of Z as a function of $J, \mathbf{a}$ and $\mathbf{b}$ (also called the partition function, because of an analogy with statistical physics [Jaynes, 1957]) cannot easily be computed for large models, because its calculation involves summing over all

possible states of the model:

$$Z = \sum_{\{\mathbf{v}\}} \sum_{\{\mathbf{h}\}} e^{-E(\mathbf{v},\mathbf{h})},$$

where $\{\mathbf{v}\}$ indicates all 2^{N_v} possible values of $\mathbf{v}$, and analogously for the hidden layer. However, directly from (2.1) and (2.2), it is easy to see that

$$P(v_i|\mathbf{h}) = \frac{1}{1 + e^{-a_i - \sum_j J_{ij} h_j}}. \tag{2.3}$$

In other words, even when unable to explicitly compute Z, we do have an explicit and computationally cheap way to obtain values for the visible units, given the hidden, and vice versa, when the parameters are given. This is useful because, once a sample for $\mathbf{h}$ is available, we can obtain one for $\mathbf{v}$, and vice-versa, and thereby generate an arbitrarily long Markov chain of samples both for the visible and hidden units. This is a particular case of a more general Monte Carlo method called Gibbs sampling: it can be proven that, at convergence, this chain provides a correct sample of (2.1) even starting from arbitrary states [Tierney, 1994].

We have a sampling procedure, but we still need to find a way of fitting the RBM, i.e. to find values of $\mathbf{a}, \mathbf{b}$, and J such that the marginal probability for the visible units' activations, $P(\mathbf{v})$, match a given distribution $p_0(\mathbf{v})$ (the distribution of the data we want to fit). The most obvious way is maximising the likelihood by gradient descent. The negative log-likelihood of a given $(\mathbf{v^0}, \mathbf{h^0})$ pair reads

$$\mathcal{L}(\mathbf{v^0}, \mathbf{h^0}) = E + \log Z = -\mathbf{a}^\intercal \mathbf{v^0} - \mathbf{b}^\intercal \mathbf{h^0} - \mathbf{v^0}^\intercal J \mathbf{h^0} + \log \sum_{\{\mathbf{v}\}} \sum_{\{\mathbf{h}\}} e^{\mathbf{a}^\intercal \mathbf{v} + \mathbf{b}^\intercal \mathbf{h} + \mathbf{v}^\intercal J \mathbf{h}}.$$

This can be derived from (2.1) and (2.2) by applying the definition of likelihood. It is then easy to compute the gradient w.r.t. the parameters:

$$\frac{\partial \mathcal{L}(\mathbf{v^0}, \mathbf{h^0})}{\partial a_i} = \langle v_i \rangle - v_i^0; \qquad \frac{\partial \mathcal{L}(\mathbf{v^0}, \mathbf{h^0})}{\partial b_j} = \langle h_j \rangle - h_j^0; \tag{2.4}$$

$$\frac{\partial \mathcal{L}(\mathbf{v^0}, \mathbf{h^0})}{\partial J_{ij}} = \langle v_i h_j \rangle - v_i^0 h_j^0.$$

Here, the averages are computed according to the current values of the RBM parameters, and $\mathbf{v^0}$ indicates the pattern we are learning, given by the data. By definition, however, hidden units are not included in the training data, so the value of $\mathbf{h^0}$ is simply substituted with the average value of the propagation of the observed pattern into the hidden layer according to the current parameters:

$$\langle h_j^0 \rangle = \frac{1}{1 + e^{-b_j - \sum_i J_{ij} v_i}}.$$

However, this needs a final ingredient: computing the averages in (2.4) requires sampling of the RBM defined by the current parameters. In fact, it turns out that waiting for the Gibbs chain to converge is not necessary. The chain can be started by using an element of the training set, or using the chain state at the previous training step (a "persistent" chain, [Tieleman, 2008]). In this case, very few Gibbs steps are needed for an estimation of the averages: even a single sample can be reliably used [Hinton, 2002, Bengio et al., 2009]. This technique is called *contrastive divergence minimisation* (CD-k, where k is the number of sampling steps). In summary, thanks to Gibbs sampling and this approximation, RBMs are equipped with efficient reliable sampling and fitting algorithms.

2.1.2 Application to neural recordings

Fully visible Boltzmann machines (FVBMs) are equivalent to the pairwise maximum entropy model discussed in the Introduction, and correspond, in physics, to a spin glass (a general case of Ising model, with arbitrary couplings). They have been used for fitting neural data since the mid-2000s [Shlens et al., 2006, Schneidman et al., 2006]. As explained in section 0.2.3, the idea behind that work is to model the statistical properties of neuronal networks activity. While RBMs and FVBMs perform a similar task (fitting a dataset), the research questions asked in the two cases can be very different, as the information the models provide is different.

RBMs have been used to model neural activity before: [Köster et al., 2014] showed that RBMs can model visual cortex activity significantly better than pairwise maximum entropy models (that is, fully visible Boltzmann machines), leading to the conclusion, in line with previous research, that higher order correlations play a role in shaping population-level activity. Additionally, they report that some hidden units connect with specific neurons in a way that reflects the actual organisation of a cortical column. However, the drawback of their study was the low number of neurons involved ($N = 10$), which was constrained, due to the experimental methods.

The use of high-density multielectrode arrays allowed for an increase in the number of recorded neurons. [Spicher, 2014] analogously showed that RBMs outperform pairwise maximum entropy models in modelling mouse retinal ganglion cells (RGCs) recordings. Using similar recordings, [Zanotto et al., 2017] demon-

strated that the activity of the latent units carry information about the stimuli presented to the retina. The same research group applied RBMs to electro-encephalographic and magneto-encephalographic recordings in order to identify sub-stages of sleep in mice in an unsupervised way [Katsageorgiou et al., 2018].

Finally, very recently, [Gardella et al., 2018] designed a neural metric, that is, a measure of the distance between two spike trains, which can be learned directly from data by training an RBM, and showed it performs significantly better, in terms of discriminability, than previously popular neural metrics. In their article, they also define this distance for t-RBMs, which are RBMs that involve additional visible units in order to simultaneously describe the activity at two or more consecutive times, so that they are also able to account for correlations in time. Working with t-RBMs was outside of the scope of this chapter, and I have not experimented with their use. However, they are a stronger and more realistic model, which is surely among the possible future directions of the work presented here.

The mere observation that RBMs are capable of fitting the dataset is not sufficient to show how they could be used to gain insight on the structure and function of neuronal systems: the essential characteristic of a model in the applications is interpretability. Although quite a few articles, as reported above, have also shown practical uses for RBMs, their potential as statistical models for the analysis of neural activity has not been completely exhausted — in particular, their ability to apply a form of dimensionality reduction to the data, to discover underlying structure.

In this chapter, I will first replicate the results that show how RBMs are capable of an accurate fit of neural data, particularly using retinal recordings like the ones I dealt with in the previous chapter; this is a necessary step in order to ensure the correctness of any other result. In section 2.3, I will then use more retinal recordings to interpret the roles of hidden units in reproducing the statistical structure of the activity. I will show that hidden units perform a partial unsupervised decoding of the stimulus, at least in simple cases. Most importantly, they are useful for identifying neurons with similar characteristics: neurons that correlate because of input correlations; neurons that share the same ON/OFF response pattern; neurons with similar temporal profiles of response. Because of this ability of RBMs to factorise the activity into components, reducing dimensionality, I propose they could be used to identify ensembles of co-activating neurons, or

the relevant dimensions underlying neural population activity. Recent literature has found that despite the large state space accessible to the dynamics of any neuronal network, only a small number of dimensions, referred to as *modes*, are actually spanned by experimentally observed activity [Gao et al., 2017, Gallego et al., 2017]. In section 2.4, using a simple computational model of mode-driven activity, I verify that the RBM hidden units correlate with latent states, recovering this structure.

2.2 Evaluating fits

The first, essential quality that any generative model should have is, obviously, the ability to generate samples that are representative of the data it was trained on. Since we are training by contrastive divergence, we do not have access to an exact likelihood value to use for evaluation. However, a number of measures can help us understand how accurate the fit is, and what properties of the dataset are reliably reproduced. I have checked several of these, similarly to what is commonly done in the literature [Köster et al., 2014, Gardella et al., 2018]:

- Average activity and pairwise correlations, that is, in RBM notation, the average activation of the visible units, $\langle v_1 \rangle, ..., \langle v_{N_v} \rangle$, and Pearson correlations between neurons:

  $$\rho_{i,j} = \frac{\mathrm{Cov}[v_i, v_j]}{\sqrt{\mathrm{Var}[v_i]\,\mathrm{Var}[v_j]}}, \quad i, j = 1, ..., N_v.$$

  These measures are by definition correctly fitted by a successful Pairwise Maximum Entropy model (PME, equivalent to a FVBM). Our RBMs must provide sufficient accuracy on them in order to be comparable with PME [Tkačik et al., 2014].

- Higher-order correlations (or cumulants). Due to computational constraints, I considered cumulants up to the third order, using the definition

  $$C^{(3)}_{ijk} = \langle\, (v_i - \langle v_i \rangle)(v_j - \langle v_j \rangle)(v_k - \langle v_k \rangle) \,\rangle\,.$$

  It can be proven that any distribution over a set of binary variable can be uniquely determined by knowing cumulants of all orders. This measure shows that the model fits the distribution better than by just fixing average rates and pairwise correlations [Tkacik et al., 2006].

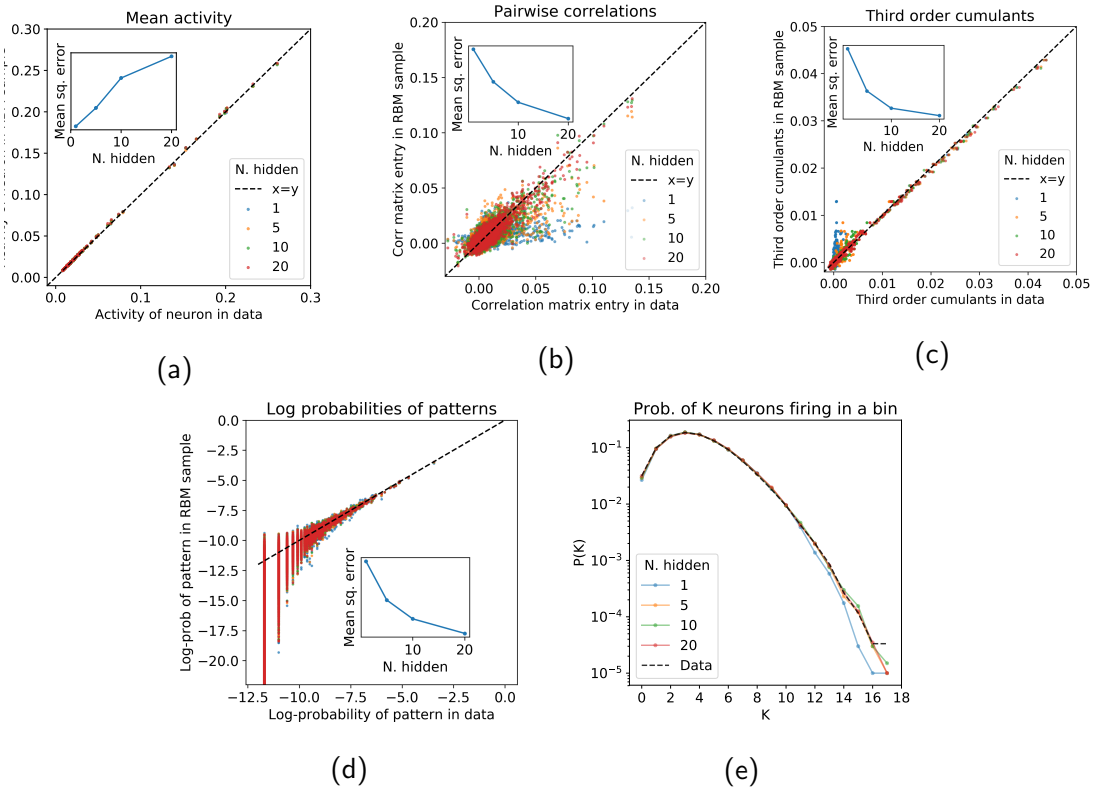


Figure 2.2: Measures of goodness of fit for RBMs with different numbers of hidden units (here, $N_h = 1, 5, 10, 20$), for $N_v = 100$ visible units (fits are significantly better for lower N_v). (a) mean activities of units $\langle v_i \rangle$; (b) pairwise correlations; (c) third order cumulants $C^{(3)}_{ijk}$; (d) Log probabilities of each observed pattern, in a sample vs. in data; (e) probability distribution of all-neurons activity in a time bin $P(k)$. Insets show the mean squared errors between data and sample for each measure, as a function of N_h. For each panel, the insets show the mean squared error between data and RBM sample, as a function of N_h.

- Distribution of same-time population activation. Let $K = \sum_{i=1}^{N_h} v_i$, that is, the number of neurons firing in a given time bin. The distribution $P(K)$ is a descriptor of population activity, and has also been used to fit maximum entropy models [Tkačik et al., 2014]. Comparing $P(K)$ in the model and in a sample shows if RBMs can perform comparably well with respect to these other models, and provide a measure of how well they represent the network activity at the population level.

The evaluation of all these measures is illustrated in detail in figure 2.2 for RBMs fitted to the binned neural activity of 100 retinal neurons. For this dataset,

the retina was stimulated with four different protocols, corresponding to different spatial and temporal correlations (see section 2.5). For all reported measures, it can be seen that the RBM does not show evident biases in estimating the desired feature of the dataset. Firing rates are fitted perfectly already at $N_h = 1$. The MSE of this statistics unexpectedly increases slightly for larger models, but its absolute value stays so small ($\approx 10^{-3}$) that this should not be taken as a hint of poor fit quality, and the values are all within 33% of each other. In fact, other experiments (not shown) exhibit a MSE value that fluctuates randomly with N_h: I conclude that this pattern is due to chance. Pairwise correlations are harder to fit properly, and smaller models ($N_h = 1, 5$) systematically underestimate them. At $N_h = 20$, there is no systematic bias, although random errors are always present: this is because RBMs, unlike maximum entropy models, do not explicitly fit correlation values, but optimise for similarity of the whole distribution, and this may prioritise different properties. A further increase in N_h would reduce these errors, but would require additional training data for an unbiased estimation of the correlations. Third-order cumulants are also very well reproduced by more complex RBMs. Figure 2.2d shows RBMs train themselves to correctly fit patterns at higher probability, their errors being larger and larger at lower probabilities. This is expected, as probabilities for rare events cannot be estimated well from the training data. Finally, the estimation of $P(K)$ does not seem to cause problems, even for one hidden unit.

2.3 The role of hidden units in RBMs fitted to retinal recordings

We know from section 2.2 that RBMs properly learn the probability structure of our neural population recordings. Both [Köster et al., 2014] (for cortical recordings) and [Spicher, 2014] (for the retina, although with an unsorted dataset) additionally showed that RBMs perform better than pairwise models in this modelling task. The RBM provides access to the whole structure of the distribution, not limiting itself to pairwise interactions — higher orders are mediated by the latent factors. But what about interpretation? Certainly, knowing that they fit better than a PME model puts a lower bound on the importance of higher-order correlations for neural coding; however, this could have been achieved in other

ways that took population-level metrics into account. In [Köster et al., 2014], it is found that some hidden units are specifically connected with neurons in one particular location, although this is a weak effect. So, the RBM can hint at the local structure of the cortex. What about the retina? I proceeded by fitting RBMs to various types of retinal recordings; by looking at the role of hidden units, I tried to understand what the model is actually... modelling.

For all the experiments on retinal data, illustrated in this section, I used RBMs with $N_h = 16$ hidden units. Fit quality evaluation showed that this was comfortably enough for a valid reproduction of the experimental pattern probability distribution. In fact, the redundancy of the response patterns of some hidden units may pose a risk of overfitting. However, note that we are interested in the analysis of a single dataset, and the ability of the model to generalise is not a concern.

Responses to on/off stimulation I started with a mouse retina that was subjected to full-field flashes of light: 2 seconds of light (white stimulus), followed by 2 seconds of darkness (black stimulus), all over the recorded area. The responses of retinal ganglion cells to this kind of stimulus, averaged over 30 repetitions of the stimulation protocol, are shown in the bottom panel of figure 2.3. Some responses are sustained for the duration of the dark or light period, even if they decay from the stimulus onset; others peak when the stimulus changes, and quickly become silent again. Some cells are sensitive to ON stimuli only, some to OFF, and some to both. In other words, we can classify RGCs based on two main criteria: the transiency of their response, and their ON/OFF index.

The corresponding plots for hidden units are shown in the top panel of figure 2.3. They were obtained by training an RBM to the data, and, after training, performing a simple forward pass from the recorded response to the hidden layer. It is immediately clear that the hidden units "decode" the stimulus, at least in a simple case like full-field stimulation — in the sense that their activation profile closely reproduces the on/off profile of the light shone on the retina, with very low levels of noise compared to the RGCs themselves (whose response can be seen in the bottom panel). In this protocol, the stimulus variable is binary (light on/light off) and carries one bit of information. A subset of the RGC's hidden units fully correlate with this variable, and carry complete information about the current stimulus value. Although this works well for a binary stimulus, it is not

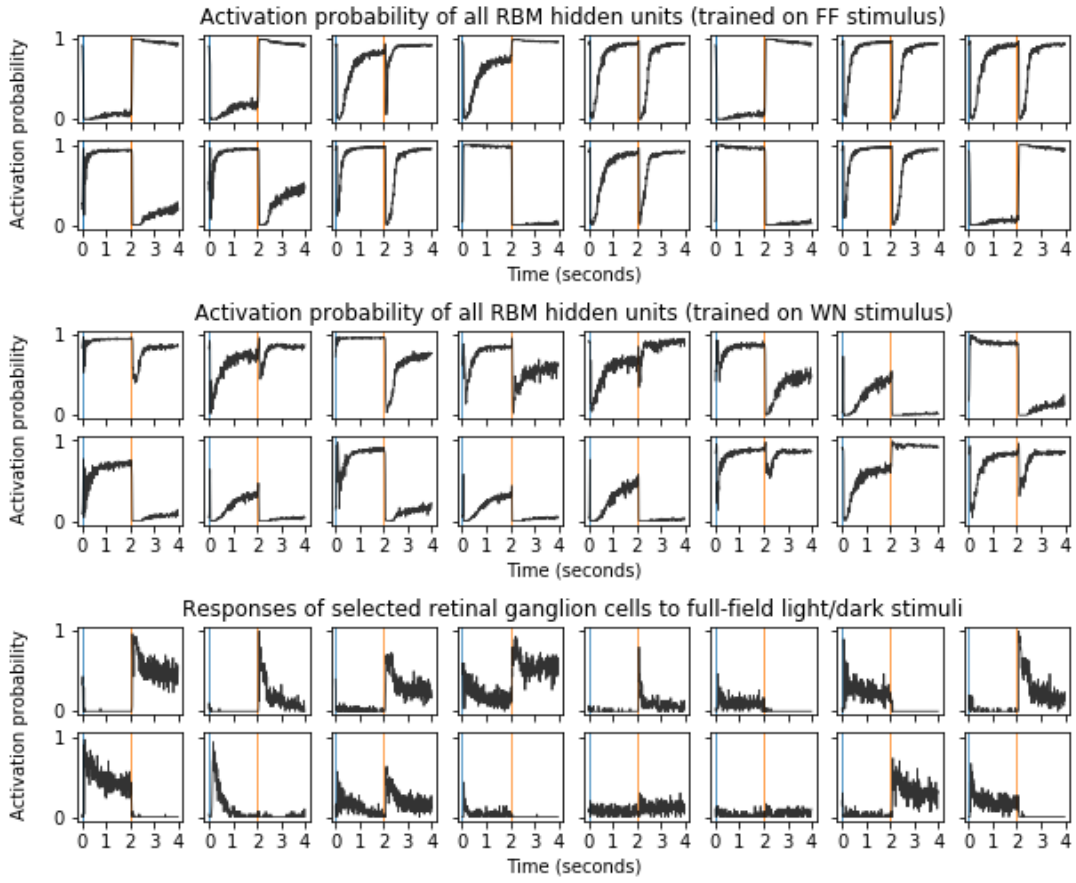


Figure 2.3: Responses to a stimulus consisting of full field light/dark flashes. At time 0, light is projected on the whole retina (blue line); at $t = 2$ s, the light is turned off (orange line). Top: hidden units of an RBM trained on RGCs subjected to full-field stimulus. Middle: hidden units of an RBM trained on RGCs subjected to a random checkerboard stimulus. Bottom: responses of a selection of retinal ganglion cells; sustained and transient responses can be seen, as well as ON and OFF behaviour.

guaranteed to scale well for high-dimensional stimulation protocols.

Alongside units that follow the current stimulus value, there is another main class of hidden units: the ones that exhibit a transient response to a *change* in the stimulus. We can see that the hidden layer has mostly decoupled the two dimensions along which RGCs differ: most units either discriminate between light and dark, or take note of how much time has passed since the last stimulus change. These two ingredients are sufficient in order to describe the responses of most RGCs.

However, the full-field stimulus very strongly drives the retina as a whole, and it is not surprising that this can be easily modelled by RBMs. A more interest-

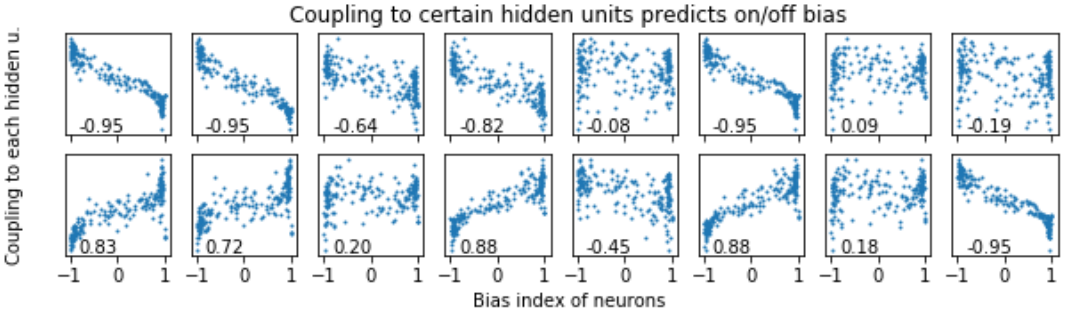


Figure 2.4: The ON/OFF bias index b of all visible units is here compared to the RBM weight between them and each hidden unit. Each panel shows couplings to a single hidden unit, the same one as in the corresponding panel of figures 2.3 (top and middle) and 2.5. Numbers indicate the Pearson correlations.

ing problem is to see if the role of hidden units stays the same independently of stimulus. To this end, I trained an RBM on the RGC responses to a checkerboard stimulation protocol, where each of the 160 µm squares can take a "dark" or "light" value for every stimulus presentation (i.e. every 33.3 ms). I then looked at the values that hidden units take when the RBM is presented the responses to the full-field stimulation dataset. The results are in the middle panel of figure 2.3. Arguably, the hidden units' responses to stimulus do not contain any more information than what we can already read in the responses of the visible units, i.e. of the neurons themselves. However, these results do show that the RBM is able to understand two main factors that drive the activity of the RGC population: stimulus value and the recency of a significant stimulus variation; and it decouples these factors by assigning them to different hidden units. This is interesting for two reasons: it constitutes a way of decoding elementary stimuli, as was done in [Zanotto et al., 2017], although for different stimulation protocols and with a slightly different model, mean-variance RBM. But it also hints at how the RBM is fitting the data, suggesting another application, which will be discussed in section 2.4. Incidentally, it should be noted that some hidden units exhibit nearly identical responses: this is a sign that the number of hidden units is more than sufficient to express the variability in the data. The opposite would likely be a sign of overfitting: when working on an intrinsically low-dimensional dataset, it is desirable that any number of hidden units in excess of the number of modes redundantly reproduce the behaviour of others, rather than fitting random aspects of the data.

RBM hidden units, in conclusion, select groups of units with similar activity patterns. Further evidence of this is shown in figure 2.4. Some of the hidden units have consistently positive connections with OFF cells, and negative connections with ON cells, or vice-versa. The coupling between these hidden units and a given visible unit is strongly related to the on/off bias index b of the cell. The bias index is simply the difference between the cell's response to light and its response to dark, normalised by the total response:

$$b = \frac{r_L - r_D}{r_L + r_D}.$$

r_L and r_D are the average firing rates of the neuron after the onset of white or black stimulus respectively. Therefore, b takes a value of 1 for purely ON cells and -1 for purely OFF, with intermediate values possible [Carcieri et al., 2003, Hilgen et al., 2017a].

Spatial characterisation Let us now look at whether hidden units take a spatial role, connecting to specific parts of the retina. Figure 2.5 shows RGCs in their actual locations within the retina, flattened out on the MEA. For each cell, I computed its correlation with every hidden unit. The top panel, which shows the case of the checkerboard stimulus, shows that several hidden units correlate strongly with cells located in small areas. Naturally, because RGCs often have overlapping receptive fields, and the size of the checkerboard squares is 160 µm, much of the correlations between neighbouring cells are driven, in this stimulation protocol, by the local nature of the common input. This is indeed confirmed by the bottom panel of figure 2.5, which shows the same analysis for full-field stimulated cells. Here, there is no strong evidence of spatial structure, likely because stimulus-independent network correlations are too weak to be observed in this way.

Nevertheless, the local role taken by hidden units under the checkerboard stimulus shows again how the RBM can identify groups of neurons with the same response patterns. In the more general case, it can be said that they group neurons based on their co-activation: under a full-field stimulus, this implies grouping ON cells and OFF cells; under a checkerboard stimulus, it implies grouping together cells that receive the same input. With a more general stimulation pattern, I argue that an RBM could, in principle, be used to systematically study RGC responses in an unsupervised way. Even more generally, in other datasets, the co-activation of cell groups need not be caused by the stimulus, and in fact, it is well

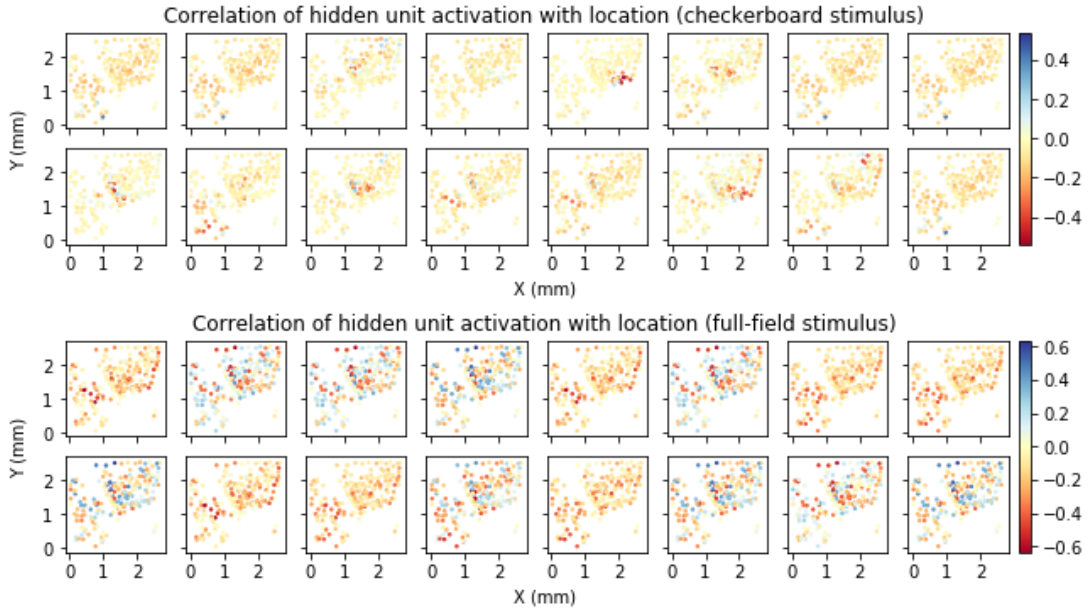


Figure 2.5: Spatial characterisation of the activity of RBM hidden units. Each panel corresponds to a hidden unit; each coloured dot shows the physical location of a neuron (visible unit) in the retina, with the colour indicating the correlation of its activity to the hidden unit. The RBM was trained and tested on responses to a random checkerboard stimulus (top), and to a full-field stimulus (bottom).

known that cell assemblies working together with similar activation patterns exist in various part of the brain. The next section takes a more principled approach, verifying this hypothesis on synthetic data where the activity is structured with the presence of correlated groups of cells, which are known a priori.

2.4 Separation of neural modes

RBMs can be used as a tool for unsupervised factor analysis, i.e. to find underlying structure in data, by identifying latent variables (also called factors, or components). In this case, the RBM hidden units encode the latent factors [Clevert et al., 2015]. In other words, they can be used to apply a form of dimensionality reduction to binary datasets. To my knowledge, this has not yet been exploited in neuroscience applications.

In recent years, however, there has been a growing interest in low dimensional representations of neural activity. Consider a large-scale neural network, governed by any kind of dynamics. The number of states of the network that are possible, in principle, at any given time, is very large — in fact, if all neurons can indepen-

dently be active or inactive, it is exponential in their number. However, experimental evidence shows that the high-dimensional space of theoretically possible network states is never entirely spanned by the observed activity; on the contrary, the latter covers only a few effective dimensions, which are often termed *neural modes* [Sadtler et al., 2014]. The activation of each mode consists in the correlated activation of a number of neurons; conversely, each neuron can contribute to one or more modes. [Sadtler et al., 2014] demonstrated that the accessible low-dimensional space is shaped by the network structure, since it is not easily changed through learning. Later, modes were found to be related to behaviour more closely than the activity of any single neuron in the motor cortex [Gallego et al., 2017], and their nature and activation is preserved during different tasks [Gallego et al., 2018]. The dimensionality reduction techniques used included Factor Analysis (FA) [Byron et al., 2009], Demixed Principal Component Analysis (dPCA) [Kobak et al., 2016] and Nonnegative Matrix Factorisation (NMF) [Onken et al., 2016]. Very recently, the idea was extended to multi-timescale components, using Tensor Component Analysis [Williams et al., 2018]. Figure 2 of [Gao et al., 2017] and the references therein contain a comprehensive summary of other related work.

The previous section has experimentally shown that RBMs can correctly identify groups of neurons that tend to activate together, which is a sign of how they exploit the low dimensionality of neural activity in the data. For a more general approach, I designed a simple model that can generate low-dimensional neural activity described by a small number of modes, and analysed the result using RBMs. First, I will describe how the synthetic data is generated, and its properties. Then, I will show how RBMs are capable or retrieving the "modes": in this case, binary latent variables that drive the activity of the population. I will also compare the results with non-negative matrix factorisation (NMF) and independent component analysis (ICA). Finally, I will consider the case of interacting hidden factors.

2.4.1 Mode-driven binary neuron model

Let N be the number of simulated neurons we are interested in, and M the number of nodes that drive their activity. This model consists, first of all, of an $N \times M$ matrix, whose entries $0 \leq u_{nm} \leq 1$ describe the coupling of neuron n

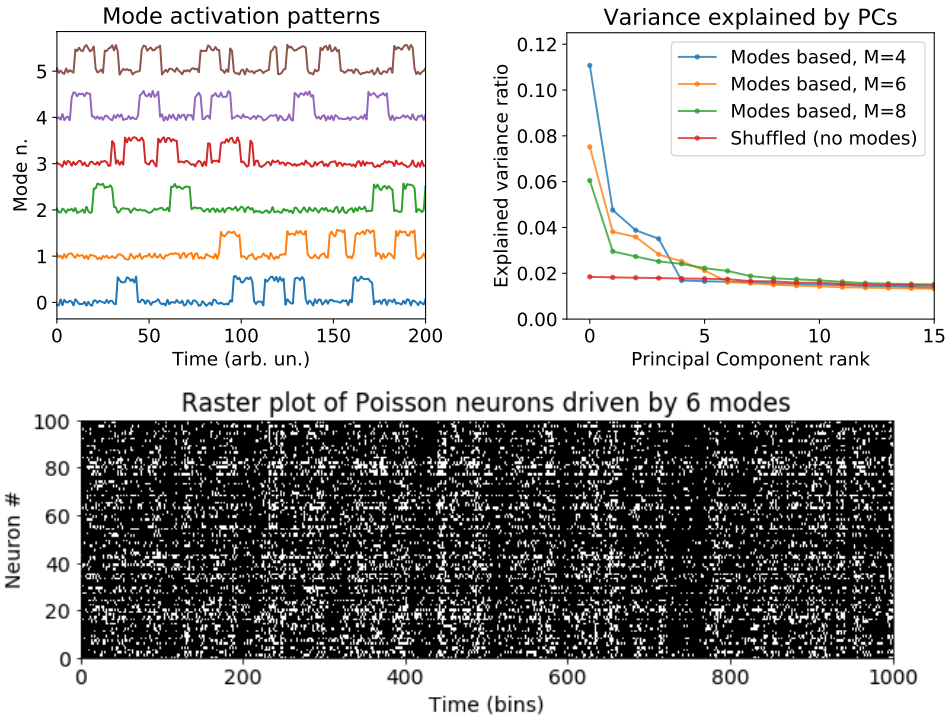


Figure 2.6: Dynamics of Poisson neurons driven by neural modes. Top left: activation and deactivation of the modes in time, $L_m(t)$. Bottom: a snippet of activity shown as a raster plot. Top right: variance explained by the first 15 principal components. Principal component analysis shows mode-driven activity has lower dimensionality.

to mode m. I chose to draw the values of these entries randomly from a beta distribution $B_{\alpha,\beta}(u) \propto u^{\alpha-1}(1-u)^{\beta-1}$, with $\alpha < 0$. This condition ensures a high probability that a neuron is either well coupled to a node, or not coupled at all. The second parameter is chosen as $\beta = \alpha(M-1)$, in order to enforce $\langle u \rangle = 1/M$.

Neurons are simple binary units, with probability of activation given by

$$p_n(t) = f\left(\sum_m u_{nm} L_m(t) + \theta_n\right)$$

where $L_m(t)$ is the state of activation of node m at time t, and f is a nonlinear function (I chose a clipped hyperbolic tangent). θ_n is a constant, different for each neuron, that accounts for individual variability in the firing threshold. The equation above closely resembles the one presented in [Gallego et al., 2017], but, in this case, gives the probability of activation of a binary neuron, as opposed to a value for an instantaneous firing rate as in their case. Time is taken in discrete timesteps.

Parameter		Value
Number of neurons	N	100
Number of modes	M	6
Simulated timesteps	T	10^5
Probability of mode activation	r	0.04
Amplitude of modes	A	0.5
Duration of modes	Δt	21
Parameter of u distribution	α	0.2
Parameter of u distribution	β	$\alpha(M-1)$
Neurons variability	θ_n	$\sim N(0, 0.1)$

Table 2.1: Parameter values for the Poisson neurons model with neural modes.

Mode dynamics Each mode activates independently with probability r at random times, staying active for a fixed number of timebins Δt. The variable describing a mode takes the value $L_m(t) = A$ when the mode is active at time t, and 0 when the mode is inactive, to which a source of noise is added. This model for the modes is shown in figure 2.6 (top left). The parameter values are reported in table 2.1. Note that we are not particularly concerned with being realistic in the dynamics of activation and inactivation of modes, because RBMs will work on isolated single-time activation patterns.

To show the low dimensionality of the activity generated by this model, we can perform a Principal Component Analysis of the data, where features correspond to the activation of single neurons and samples correspond to the states at single timesteps (the dynamics, in the sense of temporal succession of states, is disregarded). The resulting PCA can be compared with that of a dataset with identical firing rate statistics, but where the correlations between neurons are destroyed. This is accomplished by shuffling the activity of each neuron in time, so that they all follow pure Poisson processes. In the latter case, the relative importance of principal components, as measured by the fraction of variance they explain, decays smoothly and slowly. In the activity generated by the model described above, on the contrary, there is a number of principal components that explain significantly more variance than in the baseline, and this number is linear in M (figure 2.6, top right).

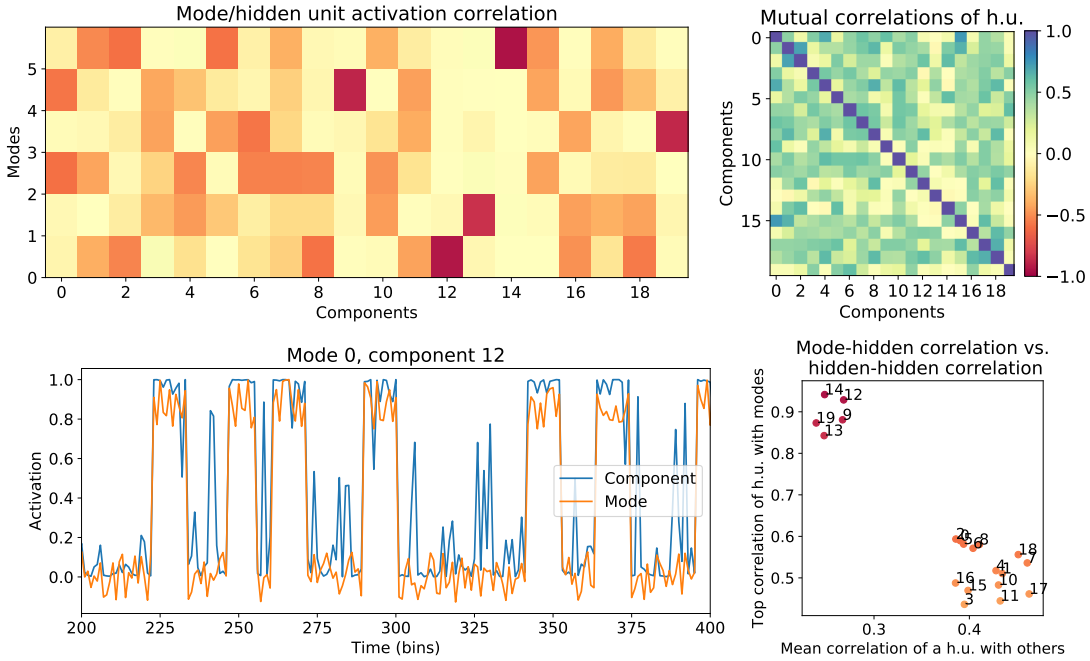


Figure 2.7: Analysis of the role of RBM hidden units in describing neural modes. Here, $N_h = 20$ and $M = 6$. Top left: correlation between hidden unit activations and the modes states used in generating the training data. Top right: mutual correlations between hidden units. Bottom left: snippet of timeseries for the activation of a mode and the correlated hidden unit. Bottom right: relation between the highest correlation of a unit with a mode (in modulus) and the correlation of that unit to the others. As explained in the main text, this enables the identification of mode-tracing units.

2.4.2 RBM hidden units can trace mode activation

I simulated the model illustrated in the previous section for $T = 10^5$ timesteps, and fitted restricted Boltzmann machines to it using contrastive divergence, as explained in the Methods section. Then, for every pattern of neurons firing, I performed a single forward pass through the RBM and obtained the probability of activation of the hidden units given those values for the visibles.

To study whether hidden units were capable of tracing modes activations, I computed the correlation of every hidden unit with every mode. The results in figure 2.7 (top left) show that every mode has at least one hidden unit that highly correlates to its activation. Moreover, the hidden unit that best correlates with it (which I will henceforth call the mode-tracing hidden unit) shows near-zero correlation with all the other modes. Note that because of the 0/1 symmetry of RBMs, this correlation may be negative: for this reason, I will always consider

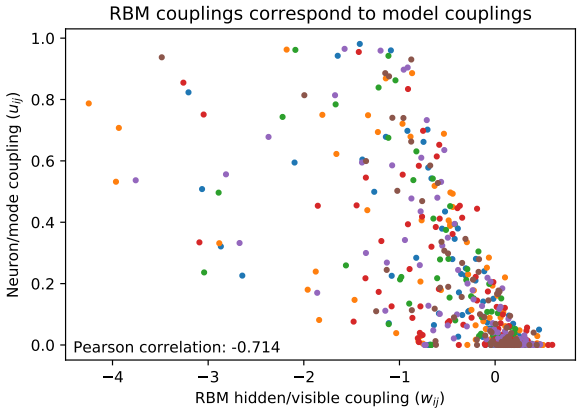


Figure 2.8: The couplings of a neuron to each mode (u_{ij}) are loosely related to the hidden-visible weights of the RBM (w_{ij}) for the corresponding mode-tracing hidden unit. Thus, the RBM could be used for a rough estimate of the contribution of neurons to a mode. Colours correspond to modes.

absolute values of correlations in the following paragraphs.

The results are clearly positive. However, in order to be able to trace modes in a real-world application, we would need a way to identify which hidden units are mode-tracing and which aren't — without prior knowledge of mode states. If the number of hidden units coincides with M, we expect a one-to-one correspondence. But what if M is unknown too? As it turns out, it is still possible: since modes are statistically independent from each other, the corresponding mode-tracing units must also be. Additionally, if the tracing is precise, we expect them not to be conditioned by other factors. Indeed, this idea is confirmed by the relation shown by figure 2.7 (bottom right). Here, one can see there is a dependence between the mean correlation of a hidden unit with the others and the highest among its correlations with a mode. The M units with the lowest hidden-to-hidden correlation precisely coincide with the mode-tracing units.

Finally, one can ask whether the RBM is able to predict the original coupling u_{mn} between neuron n and mode m. In other words, can we interpret the RBM weight matrix? Does it teach us about the contribution of a mode to the activity of a neuron (or vice-versa)? Figure 2.8 shows the values of RBM weights w_{nj} compared to the model couplings u_{mn}, where the hidden unit j is the one that was found to trace mode m. It can be clearly seen that there is a strong relation between the two, although variability still exists. It seems that knowing

the value of the RBM weight gives an upper bound over the neuron/mode coupling; in particular, a near-zero weight implies the neuron has a near-zero role in the mode. Further investigation may be needed to support this claim. Moreover, it is worth mentioning that the probabilities $P(v_i \text{ is active} \mid h_j \text{ is active})$ and $P(h_j \text{ is active} \mid v_i \text{ is active})$ are readily available for a fitted RBM, and can be used to estimate $P(\text{neuron } n \text{ is active} \mid \text{mode } m \text{ is active})$ and its dual.

Considering RBMs were not originally designed as a dimensionality reduction tool, it is interesting to see how efficiently they find the relevant factors in this dataset and in the retinal recordings. This is a consequence of the "information bottleneck" effect caused by having fewer hidden units than visible units. The RBM is trained to reproduce the statistics of patterns of N_v bits using only N_h bits: in order to do so, it implements a compression strategy which exploits the dependencies between neurons, with the effect of linking together those that share similar patterns of behaviour.

2.4.3 Comparison with NMF and ICA

Non-negative matrix factorisation (NMF) is an algorithm designed for factor analysis of matrices with non-negative entries, which are decomposed in the product of two other non-negative matrices, which share a dimension of arbitrary size C [Lee and Seung, 1999]. This algorithm is particularly suitable for the factor analysis of the model I presented above, as an alternative to RBMs.

Indeed, the results in figure 2.9 (top) show exactly M of the C component correlate very well with the activity of single modes, and aren't contaminated by the others. In this sense, it works at least as well as RBMs in detecting modes. However, I once again worked under the assumption that M is unknown, and therefore chose a large and arbitrary value for C. Unlike for the RBM, I did not find a way to identify mode-tracing NMF components without prior knowledge of how the data was built. Mode-component correlation does not seem to be related to component-component average or maximum correlation, nor to the frequency of activation of components. At this stage, there is no way to use NMF components to identify modes.

It should be stressed, however, that this is not guaranteed to apply in all conditions. A way to identify mode-tracing NMF factors may exist elsewhere, or perhaps RBMs can fail in other situations, such as a different statistics of u,

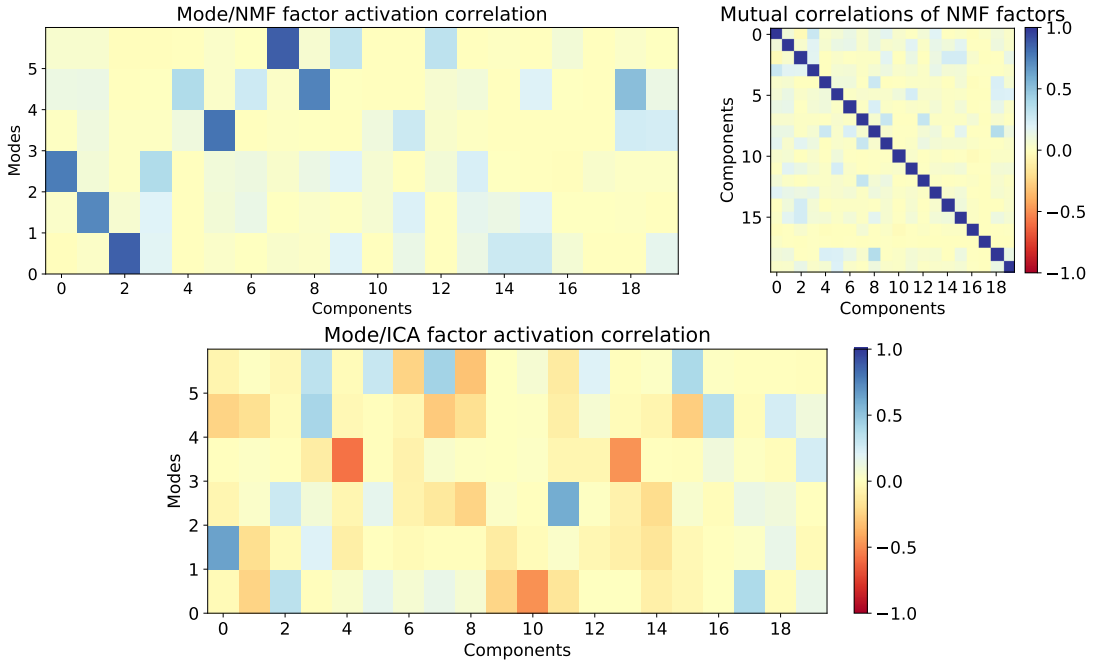


Figure 2.9: Top: the same analysis as in figure 2.7, performed by Non-negative matrix factorisation (NMF). Left: correlation of NMF factors with modes activation. Right: correlation matrix between NMF factors. Bottom: correlation between factors found by Independent Component Analysis (ICA) and modes. NMF and ICA were fitted using the scikit-learn free library [Pedregosa et al., 2011], with default settings.

different mode activation patterns, or increased randomness.

Conversely, Independent Component Analysis [Hyvärinen and Oja, 2000] does not adequately detect modes (figure 2.9, bottom). The correlations between ICA components and modes are weak, when present, and often involve more than one mode, making the discrimination impossible.

2.4.4 Interacting modes

If we regard modes not as the independent dimensions of the subspace spanned by neural activity, but as a general states of the network, such as the activation of particular ensembles of neurons, then there is no need to assume they do not interact with one another. The model described above can straightforwardly be modified to account for nonlinear interactions. Such a change, incidentally, would make it impossible for methods such as independent component analysis (ICA) to identify them: ICA relies on a hypothesis of component independence.

Imagine the simplest possible interaction between modes, in which some of

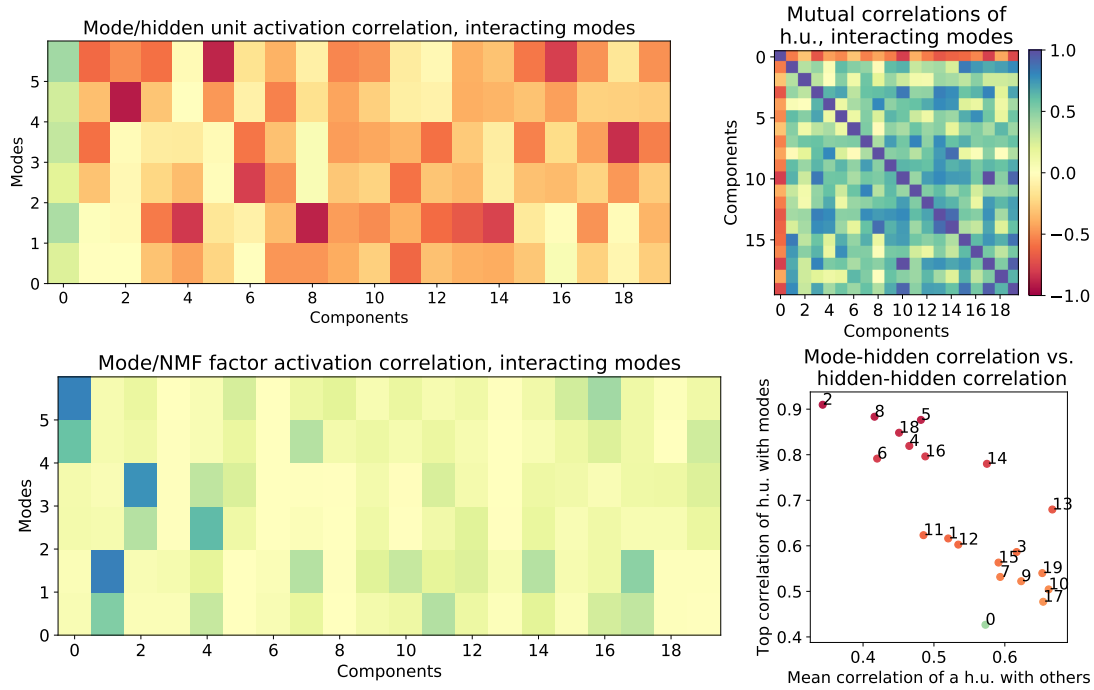


Figure 2.10: The same analysis as in figure 2.7, for interacting modes. Top left: correlations between modes and RBM hidden units. Top right: mutual correlations between hidden units' activations. Bottom left: correlation between modes and NMF factors. Bottom right: mode-tracing RBM hidden units can still be identified thanks to their independence from the others. I predict this effect may disappear when the interaction between modes is more complex.

them are active if, but not only if, another mode is active. "Primary" modes 1, 3, and 5 follow the same dynamics of activation and inactivation as in the previous case; modes 0, 2, and 4, however, are switched off whenever their corresponding primary mode is inactive, but the opposite is not true, so that the dependence is not complete. I repeated the experiments in the previous sections for this case, and reported the results in figure 2.10.

The RBM's hidden units still correlated with modes' activations, even if one of the modes is not picked up by any hidden unit in this particular instance of the experiment. It is still true, as shown before, that the mean correlation of a hidden unit with the others constitutes a hint of whether that unit is or isn't tracing a mode (figure 2.10, bottom right). However, intuitively, I predict this effect may disappear when the interaction between modes become very complex, since that case may determine a different correlation structure between hidden units.

NMF, on the other hand, seems to adopt a different strategy. Its components trace only the three most active modes (1, 3, and 5: the "primary" ones that can be arbitrarily active or inactive, whereas 0, 2, and 4 are "secondary" and can be active only at the times determined by the former three). Then, by construction, these components will also correlate with the secondary modes — however, they will lose information on when the former are inactive. I would argue, then, that this is another reason the RBM provides a better tool. Once again, it should be stressed that further investigation with more complex models, or, even better, on real data, is necessary before this is considered a general result.

2.5 Methods

For all ex-vivo recordings, the experiments were performed at the University of Newcastle, with the setup, spike detection, and sorting method explained in chapter 1. Details on stimulation protocols and RBM fitting are reported below.

For the results shown in section 2.2, the datasets consisted of recordings from 100 retinal neurons from the retina of an adult mouse (postnatal day 91), binarised into 10 ms time bins, for a total of 120000 bins (a 20-minute recoding). During the recording, the retina was stimulated projecting a random black and white checkerboard, with varying spatial correlations. The same dataset will be used in chapter 3: figure 5 of that chapter illustrates the spatial correlations of the stimulus. The results were shown for fits of RBMs with a variable number of hidden units $N_h = 1, 5, 10, 20$. These RBMs were trained by 1-step persistent contrastive divergence (CD-1). The whole dataset was presented 60 times (epochs) in minibatches of size 50, with a decreasing learning rate each epoch. Samples were extracted by Gibbs sampling: 20 randomly initialised chains, each yielding 10000 samples, taken 10 steps apart from each other to reduce correlations. The RBM code used is a customised version of deeplearning.net's Theano program [Theano Development Team, 2016].

The experiments described in section 2.3 were based on recordings of the retina of an adult mouse (postnatal day 63), binarised into 10 ms time bins. For the full-field stimulation protocol, the retina was stimulated with 2-second flashes of light followed by 2-second periods of darkness, for 30 repetition (a total of 12000 bins, equivalent to 2 minutes of recording). The checkerboard stimulus consisted of random black or white, 160 µm squares, with grid boundaries shifting by 40 µm

across presentations; the checkerboard changed at 30 Hz. The recording lasted 11 minutes. The datasets consisted of a choice of $N_v = 200$ neurons randomly selected across the recorded area, and the number of hidden units was chosen as $N_h = 16$. RBMs were trained with 1-step persistent contrastive divergence for 50 epochs (batch size 50, learning rate 0.01) using the code available in the scikit-learn machine learning library for Python [Pedregosa et al., 2011].

The RBMs used to analyse the binary neuron models in section 2.4.2 were trained with the same code; the training was repeated for 20 epochs, at learning rate 0.05 and batch size 50. The correctness of the fit was checked by taking a sample of size 18000, comparing means and pairwise correlations with the original data.

2.6 Discussion

Although Restricted Boltzmann machines are not a new tool, they have been used in neuroscience applications only in a few occasions. I demonstrated again that they are an interesting tool for fitting and studying neural data, arguing their potential in this field has not yet been entirely explored.

Like pairwise maximum entropy models (also called fully visible Boltzmann machines, FVBM), they are generative models and can be used to produce samples. The sampling method is the same, Gibbs sampling, which belongs to the family of Markov chain Monte Carlo algorithms. A relatively efficient fitting method, contrastive divergence estimation, is available for RBMs, and this is an advantage, now that large recordings (up to hundreds and even thousands of neurons simultaneously) are available: RBMs are less computationally expensive to fit compared to naive Boltzmann learning, and can achieve better fits compared to pairwise maximum entropy models, with a similar or smaller number of parameters ($N_v N_h$ as opposed to N^2). A systematic study of how the training time and the amount of data needed for a good fit scale with the number of neurons studied was not attempted by the previous literature about RBMs in computational neuroscience, and is also outside of the scope of this work. However, unlike FVBMs, it is clear that RBMs with hundreds of visible units can be easily trained on a personal computer. The results shown in section 2.3, where RBMs were trained on 200-neuron patterns, required between 10 s and 1 minute of training on a modern laptop, without an attempt at optimising for the number

of training steps. It should be noted that the computational complexity of the model also depends on the number of hidden units, and this can be tuned based on the required precision and the computational power available to the user. For $N_h \lesssim 20$, the partition function can even be explicitly computed without the need for approximations. Algorithms for RBM fitting are available in many machine learning packages (Tensorflow, Theano, scikit-learn, to name a few).

In this chapter, I first investigated the role of hidden units in RBMs fitted to retinal recordings — asking whether this statistical model is interpretable, i.e. can extract meaningful information about the dataset. I find they are able to decode simple stimuli, and that each hidden unit couples specifically to groups of neurons that share similar response patterns, notably ON cells and OFF cells (for full-field stimulations), local groups with neighbouring receptive fields (for checkerboard stimulations), or cells with sustained, as opposed to transient, responses.

Given this finding, which is a consequence of how RBMs can perform factor analysis, I adopted a simple model to generate binary data of a constructed neural population, where the neurons activate according to the activation patterns of binary latent variable. I confirmed that RBMs can typically recover the underlying structure of this activity: some hidden units closely trace the latent factors introduced by the model. There has been a growing interest, in recent years, in finding low-dimensional representations of neuronal activity, and interpreting the role of neural modes (for encoding and behaviour) is an area of ongoing research: in light of the results presented in this chapter, I propose RBMs could play a role in this analysis.

The model I chose, binary neurons linearly coupled to binary modes, is very simple, and RBMs should be tested in a more general framework. I suggest the next step should involve more realistic neuron models, which could better account for the intrinsically noisy properties of firing. The computational neuroscience literature does not lack examples of spiking networks exhibiting modes, attractors or sequences, which could be considered for this purpose.

Since I used only single-time RBMs, the models are unable to learn knowledge about the dynamics of modes or other temporal aspects of the data. A natural extension is to use t-RBMs, where more visible units are added in order to model each neurons at multiple time steps. This chapter has shown proof-of-principle usage of RBMs in factor analysis of neural recordings; the next step should be to compare this and other methods to real use cases where the objective is either

dimensional reduction or the identification of neural ensembles, such as in the recent [See et al., 2018].

There is a considerable literature on dimensionality reduction tools, and none of them is unreservedly better than the others in all situations. PCA is fast and interpretable, but is a linear transformation, which may not be able to capture more complex mixing. ICA is one of the most reliable and widely used algorithms, but cannot find Gaussian components, and does not seem to work well on the simple model I proposed (figure 2.9). NMF is not applicable to negative-valued data, and cannot distinguish dependent modes in said model (figure 2.10). Notably, RBMs seem to provide a direct way of determining which hidden units correspond to modes, based on their correlations with other modes. A possible problem is that they are designed to work on binary data: this renders them well suited for binarised spike trains, but not for continuous data such as calcium imaging recordings, unless somewhat unnatural transformations are applied. Moreover, RBM fitting can be seen as a "black box" process which makes the outcome difficult to interpret. In summary, RBMs constitute a further alternative that can be experimented with when different approaches are needed, with advantages and disadvantages to be evaluated on a case-by-case basis. Finally, unlike most other dimensionality reduction methods, they have the advantage of also being a generative model, that can be used to simulate data with similar statistical properties as the training set. This links them not only to the literature about neural modes, but also to the large body of work regarding Ising models and the thermodynamics of neural networks, as the next chapter, among other themes, will illustrate.

In the last few years, many machine learning methods have become increasingly popular, and the relationship between machine learning, the analysis of biological neural networks and their encoding properties is stronger than ever. It would be very limiting to investigate only RBMs as possible models — in the future, I expect neuroscience applications of many other models, in two different ways: as a tool for analysing data, and as a model of learning or encoding, to be compared with experimental evidence. However, while applying machine learning to neuroscience, we should never lose track of what our aims are. The simple fact that a model reproduces some statistical properties of neural activity does not mean that we have achieved some interesting result, and interpretability should be the guiding principle behind this investigation.

Chapter 3

Statistical models and criticality

This chapter consists of a paper, now in press as a book chapter: Sorbaro, M., Herrmann, J., and Hennig, M. (2019). Statistical models of neural activity, criticality, and Zipfs law. In Tomen, N., JM, H., and Ernst, U., editors, *The Functional Role of Critical Dynamics in Neural Systems*. Springer. [Sorbaro et al., 2019] *As required, the following introduction motivates the work and delineates my contributions. Part of this work had been presented in the form of a poster at the Computational and Systems Neuroscience (Cosyne) conference in 2017.*

As discussed in the previous chapter, restricted Boltzmann machines are a special case of spin models, with arbitrary binary connections — a generalised spin glass. The Ising model was the earliest of this class, and is still widely used as the simplest example of a magnetic phase transition. Therefore, understanding in what phase the model lies, once fitted to neural data, is a natural question. In the general field of statistical modelling, this question was asked before, and there were multiple reports that such models always seem to be poised in the vicinity of a critical point when fitted to neural data [Mora et al., 2015, Tkačik et al., 2015]. While this question was typically answered using pairwise maximum entropy models, we have reformulated it by using RBMs as presented in the previous chapter, and found similar results.

Importantly, however, we found a lack of convincing explanation for why what has been called "statistical criticality" should happen — is it a property of the model or of the data it is fitted to? If it is a property of neural data, does it hold because of a desirable property of critical points? How does it relate to the concept of critical dynamics, which has been extensively studied? We tried

to answer these questions, whenever possible, and otherwise review the current literature on the topic in order to frame it more clearly. The result is the book chapter *Statistical models of neural activity, criticality, and Zipf's law*, which is included in this chapter of the thesis. This paper is ready to be submitted for peer review.

This work also includes a brief account of experimental results: RBM specific heats are shown, explaining how they were computed; this is, to my knowledge, the first time RBM specific heats are studied in the context of statistical modelling of neural data. I also presented more examples of the Zipf laws we found in retinal firing patterns, discussing their dependence on various factors. The data for this section was obtained with the methods described in chapter 1.

My contributions A large part of the work consisted in reviewing literature. This was possible thanks to extended discussion about the questions involved, between me and the other authors. The general structure of the paper, the main writing work, and the final editing were done by myself, with contributions of text from the other authors, especially in sections 1, 3.1, 4.3, and 5. All figures in this chapter, including writing the relative code, the analysis of Zipf laws in neural data and the RBM fits, were made by me, using data provided by Dr. G. Hilgen and Prof. E. Sernagor of the University of Newcastle.

Statistical models of neural activity, criticality, and Zipf's law

Martino Sorbaro, J. Michael Herrmann and Matthias Hennig
University of Edinburgh, School of Informatics
10 Crichton St, Edinburgh, EH8 9AB, U. K.

Abstract We discuss the connections between the observations of critical dynamics in neuronal networks and the maximum entropy models that are often used as statistical models of neural activity, focusing in particular on the relation between *statistical* and *dynamical* criticality. We present examples of systems that are critical in one way, but not in the other, exemplifying thus the difference of the two concepts. We then discuss the emergence of Zipf laws in neural activity, verifying their presence in retinal activity under a number of different conditions. In the second part of the chapter we review connections between statistical criticality and the structure of the parameter space, as described by Fisher information. We note that the model-based signature of criticality, namely the divergence of specific heat, emerges independently of the dataset studied; we suggest this is compatible with previous theoretical findings.

1 Introduction

The debate about criticality in neural systems began with the observation of power laws in a number of experimentally measured variables related to neural activity. The first experimental observation of neuronal avalanches [6] found that their size distribution follows a power law with exponent of about $-3/2$, and their duration distribution follows one of exponent near -2 in cortical slices. These values are compatible with the exponents expected in critical branching processes — a well-studied topic in the field of complex systems physics [2]. Similar observations have been consistently reported in literature; moreover, the presence of power-law avalanche statistics was found to be theoretically justified by functional arguments on numerous occasions [5, 45, 44, 13, 51], and was shown to differ in different brain states [41, 16]. For an excellent high-level discussion of the topic, see [7].

An equally interesting instance of a power law is the finding that, in the population statistics of a neural network's activity, the rank of a state (the first being the

most frequently observed, and so on) and its frequency are inversely proportional. This phenomenon, known as Zipf's law, was first observed by Auerbach in 1913: "If one sorts individuals by a given property in a descending fashion and stops doing so at rank n_1, or at n_2, or generally at rank n_x, where the property has gone down to values p_1, p_2, p_x, then a certain law exists between n_x and p_x. In our case, this law is especially simple, it is expressed by the formula: $n_x \cdot p_x = \text{constant}$" [3]. This author already alluded to the possibility of more complex forms of the same law (e.g. for the distribution of wealth), but did not speculate why it assumes its simplest form in the studied data. In the mid-1930s, the American linguist George Kingsley Zipf discovered that the frequency of occurrence of words in Joyce's Ulysses and American newspapers follows the same law [54], which is today called Zipf's law: the frequency of each word decays as a power law of its frequency rank. After Auerbach's original example, city sizes [10, 23], Zipf's law was confirmed in a variety of fields, including citation counts in scientific literature [42], earthquake magnitudes, wealth, solar flare size, number of emails and phone calls received, and many others [35] .

Over the years several attempt have been made to understand Zipf's law. Zipf himself explains it by the *principle of least effort*: If words are stored in a linear array, then the low-frequency items are optimally located in a more distant place than more often used ones. The product of distance and frequency can be considered as a measure of the effort necessary to retrieve the word which he claims to be a constant. However, the assumption of an array, where the effort needed for retrieving an item is linear, which is necessary in order to obtain an inverse relationship rather than a general power law, seems unnatural when considering how items are stored in a neural network.

Another potential cause for the law can be seen in the idea of preferential attachment [4]. If, for example, the probability to move to or away from a city is assumed to be independent of its size, then Zipf's law for city sizes emerges. Other assumptions have been discussed and been shown to provide a better match for the distribution of city sizes [52], but again this may not easily carry over to states in a neural network.

Li [27] demonstrated that the words of an artificial language that simply consists of randomly chosen letters including a space sign tend to obey Zipf's law. However, the 'space' sign, which separates words in Li's approach, plays no such role in the analysis of neural data.

The authors of Ref. [1] aim at an explanation of Zipf's law by the existence of latent variables. Differently from the above attempts, this study is directly relevant for the analysis of neural activity. It also subsumes the scheme proposed by Li [27].

Although all of these attempts have their interest, there is some agreement that a deeper understanding is still lacking. In addition, there seems to be no clear justification on why criticality in the *statistical* sense and Zipf's law have been observed in neural data, or what brain function might benefit from it. It is interesting in this context that Zipf's law is a system property, i.e. it depends on the number of elements in the system and does not automatically apply to subset or unions of Zipfian sets. It can not be reduced to the mere presence of a particular probability distri-

bution (such as $P(x) \sim x^{-2}$), but requires a conditional sampling procedure to be reproduced in a simulation [8]. The observations in neural data as well as a number of unsolved problems with this subject make it a very interesting subject of further investigation.

In what follows, we will discuss the connections between the observations of critical dynamics and maximum entropy models that are often used as statistical models of neural activity, reviewing the recent literature on the matter, and debate the possible relationship between this *statistical* concept of criticality and the *dynamical* criticality related to avalanche statistics. First, we will illustrate the concept of Zipf's distribution, its origin, and its applicability to neural data. In section 2, we will introduce maximum entropy models, and show their connection to criticality and Zipf's law. In section 3, we will make three observations that emphasise the difference between statistical and dynamical criticality: (3.1) a system that shows dynamical, but not statistical criticality, (3.2) the process of fitting an energy-based model, and (3.3) the application of the theory of a large-scale corpus of biological data, where Zipf's law appears to hold, although the system is not dynamically critical. In section 4, we will show connections between statistical criticality and the structure of the parameter space, as described by Fisher information. Finally, in Section 5, we will return to the question debate whether there is a relationship between statistical and dynamical criticality and conclude with an outlook on the problem.

2 Statistical description of spike trains

2.1 Zipf's law in neural data

For the specific case of neural activity, Zipf's law refers to the rank-probability law for the occurrence of each possible *pattern* of activity, which has been observed to follow a power law in the same sense as for words in the English language [50]. To understand what we mean by *pattern* or *state*, we need to adopt a simplified way of representing spike trains that we can call *digital*: discretising time in bins of equal size δt, we can define a Boolean variable

$$\sigma_n(t) = \begin{cases} 1 & \text{if neuron } n \text{ spikes between } t \text{ and } t+\delta t \\ 0 & \text{otherwise.} \end{cases} \tag{1}$$

At any given time, then, the population activity is described by a *codeword*

$$\sigma(t) = (\sigma_1(t), \ldots, \sigma_N(t))$$

which describes, up to a precision of δt, the spiking state of the N neurons considered (Figure 1). Modelling the system statistically, in this framework, means giving a full account of the probability of each possible codeword to appear. Note that, typically, we are not concerned with the dynamics of the system, and we disregard

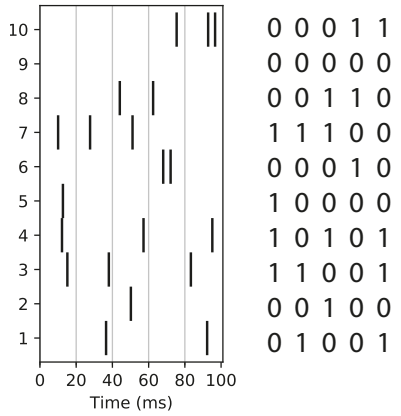


Fig. 1 Digitisation of spike trains from 10 neurons into a boolean matrix with bin size $\delta t = 20$ ms. Each row corresponds to the spike train of an individual neurons, with spikes represented as bars. Vertical gray lines indicate the boundaries of each bin. On the right, the resulting boolean matrix.

temporal correlations on scales larger than δt: this approach is suited to describe short-time correlations across space or properties of the encoding. Needless to say, the choice of δt can have important consequences on the results: in the limit of very large bin size, the pattern where all neurons fire simultaneously will be the only one to be observed; in the opposite limit of small δt, the *silent* pattern will be the most common, patterns with a single active neuron arbitrarily rare, and multi-neuron patterns absent. The results we discuss hold for bin sizes of the order of 5–20 ms, i.e. of the same order of magnitude as the typical correlation length between neurons; this value is commonly adopted in the literature [43].

To understand why Zipf laws are considered a signature of criticality, we will now illustrate the relationship, exposed by recent literature, between them and the critical points of models that have been used to describe neural activity, and are well known in physics.

2.2 Statistical modelling

The activity patterns of individual neurons and neural networks invariably display stochastic characteristics. A common approach, which we can call *top-down*, of modelling the nature of this activity is to make (simplifying) assumptions on the actual workings of neurons, synapses, and networks, in order to set up a computational model the results of which can then be compared with experimental observations.

A large part, perhaps the largest, of computational neuroscience is based on this paradigm, predominantly by simulations of spiking neural networks.

Here, on the contrary, we are concerned with what we call *bottom-up* modelling, which seeks to infer properties of neural activity in an entirely data-driven way. Understanding the correlation structure, the distribution of firing rates, or the repetition of identical patterns from experimental data are examples of this approach. In other words, the data is described in terms of probabilities and other statistical descriptors, instead of parameters directly implied by the biological or physical theory.

In the bottom-up approach, a very broad family of models is available. We will restrict ourselves to *energy-based statistical models*, a number of models developed in the last decade which adopt a log-linear relation between probability and state variables. In an *energy-based* model probabilities are expressed in terms of an energy function E, in analogy with statistical physics:

$$P(\sigma) = \frac{1}{Z} e^{-E(\sigma)},$$

where Z is the relevant normalisation factor. Many energy-based models used in neuroscience adopt the aforementioned *digital* description of spike trains in terms of binary variables: we will focus on these. In this case, for N neurons, σ can take 2^N different values, and determining the full population probability distribution requires specifying 2^N probabilities, which is an unrealistic task even for modest population sizes. Assumptions on the analytical form of the distribution are therefore required in order to infer a complete distribution from a relatively small number of samples.

2.3 Maximum entropy models

The first, and perhaps more elegant, strategy developed to this end is to adopt a *maximum entropy* approach [22], in which one first selects what features of the data should be exactly reproduced, and determines then the highest-entropy probability distribution consistent with those constraints. Schneidman et al. [43] and Shlens et al. [46] first applied this approach to neural data, using a Pairwise Maximum Entropy (PME) model, which exactly fits all $\langle \sigma_i \rangle$ and $\langle \sigma_i \sigma_j \rangle$, i.e. firing rates and pairwise correlations. Indeed, the question behind that research was primarily related to the importance of correlations in the vertebrate retina, including the study of higher-order interactions.

The PME probability distribution over all codewords has the following form:

$$P(\sigma) = \frac{1}{Z(h,J)} \exp\left(\sum_{i=1}^{N} h_i \sigma_i + \sum_{i \neq j} J_{ij} \sigma_i \sigma_j \right). \tag{2}$$

The expression above is mathematically identical, in statistical physics, to that of the canonical ensemble for the Ising model with arbitrary couplings, a generalisation of the model originally used to describe ferromagnetism in solids [21].

By definition, a successful fit of a PME model correctly reproduces all firing rates and pairwise correlations present in the data from the considered neural population. Fitting based solely on second-order statistics does not imply that third-order correlations and other statistical measures are correctly reproduced. Reports that higher-order correlations are largely irrelevant were thus very surprising [43, 46, 48], although these observation may be restricted to low activity and high pairwise correlations [53]. Assessing whether a maximum entropy model can capture additional statistics of the data provides a source of interpretability: If, say, a PME model can account for third-order correlations, then the latter are not constrained further by the data. If, conversely, third-order correlations diverge from the PME prediction, we learn that the neural activity uses higher-order statistics to encode information. Whether this is the case depends on the system and on the distance between the neurons considered [29, 39].

Several attempts have been made at improving the quality of the fit of statistical models, using different features as known statistics. The generalisation of equation (2)

$$P(\sigma) = \frac{1}{Z(h,J)} \exp\left(\sum_{i=1}^{N} h_i \sigma_i + \sum_{i,j} J^{(2)}_{ij} \sigma_i \sigma_j + \sum_{i,j,k} J^{(3)}_{ijk} \sigma_i \sigma_j \sigma_k + \dots \right)$$

can describe any probability distribution of binary variables exactly. However, finding the values of $J^{(n)}$ is data-hungry and computationally expensive, and the benefits typically do not outweigh the costs for $n \geq 3$.

As a different way to assess at least some aspects of the higher-order statistics, we can consider, for instance, the probability distribution of the number of neurons firing in a time bin, $p(K)$, where $K(t) = \sum_{i=1}^{N} \sigma_i(t)$, was used as a target. This can be introduced as a further constraint in a maximum entropy model in combination with firing rates and pairwise correlations, leading to the *K-pairwise* model [49, 33]. It typically produces significantly better fits than a pure PME, and is much less computationally expensive than attempting to fit higher order cumulants. Another related approach, the *population tracking model*, fits $p(K)$ together with the conditional probabilities $P(\sigma_i = 1|K)$ of each neuron firing, given the current population firing rate, providing a lightweight and interpretable model [11, 38].

An example of an energy-based model which does not rely on the maximum entropy principle, finally, is to use a restricted or semi-restricted Boltzmann machine (RBM/sRBM). Despite not directly aiming at fitting correlations, $p(K)$, and cumulants, as a maximum-entropy model would, RBMs were shown to perform at least comparably well in fitting all these aspects [25]. An advantage is that their complexity can be tuned, offering a choice of various degrees of accuracy and the corresponding computational costs. Additionally, *contrastive divergence*, the algorithm used for fitting, is an approximate but relatively fast and reliable algorithm, which lets one fit the simultaneous activity of a large number of units (up to several hundreds, whereas the exact learning algorithm for a PME model is not usable in practice over $N \approx 40$, although more efficient methods have been studied). Finally, RBMs can be interpretable models, specifically by studying the roles taken by hidden units.

Although a detailed discussion is beyond the scope of this chapter, we should at least mention the efforts to reproduce the time dynamics of the system, so that the statistical model fits both the distribution of single-time bin patterns and the conditional distribution of the pattern given the pattern in the previous time bin [30, 34, 12].

2.4 Phase transitions in models

Although this may initially seem less relevant from the point of view of research in neuroscience, we should remind ourselves that the Ising model is one of the earliest and most commonly studied paradigms of a phase transition. To understand its behaviour, let us make the temperature dependence of equation (2) explicit:

$$P_T(\sigma) = \frac{1}{Z(h/T, J/T)} \exp\left(\frac{1}{T}\left[\sum_{i=1}^{N} h_i \sigma_i + \sum_{i \neq j} J_{ij}\sigma_i\sigma_j\right]\right) \tag{3}$$

Note that, in the high temperature limit, this converges to a uniform probability:

$$P_{T\to\infty}(\sigma) = \frac{1}{2^N}, \quad \forall \sigma.$$

Conversely, when $T \to 0$, only a small number of states with non-zero probability survive, the others becoming infinitely rare. If a system that obeys $P_{T\to 0}(\sigma)$ is perturbed in any way, it will eventually converge to this stable set, under any reasonable dynamics. In other words, the distribution becomes, in the zero temperature limit, a finite set of stable attractors, the same as the stationary distribution of a Hopfield network [20], where the attractors play the role of memory patterns.

Clearly, neither of the two limiting cases can be a realistic description of neural statistics, and the truth lays in between them, in a regime where the model is much more informative. The physics literature shows that there is sharp phase transition between a *disordered* phase and a *spin glass* phase, with the exact location of the critical point depending on the statistics of h and J [36]. It is then a natural question to ask whether the Ising model that results from a fit to neural activity is in one of the two phases, or poised near the critical point, and whether this relates to other concepts of criticality in neural systems.

The divergence of specific heat, also called heat capacity, in a macroscopic system is a classic signature of discontinuity in the properties of the system upon variation of a single parameter, typically temperature (generalisations of this idea will be discussed in the next section). The most classic example is the case of a change in the state of matter — solid to liquid, liquid to gas, etc. — where an infinitesimal change in temperature through the critical point requires a finite amount of energy (the *latent heat*). This is equally true for spin systems of the form we examined above. Tkačik et al. [50] fitted a model of the form (2) to binned spike trains from recordings of the salamander retina subjected to movies of naturalistic stimuli. They

varied the temperature of the model around $T = 1$ (this value corresponding to the fit to neural data), and studied the specific heat as a function of T for an increasing number of neurons.

Their result clearly showed a peak in the specific heat of their models, with the peak temperature approaching T=1 as N is increased. This is evidence that T=1 coincides with the critical point, and therefore, the model is poised at criticality for parameter values exactly corresponding to those that fit the neural data. Similar observations were independently repeated, e.g. in [33, 37, 16], generating a debate on the nature of this observation and its biological interpretation, as will be discussed in later sections.

2.5 Model criticality and Zipf's law

Zipf laws can be related to statistical criticality in the sense of models, as shown in [50] (supplementary information), as follows. Call $p_1, ..., p_k, ..., p_{2^N}$ the probability of occurrence for each of the 2^N possible codewords. In statistical physics, microcanonical entropy can be defined as $S = \log \Omega$, where $\Omega = \Omega(E)$ is the number of states with energy lower than E. On the other hand, the energy level associated with a pattern is a function of its probability:

$$E_k = -\log p_k + \text{const.}$$

Now, Zipf's law states that, for every pattern, its rank $r_k \propto 1/p_k$. In the notation used above, note that $r_k = \Omega(E_k)$. Therefore, Zipf's law implies

$$\begin{aligned} \log p_k &= -\log r_k + \text{const.} \\ E_k &= S_k + \text{const.} \end{aligned} \tag{4}$$

If the above linear relation holds, then d^2S/dE^2=0. Since both specific heat and the variance of energy are inversely proportional to d^2S/dE^2, these thermodynamic quantities diverge. This is the classic signature of a second order phase transition.

The rank-probability relation defined by Zipf's law, therefore, is a model-independent way of showing criticality in this statistical sense. Its appearance guarantees the divergence of the specific heat of a PME model fit to the same data, but does not require complex and computationally expensive fitting procedures, and relies only on the statistical properties of the data.

3 Statistical and dynamical criticality

As we have mentioned, most energy-based models do not account for dynamics, as they are concerned only with fitting a single-time bin distribution. The formulation of the Ising model in physics describes a stationary distribution and does not in-

clude any dynamics. Transition probabilities from a state to another can be added through additional assumptions about the dynamics of the system. For instance, Glauber dynamics [14] generates a Markov chain whose stationary distribution coincides with the distribution (2). This is useful, for example, when sampling states from that probability distribution. Avalanches can be observed in high-dimensional Ising systems when they are driven out of stationarity by a change in temperature or applied magnetic field. Therefore, it is not expected that maximum entropy models reproduce any aspect related to avalanche dynamics.

It is not clear a priori, then, whether the observation of Zipf laws and diverging specific heats should be related to power-laws in the dynamics. In fact, finding a connection between the two concepts seems challenging. In the next sections, we will provide examples of how the two might be entirely distinct, which prompts questions on the nature, meaning, and relevance of statistical criticality.

3.1 The Eurich model is dynamically, but not statistically critical

For a discussion of the relationship between the two concepts of criticality, it is interesting to consider the Eurich model for neural avalanches as a "testbed" [9]. In this model, N identical units, analogous to non-leaky integrate-and-fire neurons, are stimulated both by a random input and by the interaction with other units, mediated by a coupling factor α. Once the integrated value of the input exceeds a threshold, the neuron resets, keeping a fraction of its energy. The parameter α can drive the system to two different regimes, with different avalanche statistics in space and time, separated by a phase transition.

This model is mathematically well-understood, and can be conveniently tuned, because the parameter values for (quasi-)critical as well as sub- or super-critical behaviour are known analytically (even for finite systems). The very definition of the model is such that all neurons have identical properties, and the same for pairs, triplets, etc., of neurons. As a consequence, all N patterns with exactly one active neuron appear with the same frequency; all $N(N-1)/2$ patterns with exactly two active neurons appear with the same frequency, and so on, giving the rank-probability plot a step-like appearance, which cannot follow a Zipf law (Figure 2).

It is tempting to consider the tail of the rank distribution. Although the number of states increases with the activity (for neurally plausible activity levels), their probability decreases strongly if the firing rate (per time bin) is low. Therefore, steps will disappear for higher ranks, which may or may not produce a power-law-like behaviour. However, in the statistical approach, typically small values of $N\delta t$ (see equation 1) are used, such that the potentially Zipf-like tail (figure 2, right) will be statistically irrelevant. It has also been claimed that the statistical approach is most likely restricted to low-activity patterns [53].

Note that the rank curve would be less step-like if some form of heterogeneity is introduced, as opposed to the complete symmetry between neurons that characterises the Eurich model. However, this would amount to an additional assumption

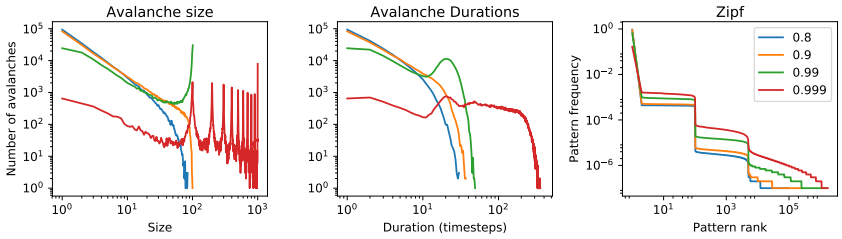


Fig. 2 Avalanche statistics compared to Zipf law for the Eurich model in different regimes [9]. Here $N = 100$, and the critical point is at $\alpha_c = 0.90$: in this case, the avalanche size and duration distributions most closely approach a power law with exponent $-3/2$, except for a cutoff due to the finite size. The subcritical case (blue), on the other hand, shows short-tailed distributions, and the supercritical cases (green and red) exhibit respectively one or many peaks at large values. In all cases, the Zipf plot does not show power-law dependence. Note that the smoothing of the step-like function is due to the finite sample size.

that is, as the Eurich model shows, not necessary for criticality. In particular, we would need to assume that the patterns with a single active neuron already follow Zipf's law. This is a particular, but not unreasonable assumption, as this can be expected, for example, in a scale-free neural network. In fact, we cannot rule out that a Zipf profile, or at least an approximation, could be found just by tuning the distribution of firing rates, even in the absence of correlations. Such a finding would entirely rule out any relation to dynamical criticality, which appears exclusively as a consequence of emergent phenomena deriving from complex interactions. However, it would still require a specific distribution of firing rates among the neurons, an assumption that in itself would prompt questions about its functional reasons. Firing rate distributions have been studied extensively [32], and found to be highly skewed, with a small fraction of neurons responsible for the majority of emitted spikes. A clear theory on why this is an advantage for the encoding is still missing. In section 3.3, we will consider a case in which the Zipf relation holds even when correlations are destroyed, which suggests the long-tailed firing rate distribution is sufficient for it to hold.

3.2 Fitting energy-based models to critical activity

A natural way of checking if dynamical and statistical criticality are related could involve fitting a statistical model to neural models that exhibit various kinds of dynamics, and can be tuned to a supercritical (noisy), subcritical or critical regime. This was one of the goals of in Ref. [16]. The authors identified five different dynamical states of the cat and monkey cortex, studied their avalanche statistics, and evaluated the temperatures corresponding to peak specific heats of Ising models fitted to each dataset. The results did show a small but significant relationship be-

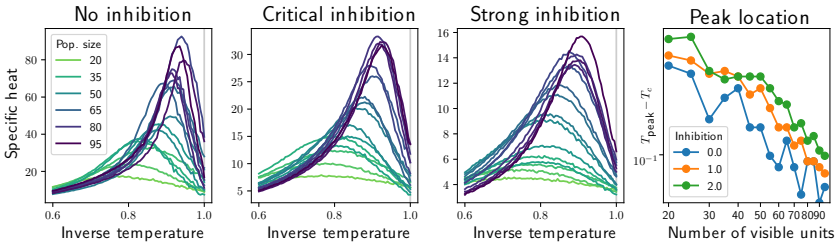


Fig. 3 RBM specific heats peak when fitted to a variety of datasets, with the peak approaching the temperature of the fit as the model size increases. RBMs were fitted to simulated data with different avalanche statistics: supercritical (left), critical (centre) and subcritical (right).

tween avalanche dynamics and specific heat peak location. However, it should be noted that some of the results in this work were obtained with small datasets of six neurons only, which may not offer insight on what happens in the thermodynamic limit.

We attempted a similar task fitting Bernoulli RBMs to the activity generated by a tunable model of a neural network, similar to the binary, non-leaky, integrate and fire model used by [13]. In this model, the strength of inhibition can be tuned, leading to a network with low, random-looking activity and a short-tailed avalanche distribution (high inhibition); or a network generating activity in large bursts (low or no inhibition). The critical regime lies in between the two. Details about the implementation of the model are given below.

We found that although the absolute value of the peak does depend on the correlations of the data, its location is always at a temperature near $T = 1$ (which is the value corresponding to the original fit), and further approaches this temperature as the number of units increases, i.e. in the thermodynamic limit. These results are compatible with what was shown by [37], using a different dataset, for the K-pairwise model.

It seems, then, not only that the statistical model that we fitted does not accurately detect criticality in the dynamical sense, but it also exhibits statistical criticality no matter the dataset it was fitted to. This implies, on the one hand, that the dynamical criticality of a dataset and the statistical criticality of a model fitted to it are unrelated, and, on the other hand, that a model fitted to datasets of very different nature all tend to exhibit statistical criticality. This is compatible with an argument that was put forward by theoreticians [31], as will be discussed in section 4.2.

Methods: network model

The binary neuron model used for the simulations is similar to the one presented by [13]. In this model, each neuron has a probability of firing given by a weighted sum of its inputs, divided by a factor dependent on its own firing history. A fifth

of all neurons are inhibitory, while the rest are excitatory; the value of inhibition was tuned to 0.0, 1.0, or 2.0, to enforce different regimes, corresponding to different correlations and avalanche statistics. While in the original work the connectivity was all-to-all, with weights drawn from a uniform distribution, we modified it to identical couplings, but set on a network with scale-free degree distribution. This enforced larger variability of firing rates between neurons; both experimental evidence and the theory in [26] suggest this choice does not affect the location of the critical point. We simulated 1000 neurons, from which we took subsets of the sizes required for analysis, for 1 million time steps (conventionally taken to equal 1 ms). The resulting activity was re-binned in 5 ms bins, to reduce sparseness.

Methods: RBM specific heats

As we will argue in section 4.2, the direction that best indicates the critical point coincides with the fist eigenvector (the one corresponding to the largest eigenvalue) of the Fisher information tensor. However, in practice, this is never orthogonal to the direction of increasing/decreasing temperature: thus, varying temperature is an acceptable way to look for a phase transition.

In statistical physics, the general expression for the probability of a pattern in an energy-based model at temperature T is

$$P_T(x) = \frac{1}{Z(T)} e^{-E(x)/T}.$$

The expression for the energy in the case of RBMs is

$$E(\mathbf{v},\mathbf{h}) = -\mathbf{a}^\intercal \mathbf{v} - \mathbf{b}^\intercal \mathbf{h} - \mathbf{v}^\intercal J \mathbf{h}.$$

Where $\mathbf{v}$ and $\mathbf{h}$ are vectors of visible and hidden binary variables respectively. Since this expression is linear in the parameters a_i, b_j and J_{ij} for all i, j, changing the temperature of a model coincides with rescaling these parameters by a linear factor $\beta = 1/T$. In the following, we have adopted the standard strategy of fitting an RBM to neural data, obtaining values for its parameters, and then rescaling them — this means $T = 1$ (no rescaling) coincides with the parameters as they were fitted, the values corresponding to a model that correctly reproduces the given data. Fits were obtained by 1-step persistent contrastive divergence.

We can then compute the specific heats at different temperatures. The marginal probability of $\mathbf{v}$ is

$$P_T(\mathbf{v}) = \frac{1}{Z} \sum_{h_{1\ldots N}=0,1} e^{\frac{1}{T}(\mathbf{a}^\intercal \mathbf{v} + \mathbf{b}^\intercal \mathbf{h} + \mathbf{v}^\intercal J \mathbf{h})} = \frac{e^{\mathbf{a}^\intercal \mathbf{v}/T}}{Z} \prod_{j=1}^{N_h} \left(1 + e^{b_j/T + (\mathbf{v}^\intercal J)_j/T}\right)$$

Disregarding an additive constant, the energy of a visible pattern can be expressed as the logarithm:

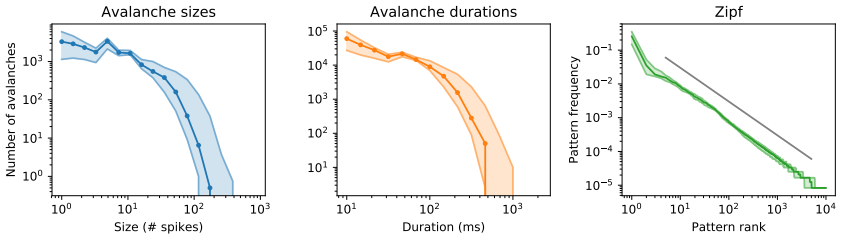


Fig. 4 Avalanche statistics compared to Zipf law in the neural activity of a healthy, adult (postnatal day 91) mouse retina stimulated by projection of a white noise checkerboard pattern. The detection of avalanches and the pattern count were repeated over 30 sets of 100 neighbouring neurons, each of which was recorded for 20 minutes. The sets may overlap. The solid lines are medians over these sets; the shaded area is delimited by the first and third quartiles. The grey line in the rightmost plot is for comparison with Zipf's law. The data were made available by G. Hilgen and E. Sernagor, University of Newcastle. We refer to [19] for experimental and data analysis methods.

$$E_T(\mathbf{v}) = \frac{\mathbf{a}^\intercal \mathbf{v}}{T} + \sum_{j=1}^{N_h} \log\left(1 + e^{b_j/T + (\mathbf{v}^\intercal J)_j/T}\right)$$

In accordance with statistical physics, we can define

$$c(T) = \frac{\mathrm{Var}(E_T)}{N_v T^2}.$$

This quantity can be computed from a sample. For each temperature value, in each dataset, we extracted 20 chains of 2000 samples, taken every 10 steps in order to reduce spurious correlations, and computed $c(T)$ by the expression above.

3.3 Retinal activity obeys Zipf's law, but is not dynamically critical

The mammalian retina, a system that is often chosen when studying the statistics of neural activity, and whose encoding and dynamical properties are well known, is an example of the opposite case: It was the first system in which statistical criticality was observed, but it does not exhibit dynamical criticality.

Avalanches arise in the mammalian retina only during the period of development: for mice, in the first few days after birth, before eye opening, when the retina does not respond to light and the network activates spontaneously. During this stage, the activity of the retina consists of the so-called *retinal waves*, which are effectively power-law distributed avalanches. Direct comparison with a computational model showed that these are indeed the signature of a critical state between locally and globally connected activity [17]. However, these disappear in a functional retina: Figure 4 shows the statistics of a 20-minute recording of an untreated, adult mouse

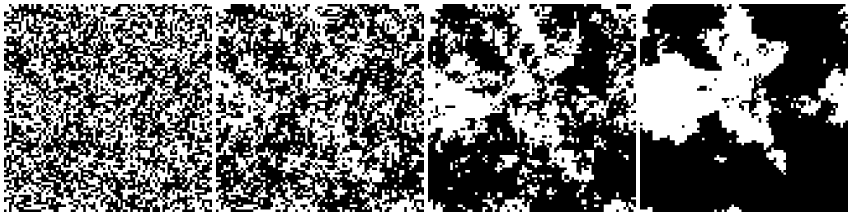


Fig. 5 Examples of stimulation frames. Correlations increase from left to right (dataset ID 0 to 3): the frequency spectra follow f^{-a} with $a = 0.5, 1.0, 1.5, 2.0$, i.e. from noise to the statistics of natural images. The correlation statistics extend to time.

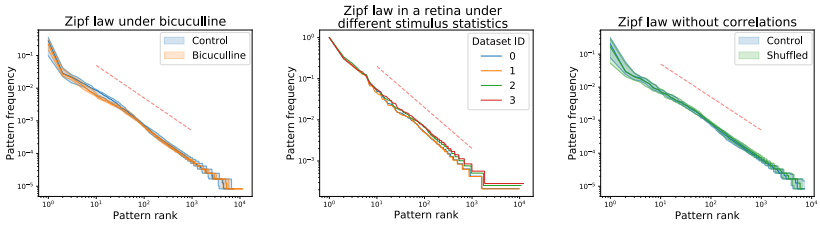


Fig. 6 Left: Zipf plots, before and after treatment with bicuculline. 30 groups of 100 neurons, selected as explained in the Methods paragraph. Centre: Zipf plots for a unique group of 100 neurons under stimuli of different statistics; the difference between datasets 0-3 consist in the different spatial frequency — from near-white noise to natural stimulus statistics. Right: the same data as in the left panel (control), and its shuffled version, where correlations have been destroyed, while keeping the same firing rates. The red dashed lines correspond to $1/x$ laws.

retina under an uncorrelated black-and-white checkerboard stimulation. It is evident that the avalanche statistics is short-tailed, and, at the same time, the probability-rank plot of pattern frequencies is well compatible with a Zipf law. Note that correlations between the activities of retinal ganglion cells change significantly with the statistics of the stimulus, and the avalanche statistics will consequently appear different. The example of adult retinas is complementary to Section 3.1 in the sense that, here, a system that does not show dynamical criticality can well obey Zipf's law.

It is worth mentioning that the observation of Zipf law in retinas is very robust to a number of external factors. We found no significant differences in the rank-frequency plots of patterns observed when the retina was treated with bicuculline (a GABA blocker) compared to a control; analogously for retinas under stimuli characterised by very different level of spatial correlations.

If Zipf's laws have a functional role, there is no expectation this phenomenon would survive in a non functioning neural system, such as a retina that has been pharmacologically treated in a way that breaks its normal operative mode. Here, we took data from the same mouse retina, before and after treatment with a 20 μM solution of bicuculline, which is a $\mathrm{GABA_A}$ antagonist. The results are shown

in figure 6 (left): as it is evident, there is no clear difference between the two rank-probability plots. Of course, the only strong argument against the functional role of Zipf laws would be finding a functional retina in which this law is broken, which is not the case here. However, we can notice that even an intervention that significantly disrupts the retina's activity, by blocking inhibitory interactions, doesn't prevent this phenomenon to arise. This is despite the large change in the correlation between neurons induced by bicuculline.

Likewise, one may expect a dependence of pattern frequency-rank statistics on stimulus statistics. The retina, after all, is a neural system design to encode a stimulus — and the correlation structure of its neurons' activity strongly depends on the correlations in the stimulus. However, we found no significant difference in Zipf laws under different stimulations. Figure 6 (centre) shows a single group of 100 neurons selected in a retina that was stimulated with light patterns of different kinds. All stimulus presentations consisted of black-and-white random checkerboards, which are binarised versions of random noise of given frequency spectrum f^{-a} with $a = 0.5, 1.0, 1.5, 2.0$ in space and time: from near-white noise to the statistics of natural images (figure 5).

The independence from correlations is evident in the right panel of figure 6: here, the "control" curve is the same as in the left panel, and is compared with the rank-probability plot for a "shuffled" version of the same data, where the firing rates were kept the same, but spikes were moved in time in order to cancel neuron-neuron correlations. The difference between the two curves is clearly not significant. This demonstrates how a firing rate distribution which is long-tailed (approximately log-normal) can in itself produce a Zipf-like plot. More research is needed to show whether this holds in general.

Methods

The handling of the retinas, experimental apparatus, and the first part of the data analysis pipeline were performed as illustrated in [19]. Starting from detected and sorted spikes, we removed those with very low amplitudes, by selecting a threshold corresponding roughly to the lowest 10%. This was to ensure only good-quality events were left. Then, we selected, for each Zipf plot, $N = 100$ clusters all pertaining to the same area of the retina (figure 7).

At this stage, spikes were binarised into a $N \times T$ matrix S of boolean variables, with $S(n,t) = 1$ if neuron n spiked between times t and $t + \delta t$ and $S(n,t) = 0$ otherwise. When multiple spikes from the same neuron occurred in a single time bin, the extra spikes were disregarded. For recordings shown in this chapter, $T = 120000$ or more, and $\delta t = 10$ ms, implying at least 20 minutes of neural activity were recorded.

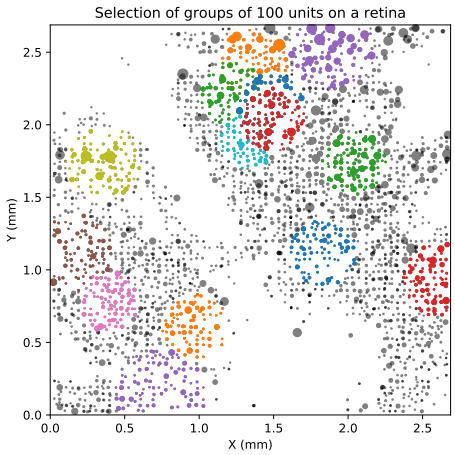


Fig. 7 All spike clusters in a dataset (P91 mouse retina under white noise checkerboard stimulation), arranged spatially. For Zipf analysis, a random cluster was selected, and the 100 nearest ones picked along it (coloured patches on the figure are examples) to form a 100-neuron group. The process was repeated 30 times to study error intervals. The size of the dot scales with the number of spikes in the cluster. Even if this image only represents detected spikes, the optic disc is noticeable at the bottom end; other inactive areas corresponds to cuts in the retina, unavoidable when placing it on a flat surface.

4 Parametric sensitivity

The basic fitting procedure of a maximum entropy model minimises the quadratic difference between the data moments and the moments predicted by the model. During fitting, any model is updated by exploring the parameter space, following a direction given by the loss function. When a model admits a phase transition, the parameter space is characterised by (at least) two regions, corresponding to the phases, separated by a critical surface. From a theoretical point of view, asking why a model is poised at criticality coincides with asking why the fitting process tends to lead towards the critical surface in the parameter space. This has been discussed by [31]; before introducing their argument, we provide some theoretical background.

4.1 Model distance

Intuitively, a phase transition occurs at a location (the critical point or critical surface) where an arbitrarily small change in the parameters yields a sharp, qualitative change in the behaviour of the model. In this section, we will formalise this idea,

and link the notion of model distance to the statistical physics framework that we have introduced above.

A common measure of the distance (in model space) of a probability distribution p from a given one q, both defined on a set S, is the Kullback-Leibler divergence

$$D_{KL}(p;q) = \int_S p(x) \log \frac{p(x)}{q(x)} dx.$$

It measures the amount of information that is lost when approximating p by q. The name *divergence* stresses that this quantity does not have the mathematical properties of a distance, namely not being symmetric and not obeying the so-called triangle inequality. If the model space is parametrised by θ, and p and q are close to each other in this space, so that $q = P_\theta$ and $p = P_{\theta+\delta\theta}$, at second order in $\delta\theta$, the divergence can be approximated as

$$D_{KL}(P_{\theta+\delta\theta}; P_\theta) \approx \frac{1}{2} \delta\theta^T F(\theta) \delta\theta,$$

where F is called Fisher information tensor (FIT) which is given by

$$F_{ij}(\theta) = -\int P_\theta(x) \frac{\partial \log P_\theta(x)}{\partial \theta_i} \frac{\partial \log P_\theta(x)}{\partial \theta_j} dx.$$

Fisher information is here expressed as a statistical quantity, but it has an important relation to the physics of statistical models. Consider a Hamiltonian model, where the probability distribution is given by

$$P_\theta(x) = \frac{e^{-H_\theta(x)}}{Z_\theta}, \qquad H_\theta(x) = \sum_{k=1}^{n} \theta_k f_k(x), \tag{5}$$

which is an obvious generalisation of maximum entropy models. Calculating the Fisher information for this form of P (5), we retrieve the direct connection between the covariance (with respect to P_θ) of the physical quantities f and the FIT that is characteristic for probability distributions of the exponential type:

$$F_{ij}(\theta) = \mathrm{Cov}[f_i(x), f_j(x)].$$

This means that the FIT characterises the variances and correlations of the functions f which are now considered as stochastic variables and depend on the state x of the system.

Additionally, note that changing the temperature in the traditional canonical ensemble corresponds to scaling the Hamiltonian by a factor $\beta = 1/T$, similarly to equation (3). In the formulation above (5), this is equivalent to scaling all the θ_i by β. Given a point θ on the parameter manifold, the direction $\partial/\partial\beta$ can be expressed as

$$\frac{\partial}{\partial\beta} = \frac{1}{n} \sum_{k=1}^{n} \frac{\partial}{\partial\theta_k}$$

which is just a linear combination. The specific heat is given by

$$c(\beta) = \frac{\beta^2}{N} \mathrm{Var}[E].$$

Thus, we can arrange for the specific heat to be one of the entries of the Fisher information matrix, with a change of basis, which includes the β direction as a base vector together with other $n-1$ linearly independent ones. Analogous considerations can be made for the magnetic field and magnetic susceptibility. In this sense, the Fisher information tensor is a generalisation of specific heats and susceptibilities.

4.2 Fisher information and criticality

We can now look at the relationship between statistical criticality and the model's parameter space. Suppose any *generalised susceptibility* (i.e. a component of the Fisher tensor) diverges at a point θ_0. Then an eigenvalue of the Fisher information, say λ_k, diverges at θ_0. Call v_k the corresponding normalised eigenvector. For small α,

$$D_{KL}(P_{\theta_0+\alpha v_k}; P_{\theta_0}) \approx \frac{\alpha^2}{2} v_k^T F(\theta_0) v_k = \frac{\alpha^2}{2} \lambda_k,$$

and the r.h.s. diverges. This means that, moving from θ_0 in the v_k direction by an arbitrarily small step yields a model $P_{\theta_0+\alpha v_k}$ that is completely different from P_{θ_0}, as indicated by an infinite KL divergence.

We introduced this description in terms of Fisher information in order to give an interpretation of criticality from the point of view of modelling. A model is at a critical point whenever there is a direction in parameter space that leads to an infinitely different model by a finite change in parameters. This, incidentally, shows that the best way of measuring the distance from a critical point is not to vary temperature, but to use the first eigenvalue of the FIT and move in the direction of the corresponding eigenvector. Temperature is not always the most relevant control parameter.

Mastromatteo and Marsili [31] have argued that, because of this special property critical points have in the parameter space, they are particularly favoured by model fitting. In particular, they show that *distinguishable* models accumulate near critical points, whereas models farther away from these are largely indistinguishable. Their argument, in brief, goes a follows. Two models are considered indistinguishable if their Kullback-Leibler divergence is less than a given value ε. For small ε, D_{KL} is approximated by Fisher information, and the volume of parameter space occupied by models indistinguishable from θ_0 turns out to be proportional to $(\det F(\theta_0))^{-\frac{1}{2}}$. This quantity diverges at critical points due to the first eigenvalue diverging as explained above. Thus, *most models* actually are poised near a critical point, according to this metric. They conclude that criticality may be a feature induced by the inference process, rather than one intrinsic to the real system being studied by the model.

This may be the reason why statistical models seem to be poised at a critical point, for a variety of training datasets, as we showed in section 3.2. However, it does not affect Zipf laws, which are directly observed in the data.

4.3 Criticality and parameter 'sloppiness'

It is well known that the parameter spaces of many models often show only a small number of directions (linear combinations of parameter changes) along which the overall properties of the model strongly change ("stiff"), and a large number of directions which have little influence on the model ("sloppy"). This phenomenon, termed "model sloppiness", has been observed in a wide number of cases in systems science [15, 28].

For the specific case of neuronal networks, in Ref. [40], although for small numbers of neurons, "stiff" dimensions corresponding to large FIT eigenvalues were identified. The remaining "sloppy" dimensions, on the other hand, can change without much effect on the goodness of fit of the model. A further development of this approach has been reported in Ref. [18], where it was shown that about half of the dimensions in the data manifold are irrelevant for the modelling. As shown in paragraph 4.2, near a critical point, the direction pointing towards the critical surface has a diverging FIT eigenvalue, while the others are smaller. This hints there may be a connection between sloppiness and criticality, which, at the moment, we can only leave at the level of speculation.

Additionally, however, sloppiness indicates that a fitting algorithm for the data may be improved if different dimensions are differently weighted during the optimisation process. We can then ask whether using a natural gradient in the fitting procedure would lead to a different result while evaluating model criticality. In natural gradient optimisation, the components of the gradient are compensated by the inverse Fisher information, i.e. the divergence near a critical point of the model would disappear, at least theoretically when the Fisher information is exactly known. As a result, the fitting procedure is not homogeneous with respect to the set of the parameters, but with respect to the space of the parameters, taking into account its geometrical properties, and parameters can be identified equally well in all regions. In this way, the problem discussed in Ref. [31] may disappear — more research will be needed in order to verify this.

5 Discussion

Neuronal avalanches are an experimentally well-studied phenomenon, that can be explained as a consequence of the optimisation of information processing in the brain. It should be noted that an understanding of how the potential functional ben-

efits of this "dynamical" criticality are realised is missing [44] — however, it has been shown that the maximisation of the dynamical range happens at criticality [24].

Statistical criticality is an equally complex phenomenon to explain theoretically. Like dynamical criticality, it can be taken to indicate the complexity of the neural data and the relevance of higher-order correlations or latent variables, but its functional implications are less clear. In this chapter, we have reviewed the concept, both in the context of fitted statistical models, and as a direct observation of Zipf laws in neural population data. Through experiments on restricted Boltzmann machines, we suggested that the divergence of model specific heat is not a reliable way to infer properties of the data. We mentioned how Fisher information provides the correct description of the parameter space and the critical surfaces, and reviewed a possible explanation of why statistical models tend to poise themselves at a critical point. Then, we tried to describe the connection between statistical and dynamical criticality, and argued there is no clear connection, by showing examples where one of the two was present without the other. Further insight on this matter might come from models that are capable of both, provided they can reproduce not only the equilibrium distribution of the data, but also the dynamics. A multi-time maximum entropy model might provide a starting point for this work.

Of course, it may well be that the observation of Zipf laws is simply a consequence of problems related to how we describe the data — these include the typically small sets of observables, the choice of binning size, failure to account for the real dynamics, and biases introduced by sampling. However, the ubiquity of Zipf laws in complex systems means its emergence in biological neural networks should not surprise us, and it could be explained in terms of mechanisms such as the one described by [1], or perhaps with preferential attachment. Conversely, an important open problem is an explanation on whether statistical criticality is something that is actively sought by the system because of some functional relevance. On this matter, we tried to analyse the Zipf profile of retinal activity under various conditions (various stimulus statistics, pharmacological treatment), but we found no significant differences in the cases examined. Interestingly, it seems to be possible to generate a Zipf profile simply by enforcing a long-tailed firing rate distribution, despite the absence of correlations. Even if this observation were confirmed, the question would simply shift towards finding a reason for such a skewed distribution of firing rates, which has not yet found a justification in terms of function.

Notably, recent research has started showing how Zipf laws appear in different kinds of parametric models, including "deep" ones, as soon as learning occurs. It has been shown that the Zipf property arises to different degrees in different layers of a deep network, and is maximal in the layers that attain an optimal trade-off between resolution and accuracy in generating samples [47]. This is a starting point in linking statistical criticality to function. It is not known whether similar principles are relevant in the case of biological neural networks, and finding such a link could be an interesting direction of future research.

References

1. Aitchison, L., Corradi, N., Latham, P.E.: Zipfs law arises naturally when there are underlying, unobserved variables. PLoS Computational Biology **12**(12), 1–32 (2016)
2. Athreya, K.B., Jagers, P.: Classical and Modern Branching Processes, *IMA*, vol. 84. Springer (1997)
3. Auerbach, F.: Das Gesetz der Bevölkerungskonzentration. Petermanns Geographische Mitteilungen **59**, 74–76 (1913). (Quote translated by J.M.H.)
4. Barabási, A.L., Albert, R.: Emergence of scaling in random networks. Science **286**, 509–512 (1999)
5. Beggs, J.M.: The criticality hypothesis: how local cortical networks might optimize information processing. Philosophical Transactions of the Royal Society of London A: Mathematical, Physical and Engineering Sciences **366**(1864), 329–343 (2008)
6. Beggs, J.M., Plenz, D.: Neuronal avalanches in neocortical circuits. Journal of Neuroscience **23**(35), 11,167–11,177 (2003)
7. Beggs, J.M., Timme, N.: Being critical of criticality in the brain. Frontiers in Physiology **3**, 163 (2012)
8. Cristelli, M., Batty, M., Pietronero, L.: There is more than power law in Zipf. Scientific Reports **2**, 812(7) (2012)
9. Eurich, C.W., Herrmann, J.M., Ernst, U.A.: Finite-size effects of avalanche dynamics. Physical Review E **66**(6), 066,137 (2002)
10. Gabaix, X.: Zipf's law and the growth of cities. American Economic Review **89**(2), 129–132 (1999)
11. Gardella, C., Marre, O., Mora, T.: A tractable method for describing complex couplings between neurons and population rate. eneuro **3**(4) (2016)
12. Gardella, C., Marre, O., Mora, T.: Blindfold learning of an accurate neural metric. Proceedings of the National Academy of Sciences p. 201718710 (2018)
13. Gautam, S.H., Hoang, T.T., McClanahan, K., Grady, S.K., Shew, W.L.: Maximizing sensory dynamic range by tuning the cortical state to criticality. PLoS Computational Biology **11**(12), e1004,576 (2015)
14. Glauber, R.J.: Time-dependent statistics of the Ising model. Journal of Mathematical Physics **4**(2), 294–307 (1963)
15. Gutenkunst, R.N., Waterfall, J.J., Casey, F.P., Brown, K.S., Myers, C.R., Sethna, J.P.: Universally sloppy parameter sensitivities in systems biology models. PLoS Computational Biology **3**(10), e189 (2007)
16. Hahn, G., Ponce-Alvarez, A., Monier, C., Benvenuti, G., Kumar, A., Chavane, F., Deco, G., Frégnac, Y.: Spontaneous cortical activity is transiently poised close to criticality. PLoS Computational Biology **13**(5), 1–29 (2017)
17. Hennig, M.H., Adams, C., Willshaw, D., Sernagor, E.: Early-stage waves in the retinal network emerge close to a critical state transition between local and global functional connectivity. The Journal of Neuroscience **29**(4), 1077–1086 (2009)
18. Herzog, R., Escobar, M.J., Cofre, R., Palacios, A.G., Cessac, B.: Dimensionality reduction on spatio-temporal maximum entropy models on spiking networks. Preprint *bioRxiv*:278606 (2018)
19. Hilgen, G., Sorbaro, M., Pirmoradian, S., Muthmann, J.O., Kepiro, I.E., Ullo, S., Ramirez, C.J., Encinas, A.P., Maccione, A., Berdondini, L., et al.: Unsupervised spike sorting for large-scale, high-density multielectrode arrays. Cell reports **18**(10), 2521–2532 (2017)
20. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences **79**(8), 2554–2558 (1982)
21. Ising, E.: Beitrag zur Theorie des Ferromagnetismus. Zeitschrift für Physik **31**(1), 253–258 (1925)
22. Jaynes, E.T.: Information theory and statistical mechanics. Physical Review **106**(4), 620–630 (1957)

23. Jiang, B., Jia, T.: Zipf's law for all the natural cities in the united states: a geospatial perspective. International Journal of Geographical Information Science **25**(8), 1269–1281 (2011)
24. Kinouchi, O., Copelli., M.: Optimal dynamical range of excitable networks at criticality. Nat. Phys. **2**, 348–352 (2006)
25. Köster, U., Sohl-Dickstein, J., Gray, C.M., Olshausen, B.A.: Modeling higher-order correlations within cortical microcolumns. PLoS Computational Biology **10**(7), e1003,684 (2014)
26. Larremore, D.B., Shew, W.L., Restrepo, J.G.: Predicting criticality and dynamic range in complex networks: effects of topology. Physical Review Letters **106**(5), 058,101 (2011)
27. Li, W.: Random texts exhibit Zipf's-law-like word frequency distribution. IEEE Transactions on Information Theory **38**(6), 1842–1845 (1992)
28. Machta, B.B., Chachra, R., Transtrum, M.K., Sethna, J.P.: Parameter space compression underlies emergent theories and predictive models. Science **342**(6158), 604–607 (2013)
29. Macke, J.H., Opper, M., Bethge, M.: Common input explains higher-order correlations and entropy in a simple model of neural population activity. Physical Review Letters **106**(20), 208,102 (2011)
30. Marre, O., El Boustani, S., Frégnac, Y., Destexhe, A.: Prediction of spatiotemporal patterns of neural activity from pairwise correlations. Physical Review Letters **102**(13), 138,101 (2009)
31. Mastromatteo, I., Marsili, M.: On the criticality of inferred models. Journal of Statistical Mechanics: Theory and Experiment **2011**(10), P10,012 (2011)
32. Mizuseki, K., Buzsáki, G.: Preconfigured, skewed distribution of firing rates in the hippocampus and entorhinal cortex. Cell reports **4**(5), 1010–1021 (2013)
33. Mora, T., Deny, S., Marre, O.: Dynamical criticality in the collective activity of a population of retinal neurons. Physical Review Letters **114**(7), 078,105 (2015)
34. Nasser, H., Marre, O., Cessac, B.: Spatio-temporal spike train analysis for large scale networks using the maximum entropy principle and monte carlo method. Journal of Statistical Mechanics: Theory and Experiment **2013**(03), P03,006 (2013)
35. Newman, M.E.: Power laws, Pareto distributions and Zipf's law. Contemporary physics **46**(5), 323–351 (2005)
36. Nishimori, H.: Statistical physics of spin glasses and information processing: an introduction, vol. 111. Clarendon Press (2001)
37. Nonnenmacher, M., Behrens, C., Berens, P., Bethge, M., Macke, J.H.: Signatures of criticality arise from random subsampling in simple population models. PLoS Computational Biology **13**(10), e1005,718 (2017)
38. O'Donnell, C., Gonçalves, J.T., Whiteley, N., Portera-Cailliau, C., Sejnowski, T.J.: The population tracking model: A simple, scalable statistical model for neural population data. Neural Computation **29**(1), 50–93 (2016)
39. Ohiorhenuan, I.E., Mechler, F., Purpura, K.P., Schmid, A.M., Hu, Q., Victor, J.D.: Sparse coding and high-order correlations in fine-scale cortical networks. Nature **466**(7306), 617–621 (2010)
40. Panas, D., Amin, H., Maccione, A., Muthmann, O., van Rossum, M., Berdondini, L., Hennig, M.H.: Sloppiness in spontaneously active neuronal networks. Journal of Neuroscience **35**(22), 8480–8492 (2015)
41. Priesemann, V., Valderrama, M., Wibral, M., Le Van Quyen, M.: Neuronal avalanches differ from wakefulness to deep sleep–evidence from intracranial depth recordings in humans. PLoS Computational Biology **9**(3), e1002,985 (2013)
42. Redner, S.: How popular is your paper? an empirical study of the citation distribution. The European Physical Journal B-Condensed Matter and Complex Systems **4**(2), 131–134 (1998)
43. Schneidman, E., Berry, M.J., Segev, R., Bialek, W.: Weak pairwise correlations imply strongly correlated network states in a neural population. Nature **440**(7087), 1007–1012 (2006)
44. Shew, W.L., Plenz, D.: The functional benefits of criticality in the cortex. The Neuroscientist **19**(1), 88–100 (2013)
45. Shew, W.L., Yang, H., Petermann, T., Roy, R., Plenz, D.: Neuronal avalanches imply maximum dynamic range in cortical networks at criticality. Journal of Neuroscience **29**(49), 15,595–15,600 (2009)

46. Shlens, J., Field, G.D., Gauthier, J.L., Grivich, M.I., Petrusca, D., Sher, A., Litke, A.M., Chichilnisky, E.: The structure of multi-neuron firing patterns in primate retina. The Journal of Neuroscience **26**(32), 8254–8266 (2006)
47. Song, J., Marsili, M., Jo, J.: Emergence and relevance of criticality in deep learning. arXiv preprint arXiv:1710.11324 (2017)
48. Tang, A., Jackson, D., Hobbs, J., Chen, W., Smith, J.L., Patel, H., Prieto, A., Petrusca, D., Grivich, M.I., Sher, A., Hottowy, P., Dabrowski, W., Litke, A.M., Beggs, J.M.: A maximum entropy model applied to spatial and temporal correlations from cortical networks in vitro. Journal of Neuroscience **28**, 505518 (2008)
49. Tkačik, G., Marre, O., Amodei, D., Schneidman, E., Bialek, W., Berry II, M.J.: Searching for collective behavior in a large network of sensory neurons. PLoS Computational Biology **10**(1), e1003,408 (2014)
50. Tkačik, G., Mora, T., Marre, O., Amodei, D., Palmer, S.E., Berry, M.J., Bialek, W.: Thermodynamics and signatures of criticality in a network of neurons. Proceedings of the National Academy of Sciences **112**(37), 11,508–11,513 (2015)
51. Vázquez-Rodríguez, B., Avena-Koenigsberger, A., Sporns, O., Griffa, A., Hagmann, P., Larralde, H.: Stochastic resonance at criticality in a network model of the human cortex. Scientific Reports **7**(1), 13,020 (2017)
52. Vitanov, N.K., Ausloos, M.: Test of two hypotheses explaining the size of populations in a system of cities. Journal of Applied Statistics **42**(12), 2686–2693 (2015)
53. Yu, S., Yang, H., Nakahara, H., Santos, G.S., Nikolić, D., Plenz, D.: Higher-order interactions characterized in cortical activity. Journal of Neuroscience **30**(48), 17,514–17,526 (2011)
54. Zipf, G.K.: Human behavior and the principle of least effort. Addison-Wesley, Cambridge (1949)

Chapter 4

Local learning rules to attenuate forgetting in neural networks

This chapter consists of a submitted journal article. The paper is available as a preprint as Deistler, M., Sorbaro, M., Rule, M. E., & Hennig, M. H. (2018). Local learning rules to attenuate forgetting in neural networks. arXiv preprint arXiv:1807.05097 [Deistler et al., 2018]. *The first two authors were reported as contributing equally and are listed in alphabetical order. As required, the following introduction motivates the work, delineates my contributions, and illustrates some further work done after publication.*

The concept of model sloppiness, that is, of insensitivity to a large number of directions in the parameter space, was introduced in the previous chapter. Starting from the observation that this phenomenon appears in a wide variety of models in systems science [Gutenkunst et al., 2007, Machta et al., 2013] and specifically in neuroscience [Panas et al., 2015], one can wonder whether knowledge of Fisher Information, and therefore of the geometrical properties of a parameter space, can improve learning — both in models and in brains.

In machine learning, in particular in model optimisation, using the "natural" gradient, which is weighted by the inverse Fisher information, takes into account the geometry of the parameter space. This was shown to be a better optimisation strategy when minimising Kullback-Leibler divergence [Amari, 1998], although the problem of efficiently computing, storing and inverting the Fisher Information Tensor (FIT) is a disadvantage in some practical applications, including deep neural networks, where the parameter space is extremely high-dimensional. Re-

search in second-order optimisation methods has continued, essentially by looking for clever ways to estimate the best learning rate for each parameter. For example, the current state-of-the-art optimiser in deep learning is ADAM [Kingma and Ba, 2014], which automatically reduces the learning rate for parameters that have oscillated in the previous iterations.

It it a natural question to ask, then, whether biological neural networks may exploit their own sloppiness, or knowledge of their own Fisher information, in order to drive their plasticity and consequently learn more optimally, in any sense we wish to attribute to the word "optimal". The question we ask in this chapter is very different from the problem, mentioned above, of finding the fastest path to a minimum, and was inspired by a result in deep reinforcement learning (which, arguably, is the form of machine learning that is closest to human learning, although it is still very far from biologically realistic). Kirkpatrick et al. studied how an artificial neural network can learn two or more tasks sequentially, without forgetting the previous ones, by carefully choosing which parameters need to be kept constant [Kirkpatrick et al., 2017].

We asked the same question about a simple, early model of memory: Hopfield networks. In these networks, an all-to-all matrix of synaptic connections is learned in order to store patterns of neural activations which represent memories: the number of patterns that can be stored in a network of N neurons is known to grow linearly in N. Hopfield networks are not designed to learn sequentially, but a look at the eigenvalues of the Fisher information tensor suggests there are "sloppy" parameters that can be exploited in order to preserve the memories stored in the "stiff" ones for a longer time. Indeed, the number of non-zero eigenvalues is proportional to the number of patterns stored, as will be shown in an appendix to this chapter (Section 4.1). Note that the presence of sloppy dimensions is not sufficient to guarantee the model is still able to learn new tasks or memories. However, leaving stiff parameters untouched should indeed preserve the performance of the model on what was previously learned.

We experimented with this idea and found a positive answer, a learning rule that delays forgetting, and has the additional desirable property of being implemented locally: it does not require knowledge of the whole network when updating a single synaptic weight.

Hopfield networks are analogous, in a specific mathematical sense, to zero-temperature Boltzmann machines. The understanding of the properties of Boltz-

mann machines, built in the previous chapters, will be necessary in order to derive the form of Fisher information we adopted for Hopfield nets.

My contributions The general problem of finding a Fisher-based learning rule that could exploit sloppiness in a computational neuroscience context was posed by Matthias Hennig, Michael Rule, and myself. The initial exploration of possible models for which such a learning rule could be designed was done by me, and the idea of using Hopfield networks was mine, in the context of helpful discussions with Ramón Martinez, Wioleta Kijewska, Cole Hurwitz, and Nathalie Dupuy, who helped setting the project on the right track. A large part of the coding work and of the theoretical findings was done by a talented master's student, Michael Deistler, under my supervision. I contributed an initial minimal code example, and I revised, tested and optimised his code for speed. This code was run and the plots and results were obtained by Michael and myself, in the first phase, and by Matthias, in the final phase, with extensive discussions with all authors. All authors contributed to the theoretical results and in writing the paper. The results in appendix 4.1 are mine.

Local learning rules to attenuate forgetting in neural networks

Michael Deistler[1], Martino Sorbaro[2,3], Michael E. Rule[2], Matthias H. Hennig[2,*]

May 20, 2019

1 Technical University of Munich, Germany
2 Institute for Adaptive and Neural Computation, School of Informatics, University of Edinburgh, UK
3 Computational Science and Technology, KTH Royal Institute of Technology, Stockholm, Sweden

* m.hennig@ed.ac.uk

Abstract

Hebbian synaptic plasticity inevitably leads to interference and forgetting when different, overlapping memory patterns are sequentially stored in the same network. Recent work on artificial neural networks shows that an information-geometric approach can be used to protect important weights to slow down forgetting. This strategy however is biologically implausible as it requires knowledge of the history of previously learned patterns. In this work, we show that a purely local weight consolidation mechanism, based on estimating energy landscape curvatures from locally available statistics, prevents pattern interference. Exploring a local calculation of energy curvature in the sparse-coding limit, we demonstrate that curvature-aware learning rules reduce forgetting in the Hopfield network. We further show that this method connects information-geometric global learning rules based on the Fisher information to local spike-dependent rules accessible to biological neural networks. We conjecture that, if combined with other learning procedures, it could provide a building-block for content-aware learning strategies that use only quantities computable in biological neural networks to attenuate pattern interference and catastrophic forgetting. Additionally, this work clarifies how global information-geometric structure in a learning problem can be exposed in local model statistics, building a deeper theoretical connection between the statistics of single units in a network, and the global structure of the collective learning space.

Significance

How can neural networks avoid interference and forgetting when sequentially learning different yet overlapping memory patterns? In artificial neural networks, this problem has been solved using the geometric structure of parameter space conveyed by the Fisher information matrix (FIM),

which reveals weights in the network that are important for encoding previously learned patterns. However, these weight consolidation rules are biologically implausible as they require global information about the parameter space and the history of learned patterns. Here we show mathematically and in simulations that an attractor network can approximate such learning rules with locally available information. This work suggests a novel interpretation of weight-dependent synaptic modifications observed experimentally, and purely local learning rules that mitigate against catastrophic forgetting in artificial neural networks.

Introduction

Artificial Neural Networks (ANNs) have become adept at solving both supervised and unsupervised machine-learning tasks. Unlike biological neural networks however, ANNs are vulnerable to catastrophic forgetting [19]: ANNs forget their original trained structure if re-trained on new inputs. Recent studies have addressed catastrophic forgetting by constraining learning through globally-computed information about the importance of network parameters [22, 17, 27, 26, 1, 23, 16, 15]. However, biological neural networks must achieve the same through locally available information: neither the backpropagation algorithm [5], nor the creation of new units [28], nor non-local calculations of weight importance, can be implemented in biological networks as we currently understand them.

Here we introduce an approach that requires no information about previously stored memories and uses the measure of importance not as part of a loss function, but as a scaling factor for the learning rate. Addressing catastrophic forgetting in sequential learning in a Hopfield network, we derive a local Hebbian learning rule that calculates weight importance via a simple weight transformation. We show that this transformation is equivalent to computing Fisher Information Matrix (FIM) entries in a statistical model and that it provides a biologically plausible means to implement FIM-based solutions to catastrophic forgetting [15, 22, 17].

Results

Hopfield Networks

A Hopfield network is a network consisting of M binary nodes x_i, which are fully connected through symmetric weights w_{ij}. We use this network to store and retrieve a set of patterns $\mathbf{p}^1 \dots \mathbf{p}^N$, with $p_i^n \in \{0, 1\}$. The sparsity of these patterns s is defined as the ratio of bits being 1: $s = \langle p \rangle$. Classically [13, 30], Hopfield networks are trained using a local Hebbian learning rule, in which the weights are set to

$$w_{ij} = \frac{1}{N} \sum_{n=1}^{N} (p_i^n - s)(p_j^n - s) = \frac{1}{N} \sum_{n=1}^{N} \xi_i^n \xi_j^n, \tag{1}$$

with N the number of stored patterns, and where we defined $\xi_i^n = p_i^n - s$. Encoding patterns in terms of ξ instead of p guarantees the mean of all encoded patterns is 0, as required for optimal pattern separation [30]. The capacity of a Hopfield Network with a sparsity of s=0.5 is about $0.138 \times M$ [2], while sparser patterns lead to a higher capacity of the network, proportional to $(s|\ln(s)|)^{-1}$ [30].

Using the learning rule given above, we initialize the network by inducing local minima into the energy surface corresponding to our stored patterns. The energy of a given pattern x is defined as

$$E(x) = -\sum_{i,j=1}^{M} x_i x_j w_{ij}.$$

Given a weight matrix, if x^t is the network state at time t, the network dynamics is defined as

$$x_i^{t+1} = \Theta\left(\sum_{j=1}^{M} w_{ij} x_j^t - \theta\right), \tag{2}$$

where Θ is the Heaviside step function and θ is a bias accounting for the sparsity, known as the neural threshold [30]. Repeating this several times, either synchronously for all neurons or asynchronously for a randomly chosen neuron, leads the network to converge into the energy minimum closest to its initial configuration.

Parameter importance and sequential learning

In real-world learning, an agent is not presented all at once with all the information it needs to remember, nor does it have the chance of interleaving training on one memory with training on another and vice versa. Memories may have to be stored, and can be stored, one after the other, in sequence. If we want to investigate sequential learning in a Hopfield network, we can introduce an incremental rule

$$\Delta w_{ij} = \eta \cdot \left(\frac{1}{N}\sum_{n=1}^{N} \xi_i^n \xi_j^n - w_{ij}\right) \tag{3}$$

with learning rate η. It's easy to see that, following this rule until convergence, the weight matrix will asymptotically take the value given in Equation (1). The problem of this approach is that, while the new pattern is learned, the weight matrix is eventually entirely overwritten, and the previously stored patterns are forgotten [20].

The general idea we are aiming to implement to address this problem is that parameters that are particularly "important" for retrieving stored memories should be changed at a lower rate, or left untouched, by learning additional patterns. This will be successful if the energy landscape is highly anisotropic with respect to parameter changes, which usually is the case in ANNs as they are overdetermined. Then, some parameter changes (or combinations) cause a strong change, while others have little or no effect and can be used for new patterns. This defines sensitive and insensitive directions in parameter space, which we expect to exist in a network not saturated close to capacity, where energy minima would be quite close to each other.

Fisher Information in a Hopfield network

In probabilistic models, the Fisher Information Matrix, which describes the geometry of the parameter space, can provide a measure of "sloppiness" for the model parameters, indicating the level of plasticity a certain weight can have. This allows learning to focus on relatively unimportant parameters, leaving important or "stiff" weights undisturbed. These terms, introduced in a more general context by Gutenkunst et al. [11], define the parameter space anisotropy we want

to exploit here to prevent forgetting. It is not immediately intuitive how to define the concept of Fisher Information, which applies to parameter-dependent probability distributions, in the case of deterministic Hopfield networks, where dynamics exist that drive activity into the attractor states that correspond to stored memories. However, a consistent definition can be attained by realising that the Hopfield system is equivalent to the fully visible Boltzmann Machine (FVBM) at the zero-temperature limit.

Given a probability distribution P_w dependent on a matrix of parameters $\{w_{ij}\}$, the Fisher information matrix is defined as

$$F_{w_{ij},w_{kl}} = \left\langle \frac{\partial \log P_w(x)}{\partial w_{ij}} \frac{\partial \log P_w(x)}{\partial w_{kl}} \right\rangle, \tag{4}$$

where the brackets $\langle\cdot\rangle$ denote averaging over all patterns stored in the network. Since we are interested in a measure of single-parameter importance, we consider only the diagonal of the FIM: $F_{ij} \equiv F_{w_{ij},w_{ij}}$, which then expresses the sensitivity of the distribution to changes in the weight w_{ij}.

The FVBM, a model identical to the fully connected Ising model, has the form

$$P_w(x) = \frac{1}{Z(T)} \exp\left(\frac{1}{T} \sum_{i\neq j} x_i x_j w_{ij}\right). \tag{5}$$

The FIM can be computed from a sample extracted from P_w, rather than from its analytical expression, exploiting the fact that (see Appendix A):

$$F_{w_{ij},w_{kl}} = \mathrm{Cov}\Big[x_i x_j, x_k x_l\Big], \quad \text{and therefore} \quad F_{ij} = \mathrm{Var}\Big[x_i x_j\Big] \tag{6}$$

In the case of a Hopfield network, however, no probability distribution of states exists, since the dynamics of the model is limited to convergence to attractors. Yet, the FVBM probability distribution (5) converges, in the $T \to 0$ limit, to null probability for all patterns except the ones of lowest energy. This coincides with the equilibrium distribution of a Hopfield network, which has finite probability on attractors (learned or spurious) and zero probability elsewhere. Analogously, it can be shown that the dynamics of the FVBM, for example the one defined by Monte Carlo sampling, are equivalent to the Hopfield time evolution defined in equation (2).

Assuming the network parameters are set such that no spurious attractors exist, the stable patterns coincide with learned patterns in a trained network. This allows computing the variance in (6) over this distribution, and the FIM as

$$F_{ij} = \frac{1}{N} \sum_{n=1}^{N} (\xi_i^n \xi_j^n)^2 - \frac{1}{N^2} \left(\sum_{n=1}^{N} \xi_i^n \xi_j^n \right)^2. \tag{7}$$

The importance of each weight, computed in an appropriate way as $\Omega_{ij} = f(F_{ij})$, with f being a monotonically decreasing function, can then be used to scale the learning rate in order to protect stored memories:

$$\Delta w_{ij} = \eta \Omega_{ij} (\xi_i \xi_j - w_{ij}). \tag{8}$$

A biologically plausible learning rule

In order to evaluate the importance of a connection locally, the network has to constantly compute the sum in equation (7), which requires constant sampling of previously memorized patterns. This process is not impossible: the recall and replay of memories, for example during sleep, has been both experimentally observed and theoretically studied as a means of memory consolidation [31, 29]. However, we will here show that there is an even simpler local way of estimating importance from the value of the weight, at least for a Hopfield network.

We use (6) to write the diagonal entries of the Fisher Information Matrix as

$$F_{w_{ij},w_{ij}} = F_{ij} = \mathrm{Var}\left[x_i x_j\right] = \left\langle x_i^2 x_j^2 \right\rangle - \left\langle x_i x_j \right\rangle^2.$$

We would like an expression for the diagonal of the FIM that depends only on locally-available weight information. We can use the fact that $w_{ij} = \left\langle x_i x_j \right\rangle$ by construction (6) to write

$$F_{ij} = \left\langle x_i^2 x_j^2 \right\rangle - w_{ij}^2, \tag{9}$$

but the question remains of how to estimate $\left\langle x_i^2 x_j^2 \right\rangle$, which is a fourth moment of the activity distribution. It is possible (Appendix B) to expand this term as a function of means and correlations:

$$\begin{aligned} \left\langle x_i^2 x_j^2 \right\rangle = {} & (1-2s)^2 \left\langle x_i x_j \right\rangle \\ & + s(1-s)(1-2s)\left(\left\langle x_i \right\rangle + \left\langle x_j \right\rangle\right) \\ & + s^2(1-s)^2 \end{aligned} \tag{10}$$

However, the expected activation rates $\left\langle x_i \right\rangle$ and $\left\langle x_j \right\rangle$ are not directly accessible during sequential learning, since previously learned patterns are not 'sampled' by the network during the learning process. These mean activations are correlated with the weights, and a simple closed-form approximation does not exist. Using the fact that patterns are centered at zero-mean activation rates, we can write the approximation

$$F_{ij} \approx s^2(1-s)^2 + (1-2s)^2 w_{ij} - w_{ij}^2. \tag{11}$$

In two limiting cases, the dependence on the expected rates vanishes, and Eq. (11) holds with equality. At s=0.5, F_{ij}=$\frac{1}{16}$−w_{ij}^2, and at s=0, F_{ij}=$w_{ij}(1-w_{ij})$. We now focus on learning near the sparse-coding limit, with $s \ll 0.5$. To simplify the online, local estimation of weight importance, we consider an approximation for a perturbation around the sparse ($s \to 0$) limit. Expanding Eq. (11) to first-order in s gives

$$F_{ij} \approx w_{ij}(1-w_{ij}) - 4s w_{ij} \tag{12}$$

Figure 1 illustrates this approximation. Assuming that the weights are in the range $w \in [0,1]$ for $s \to 0$, there is a reasonable correspondence between the predicted and actual Fisher information for $s = 0.1$, and the relationship is exactly reproduced for $s = 0.5$. Once several sparse patterns are stored, only values to the left of the maximum of the parabola given by Equation (12) appear, an effect that becomes increasingly evident as more patterns are stored. This is expected because, in order to have $w_{ij} = \frac{1}{N}\sum \xi_i^n \xi_j^n > 0.5$ and assuming $\zeta_i \zeta_j \in \{0,1\}$ in the limit $s \to 0$, we need more than half of the connections to be

$$\xi_i^n \xi_j^n = 1. \tag{13}$$

s	2 patterns	5 patterns	10 patterns	20 patterns
0.05	6.2500e-06	1.5566e-07	5.0848e-14	0
0.1	1.0000e-04	9.8506e-06	2.0289e-10	0

Table 1: Estimated Probabilities of $w_{ij} > 0.5$ for 2, 5, 10 and 20 patterns and sparsities $s = 0.05$ and $s = 0.1$.

The probability for this to happen is

$$p(\zeta_i\zeta_j = 1) = p(\zeta_i = 1, \zeta_j = 1) = p(\zeta_i = 1)p(\zeta_j = 1) = s^2. \tag{14}$$

When storing N patterns, at least $\frac{N}{2}$ have to be 1, which is a process that can be described by the Cumulative Distribution Function of the binomial distribution, for which no analytical solution exists. In table 1, some numerical values are shown.
As it can be seen, the probabilities for the weight being above 0.5 is decreasing with the number of patterns and is negligible small for a sufficient number of patterns.

These results show that for a model storing sparse patterns, the weight sensitivities increase monotonically with the value of the weight. This relationship is well captured even with a linear function. This allows constructing heuristic, fully local and hence biologically plausible learning rules that only modify irrelevant weights during continuous learning. To this end, we can generalize Equation (8) to introduce an additional correction to the learning rate Ω_{ij} depending on the weight value.

In the following we investigate two approaches for learning rate correction. The first consists of imposing a threshold Θ_w on each weight:

$$\Omega_{ij} = \begin{cases} 1, & \text{for } w_{ij} \leq \Theta_w \\ 0, & \text{for } w_{ij} > \Theta_w \end{cases} \tag{15}$$

Second, the relation between Fisher information and weight in Equation 12 suggests any strictly positive, monotonically decreasing function of the weight should provide an appropriate learning rate correction. An interpretation of the weight value as the curvature of a Gaussian approximation to the weight posterior predicts an inverse relationship (see Appendix C for derivation). In simulations we however found a better performance using an exponential scaling of the weight with

$$\Omega_{ij} = \exp(-a|w_{ij}|). \tag{16}$$

Following the Bayesian interpretation, this rule can be further augmented by only updating weights with strong changes:

$$\Omega_{ij} = \begin{cases} \exp(-a|w_{ij}|) & \text{for } \Delta w_{ij} > \Theta_{\Delta w} \\ \Omega_{ij} = 0 & \text{else} \end{cases} \tag{17}$$

In addition to not modifying important weights, this will prevent weight changes that do not support the new pattern. In the following we show in simulations that these rules indeed prevent overwriting of stored memories, and enable continuous learning in the Hopfield network.

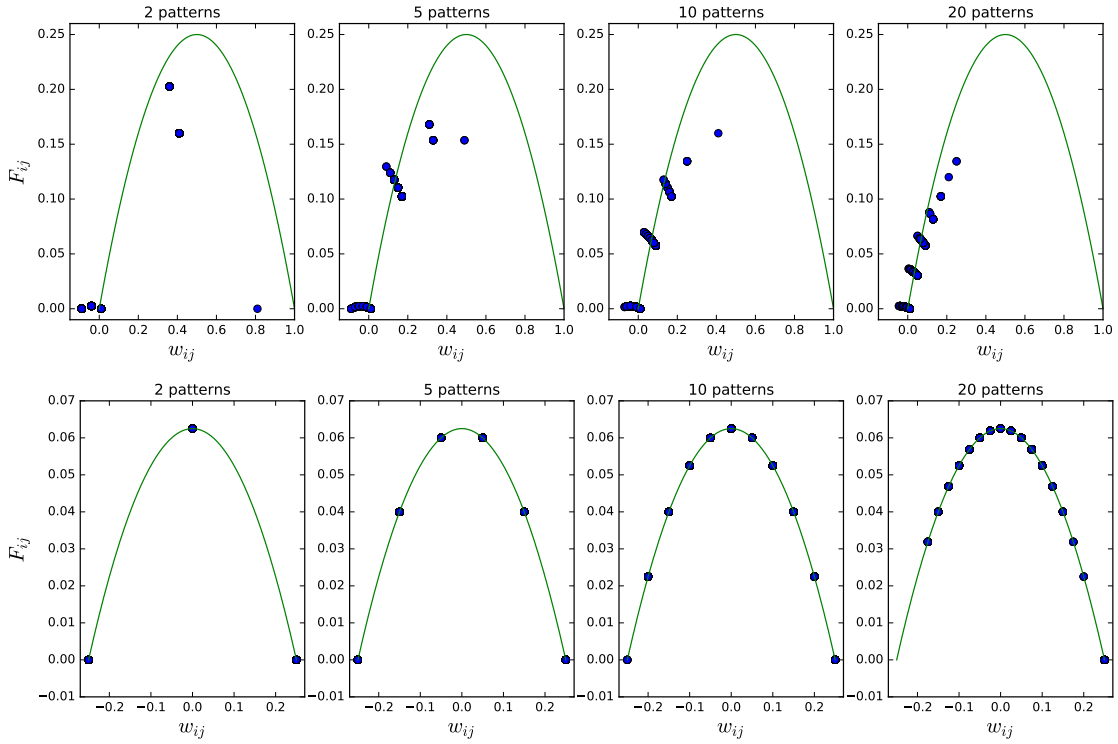


Figure 1: The Fisher information-based measure of importance as a function of the weight value in a network. Blue dots refer to the true values, computed according to equation 6, green lines to the theoretically derived relation (11). Top: measurements for $s = 0.1$ compared to the analytically available relation for $s \to 0$, Equation 12. Bottom: sparsity $s = 0.5$.

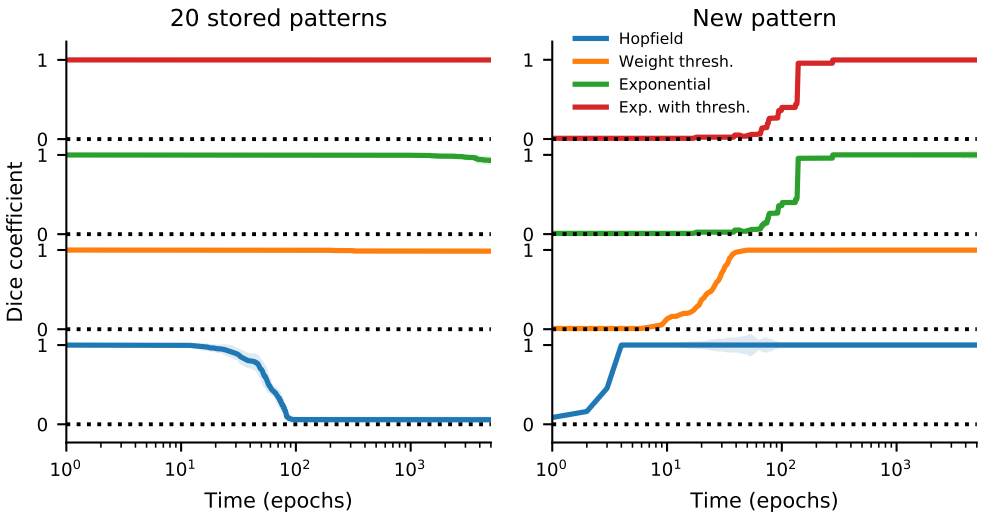


Figure 2: Modified local Hebbian learning rules prevent catastrophic forgetting. A network of 100 neurons was initialised with 20 patterns, and a novel pattern was learned with the incremental rule (Equation 3), and augmented versions of this rule. The learning rate was $\eta = 0.01$, curves show the average Dice coefficient from 20 simulations. Parameters for the augmented learning rules: weight threshold, Eqn. 15 - $\Theta_w = 0.001$; exponential, Eqn. 16 - $a = 220$; exponential with threshold, Eqn. 17 - $a = 220$ and $\Theta_{\Delta w} = 0.2 * \eta$

Simulations

We first consider a network with 20 patterns stored using Equation 1, and a new pattern learned with the incremental rule (Eqn. 3). Pattern retention is assessed by the Sørensen-Dice coefficient $D = 2 \cdot |A \cap B| / (|A| + |B|)$, where A and B reflect the binary bit-vectors reflecting a target (true) pattern and a recovered (stored) pattern, computed at each learning rule iteration by synchronously applying Equation 2 ten times.

As expected, the incremental rule rapidly removes all traces of the previously stored patterns, while the new pattern is reliably stored. Reducing the learning rate η can increase retention as overwriting is slower, but this also slows down the learning of the new pattern, always resulting in exponentially fast forgetting [4]. In contrast, for the augmented local rules the stored patterns are retained. Since the modified rule effectively shows the learning rate, storing the new pattern is slower than during normal Hopfield learning. Importantly, in these simulations the learning rate correction is re-computed at each iteration based on the current weights. Therefore successful retention is possible even for a rule operating entirely locally on each weight and in time.

Next we extend our approach to continuous learning, presenting new patterns one by one for a fixed number of iterations, and updating weights with the incremental rule. Since the Fisher information is flat for a randomly connected network, which would prevent learning with the augmented rules, the network was initialized with 20 patterns that are not tracked. For each learning rule, parameters were numerically optimized to maximize the DICE coefficient for 60 patterns with the Nelder-Mead simplex algorithm (fmin() in SciPy).

As for a single pattern, the augmented learning rules prevent forgetting also for continuous learning of multiple patterns. As can be seen from figure 3B, the hard weight threshold allows loading the network up to the theoretical capacity of about 60 patterns (100 units, sparseness $s = 0.1$),

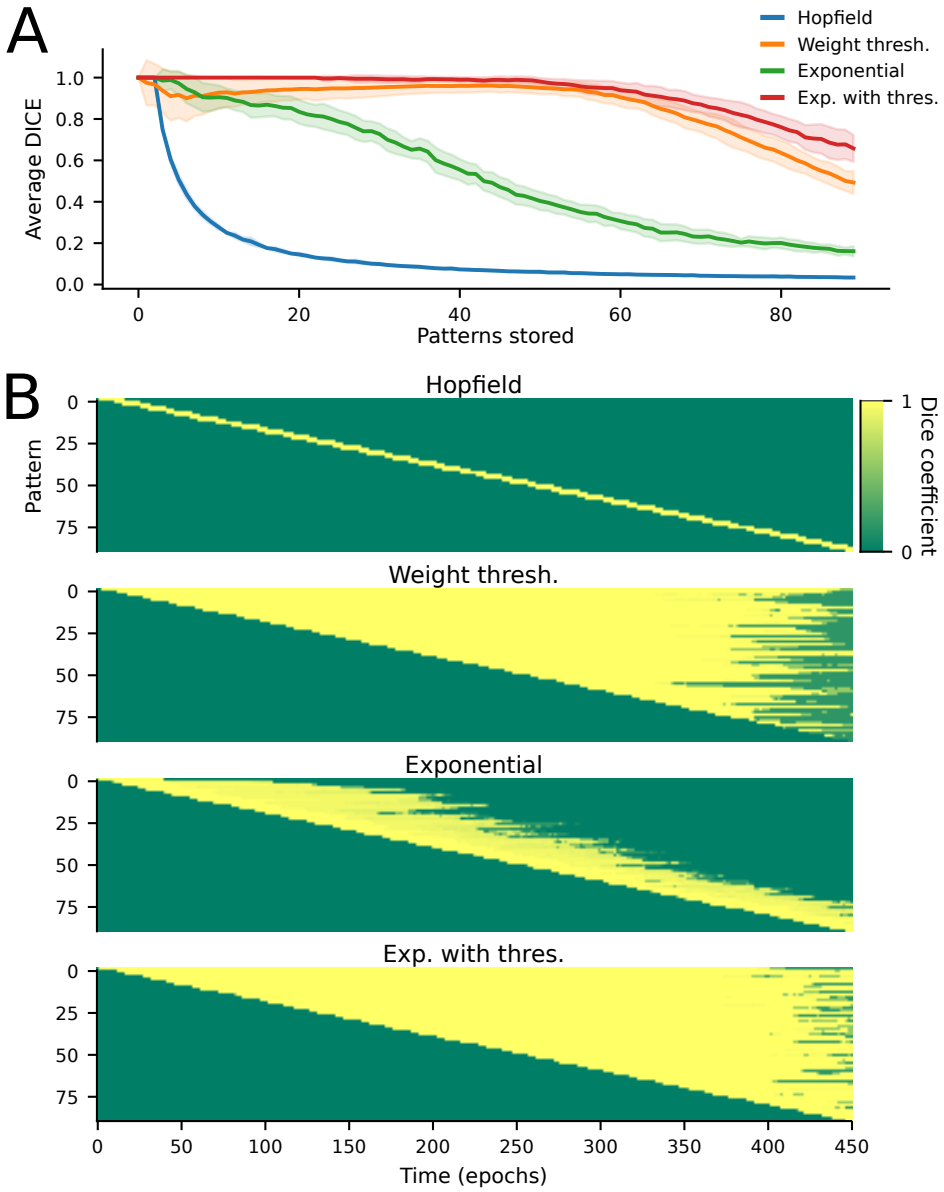


Figure 3: Continuous learning without catastrophic forgetting. In a 100 unit network, pre-trained with 20 patterns (disregarded for evaluation and not shown in these figures), 80 novel patterns were learned in succession. Each pattern was presented for five epochs, with learning rate $\eta = 0.1$, and with parameters optimized to maximize recall for 60 patterns. **A** The average DICE coefficient for all previously stored patterns, computed every time a new pattern was stored, using different learning rules. The averages are for 20 simulations, and shaded areas indicate the standard deviation. **B** The average Dice coefficient for each pattern as a function of the training epoch, for different learning rules. The simulations show that augmented learning rules have improved retention, compared to the normal Hebb rule, but their behavior differs with increasing network loading.

at around $t = 350$ in this example. Beyond this capacity, all stored patterns are erased simultaneously. In contrast, the exponential rule, which continuously modifies all weights, exhibits gradual forgetting, but can retain fewer patterns. In the relevant panel of figure 3B, a smaller number of patterns have Dice coefficient close to 1 at any given time, i.e. the network's capacity is effectively smaller than when thresholding the weights. However, this also prevents catastrophically forgetting all patterns simultaneously due to network overloading. Finally, when only larger weight changes beyond a fixed threshold are allowed for the exponential rule, full network loading is again possible.

Discussion

How neurons in the brain coordinate globally to store and retrieve information remains a major open question. In particular, we do not understand how global optimization problems can be solved reliably and robustly using only local learning rules. In this work, we have explored in a Hopfield network one aspect of this global coordination: how patterns might be routed and stored in an associative memory to reduce pattern interference and catastrophic forgetting.

FIM-based approaches to continuous learning have been adopted before [32, 15, 1]. However, these approaches employ a regularized loss function to account for previously learned tasks. Kirpatrick et al. [15] approximate the negative log-likelihood curvature using the diagonal of the Fisher Information Matrix (FIM) for each learned task, and Zenke et al. [32] propose a similar approach that can be calculated online. Recently, an approach which can be generalized to be local in supervised feed-forward networks has been described by Aljundi et al. [1]. This approach can be generalized as a local learning rule only given certain assumptions such as a Rectified Linear Unit (ReLU) activation function. However, all of those approaches rely on implementing a regularized loss function. In contrast, scaling the learning rates alters the timescales of forgetting, but does not change the asymptotic behavior of the network: at sufficiently long timescales, the accumulated information of new patterns will still overwrite previously stored patterns. Therefore, our approach attenuates forgetting and interference in the learning phase, and might be combined with other strategies to achieve more permanent stability. In addition to immediate impact in how we understand learning in biological neural networks, local learning rules have potential to accelerate machine learning as global connectivity requirements can suffer from memory transfer bottlenecks in large-scale parallel implementations running on graphics processes and clusters.

We apply this approach to a Hopfield network, a fully connected network that stores patterns via Hebbian learning, and retrieves those patterns through dynamics that minimize an energy function, moving activity into local basins of attraction [13, 6]. Its inherent instabilities and unlearning of previously learned patterns have attracted considerable interest in the past [14, 24, 8]. The Hebbian learning rule and neural-like dynamics make the Hopfield network a relevant model of biological memory, although the required symmetric weights are biologically implausible. Our model however makes two specific predictions for networks that implement similar attractor dynamics, which are both supported by experimental data. First, individual synapse stability is expected to be proportional to its strength, since strong synapses are the most important retaining memories. This effect has been reported in chronic *in vivo* experiments monitoring cortical spines [12, 18]. Second, as cortical networks mature, and more synapses are involved in maintaining stored memories, the proportion of stable synapses should increase. Chronic imaging during cortical development, which demonstrated an increase in the fraction of stable synapses after the

critical period, confirm this prediction [10, 12]. It is interesting to note that there appears to be less synapse stability in the hippocampus [3], which may be consistent with its function as a temporary episodic memory system, and suggests potential differences in the synaptic plasticity rule.

Further evidence for protection from catastrophic forgetting in cortical networks comes from studies investigating the stability of neural activity over longer time intervals. These found a small fraction of very stable neurons, which had high firing rates and were important for stabilizing the whole network dynamics, while the remaining neurons showed considerable changes [21, 9]. This pattern does not emerge in the Hopfield model, where the activity of neurons is more homogeneous than in cortical circuits, and suggests an additional organizing principle in cortical networks that conveys stability of acquired knowledge [25, 7].

In addition to biological learning, our results can be generalized to stochastic ANNs. The Hopfield model exactly replicates the behavior of a fully visible Boltzmann machine (FVBM) at zero temperature, where the structure of the weights between neurons allows only certain activity configurations corresponding to local minima in the energy landscape. Hence, the Hopfield energy function can also be interpreted as negative log-likelihood of a FVBM. In this case, a Fisher Information based learning rule will protect low-energy network configurations which correspond to high probability states. Since the learning rule protects the joint configuration of the whole network, relevant learned configurations, for instance trained through backpropagation in a deep network, are stable during continuous learning of new tasks, as demonstrated using a penalty in a global loss function by Kirkpatrick et al. [15].

Acknowledgments

Funding was provided by the Engineering and Physical Sciences Research Council grant EP/L027208/1. M.S. was supported by the EuroSPIN Erasmus Mundus Program, the EPSRC Doctoral Training Centre in Neuroinformatics (EP/F500385/1 and BB/F529254/1), and a Google Doctoral Fellowship. We thank Mark van Rossum and João Sacramento for comments.

Appendix A: Derivation of Fisher information as covariance

Under certain regularity assumptions, we can rewrite the Fisher Information as

$$F_{w_{ij},w_{kl}} = -\left\langle \frac{\partial}{\partial w_{ij}\partial w_{kl}} \log\big(p(\underline{x})\big)\right\rangle.$$

This form provides an intuitive interpretation of F_{ij} as the curvature of the energy landscape. A high Fisher Information hence corresponds to a high curvature - and hence a strong change in energy when perturbing the given parameter. in the above equation, the probability of a pattern is given by

$$p(\underline{x}) = \frac{1}{Z}\exp\big(-E\big) = \frac{1}{Z}\exp\Big(\sum_{n,m} x_n x_m w_{nm}\Big) \tag{18}$$

with Z being

$$Z = \sum_{\{\underline{x}\}} \exp\Big(\sum_{n,m} x_n x_m w_{nm}\Big)$$

We can hence write the log-probability as

$$\log\big(p(\underline{x})\big) = \Big(\sum_{n,m} x_n x_m w_{nm}\Big) - \log(Z)$$

Plugging this into (4) leads to

$$F_{w_{ij},w_{kl}} = \left\langle \left(\frac{\partial}{\partial w_{ij}}\Big(\sum_{n,m} x_n x_m w_{nm}\Big) - \log(Z)\right)\left(\frac{\partial}{\partial w_{kl}}\Big(\sum_{n,m} x_n x_m w_{nm}\Big) - \log(Z)\right)\right\rangle$$

We differentiate this using the chain rule

$$F_{w_{ij},w_{kl}} = \left\langle \left(x_i x_j - \frac{1}{Z}\frac{\partial}{\partial w_{ij}}Z\right)\left(x_k x_l - \frac{1}{Z}\frac{\partial}{\partial w_{kl}}Z\right)\right\rangle$$

We differentiate Z

$$\frac{\partial}{\partial w_{ij}}Z = \sum_{\{\underline{x}\}}\left(\exp\Big(\sum_{n,m} x_n x_m w_{nm}\Big)x_i x_j\right)$$

and use (18) leading to

$$F_{w_{ij},w_{kl}} = \left\langle \Big(x_i x_j - \sum_{\{\underline{x}\}} p(\underline{x}) x_i x_j\Big)\Big(x_k x_l - \sum_{\{\underline{x}\}} p(\underline{x}) x_k x_l\Big)\right\rangle$$

which is the definition of covariance:

$$F_{w_{ij},w_{kl}} = \mathrm{Cov}\Big[x_i x_j, x_k x_l\Big].$$

Appendix B: Expansion of FIM diagonal in terms of weights

In order to estimate the FIM diagonal entries for the weights, we must estimate fourth moments of the activity, $\left\langle x_i^2 x_j^2 \right\rangle$. The Hopfield network represents the zero-temperature limit of a pairwise spin model, which is determined entirely by the first two moments $\langle x \rangle$ and $\langle x x^\top \rangle$. It is therefore possible to derive an expression for this fourth moment, $\left\langle x_i^2 x_j^2 \right\rangle$, in terms of means and correlations. We first expand $\left\langle x_i^2 x_j^2 \right\rangle$ based on x_i=p_i−s and x_j=p_j−s, where p reflects the binary patterns being encoded, and s is the sparsity level of our encoding:

$$\left\langle x_i^2 x_j^2 \right\rangle = \left\langle (p_i - s)^2 (p_j - s)^2 \right\rangle$$

Because p is binary, $p^2 = p$, and the quadratic terms simplify on expansion:

$$\begin{aligned} (p-s)^2 &= p^2 - 2sp + s^2 \\ &= p - 2sp + s^2 \\ &= p(1-2s) + s^2 \end{aligned} \tag{19}$$

One can therefore expand $\left\langle x_i^2 x_j^2 \right\rangle$ as

$$\begin{aligned} \left\langle x_i^2 x_j^2 \right\rangle &= \left\langle (p_i(1-2s) + s^2)(p_j(1-2s) + s^2) \right\rangle \\ &= (1-2s)^2 \left\langle p_i p_j \right\rangle + s^2(1-2s)\left[\left\langle p_i \right\rangle + \left\langle p_j \right\rangle\right] + s^4 \end{aligned} \tag{20}$$

This simplification, arising the binary nature of the spins p, will allow us to express $\left\langle x_i^2 x_j^2 \right\rangle$ in terms of lower-order moments. We would like this expansion in terms of weights $w_{ij} = \left\langle x_i x_j \right\rangle$, and so substitute p_i=x_i+s and p_j=x_j+s

$$\begin{aligned} \left\langle x_i^2 x_j^2 \right\rangle &= (1-2s)^2 \left\langle (x_i + s)(x_j + s) \right\rangle \\ &\quad + s^2(1-2s)\left[\left\langle x_i + s \right\rangle + \left\langle x_j + s \right\rangle\right] + s^4 \\ &= (1-2s)^2 \left[\left\langle x_i x_j \right\rangle + s\left(\left\langle x_i \right\rangle + \left\langle x_j \right\rangle\right) + s^2\right] \\ &\quad + s^2(1-2s)\left[\left\langle x_i \right\rangle + \left\langle x_j \right\rangle + 2s\right] + s^4 \\ &= (1-2s)^2 \left\langle x_i x_j \right\rangle \\ &\quad + s(1-s)(1-2s)\left(\left\langle x_i \right\rangle + \left\langle x_j \right\rangle\right) \\ &\quad + s^2(1-s)^2 \end{aligned} \tag{21}$$

This expresses $\left\langle x_i^2 x_j^2 \right\rangle$ in terms of fist moments, $\left\langle x_i \right\rangle$, and second moments $\left\langle x_i x_j \right\rangle$. The second moments are identified with the weights w_{ij} by construction (Eq. 1). The expected activations $\left\langle x_i \right\rangle$ could be estimated via sampling, if stored patterns are re-activated, or may be approximated by their expected-value of zero.

Appendix C: Curvature-aware Hebbian learning

We begin with the incremental learning rule, dropping indices for legibility,

$$\Delta w = \Omega(\hat{w} - w), \tag{22}$$

where w is a weight, $\hat{w}$ is the new weight indicated by data, and Ω is a function that adjusts the learning rate. Time constants, step size, and learning rates have been absorbed into Ω in this case. This equation can also be written by interpreting Ω as a convex combination of the old and the new weights as

$$w_{new} = w + \Delta w = w + \Omega(\hat{w} - w) = \Omega\hat{w} + (1 - \Omega)w. \tag{23}$$

We now consider a Bayesian update of a Gaussian approximation to the posterior state for the value of a weight w. Let our current estimate have mean w and precision τ. Let our update have estimated weight $\hat{w}$ and a constant precision c. We then interpret the Fisher information as the curvature (precision) of the prior, thus equating our FIM estimate with the precision τ.

The Bayesian update to the mean of a Gaussian is the weighted sum

$$w_{new} = \frac{c\hat{w} + \tau w}{c + \tau}, \frac{c}{c + \tau}\hat{w} + \frac{\tau}{c + \tau}w. \tag{24}$$

Introducing

$$\beta = \frac{c}{c + \tau} = \frac{1}{1 + \frac{1}{c}\tau}, \tag{25}$$

we can write the weight update as the convex combination

$$w_{new} = \beta\hat{w} + (1 - \beta)w. \tag{26}$$

Interpreting τ as the FIM for weight w, and using $\tau \approx w(1 - w)$, we obtain

$$\beta = \frac{1}{1 + \frac{1}{c}w - \frac{1}{c}w^2}. \tag{27}$$

References

[1] R. Aljundi, F. Babiloni, M. Elhoseiny, M. Rohrbach, and T. Tuytelaars. Memory aware synapses: Learning what (not) to forget. *arXiv preprint arXiv:1711.09601*, 2017.

[2] D. J. Amit, H. Gutfreund, and H. Sompolinsky. Storing infinite numbers of patterns in a spin-glass model of neural networks. *Physical Review Letters*, 55(14):1530, 1985.

[3] A. Attardo, J. E. Fitzgerald, and M. J. Schnitzer. Impermanence of dendritic spines in live adult ca1 hippocampus. *Nature*, 523(7562):592, 2015.

[4] A. B. Barrett and M. C. van Rossum. Optimal learning rules for discrete synapses. *PLoS Computational Biology*, 4(11):e1000230, 2008.

[5] Y. Bengio, D.-H. Lee, J. Bornschein, T. Mesnard, and Z. Lin. Towards biologically plausible deep learning. *arXiv preprint arXiv:1502.04156*, 2015.

[6] B. Cheng and D. M. Titterington. Neural networks: A review from a statistical perspective. *Statistical Science*, pages 2–30, 1994.

[7] M. Fauth, F. Wörgötter, and C. Tetzlaff. Formation and maintenance of robust long-term information storage in the presence of synaptic turnover. *PLoS Computational Biology*, 11(12):e1004684, 2015.

[8] R. M. French. Catastrophic forgetting in connectionist networks. *Trends in Cognitive Sciences*, 3(4):128–135, 1999.

[9] A. D. Grosmark and G. Buzsaki. Diversity in neural firing dynamics supports both rigid and learned hippocampal sequences. *Science*, 351(6280):1440–1443, 2016.

[10] J. Grutzendler, N. Kasthuri, and W.-b. Gan. Long-term dendritic spine stability in the adult cortex. *Nature*, 420(6917):812–6, 2002.

[11] R. N. Gutenkunst, J. J. Waterfall, F. P. Casey, K. S. Brown, C. R. Myers, and J. P. Sethna. Universally sloppy parameter sensitivities in systems biology models. *PLoS Computational Biology*, 3(10):e189, 2007.

[12] A. J. G. D. Holtmaat, J. T. Trachtenberg, L. Wilbrecht, G. M. Shepherd, X. Zhang, G. W. Knott, and K. Svoboda. Transient and persistent dendritic spines in the neocortex in vivo. *Neuron*, 45(2):279–91, jan 2005.

[13] J. J. Hopfield. Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the National Academy of Sciences*, 79(8):2554–2558, apr 1982.

[14] J. J. Hopfield, D. Feinstein, and R. Palmer. fiunlearningfihas a stabilizing effect in collective memories. *Nature*, 304(5922):158, 1983.

[15] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, et al. Overcoming catastrophic forgetting in neural networks. *Proceedings of the National Academy of Sciences*, 114(13):3521–3526, 2017.

[16] Z. Li and D. Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[17] X. Liu, M. Masana, L. Herranz, J. Van de Weijer, A. M. Lopez, and A. D. Bagdanov. Rotate your networks: Better weight consolidation and less catastrophic forgetting. *arXiv preprint arXiv:1802.02950*, 2018.

[18] Y. Loewenstein, U. Yanover, and S. Rumpel. Predicting the Dynamics of Network Connectivity in the Neocortex. *Journal of Neuroscience*, 35(36):12535–12544, 2015.

[19] M. McCloskey and N. J. Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of Learning and Motivation*, volume 24, pages 109–165. Elsevier, 1989.

[20] J. Nadal, G. Toulouse, J. Changeux, and S. Dehaene. Networks of formal neurons and memory palimpsests. *Europhysics Letters*, 1(10):535, 1986.

[21] D. Panas, H. Amin, A. Maccione, O. Muthmann, M. van Rossum, L. Berdondini, and M. H. Hennig. Sloppiness in spontaneously active neuronal networks. *Journal of Neuroscience*, 35(22):8480–8492, 2015.

[22] B. Poole, F. Zenke, and S. Ganguli. Intelligent synapses for multi-task and transfer learning. In *International Conference on Learning Representations*, 2017.

[23] A. Rannen Ep Triki, R. Aljundi, M. Blaschko, and T. Tuytelaars. Encoder based lifelong learning. In *Proceedings ICCV 2017*, pages 1320–1328, 2017.

[24] A. Robins and S. McCALLUM. Catastrophic forgetting and the pseudorehearsal solution in hopfield-type networks. *Connection Science*, 10(2):121–135, 1998.

[25] T. Rogerson, D. J. Cai, A. Frank, Y. Sano, J. Shobe, M. F. Lopez-Aranda, and A. J. Silva. Synaptic tagging during memory allocation. *Nature Reviews Neuroscience*, 15(3):157, 2014.

[26] A. Rosenfeld and J. K. Tsotsos. Incremental learning through deep adaptation. *arXiv preprint arXiv:1705.04228*, 2017.

[27] J. Serrà, D. Surís, M. Miron, and A. Karatzoglou. Overcoming catastrophic forgetting with hard attention to the task. *arXiv preprint arXiv:1801.01423*, 2018.

[28] S. F. Sorrells, M. F. Paredes, A. Cebrian-Silla, K. Sandoval, D. Qi, K. W. Kelley, D. James, S. Mayer, J. Chang, K. I. Auguste, et al. Human hippocampal neurogenesis drops sharply in children to undetectable levels in adults. *Nature*, 555(7696):377, 2018.

[29] R. Stickgold. Sleep-dependent memory consolidation. *Nature*, 437(7063):1272, 2005.

[30] M. Tsodyks and M. Feigel'Man. The enhanced storage capacity in neural networks with low activity level. *EPL (Europhysics Letters)*, 6(2):101, 1988.

[31] M. A. Wilson and B. L. McNaughton. Reactivation of hippocampal ensemble memories during sleep. *Science*, 265(5172):676–679, 1994.

[32] F. Zenke, B. Poole, and S. Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, pages 3987–3995, 2017.

4.1 Appendix: non-zero Fisher eigenvalues in a Hopfield network

Because the assumptions of the paper are based on the sloppiness of the parameter space of a Hopfield network, in this section I will briefly illustrate the properties of the Fisher information tensor of a trained Hopfield net.

Consider a Hopfield network with M nodes, in which N patterns $\boldsymbol{\xi}^1, ..., \boldsymbol{\xi}^N$ have been stored by defining the weight matrix as $w_{ij} = \frac{1}{N}\sum_{n=1}^{N} \xi_i^n \xi_j^n$. As shown in the paper, we can compute the Fisher information as a covariance:

$$F_{w_{ij},w_{kl}} = \frac{1}{N}\sum_{n=1}^{N} \xi_i^n \xi_j^n \xi_k^n \xi_l^n - \frac{1}{N^2}\sum_{n=1}^{N} \xi_i^n \xi_j^n \sum_{n'=1}^{N} \xi_k^{n'} \xi_l^{n'}$$

Let us define $\rho_a^n = \xi_i^n \xi_j^n$, where the index $a = (i, j)$ spans all $\frac{1}{2}M(M-1)$ combinations with $i < j$. Then the above becomes

$$F_{ab} = \frac{1}{N}\sum_{n=1}^{N} \rho_a^n \rho_b^n - \frac{1}{N^2}\sum_{n=1}^{N} \rho_a^n \sum_{n'=1}^{N} \rho_b^{n'}.$$

Now, because the patterns $\boldsymbol{\xi}^1, ..., \boldsymbol{\xi}^N$ are binary and are different, they must also be linearly independent; as a consequence, the same holds for $\boldsymbol{\rho}^1, ..., \boldsymbol{\rho}^N$. Because we are interested in eigenvalues, which are invariant under a change of basis, we can pick a basis in which $\rho_a^n = \delta_a^n$ (where δ is a Kronecker delta). Note that this defines only the first N of the $\frac{1}{2}M(M-1)$ basis vectors, the others remaining arbitrary. Then we have

$$F_{ab} = \begin{cases} 0 & \text{if } a > N \text{ or } b > N \\ \frac{1}{N}\delta_{ab} - \frac{1}{N^2} & \text{otherwise} \end{cases} \tag{4.1}$$

this matrix has rank N, and therefore N non-zero eigenvalues. However, we note that an infinitesimal change that scales all weights up or down uniformly does not affect the Hopfield dynamics. This reduces the number of relevant degrees of freedom by one, leading to $N-1$. Figure 4.1 shows numerical confirmation of this result. Moreover, for large N, the non-zero submatrix of the FIT as expressed in (4.1) tends to $1/N$ times the identity. This proves its eigenvalues decrease linearly in N, and this too can be seen in Figure 4.1.

In conclusion, there are only $N-1$ "stiff" parameters when N patterns have been learned, and all the other parameters are "sloppy". While an abundance

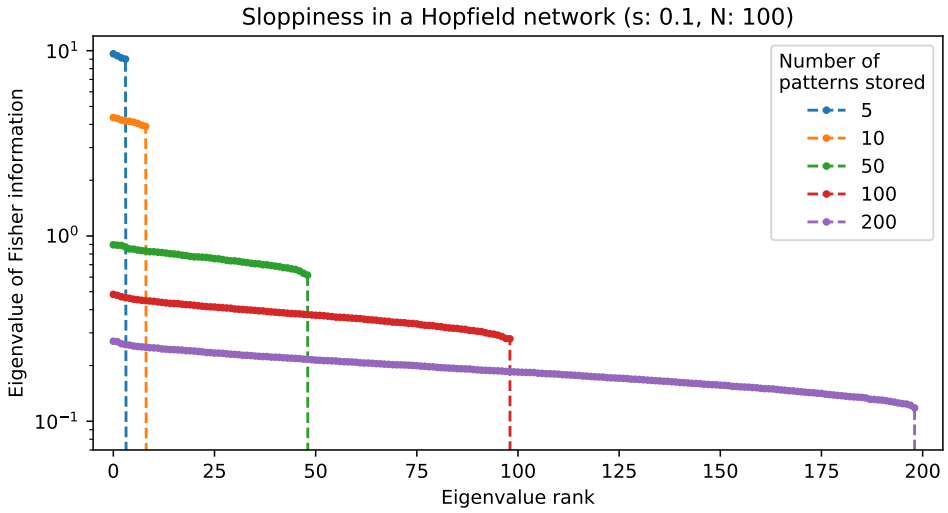


Figure 4.1: Eigenvalues of the Fisher Information for a network with sparsity $s = 0.1$ and $M = 100$ nodes, after learning N patterns as indicated. Note that the complete number of eigenvalues is $M(M-1) = 4950$, equal to the number of independent parameters of the network. All the ones that are not shown are numerically zero. The number of non-zero eigenvalues is empirically found to be always equal to $N - 1$.

of sloppy parameters does not guarantee more storage space is available for new memories (indeed, sloppy dimensions still exist when capacity is reached), delaying training on stiff dimensions can help retaining memories for longer, when this is possible, as shown in the paper.

Chapter 5

Discussion

> It is fair to say that, in general,
> no problems have been exhausted;
> instead, men have been exhausted
> by the problems.
>
> [Ramón y Cajal, 1897]

The availability of simultaneous recordings of thousands of neurons has the potential of greatly improving our understanding of how neuronal populations compute and encode information. The increasingly large scale of datasets, however, correspondingly requires research in computational methods at all levels — from the analysis of raw data (signal processing, spike detection and sorting), to the modelling of large populations, and to new theoretical frameworks. The project presented in this thesis stemmed from this problem, considering it both from the data analysis point of view, by developing a spike sorting method for large arrays, and the modelling point of view, using restricted Boltzmann machines to understand population-level data through low-dimensional representations. Although statistical models of neural activity were the main focus, the ideas I pursued were diverse and of increasing theoretical nature, starting on the data processing side, in the first chapter, and concluding with an entirely model-based research question in the last. Nevertheless, they were linked by a continuous line of research that developed naturally from a problem to the next, which can be summarised as follows.

The first chapter started with the problem of spike sorting on dense multi-electrode arrays, where the large amount of data generated by the chip requires

scalability, and the mixing of signals due to the close spacing of the electrodes requires smarter sorting techniques, which can deal with the signals from multiple electrodes at once. Following [Muthmann et al., 2015], and in collaboration with experimentalists, we proposed a sorting method that proved to be fast and reliable in many applications, and is now being used by several laboratories that adopted 3Brain's BioCam chips. Our sorting library has been tried for other MEA probes too: the new NeuroSeekers [Fiáth et al., 2018], the MEA1K from ETH [Ballini et al., 2014], and the Neuropixel probes [Jun et al., 2017b]. The constant development of denser and larger arrays will always be a source of future challenges. Our solution was not the only one that emerged in the last few years, which demonstrates how important this question was at this time. While we were working on this project, several other research groups faced the same issues and proposed their own solutions, which differ in some aspects. I have already discussed them in the introduction and in table 1; [Hennig et al., 2018] provided a more in-depth review and discussion of future perspectives. A weakness of our strategy, compared to the template-based approach, may be found in the detection step, where the resolution of temporally overlapping spikes is not guaranteed. On the other hand, learning templates requires reading the whole dataset twice, which is more computationally demanding. Soon, we may see a heavier application of contemporary machine learning research to this problem: recent alternative approaches involve the use of neural networks in the detection step [Lee et al., 2017]; my colleagues at the university of Edinburgh have recently experimented with the use of deep autoencoders, for non-linear dimensionality reduction (beyond the limitations of PCA) and more accurate spike localisation. Another direction of development should consist of comparing the different algorithms and choosing the best ones for different situations, or perhaps taking the strongest ideas from each, with the aim of creating a standard. The field would also benefit from better harmonisation on the practical level: more uniform interfaces, file formats, language, and evaluation strategies would be very valuable for an efficient way forward. For the evaluation of new and old sorting strategies, a framework was provided by the publication of the ViSAPy simulator, which generates synthetic electrode traces with realistic noise: this is an excellent tool that sidesteps the fundamental problem of the lack of ground truth data [Hagen et al., 2015]. All future research that uses MEAs can greatly benefit from the existence of a method for detection and sorting that is used and recognised as reliable by

the whole scientific community. With larger and larger datasets available, speed and scalability will be essential properties of future spike sorting methods.

This first part was necessary in order to provide reliable data for analysis in the following chapter. Here, I started addressing the main area of interest that is described in the title of this thesis: statistical modelling of neural activity. In particular, following the lead of [Köster et al., 2014, Spicher, 2014], I tried to investigate what restricted Boltzmann machines can tell us about a population of spiking neurons — in this case, the retinal cells of a mouse, recorded with an MEA and sorted as mentioned above. When trained on retinal activity, the RBM's hidden units readily couple with specific types of ganglion cells: sustained or transient, ON or OFF, or grouped by location. This use of RBMs for factor analysis looks promising as a way to identify, in a way that fully visible Boltzmann machines are unable to do, assemblies of correlated neurons, and evaluate their functional properties, such as in [See et al., 2018]. More generally, RBMs seem suitable for understanding the collective patterns that naturally arise in the activity of many neurons recorded simultaneously; the area of research that studies low-dimensional representations of neuronal firing has recently gained attention [Gao et al., 2017]. Using a very simple model of binary neurons that are driven by a few latent variables, I showed that RBMs can be a useful tool for this type of analysis. Although this is an interesting proof of concept, further research is needed: first, using more realistic simulations, then using real data, possibly comparing the activity of hidden units with functionally relevant metrics, such as stimuli or behaviour. Adding time components is also straightforward to implement, and has the potential of providing the model with much more information. Analogously, the model should be extended beyond binary mode activation, into richer dynamics. Further research in this area does not need to be limited to RBMs. As I have noted, non-negative matrix factorisation also seems a promising method, and, more generally, modern machine learning techniques may have a lot to offer; since recordings of thousands of neurons are now possible, their computational efficiency should also be considered.

Others have observed, using Ising-like models, that learned statistical models always seem to be poised in the vicinity of their critical point after training is completed [Tkačik et al., 2015]. I developed my work on Boltzmann machines in this direction: I found a suitable definition of specific heat for RBMs and reproduced the same observation, showing that they diverge in correspondence

of the temperature at which the data are fitted. This seems to happen consistently, despite the differences in the datasets used for the analysis. I argued, with [Mastromatteo and Marsili, 2011], that this is simply due to a property of the parameter space, where, in a precise mathematical sense, the density of meaningful models is much higher near the critical surface. Zipf laws, on the other hand, are a property that can be linked back to criticality, but is model-independent. I showed that Zipf laws in retinal recordings, already noted in the literature, hold for retinas subjected to stimuli with different spatial statistics, and under pharmacological treatment. I also provide preliminary evidence that, at least for finite sample size, a Zipf-like profile can appear even in the absence of correlations, simply as a consequence of a long-tailed firing rate distribution. Finally, I discussed the possible relationship between criticality in the "statistical" sense mentioned above, and in the "dynamical" sense of avalanche dynamics: checking both model specific heat and Zipf laws of the data, I showed that a relationship need not exist. However, I would argue much more theoretical insight is needed to make formal claims in this field. In general, we don't know if it is possible to obtain a "thermodynamic" theory (in the sense introduced by [Tkačik et al., 2015]) of spiking neural networks, which, like in statistical physics, would ignore microscopic detail and capture network properties on a higher level. The reasons behind Zipf laws in neural activity are still unclear, and so is their relationship with function and encoding; however, studying this phenomenon in the context of artificial neural networks is revealing new insights [Song et al., 2017, Cubero et al., 2018].

In relation to the work on statistical criticality, I also talked about model sloppiness: the observation that the vast majority of dimensions in the parameter space of most models actually leads to no changes in the overall fitness of the model. This translates to many eigenvalues of the model's Fisher information tensor being close to zero. In the final chapter, we introduced a definition of Fisher information for Hopfield networks, a classic model of memory retention, which is analogous to a zero-temperature version of the statistical models covered in previous chapters. In appendix 4.1, I show a particular kind of sloppiness applies to these networks. We therefore asked whether this can be exploited to prevent overwriting previously learned patterns, when the network is used to learn sequentially, by prioritising updates of sloppier parameters. The answer is positive: a Hebbian rule that accounts for parameter importance estimated

through Fisher information guarantees previous memories are stored for a longer time while learning new ones. Additionally, this learning rule has the desirable property of being local, in the sense that no information about the rest of the network is required when updating a single weight. The next step will be continuing this work on more sophisticated networks, maintaining a balance between functionality and biological realism. Our finding that the Fisher curvature corresponding to a synaptic weight can be computed, in certain cases, as a function of the weight itself is promising; however, more work is needed in order to show whether this can hold for spiking networks.

The thesis you have just finished reading represents my personal path of learning not only in neuroscience, but also in machine learning; it is my take on a small number of examples in which the two disciplines interact, sometimes in very different ways. Despite some shared aspects, research in artificial neural networks went largely its own way, inspired by applications. However, machine learning models are still useful in neuroscience in two different ways: as a tool for data analysis, and as a test bed for understanding common aspects of artificial and animal intelligence. The first chapter of this thesis could be seen as implementing a form of unsupervised learning for the analysis of raw data, and I believe the near future will see a heavier use of contemporary machine learning in this field. On a very different level, the second chapter also uses a classic unsupervised generative model, this time in understanding primitive properties of neural activity, following the property, that has gained attention quite recently, of the low dimensionality of neural activity. The third chapter asks whether statistical criticality should be considered a property of the model or a property of the neurons' activity. Moreover, the question of the functional relevance of Zipf laws, here discussed for neuronal populations, has now also been posed for deep learning models [Song et al., 2017]. The last chapter is an attempt at studying a problem (catastrophic forgetting) and a mathematical construct (the Fisher metric), which were originally developed for machine learning, to a Hopfield network, a classic model of old-school computational neuroscience. With the advent of deep learning, there has been a new revival of the interaction between machine learning and neuroscience. Before the two can profit from each other, large gaps between them will need to be filled, especially regarding the aspect of learning and plasticity — and it seems this is opening new, exciting opportunities.

Bibliography

[Ackley et al., 1985] Ackley, D. H., Hinton, G. E., and Sejnowski, T. J. (1985). A learning algorithm for Boltzmann machines. *Cognitive science*, 9(1):147–169.

[Amari, 1998] Amari, S.-I. (1998). Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276.

[Amunts et al., 2016] Amunts, K., Ebell, C., Muller, J., Telefont, M., Knoll, A., and Lippert, T. (2016). The human brain project: creating a european research infrastructure to decode the human brain. *Neuron*, 92(3):574–581.

[Baden et al., 2016] Baden, T., Berens, P., Franke, K., Rosón, M. R., Bethge, M., and Euler, T. (2016). The functional diversity of retinal ganglion cells in the mouse. *Nature*, 529(7586):345.

[Ballini et al., 2014] Ballini, M., Müller, J., Livi, P., Chen, Y., Frey, U., Stettler, A., Shadmani, A., Viswam, V., Jones, I. L., Jäckel, D., et al. (2014). A 1024-channel cmos microelectrode array with 26,400 electrodes for recording and stimulation of electrogenic cells in vitro. *IEEE journal of solid-state circuits*, 49(11):2705–2719.

[Barlow, 2001] Barlow, H. (2001). Redundancy reduction revisited. *Network: computation in neural systems*, 12(3):241–253.

[Barrett et al., 2019] Barrett, D. G., Morcos, A. S., and Macke, J. H. (2019). Analyzing biological and artificial neural networks: challenges with opportunities for synergy? *Current opinion in neurobiology*, 55:55–64.

[Beggs and Plenz, 2003] Beggs, J. M. and Plenz, D. (2003). Neuronal avalanches in neocortical circuits. *Journal of neuroscience*, 23(35):11167–11177.

[Beggs and Timme, 2012] Beggs, J. M. and Timme, N. (2012). Being critical of criticality in the brain. *Frontiers in physiology*, 3:163.

[Bengio et al., 2009] Bengio, Y. et al. (2009). Learning deep architectures for AI. *Foundations and trends in Machine Learning*, 2(1):1–127.

[Berdondini et al., 2009] Berdondini, L., Imfeld, K., Maccione, A., Tedesco, M., Neukom, S., Koudelka-Hep, M., and Martinoia, S. (2009). Active pixel sensor array for high spatio-temporal resolution electrophysiological recordings from single cell to large scale neuronal networks. *Lab on a Chip*, 9(18):2644–2651.

[Berry and Meister, 1998] Berry, M. J. and Meister, M. (1998). Refractoriness and neural precision. In *Advances in Neural Information Processing Systems*, pages 110–116.

[Bestel et al., 2012] Bestel, R., Daus, A. W., and Thielemann, C. (2012). A novel automated spike sorting algorithm with adaptable feature extraction. *Journal of neuroscience methods*, 211(1):168–178.

[Byron et al., 2009] Byron, M. Y., Cunningham, J. P., Santhanam, G., Ryu, S. I., Shenoy, K. V., and Sahani, M. (2009). Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. In *Advances in neural information processing systems*, pages 1881–1888.

[Carandini, 2012] Carandini, M. (2012). Area v1. *Scholarpedia*, 7(7):12105.

[Carcieri et al., 2003] Carcieri, S. M., Jacobs, A. L., and Nirenberg, S. (2003). Classification of retinal ganglion cells: a statistical approach. *Journal of neurophysiology*, 90(3):1704–1713.

[Cheung, 2007] Cheung, K. C. (2007). Implantable microscale neural interfaces. *Biomedical microdevices*, 9(6):923–938.

[Chichilnisky, 2001] Chichilnisky, E. (2001). A simple white noise analysis of neuronal light responses. *Network: Computation in Neural Systems*, 12(2):199–213.

[Chung et al., 2017] Chung, J. E., Magland, J. F., Barnett, A. H., Tolosa, V. M., Tooker, A. C., Lee, K. Y., Shah, K. G., Felix, S. H., Frank, L. M., and Greengard, L. F. (2017). A fully automated approach to spike sorting. *Neuron*, 95(6):1381–1394.

[Clevert et al., 2015] Clevert, D.-A., Mayr, A., Unterthiner, T., and Hochreiter, S. (2015). Rectified factor networks. In *Advances in neural information processing systems*, pages 1855–1863.

[Comaniciu and Meer, 2002] Comaniciu, D. and Meer, P. (2002). Mean shift: A robust approach toward feature space analysis. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24(5):603–619.

[Cubero et al., 2018] Cubero, R. J., Jo, J., Marsili, M., Roudi, Y., and Song, J. (2018). Minimally sufficient representations, maximally informative samples and zipf's law. *arXiv preprint arXiv:1808.00249*.

[Dayan and Abbott, 2001] Dayan, P. and Abbott, L. F. (2001). *Theoretical neuroscience*. Cambridge, MA: MIT Press.

[De Schutter, 2000] De Schutter, E. (2000). *Computational neuroscience: realistic modeling for experimentalists*. CRC press.

[Deistler et al., 2018] Deistler, M., Sorbaro, M., Rule, M. E., and Hennig, M. H. (2018). Local learning rules to attenuate forgetting in neural networks. *arXiv preprint arXiv:1807.05097*.

[Desimone et al., 1984] Desimone, R., Albright, T. D., Gross, C. G., and Bruce, C. (1984). Stimulus-selective properties of inferior temporal neurons in the macaque. *Journal of Neuroscience*, 4(8):2051–2062.

[Ester et al., 1996] Ester, M., Kriegel, H.-P., Sander, J., Xu, X., et al. (1996). A density-based algorithm for discovering clusters in large spatial databases with noise. In *Kdd*, volume 96, pages 226–231.

[Fiáth et al., 2018] Fiáth, R., Raducanu, B. C., Musa, S., Andrei, A., Lopez, C. M., van Hoof, C., Ruther, P., Aarts, A., Horváth, D., and Ulbert, I. (2018). A silicon-based neural probe with densely-packed low-impedance titanium nitride microelectrodes for ultrahigh-resolution in vivo recordings. *Biosensors and Bioelectronics*, 106:86–92.

[Frank et al., 2000] Frank, L. M., Brown, E. N., and Wilson, M. (2000). Trajectory encoding in the hippocampus and entorhinal cortex. *Neuron*, 27(1):169–178.

[Gallego et al., 2017] Gallego, J. A., Perich, M. G., Miller, L. E., and Solla, S. A. (2017). Neural manifolds for the control of movement. *Neuron*, 94(5):978–984.

[Gallego et al., 2018] Gallego, J. A., Perich, M. G., Naufel, S. N., Ethier, C., Solla, S. A., and Miller, L. E. (2018). Cortical population activity within a preserved neural manifold underlies multiple motor behaviors. *Nature communications*, 9(1):4233.

[Gao et al., 2017] Gao, P., Trautmann, E., Byron, M. Y., Santhanam, G., Ryu, S., Shenoy, K., and Ganguli, S. (2017). A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, page 214262.

[Gardella et al., 2018] Gardella, C., Marre, O., and Mora, T. (2018). Blindfold learning of an accurate neural metric. *Proceedings of the National Academy of Sciences*, page 201718710.

[Gollisch and Meister, 2010] Gollisch, T. and Meister, M. (2010). Eye smarter than scientists believed: neural computations in circuits of the retina. *Neuron*, 65(2):150–164.

[Gross et al., 1977] Gross, G., Rieske, E., Kreutzberg, G., and Meyer, A. (1977). A new fixed-array multi-microelectrode system designed for long-term monitoring of extracellular single unit neuronal activity in vitro. *Neuroscience Letters*, 6(2-3):101–105.

[Gutenkunst et al., 2007] Gutenkunst, R. N., Waterfall, J. J., Casey, F. P., Brown, K. S., Myers, C. R., and Sethna, J. P. (2007). Universally sloppy parameter sensitivities in systems biology models. *PLoS Comput Biol*, 3(10):e189.

[Hagen et al., 2015] Hagen, E., Ness, T. V., Khosrowshahi, A., Sørensen, C., Fyhn, M., Hafting, T., Franke, F., and Einevoll, G. T. (2015). ViSAPy: A python tool for biophysics-based generation of virtual spiking activity for evaluation of spike-sorting algorithms. *Journal of neuroscience methods*, 245:182–204.

[Hartline, 1938] Hartline, H. K. (1938). The response of single optic nerve fibers of the vertebrate eye to illumination of the retina. *American Journal of Physiology-Legacy Content*, 121(2):400–415.

[Hassabis et al., 2017] Hassabis, D., Kumaran, D., Summerfield, C., and Botvinick, M. (2017). Neuroscience-inspired artificial intelligence. *Neuron*, 95(2):245–258.

[Hennig et al., 2018] Hennig, M. H., Hurwitz, C., and Sorbaro, M. (2018). Scaling spike detection and sorting for next generation electrophysiology. *arXiv preprint arXiv:1809.01051*.

[Hilgen et al., 2017a] Hilgen, G., Pirmoradian, S., Pamplona, D., Kornprobst, P., Cessac, B., Hennig, M. H., and Sernagor, E. (2017a). Pan-retinal characterisation of light responses from ganglion cells in the developing mouse retina. *Scientific reports*, 7:42330.

[Hilgen et al., 2017b] Hilgen, G., Sorbaro, M., Pirmoradian, S., Muthmann, J.-O., Kepiro, I. E., Ullo, S., Ramirez, C. J., Encinas, A. P., Maccione, A., Berdondini, L., et al. (2017b). Unsupervised spike sorting for large-scale, high-density multielectrode arrays. *Cell reports*, 18(10):2521–2532.

[Hinton, 2002] Hinton, G. E. (2002). Training products of experts by minimizing contrastive divergence. *Neural computation*, 14(8):1771–1800.

[Hodgkin and Huxley, 1952] Hodgkin, A. L. and Huxley, A. F. (1952). A quantitative description of membrane current and its application to conduction and excitation in nerve. *The Journal of physiology*, 117(4):500–544.

[Hopfield, 1982] Hopfield, J. J. (1982). Neural networks and physical systems with emergent collective computational abilities. *Proceedings of the national academy of sciences*, 79(8):2554–2558.

[Hyvärinen and Oja, 2000] Hyvärinen, A. and Oja, E. (2000). Independent component analysis: algorithms and applications. *Neural networks*, 13(4-5):411–430.

[Jaynes, 1957] Jaynes, E. T. (1957). Information theory and statistical mechanics. *Physical review*, 106(4):620.

[Jun et al., 2017a] Jun, J. J., Mitelut, C., Lai, C., Gratiy, S., Anastassiou, C., and Harris, T. D. (2017a). Real-time spike sorting platform for high-density extracellular probes with ground-truth validation and drift correction. *bioRxiv*, page 101030.

[Jun et al., 2017b] Jun, J. J., Steinmetz, N. A., Siegle, J. H., Denman, D. J., Bauza, M., Barbarits, B., Lee, A. K., Anastassiou, C. A., Andrei, A., Aydın, Ç., et al. (2017b). Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679):232.

[Katsageorgiou et al., 2018] Katsageorgiou, V.-M., Sona, D., Zanotto, M., Lassi, G., Garcia-Garcia, C., Tucci, V., and Murino, V. (2018). A novel unsupervised analysis of electrophysiological signals reveals new sleep substages in mice. *PLoS biology*, 16(5):e2003663.

[Keat et al., 2001] Keat, J., Reinagel, P., Reid, R. C., and Meister, M. (2001). Predicting every spike: a model for the responses of visual neurons. *Neuron*, 30(3):803–817.

[Kingma and Ba, 2014] Kingma, D. P. and Ba, J. (2014). Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

[Kirkpatrick et al., 2017] Kirkpatrick, J., Pascanu, R., Rabinowitz, N., Veness, J., Desjardins, G., Rusu, A. A., Milan, K., Quan, J., Ramalho, T., Grabska-Barwinska, A., et al. (2017). Overcoming catastrophic forgetting in neural networks. *Proceedings of the national academy of sciences*, page 201611835.

[Klier et al., 2001] Klier, E. M., Wang, H., and Crawford, J. D. (2001). The superior colliculus encodes gaze commands in retinal coordinates. *Nature neuroscience*, 4(6):627.

[Kobak et al., 2016] 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:e10989.

[Köster et al., 2014] Köster, U., Sohl-Dickstein, J., Gray, C. M., and Olshausen, B. A. (2014). Modeling higher-order correlations within cortical microcolumns. *PLoS computational biology*, 10(7):e1003684.

[Laughlin, 1981] Laughlin, S. (1981). A simple coding procedure enhances a neuron's information capacity. *Zeitschrift für Naturforschung c*, 36(9-10):910–912.

[LeCun et al., 2015] LeCun, Y., Bengio, Y., and Hinton, G. (2015). Deep learning. *nature*, 521(7553):436.

[Lee and Seung, 1999] Lee, D. D. and Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788.

[Lee et al., 2017] Lee, J. H., Carlson, D. E., Razaghi, H. S., Yao, W., Goetz, G. A., Hagen, E., Batty, E., Chichilnisky, E., Einevoll, G. T., and Paninski, L. (2017). YASS: Yet Another Spike Sorter. In *Advances in Neural Information Processing Systems*, pages 4005–4015.

[Lettvin et al., 1959] Lettvin, J. Y., Maturana, H. R., McCulloch, W. S., and Pitts, W. H. (1959). What the frog's eye tells the frog's brain. *Proceedings of the IRE*, 47(11):1940–1951.

[Little, 1974] Little, W. A. (1974). The existence of persistent states in the brain. In *From High-Temperature Superconductivity to Microminiature Refrigeration*, pages 145–164. Springer.

[Maccione et al., 2014] Maccione, A., Hennig, M. H., Gandolfo, M., Muthmann, O., van Coppenhagen, J., Eglen, S. J., Berdondini, L., and Sernagor, E. (2014). Following the ontogeny of retinal waves: pan-retinal recordings of population dynamics in the neonatal mouse. *The Journal of physiology*, 592(7):1545–1563.

[Machta et al., 2013] Machta, B. B., Chachra, R., Transtrum, M. K., and Sethna, J. P. (2013). Parameter space compression underlies emergent theories and predictive models. *Science*, 342(6158):604–607.

[Macke et al., 2009] Macke, J. H., Berens, P., Ecker, A. S., Tolias, A. S., and Bethge, M. (2009). Generating spike trains with specified correlation coefficients. *Neural computation*, 21(2):397–423.

[Marblestone et al., 2016] Marblestone, A. H., Wayne, G., and Kording, K. P. (2016). Toward an integration of deep learning and neuroscience. *Frontiers in computational neuroscience*, 10:94.

[Markoff and Gorman, 2013] Markoff, J. and Gorman, J. (2013). Obama to unveil initiative to map the human brain. *The New York Times*, 2.

[Marr and Poggio, 1976] Marr, D. and Poggio, T. (1976). From understanding computation to understanding neural circuitry.

[Marre et al., 2009] Marre, O., El Boustani, S., Frégnac, Y., and Destexhe, A. (2009). Prediction of spatiotemporal patterns of neural activity from pairwise correlations. *Physical review letters*, 102(13):138101.

[Martignon et al., 2000] Martignon, L., Deco, G., Laskey, K., Diamond, M., Freiwald, W., and Vaadia, E. (2000). Neural coding: higher-order temporal patterns in the neurostatistics of cell assemblies. *Neural Computation*, 12(11):2621–2653.

[Masland, 2012] Masland, R. H. (2012). The neuronal organization of the retina. *Neuron*, 76(2):266–280.

[Mastromatteo and Marsili, 2011] Mastromatteo, I. and Marsili, M. (2011). On the criticality of inferred models. *J. of Statistical Mechanics: Theory and Experiment*, 2011(10):P10012.

[McCullagh, 1983] McCullagh, P. (1983). *Generalized linear models*. Routledge.

[McCulloch and Pitts, 1943] McCulloch, W. S. and Pitts, W. (1943). A logical calculus of the ideas immanent in nervous activity. *The bulletin of mathematical biophysics*, 5(4):115–133.

[McInnes et al., 2017] McInnes, L., Healy, J., and Astels, S. (2017). hdbscan: Hierarchical density based clustering. *The Journal of Open Source Software*, 2(11):205.

[Miller, 2018] Miller, P. (2018). *An Introductory Course in Computational Neuroscience*. MIT Press.

[Molano-Mazon et al., 2018] Molano-Mazon, M., Onken, A., Piasini, E., and Panzeri, S. (2018). Synthesizing realistic neural population activity patterns using generative adversarial networks. *arXiv preprint arXiv:1803.00338*.

[Mora et al., 2015] Mora, T., Deny, S., and Marre, O. (2015). Dynamical criticality in the collective activity of a population of retinal neurons. *Physical review letters*, 114(7):078105.

[Muthmann et al., 2015] Muthmann, J.-O., Amin, H., Sernagor, E., Maccione, A., Panas, D., Berdondini, L., Bhalla, U. S., and Hennig, M. H. (2015). Spike detection for large neural populations using high density multielectrode arrays. *Frontiers in neuroinformatics*, 9.

[Nasser et al., 2013] Nasser, H., Marre, O., and Cessac, B. (2013). Spatio-temporal spike train analysis for large scale networks using the maximum entropy principle and Monte Carlo method. *Journal of Statistical Mechanics: Theory and Experiment*, 2013(03):P03006.

[Neher and Sakmann, 1976] Neher, E. and Sakmann, B. (1976). Single-channel currents recorded from membrane of denervated frog muscle fibres. *Nature*, 260(5554):799.

[Nelder and Wedderburn, 1972] Nelder, J. A. and Wedderburn, R. W. (1972). Generalized linear models. *Journal of the Royal Statistical Society: Series A (General)*, 135(3):370–384.

[Nirenberg and Latham, 2003] Nirenberg, S. and Latham, P. E. (2003). Decoding neuronal spike trains: how important are correlations? *Proceedings of the National Academy of Sciences*, 100(12):7348–7353.

[O'Donnell et al., 2016] O'Donnell, C., Gonçalves, J. T., Whiteley, N., Portera-Cailliau, C., and Sejnowski, T. J. (2016). The population tracking model: A simple, scalable statistical model for neural population data. *Neural computation*.

[Onken et al., 2016] Onken, A., Liu, J. K., Karunasekara, P. C. R., Delis, I., Gollisch, T., and Panzeri, S. (2016). Using matrix and tensor factorizations for the single-trial analysis of population spike trains. *PLoS computational biology*, 12(11):e1005189.

[Pachitariu et al., 2016] Pachitariu, M., Steinmetz, N. A., Kadir, S. N., Carandini, M., and Harris, K. D. (2016). Fast and accurate spike sorting of high-channel count probes with KiloSort. In *Advances in Neural Information Processing Systems*, pages 4448–4456.

[Panas et al., 2015] Panas, D., Amin, H., Maccione, A., Muthmann, O., van Rossum, M., Berdondini, L., and Hennig, M. H. (2015). Sloppiness in spontaneously active neuronal networks. *The Journal of Neuroscience*, 35(22):8480–8492.

[Paninski, 2004] Paninski, L. (2004). Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262.

[Pedregosa et al., 2011] Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., and Duchesnay, E. (2011). Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830.

[Pillow et al., 2008] Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E., and Simoncelli, E. P. (2008). Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999.

[Pine, 2006] Pine, J. (2006). A history of MEA development. In *Advances in network electrophysiology*, pages 3–23. Springer.

[Posner and Gilbert, 1999] Posner, M. I. and Gilbert, C. D. (1999). Attention and primary visual cortex. *Proceedings of the National Academy of Sciences*, 96(6):2585–2587.

[Quiroga, 2012] Quiroga, R. Q. (2012). Concept cells: the building blocks of declarative memory functions. *Nature Reviews Neuroscience*, 13(8):587.

[Quiroga et al., 2005] Quiroga, R. Q., Reddy, L., Kreiman, G., Koch, C., and Fried, I. (2005). Invariant visual representation by single neurons in the human brain. *Nature*, 435(7045):1102.

[Ramón y Cajal, 1897] Ramón y Cajal, S. (1897). *Advice for a young investigator*. MIT Press.

[Ramón y Cajal, 1904] Ramón y Cajal, S. (1904). *Textura del Sistema Nervioso del Hombre y de los Vertebrados*, volume 2. Madrid Nicolas Moya.

[Repérant et al., 2006] Repérant, J., Ward, R., Miceli, D., Rio, J., Medina, M., Kenigfest, N., and Vesselkin, N. (2006). The centrifugal visual system of vertebrates: a comparative analysis of its functional anatomical organization. *Brain research reviews*, 52(1):1–57.

[Rey et al., 2015] Rey, H. G., Pedreira, C., and Quiroga, R. Q. (2015). Past, present and future of spike sorting techniques. *Brain research bulletin*, 119:106–117.

[Roelfsema and Holtmaat, 2018] Roelfsema, P. R. and Holtmaat, A. (2018). Control of synaptic plasticity in deep cortical networks. *Nature Reviews Neuroscience*, 19(3):166.

[Rosenblatt, 1958] Rosenblatt, F. (1958). The perceptron: a probabilistic model for information storage and organization in the brain. *Psychological review*, 65(6):386.

[Rossant et al., 2016] Rossant, C., Kadir, S. N., Goodman, D. F., Schulman, J., Hunter, M. L., Saleem, A. B., Grosmark, A., Belluscio, M., Denfield, G. H., Ecker, A. S., et al. (2016). Spike sorting for large, dense electrode arrays. *Nature neuroscience*, 19(4):634.

[Roudi et al., 2009] Roudi, Y., Nirenberg, S., and Latham, P. E. (2009). Pairwise maximum entropy models for studying large biological systems: when they can work and when they can't. *PLoS computational biology*, 5(5):e1000380.

[Sadtler et al., 2014] Sadtler, P. T., Quick, K. M., Golub, M. D., Chase, S. M., Ryu, S. I., Tyler-Kabara, E. C., Byron, M. Y., and Batista, A. P. (2014). Neural constraints on learning. *Nature*, 512(7515):423.

[Schneidman et al., 2006] Schneidman, E., Berry, M. J., Segev, R., and Bialek, W. (2006). Weak pairwise correlations imply strongly correlated network states in a neural population. *Nature*, 440(7087):1007–1012.

[Schneidman et al., 2003] Schneidman, E., Still, S., Berry, M. J., Bialek, W., et al. (2003). Network information and connected correlations. *Physical review letters*, 91(23):238701.

[See et al., 2018] See, J. Z., Atencio, C. A., Sohal, V. S., and Schreiner, C. E. (2018). Coordinated neuronal ensembles in primary auditory cortical columns. *eLife*, 7:e35587.

[Sejnowski et al., 1988] Sejnowski, T. J., Koch, C., and Churchland, P. S. (1988). Computational neuroscience. *Science*, 241(4871):1299–1306.

[Shlens et al., 2006] Shlens, J., Field, G. D., Gauthier, J. L., Grivich, M. I., Petrusca, D., Sher, A., Litke, A. M., and Chichilnisky, E. (2006). The structure of multi-neuron firing patterns in primate retina. *The Journal of neuroscience*, 26(32):8254–8266.

[Song et al., 2017] Song, J., Marsili, M., and Jo, J. (2017). Emergence and relevance of criticality in deep learning. *arXiv preprint arXiv:1710.11324*.

[Sorbaro et al., 2019] Sorbaro, M., Herrmann, J., and Hennig, M. (2019). Statistical models of neural activity, criticality, and zipf's law. In Tomen, N., JM,

H., and Ernst, U., editors, *The Functional Role of Critical Dynamics in Neural Systems*. Springer.

[Spicher, 2014] Spicher, D. (2014). Modeling multi-neuron spike trains with energy-based models. Master's thesis, Master Program in Computational Neuroscience, Berlin.

[Sporns et al., 2005] Sporns, O., Tononi, G., and Kötter, R. (2005). The human connectome: a structural description of the human brain. *PLoS computational biology*, 1(4):e42.

[Theano Development Team, 2016] Theano Development Team (2016). Theano: A Python framework for fast computation of mathematical expressions. *arXiv e-prints*, abs/1605.02688.

[Thomas et al., 1972] Thomas, C., Springer, P., Loeb, G., Berwald-Netter, Y., and Okun, L. (1972). A miniature microelectrode array to monitor the bioelectric activity of cultured cells. *Experimental cell research*, 74(1):61–66.

[Tieleman, 2008] Tieleman, T. (2008). Training restricted Boltzmann machines using approximations to the likelihood gradient. In *Proceedings of the 25th international conference on Machine learning*, pages 1064–1071. ACM.

[Tierney, 1994] Tierney, L. (1994). Markov chains for exploring posterior distributions. *the Annals of Statistics*, pages 1701–1728.

[Tkačik et al., 2014] Tkačik, G., Marre, O., Amodei, D., Schneidman, E., Bialek, W., and Berry II, M. J. (2014). Searching for collective behavior in a large network of sensory neurons. *PLoS computational biology*, 10(1):e1003408.

[Tkačik et al., 2015] Tkačik, G., Mora, T., Marre, O., Amodei, D., Palmer, S. E., Berry, M. J., and Bialek, W. (2015). Thermodynamics and signatures of criticality in a network of neurons. *Proceedings of the National Academy of Sciences*, 112(37):11508–11513.

[Tkacik et al., 2006] Tkacik, G., Schneidman, E., Berry, I., Michael, J., and Bialek, W. (2006). Ising models for networks of real neurons. *arXiv preprint q-bio/0611072*.

[Truccolo et al., 2005] Truccolo, W., Eden, U. T., Fellows, M. R., Donoghue, J. P., and Brown, E. N. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089.

[Ungerleider and Pessoa, 2008] Ungerleider, L. G. and Pessoa, L. (2008). What and where pathways. *Scholarpedia*, 3(11):5342.

[Verkhratsky and Parpura, 2014] Verkhratsky, A. and Parpura, V. (2014). History of electrophysiology and the patch clamp. In *Patch-Clamp Methods and Protocols*, pages 1–19. Springer.

[Wässle et al., 2009] Wässle, H., Puller, C., Müller, F., and Haverkamp, S. (2009). Cone contacts, mosaics, and territories of bipolar cells in the mouse retina. *Journal of Neuroscience*, 29(1):106–117.

[Whittington and Bogacz, 2019] Whittington, J. C. and Bogacz, R. (2019). Theories of error back-propagation in the brain. *Trends in cognitive sciences*.

[Williams et al., 2018] Williams, A. H., Kim, T. H., Wang, F., Vyas, S., Ryu, S. I., Shenoy, K. V., Schnitzer, M., Kolda, T. G., and Ganguli, S. (2018). Unsupervised discovery of demixed, low-dimensional neural dynamics across multiple timescales through tensor component analysis. *Neuron*.

[Yger et al., 2018] Yger, P., Spampinato, G. L., Esposito, E., Lefebvre, B., Deny, S., Gardella, C., Stimberg, M., Jetter, F., Zeck, G., Picaud, S., et al. (2018). A spike sorting toolbox for up to thousands of electrodes validated with ground truth recordings in vitro and in vivo. *eLife*, 7:e34518.

[Zanotto et al., 2017] Zanotto, M., Volpi, R., Maccione, A., Berdondini, L., Sona, D., and Murino, V. (2017). Modeling retinal ganglion cell population activity with restricted Boltzmann machines. *arXiv preprint arXiv:1701.02898*.

Acknowledgements

> Tanto, salendo inverso l'erta, acquista,
> che vede dove aperta era la grotta;
> e l'aria, già caliginosa e trista,
> dal lume cominciava ad esser rotta.
> Al fin con molto affanno e grave ambascia
> esce de l'antro, e dietro il fumo lascia.
>
> L. Ariosto, Orlando Furioso, XXXIV, 45

Four and a half years have passed since I moved to Edinburgh and started this PhD, and what's written in the thesis you have (or haven't) just read is only a small part of what I learned in this time. The other part is made of people.

I have had the fortune of having a wonderful supervisor, Matthias Hennig, who never once failed to be wholly, almost unconditionally supportive of my work and ideas. And another one, Arvind Kumar, who is one of the most interesting and knowledgeable people I know. They both did an amazing human and scientific job, and I must thank them before everybody else. Michael Rule, Michael Deistler, and Gerrit Hilgen were also essential collaborators without whom much of this work wouldn't have been possible. Thank you also to David Barrett, who mentored me, listened to me, and was responsible for my Google Fellowship, which boosted my self-confidence. Simone Bronzin, Federica Gregori, and Anna Maraga welcomed me for a few months at Metaliquid, where I could learn something new, and free my mind from thesis writing for a while.

Together with Arvind, I would like to warmly thank Ramón Martínez Mayorquin, Ylva Jansson, Emil Wärnberg, Luiz Tauffer, Florian Fiebig, and all the others that shared deep, stimulating lunchtime discussions at KTH and the computational biology journal club. I am proud of having been around you. It was a great learning environment for academic and non-academic topics, and I will certainly miss our debates about carbon offsetting, global warming, consciousness uploading, fake news, academic misconduct, cultured meat, open data, open review, open access, general intelligence, contemporary meditation, teaching evaluation, psychoactive drugs, and so much more. Nikhil Nair was of great support during my time in Stockholm, and I thank him for the time he shared with me.

I spent an amazing time in Bangalore in 2015 and in 2017. I saw a great learning environment, participated in great discussions, visited a wonderful country, and met some incredibly smart and friendly people. I cannot mention them all, but thank you too. I sincerely hope to see you all again.

When I arrived in Edinburgh, two people offered me a great flat, and with it, what later became an unexpectedly intimate friendship. Thank you, Howard Lin, and thank you, Filipa Sousa, for your space, time, and trust.

It is almost impossible to thank all of the people that shared good times and bad times with me in Scotland in all these years. An honourable mention goes to all the people of Edinburgh University Fencing Club: the club gave me a way to spend my time and efforts in something different from research. So did the D&D sessions with my friends Chiara Cupini, Manuel Nobili, Anna Francesca, Carlo Miccolis, Daniela Borasco, Nunzia Napolitano, and the others that have left. Oh, and the climbing people!

Sam Laing, Sander Keemink, Nolwenn Donsimoni and later Martijn Kelder were my first "family" when I arrived in the UK, and I'm very grateful for the safety I felt around them; for the dinners, the trip to Orkney, the games, the conversations, and all the whisky. Among the Forum people I must mention my old officemates of 5.08 (for the tea and endless source of snacks) and 2.53 (for the music and all that), Balázs Szigeti for his contagious passion for science, Katharina Heil for her friendliness and consistent trustworthiness, and Wioleta Kijewska for her unstoppable laughing crises. And thank you too, Pavlos Andreadis, for welcoming me to your wonderful flat, which is now home.

And for the lunches, the barbecues in the park, the nights out, the hours-long coffee breaks, the whining, the lazy times, the poetry nights, the westworld and game of thrones screenings, the dinners at home, the proofreading, and the occasional fight, thank you Nicolas Collignon, Joseph Cronin, Kathryn O'Brien, Nathalie Dupuy, Janie Sinclair, Akash Srivastava, Pablo León, Jenny Sanger, Ed Fincham, Todor Davchev, Cole Hurwitz, Maria Astefanoaei, and Ludovica Luisa Vissat. Thank you, thank you, thank you. And with you there are so many others that *hanc marginis exiguitas non caperet*.

During difficult times, I could benefit from the help of many friends. Some didn't even know they were helping me. Some didn't even know me well. Some where next to me, some were thousands of kilometres away. So, thank you to (in random order) Jacopo Béchaz, Gian Matteo Rinaldi, Michela Gazzetto, Salvatore Gnecchi, Giulia Provasoli, Samuele Bottagisi, and *o excelentíssimo senhor* Tiago Zortea. And sorry, for occasionally pouring my distress upon you all.

And, as always, thank you, mum and dad, for encouraging me to go and explore, even if it meant seeing me only twice a year.

Finally, in the last year, one person helped more than the others, simply with his constant presence, silliness, sweetness, and understanding, and that is Earl Cuarteros.

You, you, all of you, deserve the best from life, whether it'll take us far away from each other or not. You have been amazing company, you have been my safety net, you have all been necessary.