

Distributed interactions across multiple brain circuits explain performance in a cued spatiovisual discrimination task

Alexander Morley

University College
University of Oxford

*A thesis submitted for the degree of
Doctor of Philosophy*

Trinity 2019

Abstract

Neurons transiently synchronise their spiking activity to form cell assemblies that can represent information about external sensory inputs and internal cognitive mechanisms in selective brain circuits. However, it remains poorly understood whether, and how, within-circuit spiking can be organised to form brain-wide assemblies at the service of optimal behaviour. The answer to this question, grounded in that everyday life experience is typically composed of a variety of stimuli and sensory modalities, remains elusive because population spike codes have not been investigated across multiple simultaneously recorded brain circuits.

Here we show that in a baseline period neurons in the dorsal hippocampal CA1 (CA1), the amygdala (Amy), the prefrontal cortex (PFC), the ventral tegmental area (VTA) and the nucleus accumbens (NAc) of mice can organise their activity over few tens of milliseconds to form constitutively expressed patterns of short timescale neuronal co-firing in the mouse brain. We found that in a cued discrimination task, where mice used LED-based visual cues to determine their behaviour in response to an auditory stimulus, these same brain-wide assemblies evolve with the successful translation of local sensory representations into behaviour. Moreover, their activity was predicted by a stereotyped, transient oscillatory state which appeared to facilitate the tight coordination of firing patterns between neurons in distal structures.

These insights were facilitated by state-of-the-art algorithms for spike-sorting and multivariate LFP segmentation which we were able to leverage using current best practices in software architecture. Our results thus support the notion of cell assemblies existing at the macro, inter-regional level of brain organisation and show how this may be critical for integrating disparate sources of information.

Distributed interactions across multiple brain circuits explain performance in a cued spatiovisual discrimination task



Alexander Morley
University College
University of Oxford

A thesis submitted for the degree of
Doctor of Philosophy
Trinity 2019

Acknowledgements

Specific Contributions

The development of mountaintools and SpikeForest, described in chapter 2 was done in collaboration with Jeremy Magland and others at the Flatiron Institute. All of the written content and analysis shown in the chapter is solely my own work.

Of the recordings from the 8 mice presented in this thesis, 4 of them were recorded by myself personally, the other 4 by Roman Rothaermel. The experimental design and analysis, as well as all of the written content is solely my own work.

The immunohistochemistry and confocal imaging shown here was performed by Natalia Campo-Urizza and Ben Micklem respectively.

Personal and Institutional

None of this would have been remotely possible without the support of everyone in the Dupret Lab and the MRC Brain Network Dynamics Unit as a whole. Thanks especially to David Dupret, whose extraordinary surgical and technical expertise allowed me to get up and running almost as soon as I arrived. He offered me a project with so many exciting possibilities and guided me through many of them. Thanks also to Pavel Perestenko for building and helping me maintain the rig. Who knows what would get done without his amazing support. The same applies to Natalia Campo-Urriza, who built most of the microdrives for this project and also supported the anatomical work along with Ben Micklem who took the beautiful images seen in the thesis. I am also very grateful for Roman Rothaermel's contribution in doing some of the recordings for this project, and I cannot wait to see where he takes it next. Along with others in the lab, Vítor Lopes dos Santos taught me more than I ever knew I wanted to know about data analysis and was always there to tell me why my ideas would not work, and occasionally even gave suggestions for how I could make them better.

I would like to thank the Medical Research Council both for supporting me financially for the large part of my DPhil and for continuing to support the Brain Network Dynamics Unit. I would also like to thank my friends and colleagues with whom I have had the pleasure of working with not just for "Open Science" but with the aim of building more just and equitable research systems. Of course I would not have met half of those people without the support of the Mozilla

Foundation, who through the Mozilla Science Labs connected me with an incredibly supportive global open research community. They then offered me the honour of taking up a Mozilla Fellowship for the past year, an opportunity which, though almost over, I have yet to fully digest the impact of.

Finally, and most importantly, I owe everything to my family. My parents have supported me through every endeavour I have set my eyes on and did not bat an eyelid when I told them I was going to take four years away from medical school to do research. If I did not ask too much of them over this time then I definitely did ask too much of my partner Cassie. Even with the best lab in the world research can be a lonely place at times and I do not know that I could have coped without her.

Abstract

Neurons transiently synchronise their spiking activity to form cell assemblies that can represent information about external sensory inputs and internal cognitive mechanisms in selective brain circuits. However, it remains poorly understood whether, and how, within-circuit spiking can be organised to form brain-wide assemblies at the service of optimal behaviour. The answer to this question, grounded in that everyday life experience is typically composed of a variety of stimuli and sensory modalities, remains elusive because population spike codes have not been investigated across multiple simultaneously recorded brain circuits.

Here we show that in a baseline period neurons in the dorsal hippocampal CA1 (CA1), the amygdala (Amy), the prefrontal cortex (Pfc), the ventral tegmental area (VTA) and the nucleus accumbens (NAc) of mice can organise their activity over few tens of milliseconds to form constitutively expressed patterns of short timescale neuronal co-firing in the mouse brain. We found that in a cued discrimination task, where mice used LED-based visual cues to determine their behaviour in response to an auditory stimulus, these same brain-wide assemblies evolve with the successful translation of local sensory representations into behaviour. Moreover, their activity was predicted by a stereotyped, transient oscillatory state which appeared to facilitate the tight coordination of firing patterns between neurons in distal structures.

These insights were facilitated by state-of-the-art algorithms for spike-sorting and multivariate LFP segmentation which we were able to leverage using current best practices in software architecture. Our results thus support the notion of cell assemblies existing at the macro, inter-regional level of brain organisation and show how this may be critical for integrating disparate sources of information.

Contents

List of Figures	xi
List of Tables	xiii
List of Abbreviations	xv
1 General Introduction	1
1.0.1 How can memory guide behaviour?	2
1.0.2 How are memories physically manifested in the brain?	2
1.0.3 Evidence for localisation of brain function	5
1.0.4 How can memory sub-systems function as a whole?	7
1.0.5 Questions	8
2 Spike Sorting - Progress in fully automatic methods	11
2.1 Introduction	12
2.1.1 Spike sorting is challenging	12
2.1.2 Overview of current spike sorting methods	13
2.1.3 Assessing the performance of automatic spike sorting algorithms	20
2.2 Methods	22
2.2.1 mountaintools	22
2.2.2 Automatic curation of sorting results	25
2.2.3 Validating and comparing results	29
2.3 Results	30
2.3.1 SpikeForest	30
2.3.2 Comparison between sorters without ground truth	30
2.4 Discussion	34
2.4.1 Alternative sorting metrics	34
2.4.2 Justification for our approaches to curation	35
2.4.3 Human factors affected by automatic versus manual spike sorting	35
2.4.4 Impact of the software-first approach	36

3	Experimental Design	39
3.1	Introduction	40
3.2	Methods	43
3.2.1	Recording procedures	43
3.2.2	Assessment of behaviour	47
3.2.3	Speed Profile Matching	51
3.3	Results	53
3.3.1	Anatomical verification of recording sites	53
3.3.2	Behavioural results	53
3.4	Discussion	58
3.4.1	Task design	58
3.4.2	Controlling for the effects of behaviour	59
3.4.3	Recording technologies	60
4	Integrating memories spanning multiple brain circuits	61
4.1	Introduction	63
4.2	Methods	67
4.2.1	Problems and solutions assessing single neuron modulation by external events	67
4.2.2	Assembly Analysis	72
4.2.3	Assembly Classifiers	77
4.3	Results	78
4.3.1	Differential engagement of multiple regions during successive trial stages	78
4.3.2	Distribution of single neuron responses does not suggest classes	78
4.3.3	Observing cross-structural assemblies	79
4.3.4	Intrinsic properties between cross- vs within- region assemblies	80
4.3.5	Distributed assembly activity encodes performance information	86
4.4	Discussion	91
5	Population Dynamics – Comparison & development of methods for assessing population dynamics in multi-site, multi-unit record- ings	93
5.1	Introduction	94
5.1.1	Motivations for assessing population dynamics	94
5.1.2	Defining interpretability in relation to models of population dynamics	95
5.2	Methods and Results	98
5.2.1	Overview of task-independent (single observation) methods .	98
5.2.2	Overview of task-dependent (multiple observation) methods	104

5.2.3	Example results with models of population dynamics	105
5.2.4	Finding patterns of cross-correlations using ICA	106
5.3	Discussion	109
5.3.1	How interpretable are these models?	109
5.3.2	Choosing a Model	110
6	Searching for global oscillatory states underlying long range coordination	111
6.1	Introduction	112
6.1.1	Long-range Coordination	112
6.1.2	Pairwise approaches for characterising multiple local field potentials	113
6.1.3	Multivariate approach for characterising the interaction between more than two local field potentials over time	113
6.2	Methods	115
6.2.1	Preprocessing	115
6.2.2	Spectrograms	115
6.3	Results	117
6.3.1	Transient oscillatory states are driven by task-timings	117
6.3.2	Transient oscillatory states support the emergence of brain-wide assemblies	117
6.4	Discussion	123
7	Closing Discussion	125
7.1	Overview	126
7.2	What is a cell assembly?	127
7.3	Future Directions	129
Appendices		
A	Additional Examples	133
A.1	Behaviour	134
A.2	Single-neuron responses	135
A.3	Assembly composition	136
B	Anatomy	137
C	Software Usage & Contribution	141
C.1	Software in science	142
C.2	Contributions to scientific software as part of this thesis	143
References		145

List of Figures

1.1	A simple model of brain function	3
1.2	Interaction Schema	8
2.1	Kilosort	15
2.2	SpyKING CIRCUS	16
2.3	MountainSort	17
2.4	Wave_clus	18
2.5	JRClust	19
2.6	Curation - Filter Stage	26
2.7	Curation - Merge Stage	27
2.8	Curation - Merge Clusters	29
2.9	Between Sorter Confusion Matrices	31
2.10	Sorting Comparison	32
3.1	Anatomy Schematic	44
3.2	Decision Point Heuristic	50
3.3	Trajectory Vector Method	51
3.4	Speed Profile Matching	52
3.5	Tetrode Tracks	54
3.5	Tetrode Tracks (cont.)	55
3.6	Task Schema	55
3.7	Behaviour Task Performance	57
4.1	Wavelet Information Comparison	71
4.2	Assembly Detection	75
4.3	Assembly Tracking	76
4.4	Rate Response Matrix	79
4.5	Rate Response Distributions	82
4.6	Modulation Response Distributions	83
4.7	Response Distributions Dip Test	84
4.8	Assembly weight vectors span multiple regions	84
4.9	Contribution of cross-regional interactions to assembly activity	85
4.10	Assembly Properties	87

4.11	Relationship between dynamics of assemblies and behaviour	87
4.12	Decoding behaviour from assembly activity	88
4.13	Relationship between dynamics of assemblies and surrogate behaviour	90
4.14	Evidence for different patterns of inter-regional communication . . .	92
5.1	Modelling population dynamics over times and trials with TCA . .	105
5.2	Modelling population dynamics over time using correlation matrices, PCA and LDA	106
5.3	Spike-Timing Pattern Detection using ICA	108
5.4	Choosing a model of population dynamics	110
6.1	LFP Responses	119
6.2	LFP State Responses	120
6.3	LFP lead, lag relationships in different states.	121
6.4	Assembly State Relationships	122
A.1	Example Paths During Task	134
A.2	Task Peri-Stimulus Histograms	135
A.3	Regional Assembly Contributions	136
B.1	Tetrode Tracks Extended 1	138
B.2	Tetrode Tracks Extended 2	139
B.3	Tetrode Tracks Extended 2	140

List of Tables

3.1	Stereotaxic coordinates for tetrode implantation. Distances are in mm.	44
3.2	Antibody Table	46
3.3	Summary of recordings	53

List of Abbreviations

AP	Anteroposterior Axis
ML	Mediolateral Axis
DV	Dorsoventral Axis
CA1	Dorsal Region of the CA1 Hippocampus
Amy	Amygdalar Nuclei
NAc	Nucleus Accumbens
PfC	Prefrontal Cortex
VTA	Ventral Tegmental Area
ArchT	Archaeorhodopsin from Halorubrum strain TP009
EEG	Electroencephalogram
MEG	Magnetoencephalography
LFP	Local Field Potential
PSD	Power Spectral Density
(M)AR	(Multivariate) Auto Regression
HMM	Hidden Markov Model
#	Number or Count
1-D, N-D	One- or N-dimensional.
PC(A)	Principal Components (Analysis)
IC(A)	Independent Components (Analysis)
PSTH	Peri-stimulus Histogram
SD	Standard Deviation
SE(M)	Standard Error (of the mean)
SNR	Signal-to-Noise Ratio
MI	Mutual Information
CAS	Content Addressable Storage
GPU	Graphics Processing Unit

1

General Introduction

Contents

1.0.1	How can memory guide behaviour?	2
1.0.2	How are memories physically manifested in the brain? .	2
1.0.3	Evidence for localisation of brain function	5
1.0.4	How can memory sub-systems function as a whole? . . .	7
1.0.5	Questions	8

What is your favourite colour? Would you eat a pizza if it were that colour? How about a sweet? While simple features of the environment such as colour can invoke emotional responses their meaning is determined by the context in which they are experienced.

Extracting the correct meaning from any experience thus involves integrating across multiple sources and potentially across multiple sensory modalities. What the correct meaning is and thus what behaviour will be advantageous is, for the most part, learned from past experience.

More generally, the ability to form a memory that allows an animal to learn to seek out experiences that have lead to reward in the past, and avoid those that are associated with danger, is evolutionarily advantageous. Indeed, as a result of this selection pressure, the circuits that are thought to underlie this key function are highly conserved in the mammalian brain (Janak et al. 2015).

1.0.1 How can memory guide behaviour?

Before we discuss our current understanding about the division of labour in memory-related circuits it is first helpful to define a framework under which we understand the brain to be operating. This in turn will allow us to precisely define some key terms.

We consider brain function as the process of mapping a set of external stimuli to a set of motor outputs. The exact function that performs this mapping is referred to as the "network state", which is not necessarily limited to the activity of neurons in the brain, although that remains the focus of this thesis (Figure 1.1). We refer to the process whereby a stimulus produces a change in network state as learning and the change itself as a memory. The nature of these network state updates is assumed to be driven by evolutionary processes such that the optimal motor output is one that maximises the chances of survival. Given this long-term goal, learning can often be seen as maximising a valence function which more closely reflects upstream goals such as nutritional balance, immediate safety or, in some species, social dominance. In reinforcement learning terms these intermediate rewards are what allow an agent to estimate the value of past and future actions (Sutton 1998).

1.0.2 How are memories physically manifested in the brain?

The word engram is, in our opinion, often misunderstood outside the neuroscience community likely as a result of it having different meanings for different subsets within the neuroscience community. The Oxford English Dictionary defines an engram as a "memory-trace; a permanent and heritable physical change in the nerve tissue of the brain, posited to account for the existence of memory." This definition refers to Richard Semon's original account in which he defined the words engraving, engram, and ephory as encoding, physical change, and the retrieval process of memory, respectively (Semon 1904; Semon 1921). However this does not reflect current usage of the term. Firstly our understanding is that an engram need not be permanent: it only need last as long as the memory itself. Second, the suggestion that an engram be heritable was discredited even in Semon's lifetime,

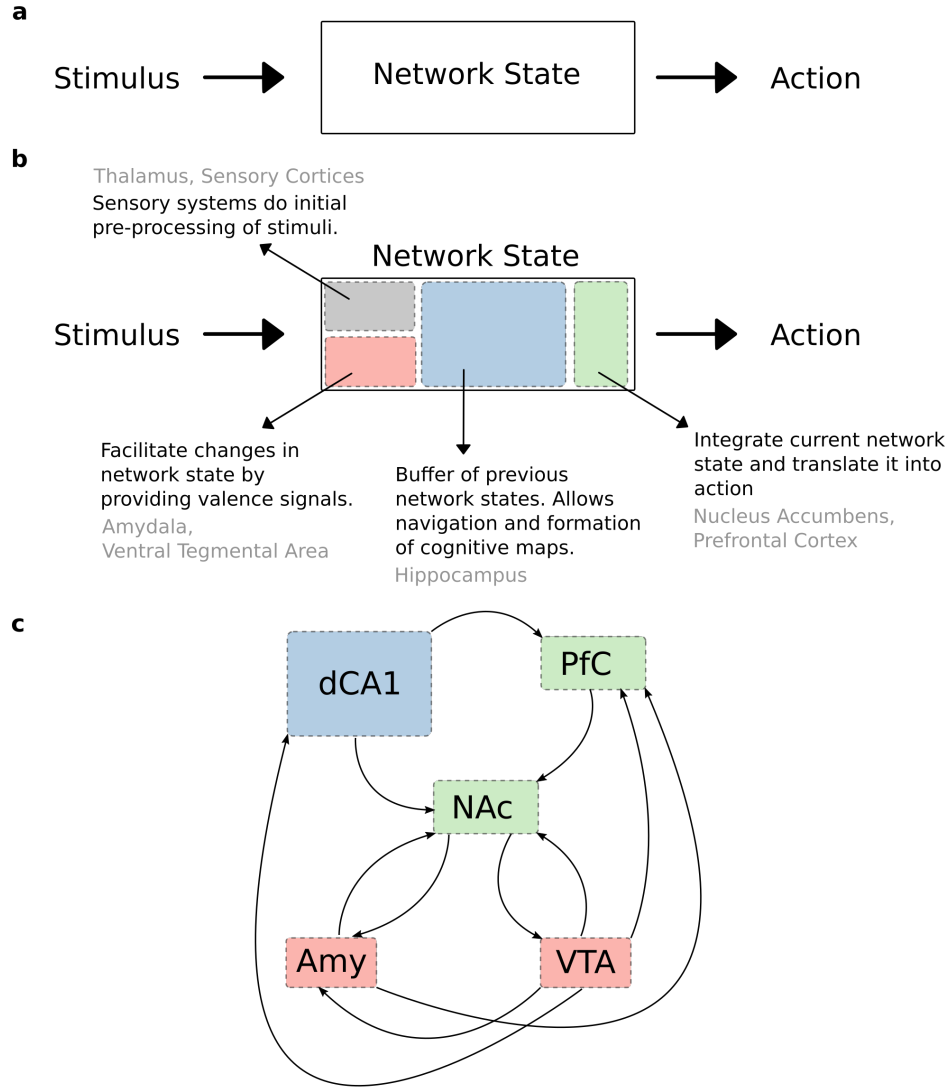


Figure 1.1: (a) Network state is the mapping between external stimuli and action. (b) The network state has evolved multiple sub-components such that it selects evolutionarily advantageous actions. (c) The (non-exhaustive) primary interconnections between the dorsal CA1, Amygdala, Ventral Tegmental Area, Nucleus Accumbens and prefrontal cortex. Dopaminergic projections from the VTA go to PfC (Williams et al. 1995), dorsal CA1 (McNamara et al. 2014), the amygdala and to the Nucleus Accumbens (Salgado et al. 2015). Medium spiny neurons (MSNs) in the Accumbens project back to the VTA (Morales et al. 2017) as well as to the amygdala (Salgado et al. 2015). The MSNs in the accumbens also receive glutamatergic inputs from the dorsal CA1 (Trouche et al. 2019), prefrontal cortex (Salgado et al. 2015) and the amygdala (McDonald 1991). The amygdala (McDonald 1991) and the CA1 region of the hippocampus also project to prefrontal cortex (Swanson 1981).

and he himself ignored this idea in his second book¹ (SEMON 1909; Semon 1923). Merriam Webster describes an engram as "a hypothetical change in neural tissue postulated in order to account for persistence of memory". This is closer to what we believe although we would contest that the existence of a change in neural tissue accounting for memory has moved beyond hypothetical even if the exact nature of the change is not fully understood. Here we use the definition provided by Collins: an engram is "the physical basis of an individual memory in the brain".

However, these are all distinct from the common usage of the word to refer to a group of cells that were activated during an experience and whose re-activation leads to memory recall. This definition has emerged from the use of activity-dependent cellular tagging mechanisms which make it possible to selectively drive only the neurons that were highly active in a given context and the observation that doing so induces behaviours associated with that earlier context (Liu et al. 2012). Although some have attempted to rename this particular idea as "engram cells" (Tonegawa et al. 2018) to reflect the narrower definition this distinction is rarely made. Those who focus on the dynamics of neural activity might instead suggest that it is in fact a specific temporal pattern of activity in a specific set of neurons leading to recall that represent a "memory trace". However, given our previous assertion that a memory is the change in network state based on past experience that alters future outcomes, neither of the above definitions are necessary or sufficient. For example, by activating cells in the retina with a pattern of stimulation to match a previous experience it would be possible to trigger recall but few would argue that the locus of memory includes the retina. Instead our definition of memory is the physical change that allows a past experience to influence a future one or, equivalently, the change in network state caused by a particular experience. This likely includes neuronal plasticity, synaptic plasticity, epigenetic responses and alterations in glial cell function. This shifts our perspective on the question of the functional localisation of memories in the brain from the question "where can we

¹Semon did have some prescient ideas about the constructive nature of memory and his work focused on recall, or "ecphory" in his terms, being an active process.

find representations of a experience in the brain?" to "what role do different regions play in facilitating these changes by which we have defined an engram?".

1.0.3 Evidence for localisation of brain function

Instead of, for example, focusing on the representations of space in the CA1 region of the hippocampus that can be found in the form of place cells (Moser et al. 2017; O'Keefe et al. 1971), we focus on what aspects of the hippocampus make it suited to playing the role of a high capacity buffer of previous network states (Figure 1.1). The two main inputs to the dorsal CA1 are directly from layer III of entorhinal cortex and via the perforant path that consists of the dentate gyrus and then the CA3. Both theoretical models (Treves et al. 1994) and experimental evidence (Leutgeb et al. 2007) suggest local competitive inhibition allows the dentate gyrus to decorrelate the cortical inputs. In a similar vein the highly recurrent connectivity of the CA3 is thought to underlie its pattern completion capacity (Hasselmo et al. 1995; Nakazawa et al. 2002). The CA1 may integrate the output of the CA3 with direct input from layer II of entorhinal cortex and provide a sparse low-footprint representation for readout by downstream regions (Barnes et al. 1990).

Beyond its architecture, there is also behavioural evidence for the CA1 acting as a buffer of prior network states. For example, it is required for allocentric navigation tasks (Morris et al. 1986; Tsien et al. 1996) which are solved by computing the difference between the current network state (egocentric representation) and past states in order to navigate to the goal location. However, it is not required for egocentric navigation (Packard et al. 1992), which would not require a buffer unless there are a number of steps required or if there was a temporal component, both of which seem to render the task CA1-dependent (Farovik et al. 2010; Huerta et al. 2000; Rondi-Reig et al. 2006).

Both the VTA and Amygdala are thought to encode valence and salience signals (Herry et al. 2008; Paton et al. 2006; Schultz 1998; Weiskrantz 1956; Wise 2004). Conceptually these two signals are orthogonal, although their implementation in the brain is unlikely to be. The valence of an event indicates whether it may be

advantageous to change the network state to seek out or avoid a similar experience in the future while the salience indicates whether it may be advantageous to alter the network state at all.

Beyond this common role the two regions have little overlap in terms of their afferent connectivity, how they transmit these signals to other regions and their core roles in memory. The amygdalar nuclei receive a wide range of primary sensory inputs from the olfactory stream (Scalia et al. 1975), cortical regions (TEIII LeDoux 1998) and thalamus (MGN LeDoux 1998, LGN Shi et al. 2001) as well as somatosensory, gustatory and visual inputs from insular cortex (Shi et al. 1998). In contrast, the VTA receives inputs from a wide range of structure (to which it is usually reciprocally connected) but mostly those that encode multi-modal sensory information such as the PfC (Geisler et al. 2007) or visceral signals from the midbrain and thalamus (Morales et al. 2017).

This difference in inputs likely reflects their different functions. While the VTA broadcasts its valence (Schultz 1998) or salience (Horvitz 2000; Redgrave et al. 1999; Ungless 2004) signals to many brain regions via its dopaminergic outputs the amygdala retains the associations between sensory cues and unconditioned stimuli drive behaviour directly. Thus the VTA may, broadly speaking, act as a teaching signal while the amygdala both stores associations between sensory stimuli and affective states as well as guiding the associated output behavioural response.

While the simpler behavioural outputs of the amygdala may be mediated by direct projections to the brainstem, midbrain and hypothalamus it is likely that instrumental behaviour is driven via the pre-frontal cortex (PfC) and nucleus accumbens (NAc; Cardinal et al. 2002). The NAc is often seen as an interface between the limbic and motor systems. In this way it integrates information from the amygdala, hippocampus and cortical regions to drive goal-directed behaviours (Ambroggi et al. 2008; Day et al. 2007; Ito et al. 2008; Trouche et al. 2019).

In monkeys and humans prefrontal cortex is important for cognitive control (Miller 2000). In the rodent PfC, prelimbic areas may be biased towards guiding the execution of behaviour (Peters et al. 2009) while infralimbic areas may be biased

towards behavioural suppression (Maren et al. 2004). More specifically studies have tended to focus for two roles for the PFC in rodents. The first is in spatial working memory tasks, where the PFC seems to interact in concert with the hippocampus to support goal-directed actions (Benchenane et al. 2010; Jones et al. 2005; Spellman et al. 2015). The other role is in encoding changes in reward contingencies. In this context the PFC is also thought to co-ordinate with other regions in order to fulfil its role. For example it may interact with the amygdala to enable reversal learning of conditioned olfactory stimuli (Schoenbaum et al. 1999) and dopamine release from the VTA in the PFC seems to be necessary for performance in a Pavlovian over-expectation task (Takahashi et al. 2009).

1.0.4 How can memory sub-systems function as a whole?

Given this functional localisation the next question is how these processes are integrated to rapidly control behaviour while using information acquired from days to years ago. The motifs that exist between pairs of these regions are discussed in the introduction to chapter 4.

Here we try and consider the system as a whole. We outline three possible highly simplified frameworks for understanding the passage from stimulus to response (Figure 1.2). In scenario **(a)** information is processed sequentially from sensory input to action. In scenario **(b)** the initial input is immediately distributed amongst all regions according to their function where it is locally processed and then converges onto an output structure. In scenario **(c)** there is some sequential structure, but activity is dominated by direct and indirect reciprocal interactions between regions, at some point the inputs to the output structure reach a threshold that drives a response. Reality likely holds a mixture of all three, but the extent of each regime may vary between circuits and behaviour. For example, the early visual stream is largely sequential while there are multiple recurrent pathways in the hippocampal-pre-frontal loop.

In order to distinguish between the modes of operation in the context of translating multi-modal sensory information into correct behaviour we simultaneously

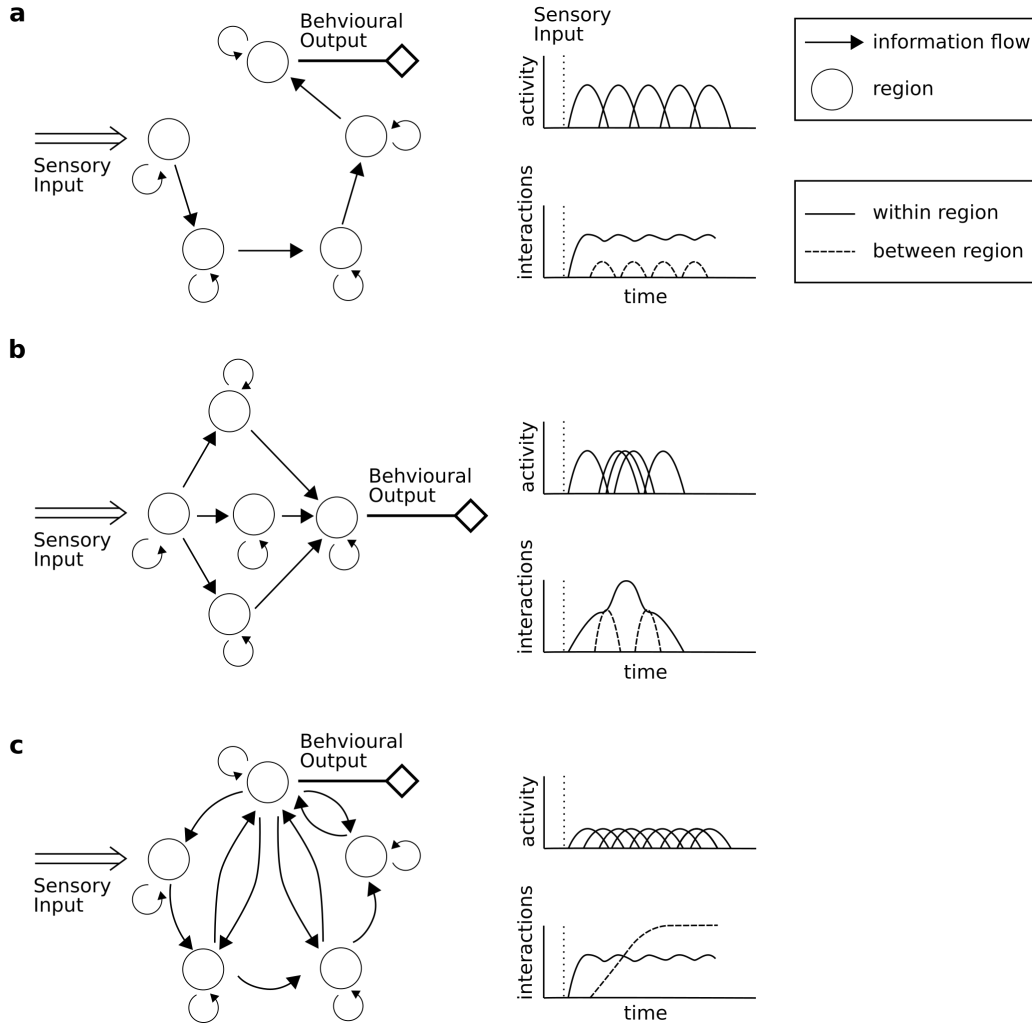


Figure 1.2: Schema representing putative patterns of communication between brain regions during the course of responding to a stimulus. Left, depicts the flow of information between regions about a sensory input that eventually drives a behavioural output. Right top, the corresponding activity within each region during the same period. Right bottom, the corresponding magnitude of interactions between neurons either within (solid line) or between (dashed line) regions.

monitored LFP and spiking activity in five anatomically disparate brain regions during execution of a cued spatio-visual discrimination task.

1.0.5 Questions

In chapter 2 we discuss progress in automated spike sorting for extracellular multi-channel recordings and show how we leveraged best practices in software development to take advantage of the recent algorithmic advances to sort hundreds of neurons across five brain regions.

In chapter 3 we detail the design of, and then characterise, a behavioural task where the same discrete stimulus triggers opposing behavioural responses.

In chapter 4 we provide an initial investigation into the neural underpinnings of these responses, drawing from both single-neuron analyses and population measures.

In chapter 5 we discuss alternative ways to assess emerging population dynamics and provide a framework for deciding which methods are appropriate under which circumstances.

Finally in chapter 6 we discuss a machine-learning approach applied to the local field potentials to detect oscillatory states and relate these to the task and neuronal population dynamics.

2

Spike Sorting - Progress in fully automatic methods

Contents

2.1	Introduction	12
2.1.1	Spike sorting is challenging	12
2.1.2	Overview of current spike sorting methods	13
2.1.3	Assessing the performance of automatic spike sorting algorithms	20
2.2	Methods	22
2.2.1	mountaintools	22
2.2.2	Automatic curation of sorting results	25
2.2.3	Validating and comparing results	29
2.3	Results	30
2.3.1	SpikeForest	30
2.3.2	Comparison between sorters without ground truth	30
2.4	Discussion	34
2.4.1	Alternative sorting metrics	34
2.4.2	Justification for our approaches to curation	35
2.4.3	Human factors affected by automatic versus manual spike sorting	35
2.4.4	Impact of the software-first approach	36

2.1 Introduction

The latest developments in algorithms for automatic spike sorting will be more effective when combined with current best practices in software development.

2.1.1 Spike sorting is challenging

Spike sorting, the transformation of the extracellular recorded signal into a series of action potentials labelled according to whether they likely originated from the same neuron, has been a challenge for electrophysiologists since the origin of extracellular recordings. The problem can appear relatively simple to solve optimally at first look, but the statistics of real-world recordings render it difficult to do so. Extracellular recordings tend to be non-stationary: both the waveforms from a given neuron and the different sources of noise can change over both short and long timescales. Muscle and other artefacts are common and overlap with the same frequency band as action potentials. Much of the background "noise" could result from a summation of many small action potentials from distant neurons whose amplitudes are too close to the noise floor to sort. Amplitude drift in these waveforms can result in neurons seeming to appear and disappear from recordings. One component of this drift occurs slowly over time as the tissue moves around the tetrodes and changes the relative positions of each neuron. More dangerously the other component is drift on a much faster timescale as a result of the movement of the mouse. The danger comes from misinterpreting the changes in the activity of a neuron as genuinely motion- or task-related when they may result from the electrodes moving a very small amount when the animal makes a certain movement.

At the turn of the millennium Harris et al. 2000 and Harris et al. 2003 showed the advantage of multiple closely-spaced recording sites in this process, and the initial promise of semi-automated techniques over manual sorting. However, this process was both computationally intensive and still required significant manual curation. A six hour recording with ten tetrodes could still take over 12 hours each for the automatic and manual processes respectively. While iterative

improvements were made in the following years, it was the upcoming popularity of high-channel count probes, for which these methods would be entirely impractical¹, that drove the development of multiple techniques aiming at fully automated solutions to spike sorting.

2.1.2 Overview of current spike sorting methods

We chose a few of these methods based on our subjective assessment of their current popularity, algorithmic novelty, and potential for adoption in the coming years. In Figure 2.1 to Figure 2.5 we establish a graphical overview of each of these methods. While early spike sorting followed a linear progression from spike-detection to feature extraction to clustering to curation, the algorithms presented here aim to take better advantage of densely spaced probes and improvements in commodity computing hardware such as GPUs.

Each of the sorters discussed here either follow the same script but improve the component parts or take a totally different approach to solving the problem. Of the latter, Kilosort and SpyKING CIRCUS are the most prominent examples both using adaptive approximations of the full time series. That is they try to find a model parameterised by a set of spatio-temporal templates and the spike times that best explain the data. In general, adaptive approximation aims to find a set of functions g_{γ_n} such that:

$$x \approx \sum_{n=0}^{M-1} a_n g_{\gamma_n}$$

where x is the time series and α is a weighting coefficient for the M functions g_{γ_n} chosen from a redundant dictionary D . Here the dictionary will contain the set of spatio-temporal templates that represent each putative cluster. While Yger et al. 2018 (SpyKING CIRCUS) estimate D by clustering a subset of spikes using density-based clustering, Pachitariu et al. 2016 (Kilosort) instead initialise D with k-means but then iteratively update the templates making several passes through

¹Not only because manual curation would take too long in the case that hundreds or thousands of neurons could be isolated each day, but also because the speed and the performance of the chosen clustering algorithms scale with inverse of the number of combinations of features i.e. exponentially.

the data to minimise the cost-function shown in Figure 2.1. The pre-clustering approach is likely what allows SpyKING CIRCUS to be very memory efficient relative to Kilosort and facilitates clustering thousands of channels simultaneously. However, without updating the templates it is possible that there will be reduced power to pick up smaller units that heavily overlap with other templates in the time domain as they may not be found in the initial sorting. Moreover, units that drift slowly over time may be split into multiple clusters in the initial step.

MountainSort (Chung et al. 2017) clustering operates on the waveforms of spikes after they have been detected: there is no modelling of the time series itself. The key routine is that each pair of clusters are merged if the density distribution between the centroids of each cluster is unimodal. This is determined using a modified version of a dip test where an index of where the distribution should be split is found as well as the dip statistic itself. If the cluster pair is not merged, it is split at that index and the principal components are recomputed for each cluster individually before the next iteration. Allowing the individual clusters their own principal component space prevents potentially uninformative features from other clusters from dominating the subspace and thus increases the separability of any sub-clusters contained within.

For both Wave_clus and JRClust the main advances are in the decomposition step. Quiroga et al. 2004 (Wave_clus) convolve each waveform with a Haar wavelet and then select the coefficients that have the least unimodal distributions (estimated via a KS-test) as these are the ones more likely to have picked up features which separate different waveforms. This type of manipulation is applicable to many clustering problems but is not heavily specialised for this domain. In contrast the feature selection implemented by Jun et al. 2017b specifically takes advantage of the latest hardware. Jun et al. 2017b first identify the recording site with the highest amplitude for a given waveform. They then project the other waveforms onto a normalised version and a time-shifted version of this waveform. The latter is used because often there is a few sample time delay between the waveforms at neighbouring sites. They found that this feature set showed less variability for each hybrid ground-truth unit compared to PCA.

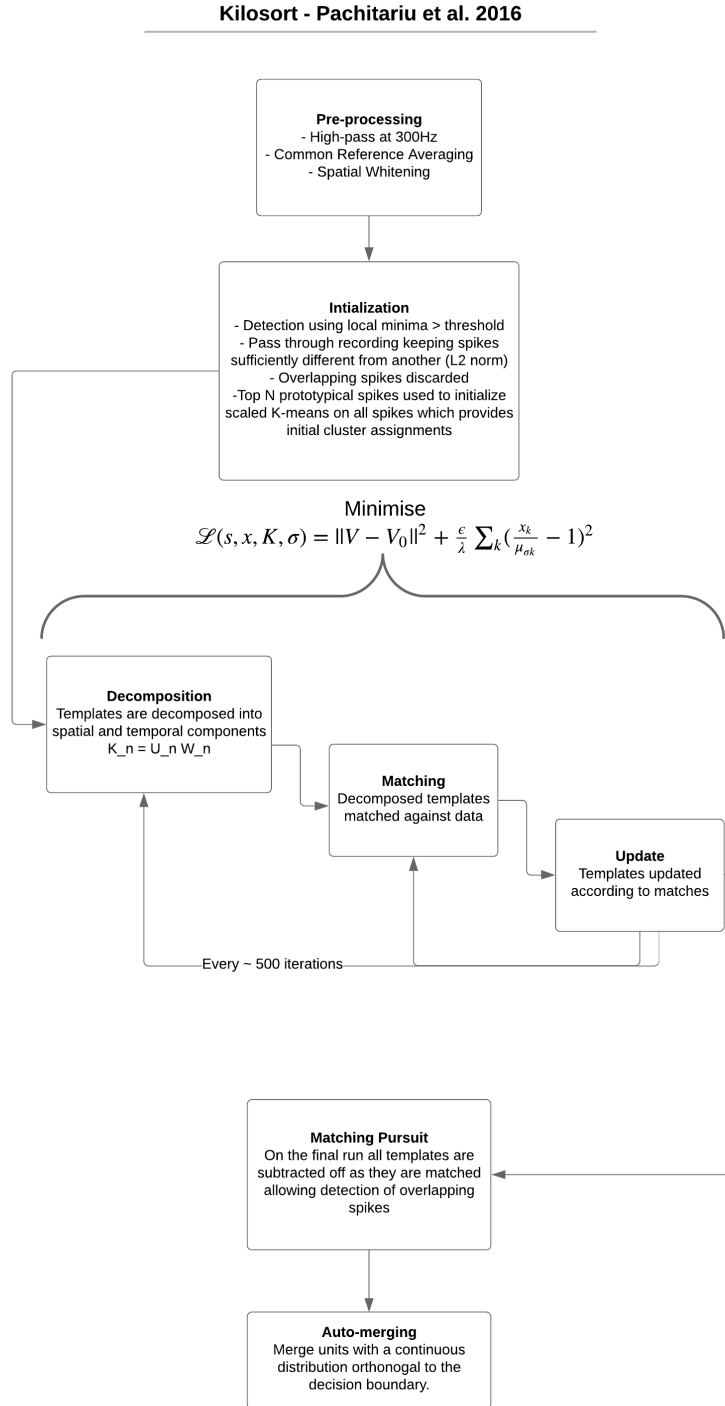


Figure 2.1: Graphical overview of the Kilosort algorithm. s are the spike times, x are the spike amplitudes, K are the waveform templates, and σ are the cluster assignments.

SpyKING CIRCUS - Yger et al. 2018

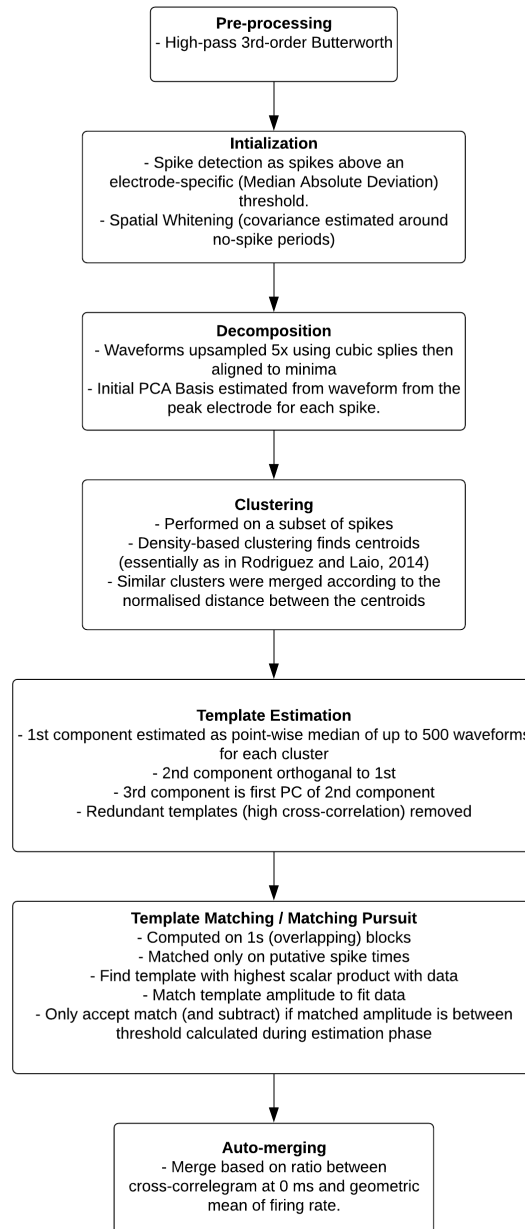


Figure 2.2: Graphical overview of the SpyKING CIRCUS algorithm.

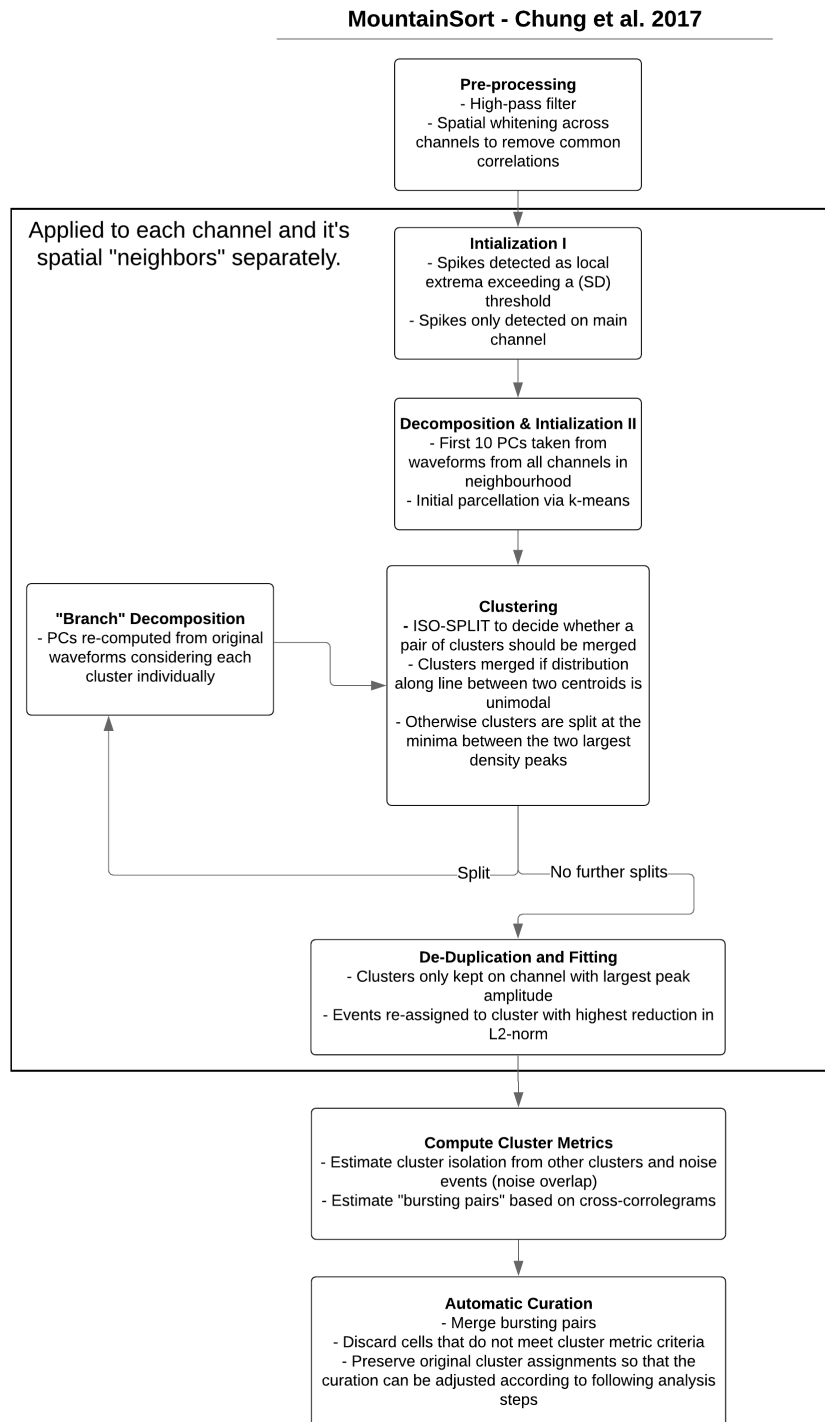


Figure 2.3: Graphical overview of the MountainSort algorithm.

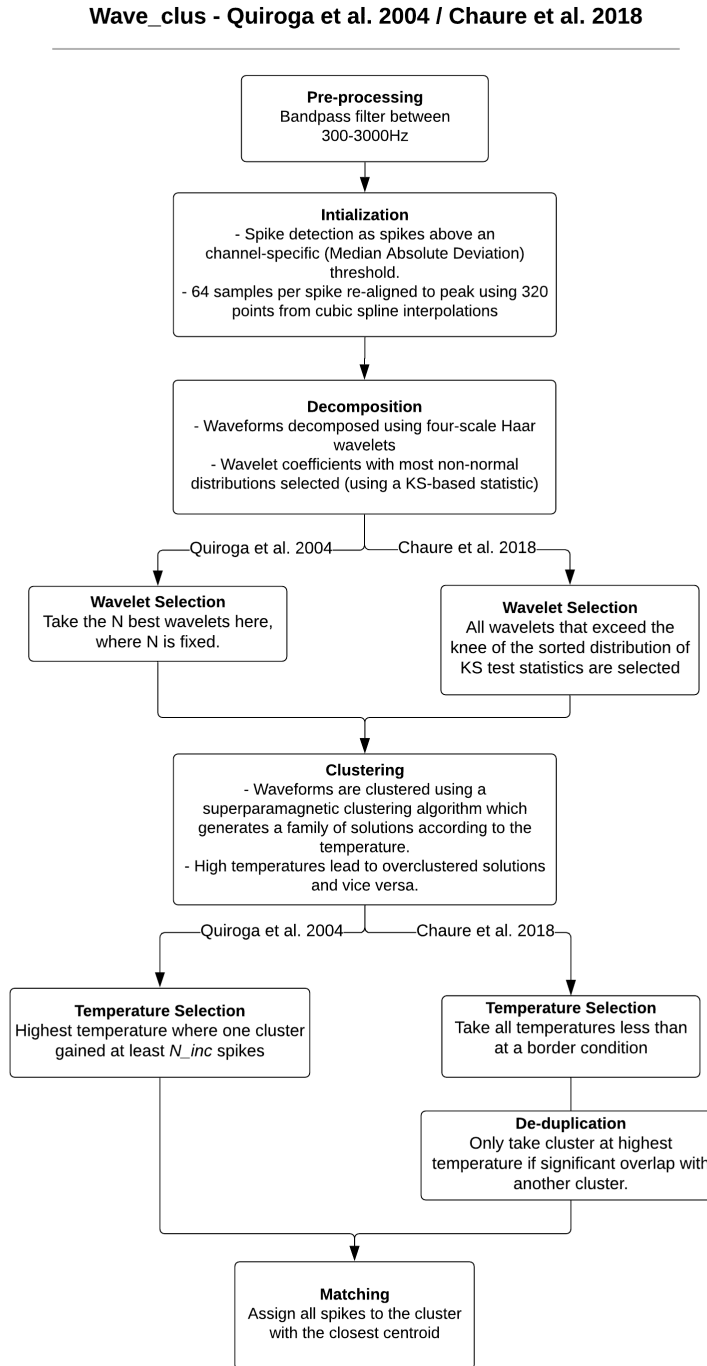


Figure 2.4: Graphical overview of the Wave_clus algorithm.

JRClust - Jun et al. 2018

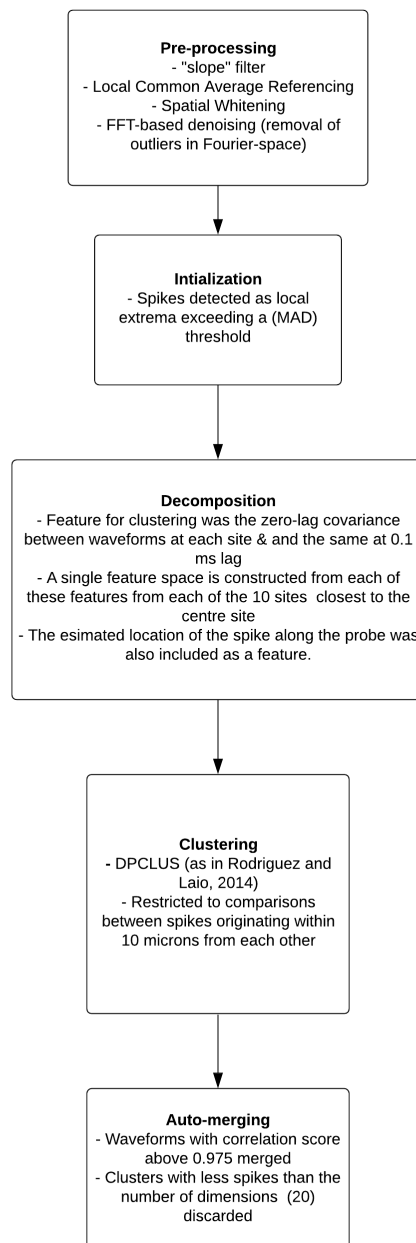


Figure 2.5: Graphical overview of the JRClust algorithm.

2.1.3 Assessing the performance of automatic spike sorting algorithms

Having discussed the theoretical advantages of each sorter what we are most concerned with here is trying to assess their real world performance. Several papers report comparisons on a case-by-case basis, but it is hard to determine the relevance of these for the individual experimenter. There is a lack of standardised measures or datasets. Comparisons are often developed by the author of a particular algorithm and as such there is the possibility of introducing bias, e.g. via suboptimal tuning of rival algorithms. It is not possible to incorporate each algorithm that is newly released into these comparisons.

An ideal systematic assessment of spike sorting algorithms should thus 1) choose some standard measures based on community input and allow these measures to be updated in the future, 2) collate all existing ground truth datasets and simulations into a common format, and 3) allow benchmarks to be re-run when new, or new versions of, software are released. However, in order to implement such an assessment there are some infrastructure challenges that need to be addressed.

Resources Electrophysiological datasets can be large, on the order of terabytes, and running spike sorting can take a long time. How can we ensure that we only run the sorting when we need to (i.e. when a change has been made) but verify that the results we have are correct? This needs to work across multiple machines with different resource-types such as those with fast storage and those with GPUs.

Software Delivery How can each sorter be reliably automatically installed, including installing multiple sorters in a single environment for comparison? How can the software be updated to reflect new releases?

Data Format Each sorter requires data and metadata encoded in a variety of formats. How can we enable users to continue using

their preferred data formats while allowing access to all the software?

In order to tackle these challenges we applied principles from current best practices in software development and applied them to develop *mountaintools*², a general framework for benchmarking computationally intensive algorithms, and *SpikeForest*³, a collection of wrappers and comparison tools for a range of spike sorting algorithms. This work was undertaken in collaboration with Jeremy Magland, who led the project, and others at the Flatiron Institute. These were then applied to build a publicly accessible interactive web interface⁴ for comparing spike-sorter performance as well as to find the best performing sorters for the data we are acquiring in this lab.

²<https://github.com/flatironinstitute/mountaintools>

³<https://github.com/flatironinstitute/spikeforest>

⁴<http://spikeforest.flatironinstitute.org>

2.2 Methods

2.2.1 mountaintools

The core principle of mountaintools is to provide a separation of concerns between data storage, processing routines and compute resources. There is little need for this separation in the simple case where one analysis only need be run once on the same computer. However, when working with big data, computationally intensive tasks, complex processing pipelines, or a combination of all three then it is advantageous to be able to handle these different aspects separately.

Data Storage

The first layer of mountaintools is a Content Addressable Storage system (CAS). Generally, data are stored as files that can be accessed by knowing the path to the data such as a file path or URL path. Files are accessed by requesting the data (either from the filesystem directly or via a network protocol) at a particular location, and whatever is in that location is returned. This is efficient to implement, especially for data that changes often, because the address of the data need not change. Perhaps more importantly, it also fits well with our mental model of how data is stored (e.g. like it would be in a library or in a office filing system). With CAS we address the data by its content directly, in mountaintools we address it by its SHA1-hash⁵. This has a different set of advantages. Firstly, the file can be referred to in the same way no matter where it is located (location independence), or on however many devices. Secondly, we can be certain⁶ that the data we retrieved is exactly the data we asked for, as any mutations of the data would change the hash (content integrity). Thirdly, we can safely ensure that there is only ever one copy of the data on a given machine (single-instance storage). Finally, CAS can

⁵Tangentially, this is also how recall occurs in biological systems. Recall isn't triggered by looking up an address, but directly by some external content. One fundamental difference is that this system, by design, is not tolerant to even the tiniest possible change in the data, whereas successful memory recall is.

⁶The chance of a random collision (where the same hash points to different data) p for n files is approximately $\frac{n(n-1)}{2^{b+1}}$ where b is 160 in the case of SHA1.

be very efficient where the data are relatively static (fast random access), which should be the case for scientific data written to disk.

In practical terms that means that instead of using local file paths such as `"/home/alexmorley/mouse1/raw.dat"` or web URLs such as `"alexmorley.me/mouse1/raw.dat"` we use SHA1 URLs like `"sha1://10fe3f8c67bf5432e9266831ab7d64d19c9e04cc/mouse1/raw.dat"`. Alternatively we can use the python client like so `"dict(study='some-unique-id', experiment='mouse1', data='raw')"`. We cannot use file paths because they are not portable across computers. Standard URLs allow multiple clients to reach the same server but if the data is moved then the URL will no longer be valid. Moreover, the same data cannot be stored in two different locations with a single index, nor is any validation regarding the integrity of the data provided. KBucket addresses these issues by allowing data to be addressed in any of the previously listed ways.

KBucket is a decentralised network of file-sharing servers that facilitates CAS across a network of computers. The network has a tree structure such that each node (known as a share) has a single parent node (known as a hub) which may or may not have parent nodes of its own (a parent node may have many children however). When you address a file via KBucket it will check your local computer and then a share, and then traverse the tree, stopping as soon as the file is found. Because each hub is aware of the content of all of the shares beneath it and hash look-up can be done in constant time the time complexity is only the order of the depth of the tree (regardless of how many files are stored in total). Moreover the file will end up being transferred via transferred via the fewest hops as possible. If one of the hubs is made accessible via a public IP address then all of its child nodes can be accessed without having to connect to an internal network. This can be very useful when sharing data or compute resources with external collaborators in a research setting. Access permissions are provided for the hubs for security purposes.

Processors

mountaintools also provides a job management framework. In this framework processors are the main unit of abstraction over the use of a piece of software.

Processors can only take input files, output files and parameters as arguments, and the output of a processor can only depend on those things. Their identity is only determined by those fields plus a name, a version number, and optionally a container ID. As such the definition of a processor is serialisable, and thus hashable, so a processor itself can also be indexed by its hash. These constraints also mean that a processor, once executed, represents a unique mapping from a set of input files to a set of output files. This mapping has very small memory footprint (typically just a few kilobytes) as it is just a set of hashes and so can be stored in a local or remote cache. When a processor is run mountaintools will query its hash in the cache. If any of the input files, parameters or the version of the processor have changed since it was last run the hash will not match any in the cache and it will be re-executed. However, if the same processor is run with the same input data and same parameters on any computer with access to the same cache, the output hash will be returned and the associated file can be fetched from KBucket (if not already present locally). This is more powerful than it first seems as it allows us to say that it does not matter where the computation takes place because the files associated with the result can still be accessed regardless. In other words because processors and their inputs can be hashed, the mapping of their hash to the result is accessible, and all files can be accessed by their hash, it is possible to find the result of any process if it has been computed before on any other computer in the network. We are therefore able to address results not only by their content but also by the process that formed them.

The next step is to allow a local computer to send a job to a remote computer to do the computation and then fetch the results. If the software installed on each machine is the same then we can simply send the source code associated with the processor (this is captured when the processor is created) to the remote machine via KBucket. However, as many useful scientific softwares require complex dependencies, this is often not feasible. Instead, one can optionally provide a container ID for the processor which points to a container image which can be either hosted on KBucket or on the web. The remote computer will download the container (if not already present) and then mount the data and required source

code inside. Containers provide an isolated environment for the execution of software but are more performant than virtual machines because they share the host kernel. In this case both computers only need have docker or singularity installed to ensure reproducibility of the processor, both of which are easily installed on all modern operating systems.

Finally, mountaintools also provides handlers for batching and the parallel execution of local and remote jobs. It is also aware of the "order of computation" in the sense that if multiple jobs are submitted and the input file of one matches the output file of another then it will process them in the correct order even across multiple machines.

2.2.2 Automatic curation of sorting results

Automated curation of results from spike sorters is done in three phases: filtering, merging and patching.

Filtering

In the filtering stage, neurons that have too many spikes in their refractory period, or whose waveforms are very similar on all channels, are discarded (Figure 2.6). To compute the refractory ratio F_{AA} we first compute the auto-correlogram for each neuron R_{AA} and then compute the ratio between the number of spikes per bin in the refractory period μ_R over the total number of spikes for all bins μ_n .

$$F_{AA} = \frac{\mu_R}{\mu_n} \quad (2.1)$$

The channel similarity was computed by, for each pair of channels, finding the 98th percentile of the element-wise difference between the mean waveforms and then taking the mean of these values for all pairs.

$$\text{channel_dissimilarity} = \frac{C_2 - C}{2} \sum_{ci=1}^C \sum_{cj=ci+1}^C P_{98}(|wv_{ci} - wv_{cj}|) \quad (2.2)$$

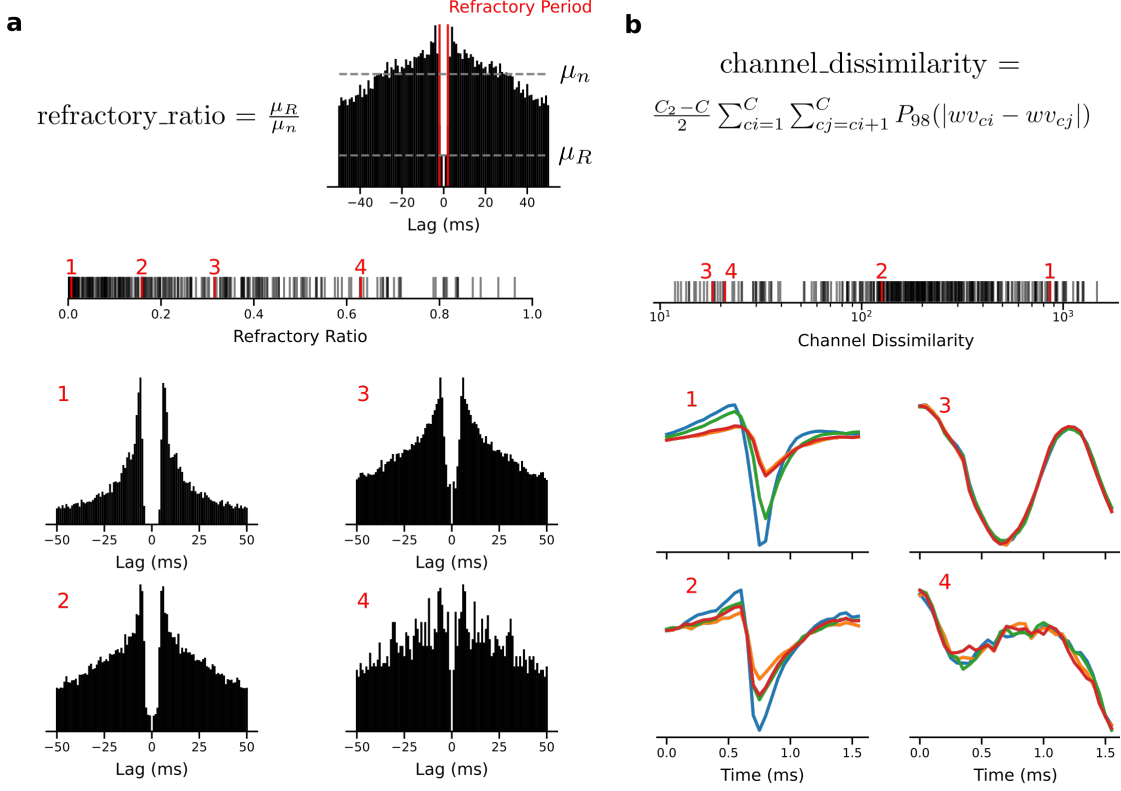


Figure 2.6: Filter out units that contain mostly artifactual spikes. (a) Refractory ratio score. Example unit 1 has a good score, unit 2 has an okay score and units 3 and 4 were rejected. Filled bar plots are auto-correlograms. (b) Same as (a) but for the channel dissimilarity metric. Traces are waveforms with each channel from a given tetrode in a different colour.

where C is the number of channels, P_{98} indicates the 98th percentile, and $wv_{ci} - wv_{cj}$ is an element-wise subtraction for each value in the mean waveform between two channels.

Identifying Merge Candidates

Each pair of neurons is considered as a merge candidate if 1) it exceeds the clean cross-correlation threshold and 2) a combination of firing pattern, amplitude correlation similarity, and clean cross-correlation scores exceed another threshold (Figure 2.7).

Clean cross-correlations are defined relative to the best "parent" auto-correlogram.

$$\frac{1 - F_{AB}}{1 - \min(F_{AA}, F_{BB})} \quad (2.3)$$

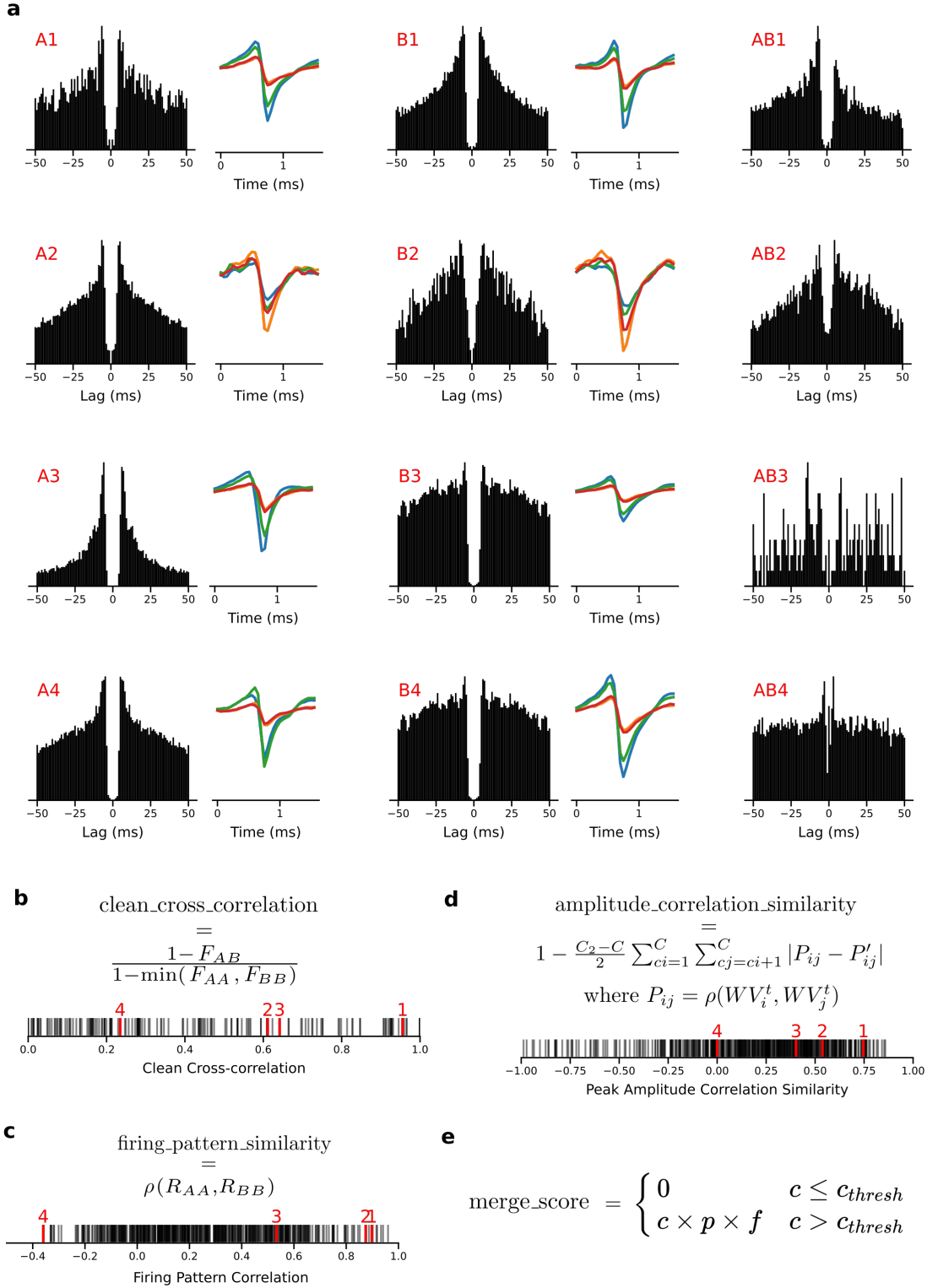


Figure 2.7: Merge units that probably originate from the same neuron. (a) Example pair 1 has a good merge score, example pair two 2 has an okay score and unit pairs 3 and 4 were not merged. (b-d) Show scores for each metric for each of the four pairs relative to the other units recorded on the same day. (e) Calculation of the final merge score where c is the clean cross-correlation score, f is the firing patterns similarity, and p is the peak amplitude correlation similarity.

where F_{AB} is the refractory_ratio for the cross-correlogram between units A and B .

The firing pattern score is simply the correlation between the two auto-correlograms of units A and B .

$$\text{firing_pattern_score} = \rho(R_{AA}, R_{BB}) \quad (2.4)$$

The amplitude correlation similarity is the mean difference in upper triangle of the correlation matrix that is formed from the correlation between the peak value from each channel for every spike in that unit.

$$\text{amplitude_correlation_similarity} = 1 - \frac{C_2 - C}{2} \sum_{i=1}^C \sum_{j=i+1}^C |P_{ij} - P'_{ij}| \quad (2.5)$$

where C is the total number of channels and P' is the same as P but for the alternate unit of the pair and $P_{ij} = \rho(WV_i^t, WV_j^t)$. WV is each of the individual waveforms and WV^t is the value of the individual waveforms at time t where t is the index of the peak on the channel with the largest mean amplitude.

Merging

Merge candidacy is computed for all pairs of cells before any merging actually takes place. In order to merge them we first need to cluster the relevant pairs. However where the original results were highly over split it is possible that a single spurious merge candidate score could result in the merging of two otherwise good sets of merges (see Figure 2.8a). This scenario is detected by re-computing the refractory violations for all of the spikes pooled together. If the refractory violations of the merged cluster is higher than the median refractory violation for all of the sub units then we try to split it. This is done by first finding the "corner unit" which is the unit featured in the most pairs (in graphical terms it is the node with the highest degree). The cluster is then split into those nodes that are neighbours with the corner unit (i.e. were identified as merge candidates with that unit) and those that were not (Figure 2.8b). This procedure is carried recursively until the refractory criteria are satisfied. If all units are connected to the first corner unit

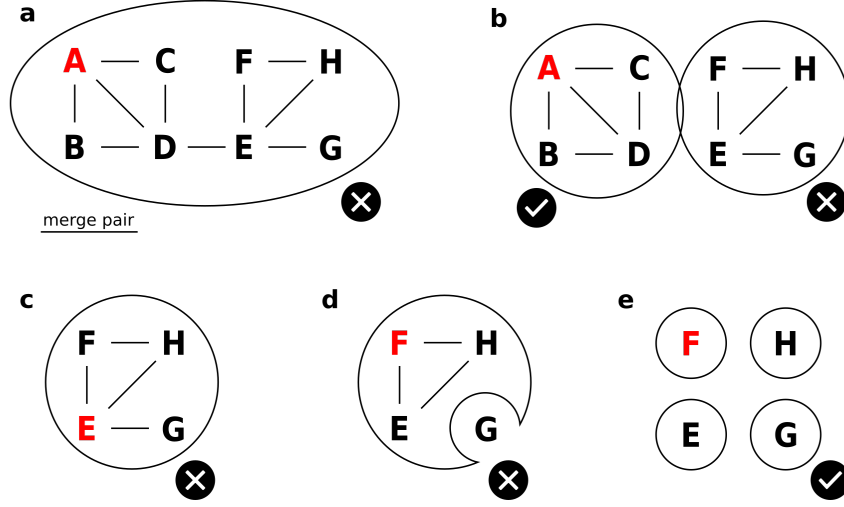


Figure 2.8: Merging clusters of merge candidates. (a) Spurious merge score between D and E results in over merging between two groups. (b) A is identified as the corner unit, and graph is split into nodes that are neighbours with A and nodes that are not. If either sub-cluster also has a poor refractory score the procedure is applied recursively. In (c) no change occurs as the corner unit is connected to all other nodes. (d) F is selected as the next corner unit and G is split off. (e) If no successful corner unit is found then the algorithm degrades to keeping all units separate.

chosen then the node with the next highest degree is chosen instead (Figure 2.8c,d). In the case that none of the corner unit splits are effective each sub-unit will end up being kept un-merged (Figure 2.8e).

2.2.3 Validating and comparing results

Confusion Matrices

Entries in the confusion matrices (unit agreement) were computed as the fraction of shared spikes for each pair of units across sortings. Shared spikes had to be within δ_t samples of each other, which we set to 20 (corresponding to 1ms on our 20 kHz recordings). The confusion matrices were permuted to maximise the diagonal using the Hungarian algorithm (implemented in scipy as the function `optimize.linear_sum_assignment`). Pairs were considered to be matched if they lay on the diagonal of the confusion matrix and had more than 50% shared spikes.

2.3 Results

2.3.1 SpikeForest

mountaintools provides the framework for organising the execution of arbitrary software on multiple datasets. SpikeForest applies this framework to the specific task of spike sorting and exposes an easy interface to explore the results. It wraps 12 different spike sorting algorithms as mountaintools processors. In order to allow all of these sorters to be applied to any data format we use spikeextractors, an interface designed by the SpikeInterface project which gives a common API for loading, manipulating and writing various raw recording and sorting result formats. SpikeForest also hosts a collection of validation datasets on which every sorter is executed, the results of which are published continuously on the internet. These comparisons can be explored using widgets, built using javascript and d3.js. The same set of widgets can also be composed to provide an interface for the manual curation of spike sorting results.

2.3.2 Comparison between sorters without ground truth

Even without knowing the ground-truth for neuron and spike identity it is still possible to compare the performance of automatic and manual sorters. For example we can compare the properties of those units that were matched to those that were not. Automatically identified units were matched with manually identified ones by first maximising the diagonal of their confusion matrices constructed from the agreement rates between each pair (Figure 2.9). The pairs on the diagonal are then considered matches if they exceed a predefined threshold, here we used 0.4.

For both manual and automatic sorters, those units that were not matched tended to have a lower firing rate, lower signal to noise ratio and a higher refractory ratio (Figure 2.10a). This follows from more difficult to identify units being less like to be found by both sorters. For those units that were matched the properties were strongly correlated between the automatic and manual sorters, except for the refractory ratio. This is likely because the latter is determined by very few spikes. However, in general the manually sorted units had a slightly higher SNR, and a

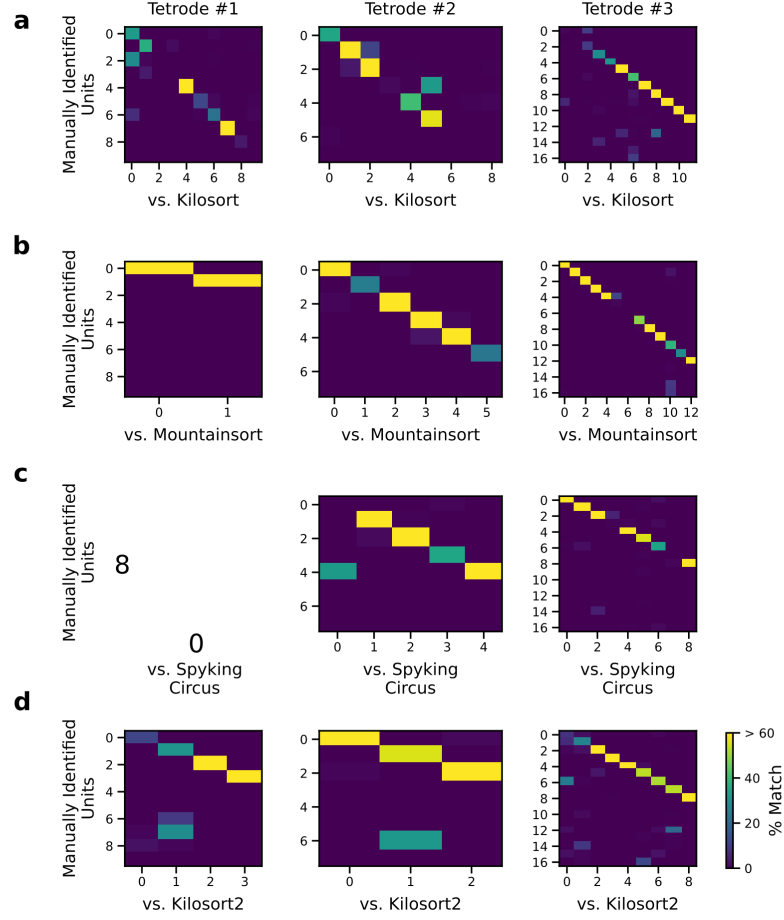


Figure 2.9: Confusion matrices showing the agreement between each automatic sorter and manual results. Z-axis indicates the percentage of units that were matched in both spike-timing and identity between sorters. The matrices are sorted to maximise the diagonal.

slightly lower rate. Because this was not explained by differences in the refractory period it is likely that it is a result of noise being manually removed from clusters at the expense of real spikes. Whether or not this is a reasonable tradeoff and whether in practice the differences would be great enough to have a significant effect on downstream analysis is a question for future investigation.

While we are unable to determine accuracy without ground-truth, we can estimate the false negative rate of each automatic sorter by assuming most of the manually identified clusters are indeed individual units. In practice, given the high correlation between automatic and manual sorting for most metrics in matched units, the false negative rate is both meaningful and important for the experimenter.

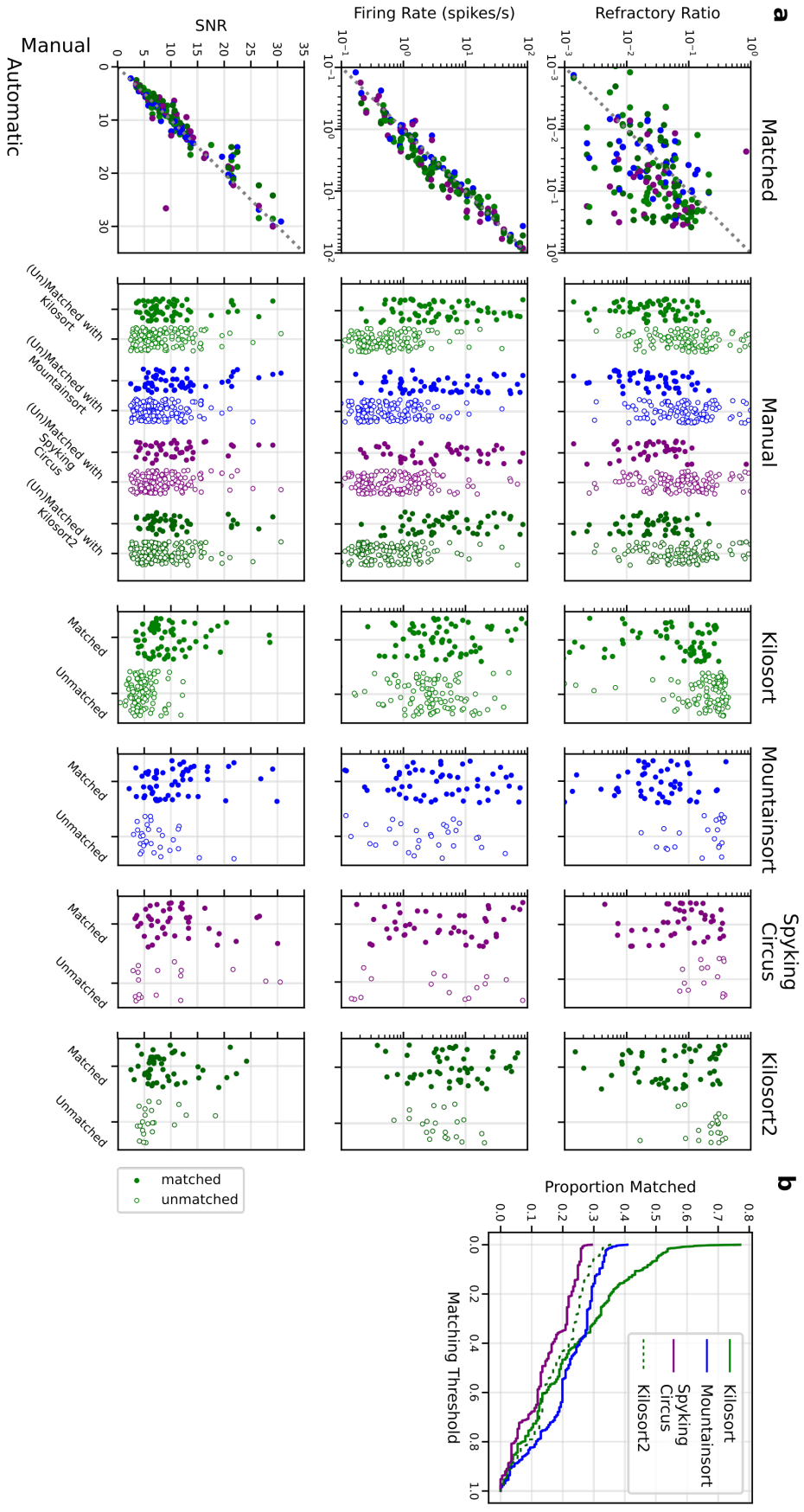


Figure 2.10: (a) Comparison of matched and unmatched clusters between manual and each automatic sorting algorithm. The far left column shows, for each matched unit, a comparison between the value of a particular metric computed on the manual versus the automatic sorter. The next column from the left shows each metric for all of the manually identified neurons, divided up by whether they were matched with a particular sorter. The following columns represent the inverse: it shows the metric for each of the automatically identified units split by whether they were matched in the manual sorting or not. **(b)** The correspondence, for each automatic sorter, of the proportion matched to the manual sorter versus the agreement threshold that needed to be met.

We found that for low agreement thresholds (less than 0.4) Kilosort matched the highest proportion of manually identified units, giving it the lowest false negative rates. At high agreement thresholds (around 0.7) MountainSort matched the highest proportion. This broadly matches our expectations that Kilosort is better able to match low amplitude units (that may be missed by other automatic approaches that rely only on an initial thresholding for spike detection) but that the iterative template optimisation is prone to over-splitting the underlying clusters.

2.4 Discussion

2.4.1 Alternative sorting metrics

There are several other metrics not shown here that we could have used to assess sorting quality, each with varying utility based on the goal of the assessment, namely: evaluation, comparison or validation. To evaluate the performance of a sorter given some estimate of ground truth (e.g. paired patch and extracellular recordings) then we can compute its accuracy and compare that to an empirical estimate of the upper-bound accuracy given by a supervised non-linear classifier. By doing so we can contextualise the accuracy value by how successful the sorter was compared to an optimal approach. This approach was introduced by Harris et al. 2000 as the "Best Ellipsoidal Error Rate" however as bursting spikes and drift violate its key assumption that all spikes of a given neuron can be separated from others using an ellipsoidal boundary in feature space it does not represent a true upper bound on accuracy.

However we cannot always do comparisons between sorters in settings with ground truth information as these recordings do not always have the same properties of other recordings in terms of the noise statistics, electrode design, and the number of neurons to be sorted. In these settings another metric we can consider for benchmarking sorting algorithms is their stability. The stability of a spike sorter under permutations that are similar to those found in the same dataset should correlate with how reliably the sorter correctly identifies unique events. Barnett et al. 2015 provide a novel method for assessing stability by "blurring" (adding noise to each identified cluster using only the noise statistics of that cluster) clusters before resorting the dataset. This allows them to compare the stability of different algorithms even where some are deterministic.

This approach can also be used in validating a sorting run. This process of validation is likely to be the one most commonly performed by an experimenter as they check that the results from their sorting process is credible. Another group of metrics that are often used for validation are cluster isolation metrics. Harris et al.

2001 define the isolation distance for a given unit as the radius for the smallest ellipsoid from the cluster center that contains all of the spikes from that unit and the same number of spikes not from that unit. Jun et al. 2017a use a different isolation metric which is based on the fraction of nearest neighbours for each spike from one unit that are classified as from the same unit. Isolation metrics, in combination with the spike timing metrics discussed earlier, can thus be useful for the experimenter to decide whether or not to include a particular unit in their analysis. However they are not suitable for use in evaluation or comparison between spike sortings as they will likely be biased in favour of sorters whose clustering algorithms make similar statistical assumptions about what defines a cluster.

2.4.2 Justification for our approaches to curation

Our goal for automatic curation was not only to be as accurate as possible but also for the process to be interpretable and easy to inspect. Each step, and each decision within that step, is accompanied by a visualisation that makes it easy to check for any mistakes or mis-set parameters. This focus on the transparency of the process also influenced our decision not to use a greedy algorithm for the merge step which avoids the joined sub-graph problem (Figure 2.8) by merging pairs immediately and then treating that unit as a whole. Note that using a greedy algorithm does not really solve the problem: it instead avoids it by making the process dependent on which order the pairs are merged in (although choosing the order such that the best pairs are merged first may help).

2.4.3 Human factors affected by automatic versus manual spike sorting

We should take advantage of automatic spike sorting algorithms that outperform approaches relying heavily on manual curation. However, we should also consider whether there are hidden costs involved, and whether these can be mitigated. For example, the act of cluster-cutting often results in the operator becoming familiar with the data in its raw format. We believe this kind of familiarity with the data in its unprocessed form is invaluable. Although automatic spike sorting does not

preclude this, it does make it easier for this step to be skipped. We suggest that it is vital that we nonetheless continue to take the time to familiarise ourselves with the raw data, and maintain the tools that allow us to do so.

2.4.4 Impact of the software-first approach

Other costs and benefits come not directly from using automatic spike sorting, but from the software-first approach we took to it. We considered the spike-sorters as black-boxes that transform a time series into a series of time stamps and labels, an approach also taken in Barnett et al. 2015. This simplification allowed us to cover a much wider range of algorithms than would have been possible otherwise and we nonetheless kept the ability to parameterise each of the sorters. In fact, one of the underexplored opportunities in this project so far is the possibility of doing a large-scale search through all of the parameters of each of the sorters in order to determine what is optimal for what types of datasets.

While we allow future users to avoid the trials of installing and testing multiple pieces of software themselves we have also introduced a soft requirement to understand another layer of tooling. The current priority for this project is finding a balance between exposing the power of these frameworks while making it as easy as possible to start. The web framework is one aspect of this process, the other is searching for ways in which we can integrate with tools that people are already comfortable with. For example, because many people in this lab use Jupyter Notebooks, we provide template notebooks for sorting data using SpikeForest.

By treating sorters as black-boxes we also have missed out on some opportunities for swapping out parts of different pipelines. For example, in this framework as it stands it would be hard to test whether the wavelet PCA used in Quiroga et al. 2004 would improve the performance of the other spike sorters if used in place of their dimensionality-reduction techniques. However we are able to apply, for example, a drift-tracking approach which involved applying sorters to overlapping segments of the recordings and then matching the units between these segments. So while we limit the types of modifications in approach that can easily be tested

when treating spike sorters as black boxes those modifications that can be tested are easily applied to many algorithms with no additional effort.

The key outcome of this chapter as it relates to the rest of the thesis was that we were able to find an automatic sorting algorithm (Kilosort) that, combined with our automated curation process, performed almost at human-level. This confirms anecdotal evidence from other labs that Kilosort has the lowest false negative rate. Moreover we have mitigated most of the false positive and false split events via our automated curation steps. Importantly we saw good performance over multiple regions with very different waveforms and firing patterns. Next we describe how we recorded from five regions in a freely-moving mouse performing our discrimination task.

3

Experimental Design

Contents

3.1	Introduction	40
3.2	Methods	43
3.2.1	Recording procedures	43
3.2.2	Assessment of behaviour	47
3.2.3	Speed Profile Matching	51
3.3	Results	53
3.3.1	Anatomical verification of recording sites	53
3.3.2	Behavioural results	53
3.4	Discussion	58
3.4.1	Task design	58
3.4.2	Controlling for the effects of behaviour	59
3.4.3	Recording technologies	60

We like to think we have
constrained an organism to the task
we have set. In practice, our
paradigms are constrained by the
way the organisms respond.

McNaughton N. et al. 2004

3.1 Introduction

Choosing optimal behaviour using information from real world stimuli often requires resolving conflicting information, disambiguating overlapping stimuli and using past experience to guide future decisions. While the simplest, realistic decisions that can be made often boil down to a binary choice, "Should I do this, or not?", they can still require integrating information across multiple sensory modalities. Moreover, the same stimulus can drive different behaviours in different environmental contexts or during different internal states. This means that it may not be efficient to assign valence or salience to individual stimuli themselves, but instead to their representation as a whole, so that the correct information can be retrieved without interference when that combination of stimuli is encountered in the future.

Modern neuroscience, moving beyond the localisation of function discussed in chapter 1, focuses more on the role of "circuit motifs" or patterns of interactions between pairs of regions. Thus while our view that the amygdala likely underpins valence coding in the rodent to a large extent we no longer consider its action in isolation (Redondo et al. 2014). For example, the sign of the valence signalled by amygdalar outputs might be divergent depending on the target of specific projections. Beyeler et al. 2016 found that neurons that project locally to the central amygdala were driven by the CS predicting quinine (an aversive stimulus) and those that project to the NAc were driven by the the CS predicting sucrose. Yet another projection from the BLA to the PFC might be used when under conditions of conflict (Bloodgood et al. 2018). Specifically, ArchT-silencing of BLA projections to prefrontal cortex suggest these projections are required for resolving cues predicting aversive stimuli as well as conflicting cues, but not those that predict reward.

A separate set of circuit motifs in the form of subcortical dopaminergic projections to other subcortical, cortical and neocortical structures (Wise 2004) may allow the selection of which experiences to store on a longer timescale. For example, optogenetic activation of dopamine afferents from the VTA can bi-directionally modulate the strength of the Schaffer collateral synapses in CA1 (Rosen et al.

2015). The same projection also promotes the reactivation, and persistence, of hippocampal spatial representations (McNamara et al. 2014) and optogenetic activation of projections from the VTA to the PFC increases future responses to conditioned auditory stimuli and their later recall (Popescu et al. 2016). The link between reactivation of patterns in the hippocampus and their persistence might be explained by their consolidation via interactions between prefrontal cortex and CA1 during sharp wave ripples. Prefrontal ensembles are reactivated in co-ordination with hippocampal sharp-wave ripples in both the resting (Peyrache et al. 2009) and awake state (Jadhav et al. 2016).

One approach to synthesising findings about the heterogeneous and overlapping circuits and sub-circuits that drive complex behaviour is to try and partition the system into subsystems that have a reduced set of interactions. McNaughton N. et al. 2004 and Gray 1970 define three systems whose interactions can explain behaviours that involve choosing whether to take an action or not: the Behavioural Approach System (BAS) the Fight/Freezing/Flight System (FFFS) and the Behavioural Inhibition System (BIS). McNaughton N. et al. 2004 assign different regions or circuits to different subsystems based on experimental evidence. The NAc and the VTA would be part of the BAS, for example, while the CA1 is activated during conflict and thus would be part of the BIS. However, even these three simple systems overlap almost fully. The BLA is required for processing conflicting and aversive cues (Burgos-Robles et al. 2017) as well as long-established role in fear memory (LeDoux 1998). This would put it in all three systems.

Thus far our approach for understanding links between the brain and behaviour has been largely composed of studying the responses of single regions or simple circuit motifs in unidimensional behavioural paradigms and under the influence of experimental manipulations such as pharmacology or optogenetics. However it is hard to piece together the fractured knowledge to form testable theories without a better understanding of the system's broader dynamics. We propose to move forward with an integrative model which contains many of these motifs simultaneously. In order to do so experimentally, we need a behavioural paradigm

that is higher-dimensional but still constrained enough to be tractable, as well as the ability to record simultaneously from enough neurons spanning enough circuits that we can observe the interactions between these motifs.

3.2 Methods

3.2.1 Recording procedures

Subjects and surgical procedures

All experiments involving animals were conducted according to the UK Animals (Scientific Procedures) Act 1986 under personal and project licenses issued by the Home Office following ethical review. All animals used in this thesis were adult male wild-type C57Bl6/J mice (between 5-7 months old at the time of implantation) bred in-house. Animals were housed with their litter-mates until used in the procedure with free access to food and water in a dedicated housing room with a 12/12h light/dark cycle. Behavioral experiments were performed during daylight. On the day of the surgery and from then on they were single housed. Food restriction was always commenced at least one week after surgery, mice under food restriction were maintained between 85 and 95% of post-surgery body weight with a target of 92%. Mice were weighed at least daily and the provided food adjusted accordingly; during experimental periods mice were always fed at the end of the day.

Surgical procedures were performed under deep anaesthesia using isoflurane (0.5-2 %) and oxygen (2 l/min). Analgesia was also provided (buprenorphine, 0.1 mg/kg). All mice were implanted with a custom-made microdrive, constructed as described previously (Dupret et al. 2010; McNamara et al. 2014; O'Neill et al. 2008) with the addition of either two or four additional tetrodes resulting in 12 or 14 tetrodes per mouse respectively. Tetrodes were constructed by twisting together four insulated tungsten wires (12 μ m diameter, California Fine Wire) and briefly heating them to bind them together in a single bundle. To enable their independent movement each tetrode was attached to a M1.4 screw and all tetrodes were assembled in a microdrive prior to implantation. The drive was implanted under stereotaxic control. Co-ordinates are shown in Figure 3.1.

Following the implantation of tetrodes into the brain the exposed parts of the craniotomies were covered with paraffin wax, after which the drive was secured to the skull using dental cement. For extra stability, four stainless-steel anchor

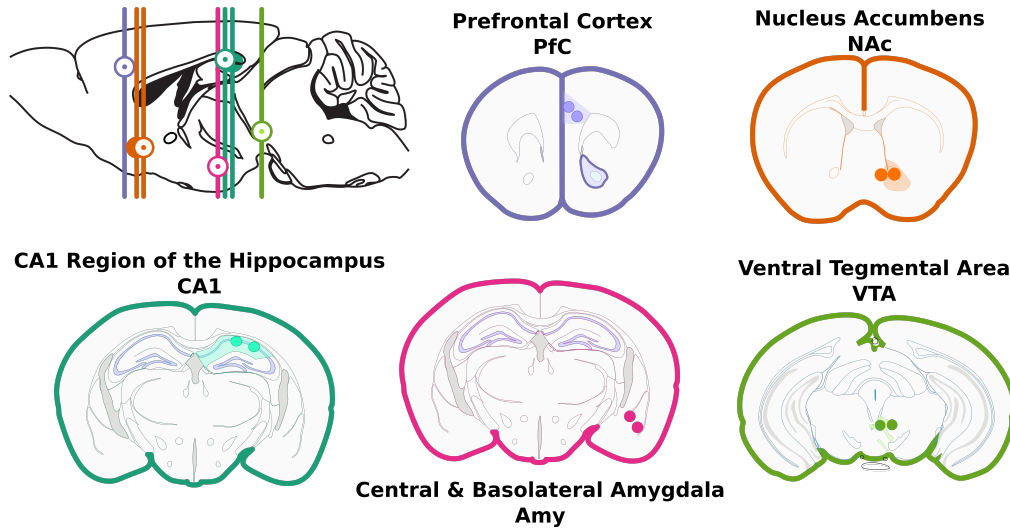


Figure 3.1: Indicative target locations for each tetrode. Stereotaxic coordinates for each region can be found in Table 3.1

Region	AP	ML	DV
CA1	-2.0	+1.8	-1.1
	-2.0	+1.4	-1.1
Amy	-1.3	+3	-4.5
	-1.65	+3.2	-4.5
NAc	+0.93	+0.8	-4.0
	+1.28	+1.0	-4.0
	+0.93	+1.2	-4.0
PFC	+1.27	+0.2	-2.0
	+1.63	+0.4	-2.0
	+1.97	+0.2	-2.0
VTA	-3.4	+0.6	-4.0
	-3.05	+0.4	-4.2

Table 3.1: Stereotaxic coordinates for tetrode implantation. Distances are in mm.

screws had first been inserted into the skull. Two of the anchor screws, which were inserted above the cerebellum, were attached to 50 μ m tungsten wires (California Fine Wire) and served as ground electrodes during the recordings. Including the dental cement the mass of the implant was approximately 2 g.

Experimental Timeline

After a recovery period of at least one week following the surgical procedures, mice were handled and familiarised with the experimentalist as well as the experimental surrounding. For approximately one week, the animals were connected to the

recording system for 1-2 hours daily and were able to move freely in an open-field enclosure. During this time we monitored the electrophysiological profile of the local field potential in the hippocampal ripple frequency band (Csicsvari et al. 1999) in order to lower the CA1 tetrodes to the pyramidal layer. All other tetrodes were lowered during this period in search of maximal multi-unit spiking activity. At the end of the first two full recording days, dCA1 tetrodes were raised in order to spare the pyramidal cell layer unnecessary mechanical damage overnight. The default for the other tetrodes was to lower them approximately 250 microns after each full day of recording. However if we already had a good yield on a tetrode we moved it less, a full table of the relative depths of each tetrode on recording day can be found in Appendix B. We kept track of the depth of each tetrode relative to the implanted depth by recording the number of turns on the screw that controlled the position of each cannula.

Anatomical confirmation of tetrode placement

At the completion of the experiment, all implanted mice were deeply anaesthetised with isoflurane/phenobarbital and transcardially perfused with 0.1 M phosphate-buffered saline (PBS) followed by cold 4 % paraformaldehyde (PFA) dissolved in PBS. Implanted tetrodes were kept in the brain for at least 4 days of post-perfusion fixation in 4 % PFA before being removed and the brain being extracted. The brains were kept in 4% PFA for at least 24 h before slicing and coronal sections (50 μ m thick) were stored in PBS-azide. Sections were prepared for the immunohistochemistry procedures: free-floating sections were rinsed extensively in PBS with 0.25% Triton X-100 (PBS-T) and were blocked for 1-2 h at room temperature in PBS-T with 10% normal donkey serum (NDS). Sections were then incubated overnight at room temperature with primary antibodies (see Table 3.2) diluted in 3% NDS blocking solution. After further washing using PBS, sections were incubated overnight at room temperature in PBS-T containing a mixture of secondary antibodies (also see Table 3.2) diluted in the blocking solution. After further washing in PBS, sections were mounted on slides, coverslipped with Vectashield mounting medium (Vector Laboratories, Cat. No. H-1000) and stored at 4°C.

Antibody Table					
Primary Antibodies			Secondary Antibodies		
Antibody	Dilution	Source	Antibody	Dilution	Source
Rb- α -wfs1	1:500	Proteintech Group Cat# 11558-1-AP, RRID:AB_2216046	Dk- α -Rb Cy3	1:500	Jackson Im- munoResearch Labs Cat# 711-165-152, RRID:AB_2307443
Rb- α -FOXP1	1:200	Abcam Cat# ab16645, RRID:AB_732428	Dk- α -Rb Cy3	1:1000	Jackson Im- munoResearch Labs Cat# 711-165-152, RRID:AB_2307443
G- α -Calretinin	1:500	Swant Cat# CG1, RRID:AB_10000342	Dk- α -G Alexa Fluor 488	1:500	Molecular Probes Cat# A-11055, RRID:AB_142672
Rt- α -Ctip2	1:500	Abcam Cat ab18465, RRID:AB_2064130	Dk- α -Rt Alexa Fluor 488	1:500	Jackson Im- munoResearch Labs Cat# 712-545-153, RRID:AB_2307443
Rb- α -TH	1:1000	Millipore Cat# 11558-1-AP RRID:AB_2216046	Dk- α -Rb Cy3	1:500	Jackson Im- munoResearch Labs Cat# 711-165-152, RRID:AB_2307443

Table 3.2: Abbreviations: α - anti, Rt - Rat, Dk - Donkey, Rb - Rabbit, TH - Tyrosine Hydroxylase

Multichannel Data Acquisition and Position Tracking

The signals from the electrodes were amplified, multiplexed, and digitised using a single integrated circuit located on the head of the animal (RHD2164, Intan Technologies; gain x1000). The amplified and filtered (0.09 Hz to 7.60 kHz) electrophysiological signals were digitised at 20 kHz and saved to disk along with the synchronisation signals from the position tracking and from the software driving the behavioural task stimuli. To track the location of the animal three LED clusters were attached to the circuit casing and captured at 25 frames per second by an overhead colour camera. Synchronisation of the tracking and the electrophysiological signals was done offline.

Spike detection and unit isolation.

Spike sorting was performed using a modified version of Kilosort adapted for tetrode recording (see previous chapter). In order to validate the automatic clustering we

first sorted a subset of recording days manually using the procedure described below.

The recorded signals were digitally band pass filtered (800 Hz to 5 kHz) offline for spike detection and unit isolation. Spikes were detected using a threshold in the RMS power (0.2-ms sliding window) of the filtered signal. Unit isolation was performed on the first principal components calculated for each channel recorded by the tetrode in question. This was achieved by automatic over clustering to up to 100 clusters (KlustaKwik, <http://klusta-team.github.io/>) followed by manual recombination and further isolation informed by cloud shape in principal component space, cross-channel spike waveforms, auto-correlation histograms and cross-correlation histograms (Csicsvari et al. 1999; Harris et al. 2000; McNamara et al. 2014). All sessions recorded on a given day were concatenated and cluster cut together to monitor cells throughout the day. Only well-isolated and stable units over the entire day were used for analysis.

Data Inclusion Criteria

For all analyses only units with mean firing rates between 0.05 and 10 Hz were used. Units from tetrodes that were identified as clearly outside of the target structure were excluded from analysis. Where tetrode position could not be clearly identified all units were included.

For assembly detection (but not tracking) only "off-trial" epochs were used. These are epochs during the same exploration session as the task but outside the trial periods i.e. before the tones were played and at least 20 seconds after the solution had been dispensed. These periods were concatenated together across a full recording day for assembly detection.

3.2.2 Assessment of behaviour

Environment Design

Mice were trained in an open-field box, 45 x 45 cm in size, with 12 cm high-walls. The open-field box was placed inside a larger box (70 x 70 x 200 cm) which was light- and sound-proof. This larger box was dimly lit by white LED strips on the top of the box. The speakers and tracking video camera were also located at

the top of the box, while the recording rig and all other electrical equipment was wired in from the outside. Except when the task tone was playing the speakers would emit a low volume white noise to mask any noise that came from outside the box. The open-field was cleaned daily after use and was only used by one mouse in a given (two week) experimental period.

Within the open-field environment mice were exposed to a range of different stimuli controlled using TTL pulses delivered via an Arduino board. Two LED panels were affixed to walls of the environment which served as visual cues when illuminated. A strip of green LEDs spanned the corner between two walls, and a circular arrangement of orange LEDs were affixed to a different wall. This configuration can be seen in Figure 3.6. A dispenser fitted with an infra-red beam that detected lick events was used to deliver the sucrose or quinine drops. The sucrose or quinine was delivered into the dispenser in a controlled manner via a syringe connected to a stepper motor by custom 3D-printed parts. The drops were automatically removed from the dispenser by opening a solenoid connected to a vacuum pump. The Arduino board controlled the LED panels, the tones and the delivery and removal of drops of the solutions. These control signals were also mirrored to the Intan board as TTL pulses for synchronisation with electrophysiology.

Training Procedure

Mice were food restricted for at least 3 days before training. The training was split into three stages. They were trained in the open field with the LED that would later predict sucrose on. During stage 1 the tone was played and then a sucrose solution was available at the spout for 30 seconds. The tone was a pure tone at 10 kHz. The side of the open field where the spout was not located was also blocked off to increase the chance of the mouse discovering the spout. During stage 2, as the animal learned the association, the full open field was opened for the mouse to explore and the availability duration was lowered to 5 or 10 s. In the final stage the animal was exposed to the task in full as described in the next section.

Task Design

Only one of the two sets of LED panels was active at any given moment during exploration of the enclosure. Every 2-4 minutes a tone would signal the delivery of a solution at the dispenser. The tone was either 5 or 10 s in duration, although this was kept constant within-subject. For each mouse, one set of LEDs predicted that the dispensed solution would be sucrose (15% solution in water) and the other quinine (0.2 mM in water). The solution was available from the end of the tone for either 5 or 10 s and then withdrawn. The LED configuration was reversed after every 1-4 tones ($\approx$ 3-5 minutes).

Analytical approaches

The decision points were defined as the periods immediately preceding commencement of approach or avoidance behaviour and were estimated as follows. First we classified the trials into approach (the animal reached the port before the solution was withdrawn) and avoidance (the animal did not approach the port AND ended up further away from the port when the solution was withdrawn). If neither of these conditions were met the trials were excluded from this analysis. For approach trials, the decision point was taken to be the last time at which the animal had zero velocity towards the port before the arrival at that location. For avoidance trials the decision point was taken to be the first point in which the animal had zero velocity towards the port before the animal began to move away from the port (see Algorithm 1).

Algorithm 1 Decision Point Heuristic Pseudocode

```

1: procedure GETBEHAVIOUR
2:   if mouse_starts_off_at_port then
3:     if mouse_remains_at_port_area then return stayed_at_port
4:   if mouse_went_to_port_during_trial then return approach
5:   else
6:     checkavoid:
7:      $dstart \leftarrow \text{distance\_from\_port\_at\_trial\_start}$ 
8:      $dend \leftarrow \text{distance\_from\_port\_at\_trial\_end}$ 
9:      $dchange \leftarrow dend - dstart$ 
10:    if  $dchange > 0$  then return avoid
11:    elsereturn noaction

```

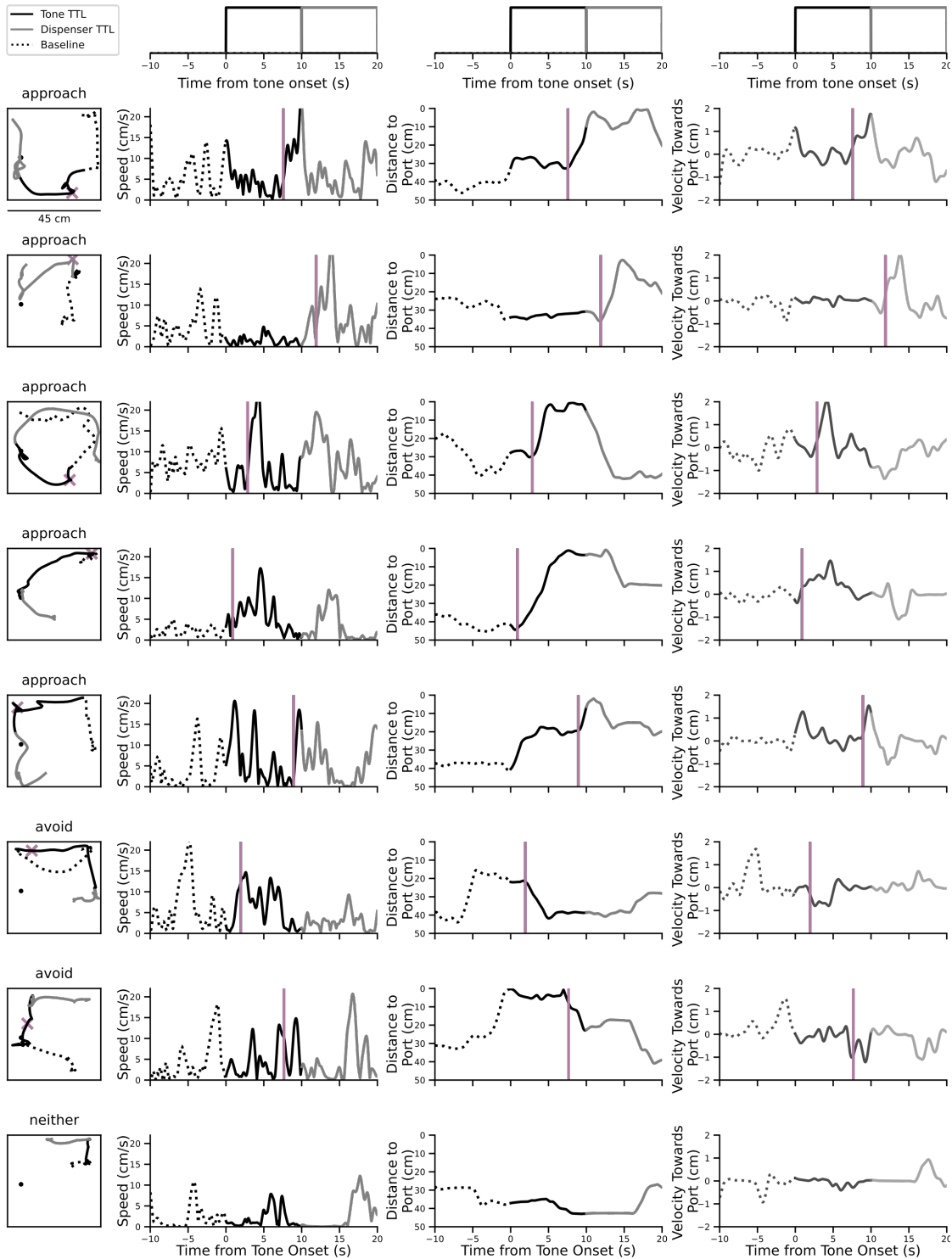


Figure 3.2: Graphical illustration of heuristic for estimating decision points in approach and avoidance behaviour. Each row represents one tone-triggered trial during a single session. In all columns the black lines represent periods during which the tone was being played, the grey lines indicate periods where the solution was available at the dispenser and the dashed lines indicate the period before the tone was played. The first column shows the trajectory of the animal within the enclosure. The purple cross indicates where the decision point was estimated to be and the black dot indicates the location of the dispenser. The class of the behavioural response during the trial (i.e. approach, avoid or neither) is shown above each box. The second, third and fourth columns show the speed, distance to point, and velocity towards the port of the animal during the respective trajectory. Here, the vertical purple lines indicates the estimated decision points.

The mean trajectory vectors during the task were computed as follows (Figure 3.3). First we divided the open-field into a 10 x 10 grid. For each spatial bin we collected all of the trajectories for the animal passing through that bin during the task. We converted each of these sub-trajectories into a vector based on the first and last points (i.e. where the animal entered and exited the bin respectively). We then rotate all of the vectors for each bin such that $\vec{i}$ is in the direction of the dispenser. All vectors from all bins were then binned in polar space to form a histogram as seen in Figure 3.7.

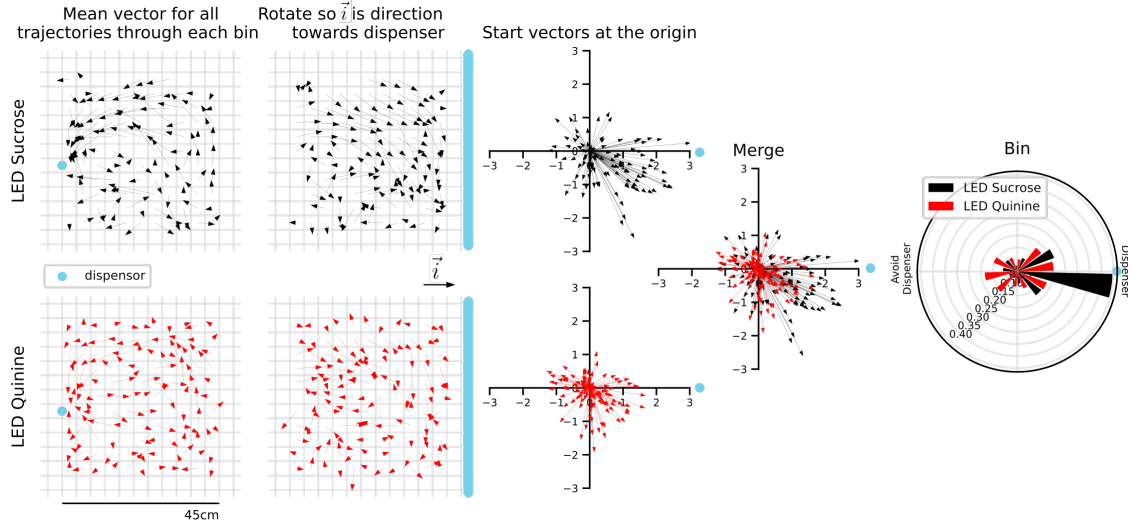


Figure 3.3: Steps for computing mean trajectory vectors. From left to right, each step is applied independently to quinine and sucrose trials. Far left: mean vector for a mouse's trajectory for each spatial bin in the arena. Middle left: the same vectors as the previous step but with each rotated such that the dispenser is at $\vec{i}$. The shape of the dispenser is stretched to maintain the angle between it and each vector. Middle right: The vectors plotted all on the same axis starting from the origin. Far right: A polar histogram where the centre of each bin lies in between the red and black segments.

3.2.3 Speed Profile Matching

To find times where the animal was displaying similar behaviour as it would during the task we took the full speed profile during the task epoch and computed a sliding similarity score as the inverse of the sum of the absolute difference between each timepoint. The tracking data were smoothed with a 1 s kernel prior to computing the score. Matching points were defined as local maxima in the similarity score exceeding 0.7.

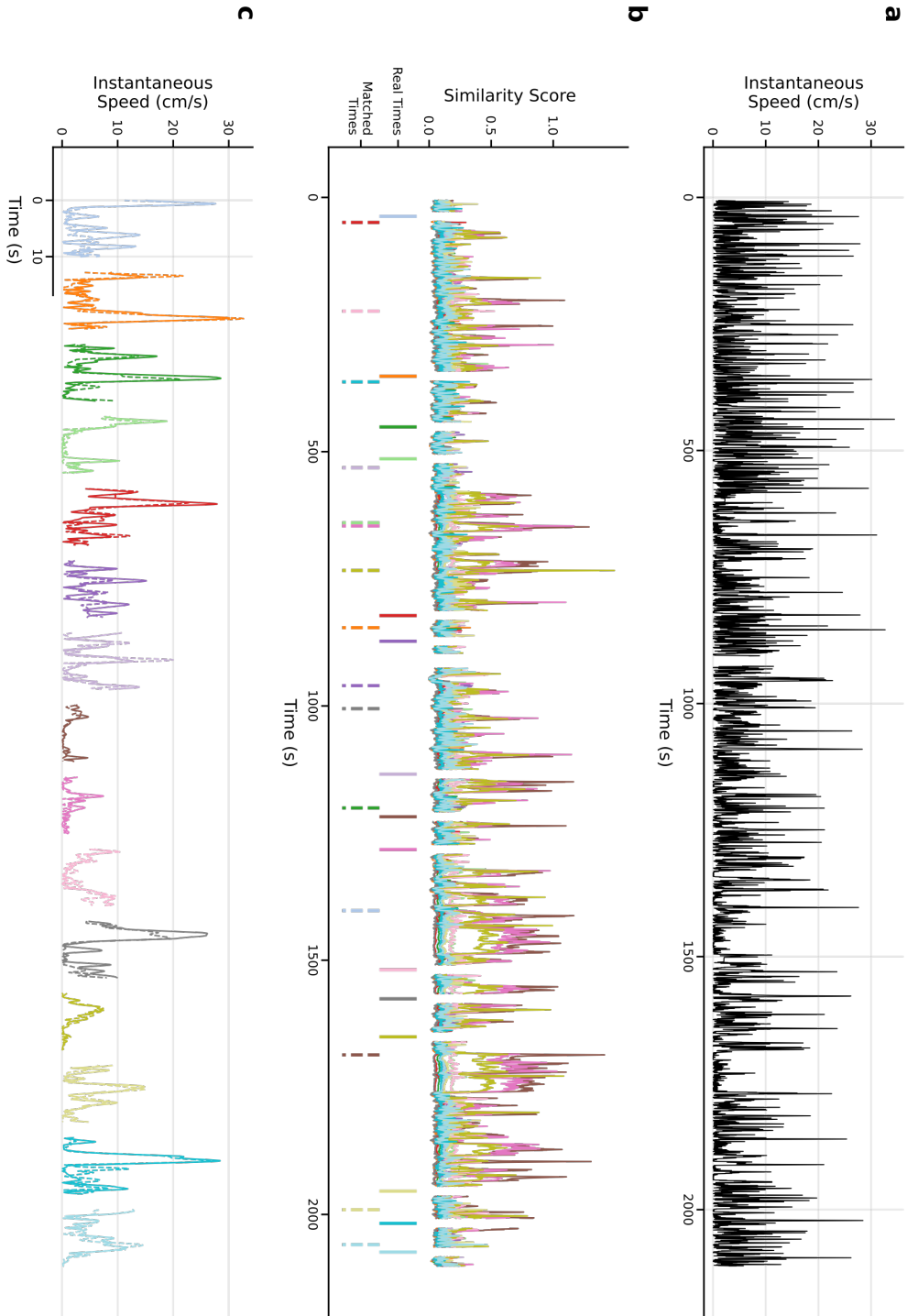


Figure 3.4: Method for finding epochs with matching speed profiles. (a) instantaneous speed over an example session. (b) an epoch around each tone in the session was taken as a speed profile that was then matched across the rest of the session. Each colour represents the similarity score for a different tone. The vertical lines represent the original tone times and the dashed lines represent the closest speed profile matches. (c) For each match in this session we have overlaid the real and surrogate time speed profiles. The colours correspond to the same epochs as above.

3.3 Results

3.3.1 Anatomical verification of recording sites

We were able to record neural ensembles and local field potentials from five regions simultaneously in freely behaving mice. Where possible, tetrode placement was verified post-mortem and images of tracks in each region can be seen in Figure 3.5 and in Figure B.1. We were able to record more than 1500 neurons overall from 8 mice.

Recording Summary					
Total Number of Neurons	1503		Mice	8	
Mean Number of Neurons per day	71.57		Recording Days	21	
By Region					
	CA1	Amy	NAc	PfC	VTA
Total Number of Neurons	376	200	264	301	362
Mean Number of Neurons per day	17.90	9.52	12.57	14.33	17.24

Table 3.3: Summary of recordings

3.3.2 Behavioural results

Mice were trained to infer the meaning of one auditory tone during two transiently-displayed (mutually exclusive) LED contexts, and accordingly opt to approach or avoid a liquid dispenser (Figure 3.6). In this task, mice explored an open-field and learnt that the tone signalled the imminent, but temporally limited, availability of a drop solution at the dispenser. After 10 s the drop was removed by automatically reversing the pressure at the spout which forced the animal to rapidly approach the dispenser to obtain a reward if one was present. The identity of the drop was contingent on the currently active visual context, with one LED display associated with sucrose delivery while the other predicted quinine. A trial was excluded from the analysis if the animal neither approached nor avoided the port (which amounted to 11.8% of trials; see Methods for more details).

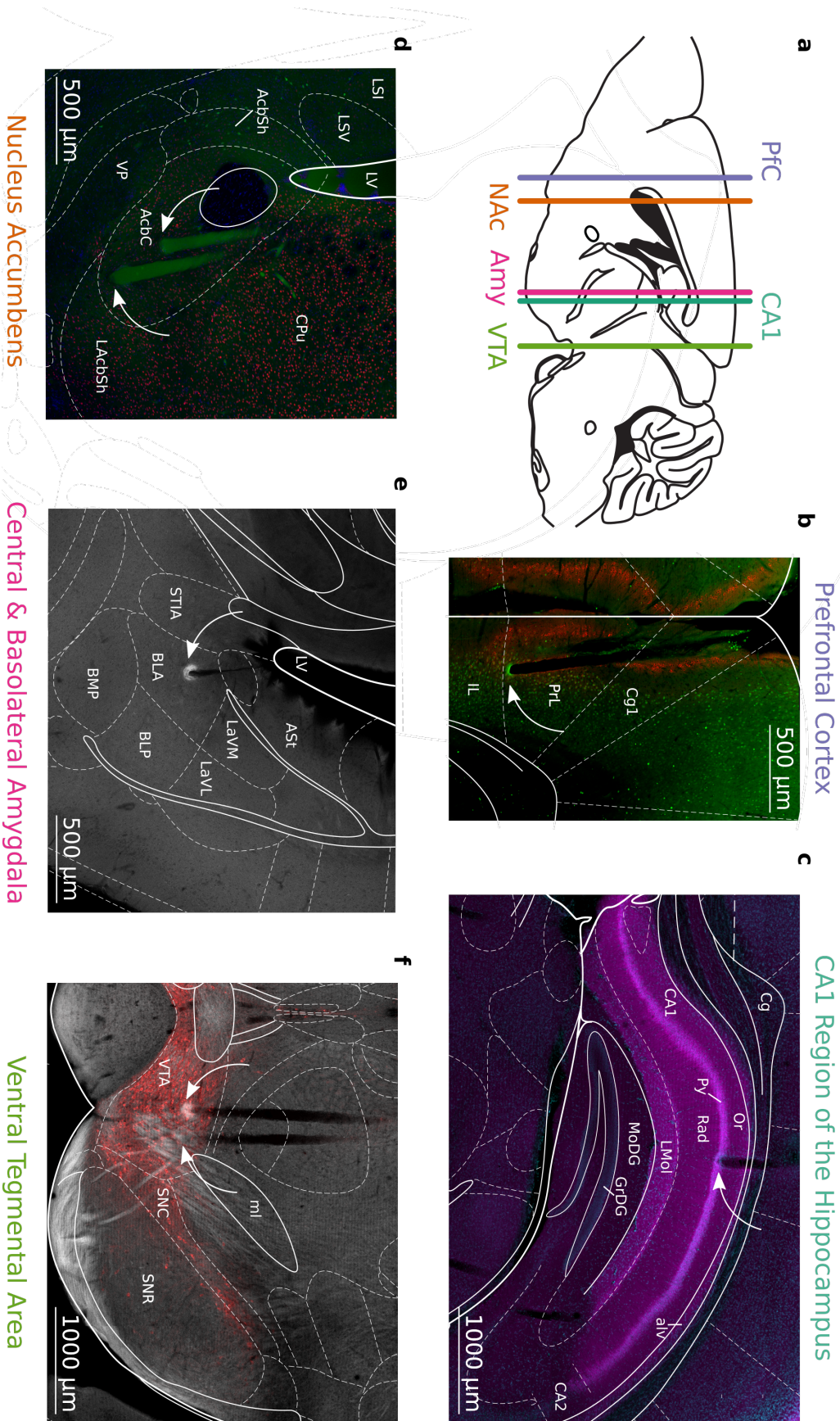


Figure 3.5: Histological verification of tetrode locations. White arrows indicate where the end of the tetrode was identified. Borders and abbreviations as in Franklin et al. 2008. (a) Approximate colour-coded location of each slice along the anteroposterior axis. (cont.)

Figure 3.5: (b) Green = wfs1, red = Ctip2. Cingulate cortex area 1 (Cg1), prelimbic cortex (PrL), infralimbic cortex (IL). (c) Magenta = wfs1, blue = DAPI. Oriens (Or), Pyramidal (Pyr), Radiatum (Rad), lacunosum moleculare (LMol) layers of the hippocampus. Molecular (MoDG), granular (GrDG) layers of the dentate gyrus. alveus (alv), cingulum (cg). (d) Red = FOXP1, green = Calretinin, blue = DAPI. Caudate putamen (CPu), nucleus accumbens core (AcbC) and shell (AcbSh), lateral nucleus accumbens shell (LAcbSh), lateral septum ventral (LSV) and intermediate part (LSI), ventral pallidum (VP), lateral ventricle. (e) Bright-field only. Basolateral amygdala anterior (BLA) and posterior part (BLP). Posterior part of the basomedial amygdala (BMP). Intraamygdaloid division of the bed nucleus of the stria terminalis (STIA), amygdalostriatal transition area (ASt), lateral amygdaloid nucleus ventrolateral (LaVL) and ventromedial (LaVM) part. (f) Red = TH, bright-field. Substantia nigra pars compacta (SNC), substantia nigra pars reticulata.

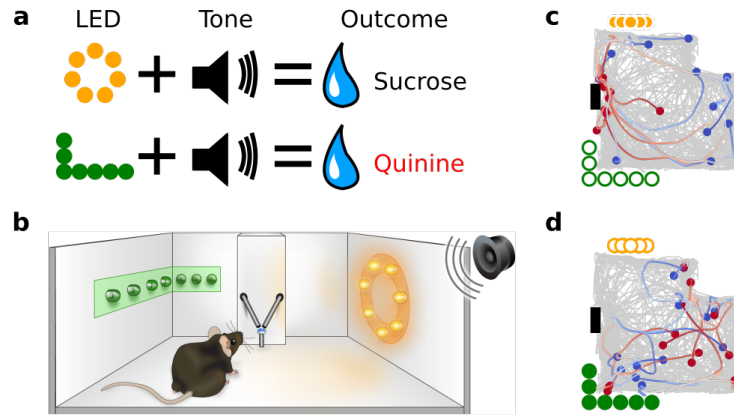


Figure 3.6: Behavioural Task. (a) Schematic of task-contingencies. (b.) Schematic of experimental behavioural set-up. Example paths for one recording session in the sucrose (c) and quinine (d) conditions. Blue dots indicate the position of the mouse at the onset of the tone, blue-red gradient indicates the passage of time, red dots indicate the position of the animal immediately prior to drop delivery.

Mice learn task contingencies

The behavioural vigour of the mice's locomotor response to the tone was similar in the two LED conditions (Figure 3.7a). That is to say the mouse increased its speed in response to the tone in both scenarios. However, the tone evoked the approach of the dispenser only when the sucrose-predicting LED was active and instead evoked its avoidance during the quinine-predicting LED. This was indicated by the probability of the mouse approaching the dispenser being higher in the sucrose versus the quinine condition on almost all recording days (Figure 3.7b; example

paths in Figure A.1). Additionally, while the change in speed was the same between the two conditions, the mean velocity towards the pump was consistently higher in the sucrose condition (Figure 3.7c). This change in the direction of the response, as opposed to the magnitude, was also reflected in the trajectories of the animal, which show that the response to the tone in the quinine condition contained an avoidance component (i.e. running in the opposite direction from the dispenser) that was absent in the sucrose condition (Figure 3.7d,e). Together these results show that the mice successfully learnt the task contingencies.

Temporal dynamics of task resolution

For both approach or avoidance behaviour the mice made their decision (see Figure 3.2) around 2 s after the onset of the tone (median 2.32 s; IQR 1.70-5 s) and arrived at the port 4 s after the tone (median 3.97 s; IQR 3.27-1.06). It is possible that some of the variability in the approach time was explained by the behaviour of the mouse immediately prior to the tone. For example, the mouse may have responded faster if it was engaged in exploratory behaviours such as climbing or running and slower if it was engaged in internally driven behaviours such as grooming.

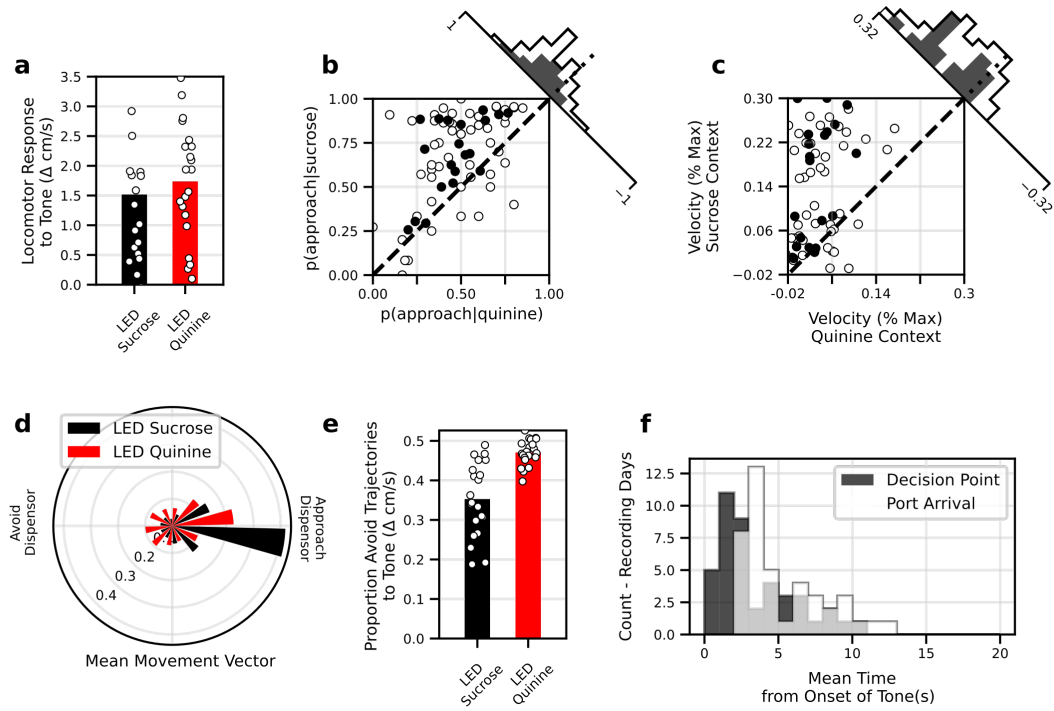


Figure 3.7: Behavioural Results. (a) Locomotor response to tone (change in speed) in each task contingency ($p = 0.38$; paired t-test) (b) Proportion of trials in which the animal approached the dispenser in one condition versus the other. Black indicates mean for a given recording day ($p < 1e-5$; paired t-test). Unfilled indicates mean for each session ($p < 1e-6$; paired t-test). Histogram above shows the difference in proportions between the two conditions. (c) As in (b) but using velocity towards port instead of proportion of trials where the port was reached ($p < 1e-6$ recording days, $p < 1e-8$ for session; paired t-test). (d) Mean vector of movement relative to the dispenser in each condition. (e) Proportion of trajectories that were in the opposite direction to the dispenser ($p < 1e-4$; paired t-test). (f) Histogram of times from tone when the mouse started towards the port (decision point), and when it arrived (port arrival).

3.4 Discussion

3.4.1 Task design

Our choice of task was designed to engage multiple stimulus modalities in which a range of sensory pathways would likely play an active role. We also wanted a task that was ethologically relevant but was still tractable for later analysis. In practice this meant that we wanted the task to be interspersed between periods where the animal was freely behaving ruling out tasks such as the Vogel test or other cued reward-aversion conflict tests (Millan 2003; Vogel et al. 1971). On the other hand, paradigms such as conditioned place preference, elevated plus maze, or the light-dark test would be inappropriate for tracking the fast-evolving neural dynamics during decision-making, as it would be difficult to determine at what time the decisions were being made.

Thus we developed the task described here, a spatio-visual discrimination task where successful resolution of the conflicting stimulus (the tone) using the surrounding spatial cues (the LEDs) led to reward. The fact that the switches in the LED-based contexts could be externally imposed allowed us to better determine the effect of context switches as we could avoid the confounds of the mouse running between two contexts or of us physically transporting it. However it is questionable whether how ethologically valid the model of a context-switch is. The use of the LED-based environment in general was inspired by Jezek et al. 2011.

The newly-designed task fulfilled all of our criteria and was successfully learnt by all mice. Additionally, the task was sufficiently difficult that we were able to collect a good sample of both correct and incorrect trials. The learning stages of the task were performed without the headstage connected and the task contingencies were the same each day for each mouse, which together precluded us from testing the development of responses over learning. Moreover, the timings of the behavioural responses were less tightly locked to the onset of the tone than can be seen in simpler tasks, or those set in more spatially restricted environments (Figure 3.7b). These are trade-offs that were acceptable in pursuit of a less constrained, and thus

more ethologically relevant, paradigm (Amir et al. 2015) than a linear track, a headfixed preparation or a task where mice are easily over-trained.

3.4.2 Controlling for the effects of behaviour

However in any freely-moving paradigm it is important to control for the effects of behavioural in order to interpret any changes in neural activity as said activity can be very different according to the gross behavioural state such as running vs. quiescence. In the CA1 for example it is known that firing rate changes are seen in place cells depending on the speed of the animal (Hirase et al. 1999; McNaughton et al. 1983).

The most common approaches to control for the speed of the animal are speed-matching and regression. In the former we attempt to find trials which have a similar mean speed for each condition, but where the labels are different, or during task-OFF periods. Then we can test to see whether the neural response exists in these trial sets as well. This simple approach is useful as a first-pass, but ignores any dynamics in the speed profile, including the acceleration of the animal, which are known to affect firing patterns (Gupta et al. 2012) and LFP signatures (Long et al. 2014).

Another common approach is to regress out speed from the neural activity (Engelhard et al. 2019). That is, fitting a model to predict the firing rate of a given neuron over the course of a recording using the instantaneous speed. By including some polynomial terms in this regression we can also take into account the acceleration of the animal at a given time-point. Using this approach we can then predict the firing responses of a given neuron for a particular speed profile - and then test whether this is significantly different for our effect of interest. However some variance in the activity of a given neuron can be predicted not only by the current instantaneous speed, but also by the past, and future time-points. We could try and include these in the regression, but this can lead to over-fitting and is inappropriate for most small data-sets.

Inspired by both of the above approaches we searched for a heuristic that would serve as an accurate control in our data-set. As described in chapter 4 instead of finding average-speed-matched time-points we looked for sections outside of the

task period which had similar speed profiles to when the mouse was responding to the tone. We found the trajectory-profile-matching based approach to be a satisfactory compromise between potentially overfitting a complex model versus underfitting based on mean speeds.

3.4.3 Recording technologies

Simultaneous multi-site recordings in anatomically disparate regions in a freely moving mouse are a recent development in the neuroscientific community. While previous multi-circuit recordings tended to use low-unit-yield techniques such as single wires (Hultman et al. 2018; Kumar et al. 2013) our tetrode microdrives allowed us to obtain higher yields with similarly customisable positions of the tetrodes. The future of multi-circuit recordings likely lie in next-generation recording probes such as CMOS-based devices (Neuropixels - Jun et al. 2017a) that have thousands of recording contacts or polymer electrode arrays which can support hundreds of channels with long-term stability (Chung et al. 2019).

4

Integrating memories spanning multiple brain circuits

[...] the idea that distributed representations are somehow in conflict with the extensive evidence for localisation of function in the brain (Luria, 1973). A system that uses distributed representations still requires many different modules for representing completely different kinds of thing at the same time. The distributed representations occur *within* these localised modules. For example, different modules would be devoted to things as different as mental images and sentence structures, but two different mental images would correspond to alternative patterns of activity in the same module. The representations advocated here are local at a global scale but global at a local scale.

Rumelhart et al. 1986

Contents

4.1	Introduction	63
4.2	Methods	67
4.2.1	Problems and solutions assessing single neuron modulation by external events	67
4.2.2	Assembly Analysis	72
4.2.3	Assembly Classifiers	77
4.3	Results	78
4.3.1	Differential engagement of multiple regions during successive trial stages	78
4.3.2	Distribution of single neuron responses does not suggest classes	78
4.3.3	Observing cross-structural assemblies	79

4.3.4	Intrinsic properties between cross- vs within- region as- semblies	80
4.3.5	Distributed assembly activity encodes performance infor- mation	86
4.4	Discussion	91

4.1 Introduction

Globally distributed representations evolve with the successful translation of local sensory representations into behaviour.

The epigraph at the start of this chapter is an excerpt from Rumelhart, Hinton and McClelland's *Parallel Distributed Processing* (Rumelhart et al. 1986). They outline how a network where computation is distributed across a population of neurons can be advantageous for a whole class of problems. Their interest in such an architecture was not primarily in its biological plausibility, but rather in how the architecture could behave in ways that matched psychological observations. For example, storing a representation in patterns across a distributed network automatically (assuming the patterns are not orthogonal) confers some ability for generalisation. This fits not only with the remarkable ability for humans to generalise between experiences, but also with the manner in which memory in humans exists on a continuum between exact recall and total reconstruction of previous events (Bartlett 1932). In this particular Rumelhart and colleagues take some care to state that their conceptualisation of these distributed representations would not be distributed across the whole brain but distributed locally within regions. Even then there was considerable evidence for functional specialisation within the brain (O'Keefe et al. 1971; Weiskrantz 1956) with which they wished to stress their theories were not incompatible. In the 40 years since this study we have continued to build on this model of distributed representations embedded within anatomically distinct regions (Miller 2000; Pennartz et al. 1994; Schultz et al. 1997). For us, the possibility of globally distributed representations¹ being another functionally relevant unit remains a tantalising, if challenging, avenue to explore.

There are two main lines of reasoning that point in this direction. The first is that the observations and experiments that outline the computational benefit of distributed representations should apply equally well at both the global and local scales. For example, in the visual stream representations become progressively

¹Note that distributed here means "not local" and thus could refer to a fully distributed code, or, more plausibly, a sparse distributed code.

more abstract, such that representations in later areas can respond to faces in a position and orientation invariant fashion (Perrett et al. 1982; Rolls 2012). Each individual layer contains its own sparse, distributed code that can represent a large number of stimuli. Each visual scene is thus encoded by a hierarchy of representations that may be bound by synchronous (Kreiter et al. 1996; Singer 1999) or polychronous (Bienenstock 1995; Isbister James B. et al. 2018; Izhikevich 2006) neuronal activity. The putative mechanisms for binding low- and high- level features in the visual system might, however, be applied elsewhere. For example, the theory of complementary learning systems suggests that episodic memories and associations encoded by the hippocampus are bound with their counterpart generalised representations in cortex (McClelland et al. 1995). A related observation is that selectivity to a given stimulus is less specific to a given neuron than is the efficacy of disrupting said neuron in altering its perception. For example, while both inhibitory and excitatory cells respond to whisker stimulation, only the experimental stimulation of inhibitory single cells in the barrel cortex affected the behaviour of mice (Tanke et al. 2018). The same principle can be applied to populations of neurons across different regions. While there are neurons in the dorsal CA1 of the hippocampus that will respond to a pure tone, it is unlikely that silencing these neurons would affect the animals perception of the tone. One explanation for these observations could be that these activated but "unnecessary" neurons are part of a wider partially redundant representation.

This idea of local representations co-existing with distributed ones has a counterpart in network topologies known as a small-world network (Watts et al. 1998). While a random network has a short path length between any two nodes and a regular network has high local clustering, a small-world network has both. This has been shown to be a property of many real world networks including the brain where it may reflect a joint optimisation over the functional length of network "wiring" and the number of "hops" needed to get from node to node (Kaiser et al. 2006). The relationship between distributed cell assemblies and small-world networks has been suggested (Zippo et al. 2013) however most investigations into neural network

topologies have focused on anatomical connectivity. As such, the relationship between the small-worldness (Humphries et al. 2008) and functional representations in the brain remains unclear. Moreover, it is not necessarily the case that this relationship is possible to determine without sampling from a much larger proportion of the network than we are currently able to do in rodent electrophysiological recordings (Perreault 2010; Stumpf et al. 2005).

In this study we instead search directly for short-timescale co-activation patterns that span multiple regions. This is the same definition that has previously been applied to explain the function of cell assemblies within local circuits (Peyrache et al. 2009; Rothschild et al. 2017; van de Ven et al. 2016). Notably this is distinct from engram representations that are experimentally manipulated using IEG-tagging techniques (Garner et al. 2012; Liu et al. 2012; Trouche et al. 2013) primarily in terms of the timescale we consider (tens to hundreds of milliseconds, rather than the hours required for IEG-expression). Our aims differ from these experiments too: we are not directly searching for the representation of the specific cues to which we expose the subject. Instead we are searching for constitutively or commonly expressed patterns of brain-wide activity during quiescent and exploratory behaviour that would form the building blocks of the representations which are dynamically re-organised during a constrained decision-making process. In practice that means the approach we take should be an unsupervised one and that we should fit our models to epochs in the recording without discrete cues. This is also distinct to studies that explicitly search for representations by fitting models during discrete task epochs (McKenzie et al. 2014). A discussion of the variety and application of different meanings ascribed to the term "cell-assembly" can be found in the discussion and in chapter 5.

The presence of such a globally distributed representation would necessitate a role for long-range synchronisation between multiple brain regions. There is evidence that such synchronisation exists and may be functionally important. During internally driven states such as REM sleep there are brain-wide haemodynamic responses following fast gamma bursts in CA1 (Bergel et al. 2018). Coordination between multiple subcortico-cortical circuits at 4 Hz has been shown to relate to

both working memory (Fujisawa et al. 2011) and fear behaviour (Karalis et al. 2016). Brain-wide coordination has even been indicated in predicting susceptibility to depression (Hultman et al. 2018). Though it is not clear whether the effect reported in this study is a result of internally driven factors, or rather confounded by the fact that the gross behavioural state of the animal (such as being previously stressed leading freezing or escape behaviours) is predictive of both the choice interaction test score and the LFP statistics. What is more certain is that both spontaneous (Stringer et al. 2019b) and motivated (Allen et al. 2019) behaviours can drive activity brain-wide. While there may even be single neurons that span all of these circuits simultaneously (Reardon 2017), it remains unclear whether there are really meaningful² brain-wide representations of events or if this simultaneous activity simply reflects local modules being driven epiphenomenally by a common feature. In other words, it is not clear whether these processes more closely reflect scenario **(b)** from Figure 1.2 where parallel processing of different aspects of sensory input happens independently before an explicit integration step, or scenario **(c)**, where the processing is distributed across areas and processing within each area is dependent on repeated interactions with other areas.

In order to determine whether globally distributed representations are used in the brain, we first characterise the responses of single neurons at the population level. In order to do so across a population of neurons potentially acting over a range of timescales and response shapes we combine decoding techniques with wavelet-based preprocessing. Then we apply assembly-detection methods so far only applied to single regions searching for signatures spanning multiple circuits. After characterising the intrinsic characteristics of these cross-structural assemblies, we compare the evolution of the corresponding local and globally distributed representation during task execution.

²Meaningful here means either functionally relevant or at least predictive of functionality.

4.2 Methods

4.2.1 Problems and solutions assessing single neuron modulation by external events

Inferential Statistics Approach

A common way to investigate groups of neurons that show differential responses to different stimuli is to classify cells based on whether or not they show a "significant firing response". There are two separate issues with such an approach that are well-known (Wohrer et al. 2013) but often ignored. The first is that the distribution of firing rate changes often does not suggest discrete classes (i.e. it is often continuous/uni-modal with a long tail) and as such "the difference between the significant and non-significant response is not itself significant" (adapted from Gelman et al. 2006). This is easily enough overcome by first examining the full distributions of firing rate changes to see if there is any evidence motivating classification into discrete classes. A statistical test such as Hartigan's dip test (Hartigan et al. 1985) with unimodality as the null hypothesis can be used if required.

The second issue is with the measure itself, using rate changes as the sole indicator of firing responses. There are three sub-issues to deal with here:

1. The timescale of response is likely unknown, and may be different for different neurons.
2. Firing rate changes do not encapsulate other changes that would still be detectable by downstream neurons, such as bursting or firing locked to the phase of a particular oscillation (Buzsáki 2010).
3. Standard hypothesis tests do not fully account for the reliability of any response, nor do they give a good indication of how much information that neuron is conveying about the stimulus.³

³For example a large enough response during just one trial can result in rejection of the null hypothesis. These types of responses can be common when for example very low firing rate neurons are normalised.

Machine Learning & Information Theory Based Approach - Wavelet Modulation

Here, we outline an approach that addresses each of the aforementioned sub-issues. The approach is a generalisation of a wavelet-based information measure from Lopes-dos-Santos et al. 2014 to quantify the modulation of a single-neuron around a given stimulus. It uses a variable set of timescales for each neuron so that we only have to set a lower bound on the bin size to use. It quantifies information in the timing of the spikes relative to the stimulus, thus providing information beyond rate. Finally, it uses a decoder-based method for quantifying information, thus accounting for the reliability of the responses.

The first step is to bin the spikes around each stimulus to obtain a matrix X ($\#_{bins} \times \#_{trials}$) where $\#_{bins}$ is a power of two. We are free to choose a bin size that represents the lower bound of the timescale we think could contain useful information: here we use 1 ms⁴. X is then split into two matrices X_{pre} and X_{post} . Each contains the bins before and after the stimulus onset respectively. We then decompose each column of each matrix using Haar wavelets.

$$wX_{pre,post} = \mathbf{H} \cdot X_{pre,post} \quad (4.1)$$

We use the wavelet transform because it gives us a representation of the signal where each set of points provides information about a different timescale while preserving information about their location in time. This can be seen more concretely by looking at the Haar Transform Matrices H for some length N signals:

when $N = 2$, we have

$$\mathbf{H}_2 = \frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 \\ 1 & -1 \end{bmatrix}$$

⁴Choosing a smaller bin size will not have a major impact on the results, assuming there is no information in spiking at that timescale, but will increase the computational burden. It's unlikely that there is information at smaller timescales as 1 ms is around the margin of error for spike detection.

when $N = 4$, we have

$$\mathbf{H}_4 = \frac{1}{2} \begin{bmatrix} 1 & 1 & 1 & 1 \\ 1 & 1 & -1 & -1 \\ \sqrt{2} & -\sqrt{2} & 0 & 0 \\ 0 & 0 & \sqrt{2} & -\sqrt{2} \end{bmatrix}$$

and when $N = 8$

$$\mathbf{H}_8 = \frac{1}{\sqrt{8}} \begin{bmatrix} 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & -1 & -1 & -1 & -1 \\ \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \sqrt{2} & \sqrt{2} & -\sqrt{2} & -\sqrt{2} \\ 2 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & -2 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 2 & -2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 2 & -2 \end{bmatrix}$$

or graphically:

$$\mathbf{H}_2 = \begin{array}{c} \text{---} \\ \text{---} \end{array}$$

$$\mathbf{H}_4 = \begin{array}{c} \text{---} \\ \text{---} \end{array}$$

$$\mathbf{H}_8 = \begin{array}{c} \text{---} \\ \text{---} \end{array}$$

Each point in the output of the signal decomposition is now effectively the convolution of the original signal with each row from the appropriately sized Haar transform matrix or each contiguous line in the graphical representation. Note that after this process we still have the same number of features (rows) for the matrices $wX_{\{pre,post\}}$ as in $X_{\{pre,post\}}$. In order to train a classifier to accurately separate these observations we first need to reduce the number of features, ideally keeping only those that are informative. We perform this feature selection by calculating the mutual information between each corresponding feature in the $wX_{\{pre,post\}}$ matrices (Shannon 1948).

$$\begin{aligned} I(X; Y) &= \sum_{x,y} P_{XY}(x, y) \log P_{XY}(x, y) \\ \frac{P_X(x)}{P_Y(y)} &= E_{P_{XY}} \log P_{XY} \frac{P_X}{P_Y} \end{aligned} \tag{4.2}$$

where X and Y are corresponding rows from matrices $wX_{\{pre,post\}}$. The k features with the highest "unbiased" mutual information are selected, where k is bounded to be within a certain range (here $[2, 25]$). Here "unbiased" mutual information was calculated by subtracting the 95th percentile of surrogates with the columns of the $wX_{\{pre,post\}}$ matrices randomly exchanged. We refer to the matrices post feature selection as $pX_{\{pre,post\}}$.

Finally, in order to estimate how much information about the timing of the stimulus we could obtain from the spike trains we assessed the performance of a linear classifier (here Gaussian Naive Bayes) on predicting whether a given observation came from pX_{pre} or pX_{post} using k-fold cross-validation ($k=5$). The performance of the decoder gives a lower-bound on the information content of the spikes (Quiñan Quiroga et al. 2009).

Validation To validate the wavelet decoder we simulated various spike time distributions that could indicate modulation by an external stimulus or internally driven event. Spike times were drawn from one of five distributions.

$$f(t) = k \quad (4.3)$$

$$f(t) = \begin{cases} k \times t & \text{for } 0 < t \\ k & \text{else} \end{cases} \quad (4.4)$$

$$f(t) = \frac{k}{2} \varphi\left(\frac{t}{2}\right) \quad (4.5)$$

$$f(t) = \begin{cases} k & \text{for } t \leq 0 \\ \sin(\pi t) + k & \text{for } 0 < t \end{cases} \quad (4.6)$$

$$f(t) = \begin{cases} 1.95k & 0 < t \leq 0.05 \\ 0.05k & 0.05 < t \leq 0.1 \\ k & \text{else} \end{cases} \quad (4.7)$$

where t is the time of the spike in seconds relative to the onset of the stimulus and k is the firing rate constant. Figure 4.1 shows how the wavelet information decoder performs in classifying simulated spike trains (ordered as above) into pre

and post classes. For comparison we also show the performance of a decoder trained on the spike counts, and one trained on the first k principal components of the matrices, where k matches the number of features chosen by the wavelet information measure. Because the performance of the PCA measure is dependent on the bin size used, it was computed for several bin sizes. In Figure 4.1 you can see that PCA approach works best when the chosen bin size is similar to the response timescale. However this is not information that we have a priori for our experiments.

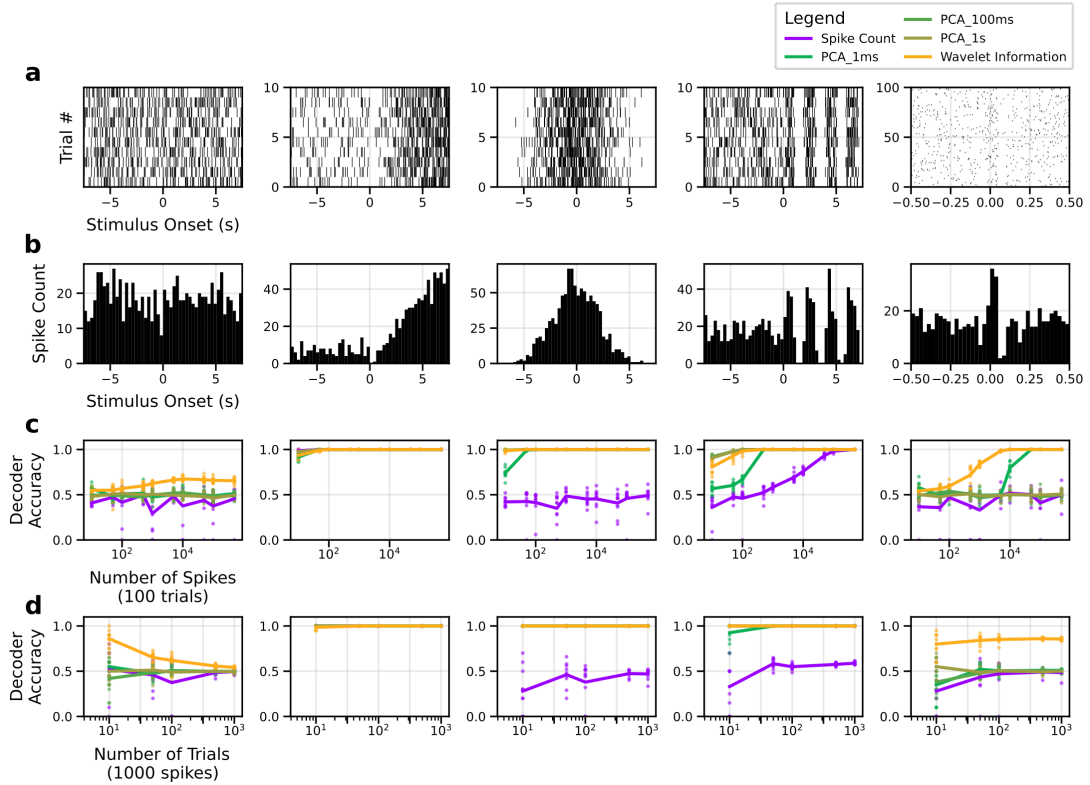


Figure 4.1: Comparison of effects of dimensionality reduction techniques on decoding performance. For each column spike times are simulated by drawing from the probability distributions from Equation 4.3 to Equation 4.7. (a) Raster plots of simulated spike times relative to stimulus. (b) Peri-stimulus histograms for simulated spike trains relative to stimulus. The cross-validated (10-fold) accuracy is reported for a naive Bayes decoder trained to predict whether a given spike train came before or after the onset of the stimulus. For (c) this is shown for each decoder as a function of the number of spike times drawn per trial (with a fixed number of trials). For (d) this is shown for each decoder as a function of the number of trials (with a fixed number of spikes per trial).

4.2.2 Assembly Analysis

Assembly Detection

To identify assembly patterns spike trains are first divided into 20 ms time bins and the number of spikes in each bin is counted for every neuron. In order not to bias the identification of assembly patterns towards neurons with higher firing rates, the binned spike counts are normalised for each neuron by a z-score transform:

$$z_{i,b} = \frac{x_{i,b} - \mu_{x_i}}{\sigma_{x_i}} \quad (4.8)$$

where $x_{i,b}$ is the spike count of neuron i in bin b , and μ_{x_i} and σ_{x_i} are respectively the mean and standard deviation of neuron i 's spike counts across bins. This ensures that the normalised activity of each neuron has null mean rate and unitary variance. With the number of neurons denoted by n and the number of time bins by B , let $\mathbf{Z}$ be the $n \times B$ binned and z-scored spike count matrix with element (i, b) equal to $z_{i,b}$. From this matrix $\mathbf{Z}$, assembly patterns are identified in a two-step procedure (Lopes-dos-Santos et al. 2013; Trouche et al. 2016).

To determine the number of significant patterns we apply PCA to $\mathbf{Z}$.

$$\sum_{j=1}^n \lambda_j \mathbf{p}_j \mathbf{p}_j^T = \frac{1}{n} \mathbf{Z} \mathbf{Z}^T \quad (4.9)$$

where p_j is the j^{th} principal component with corresponding eigenvalue λ_j (for $j = 1, \dots, n$). Note that $\frac{1}{n} \mathbf{Z} \mathbf{Z}^T$ is the correlation matrix of $\mathbf{Z}$. The Marčenko-Pastur law was used to estimate the number of significant patterns embedded within the data (Marčenko et al. 1967). This law states that for a $n \times B$ matrix whose elements are independent and identically distributed random variables with zero mean and unit variance, all eigenvalues are asymptotically (i.e., when $n, B \rightarrow \infty$ such that B/n converges to a finite positive value) bounded to the interval $[(1 - \sqrt{n/B})^2, (1 + \sqrt{n/B})^2]$. This suggests that if the firing activity of the neurons is independent from each other, then none of the eigenvalues is expected to exceed $\lambda_{\max} = (1 + \sqrt{n/B})^2$. Using simulated spike trains, this was shown to be the case even for values of n and B considerably smaller than in these recordings (Lopes-dos-Santos

et al. 2013; Peyrache et al. 2010). An eigenvalue above $\lambda_{\max}$ thus indicates that the pattern given by the corresponding principal component captures more correlation than any pattern would be expected to capture if the firing activity of all neurons was independent of each other. The number of eigenvalues above $\lambda_{\max}$ (denoted by N_A) therefore provides a lower bound for the number of distinct significant patterns.

The first N_A principal components each capture a significant amount of correlation between the firing activities of the neurons. However, principal components are restricted to be orthogonal to each other, while cell assemblies do not need to be (e.g. they can contain overlapping neurons). Principal components are also extracted from the data sequentially, which usually causes the first principal component to seemingly be a mixture of multiple assemblies (Laubach et al. 1999; Lopes-dos-Santos et al. 2011). We thus use ICA to demix these components and identify the patterns. ICA forces the linear projections of the data onto these patterns to be as independent from each other as possible. However, an ICA directly applied to matrix $\mathbf{Z}$ without prior dimension reduction would extract as many patterns as there are neurons, which leads to spurious results (Lopes-dos-Santos et al. 2013). To restrict the number of patterns identified by ICA to N_A , the data is first projected onto the subspace spanned by the first N_A principal components:

$$\mathbf{Z}_{\text{PROJ}} = \mathbf{P}_{\text{SIGN}}^T \mathbf{Z} \quad (4.10)$$

where $\mathbf{P}_{\text{SIGN}}$ is the $n \times N_A$ matrix with the first N_A principal components as columns. An ICA is then applied to the matrix $\mathbf{Z}_{\text{PROJ}}$. That is, a $N_A \times N_A$ un-mixing matrix $\mathbf{W}$ is found such that the rows of the matrix $\mathbf{Y} = \mathbf{W}^T \mathbf{Z}_{\text{PROJ}}$ are as independent as possible. This optimisation problem was solved using the fastICA algorithm implemented in Julia (Bezanson et al. 2014; Hyvarinen 1999; JuliaStatistics 2019). The resulting un-mixing matrix $\mathbf{W}$ is then expressed in the original basis spanned by all the neurons: $\mathbf{V} = \mathbf{P}_{\text{SIGN}} \mathbf{W}$ where the columns of $\mathbf{V}$ are the weight vectors of the assembly patterns. As both the sign and the scale of the ICA output are arbitrary, all weight vectors are scaled to unit length (i.e., $\sum_{i=1}^n v_k^2(i) = 1$ for $k = 1, \dots, N_A$) and the sign is fixed such that at least two of the

three largest weights were positive. An assembly pattern is described by a weight vector that contains each recorded neuron’s contribution to that pattern (Figure 4.2).

Tracking expression of assembly patterns over time

The expression of each pattern can be tracked over time as follows:

$$R_k(t) = \mathbf{z}(t)^T \mathbf{P}_k \mathbf{z}(t) \quad (4.11)$$

where $R_k(t)$ is the expression strength of assembly pattern k at time t and $\mathbf{z}(t)$ is the instantaneous firing rate of each neuron. $\mathbf{P}_k$ is the projection matrix of assembly pattern k and is constructed from the outer product of its weight vector $\mathbf{v}_k$. The diagonal entries of this projection matrix are set to zero to prevent high expression strength caused by the isolated activity of a single neuron with high weight to that pattern (Lopes-dos-Santos et al. 2013; Lopes-dos-Santos et al. 2011; Peyrache et al. 2009; Peyrache et al. 2010). With this approach, only co-firing of neurons can contribute towards the expression of an assembly pattern. To increase the temporal resolution beyond the bin size used to identify the assembly patterns, $\mathbf{z}(t)$ is obtained by convolving the spike train of each neuron with a kernel-function after which the resulting smooth curve is normalised by a z-score transform. We used a Gaussian kernel with standard deviation $w/\sqrt{12}$, so that the kernel has the same standard deviation as fixed time bins of w ms. The value w determines the width of the “integration window” in which spikes are still considered to be coincident. We set w to 20 ms to match the bin size used to identify the patterns. Peaks in R_k are thus only found when at least two neurons with some contribution to a given pattern fire close enough in time that the kernels overlap. The height of the peaks will be determined by the density of spikes discharged by each neuron as well as by the magnitude of the associated weights in assembly pattern k . Note that the scale of the expression strength can be interpreted as a “projected z-score”.

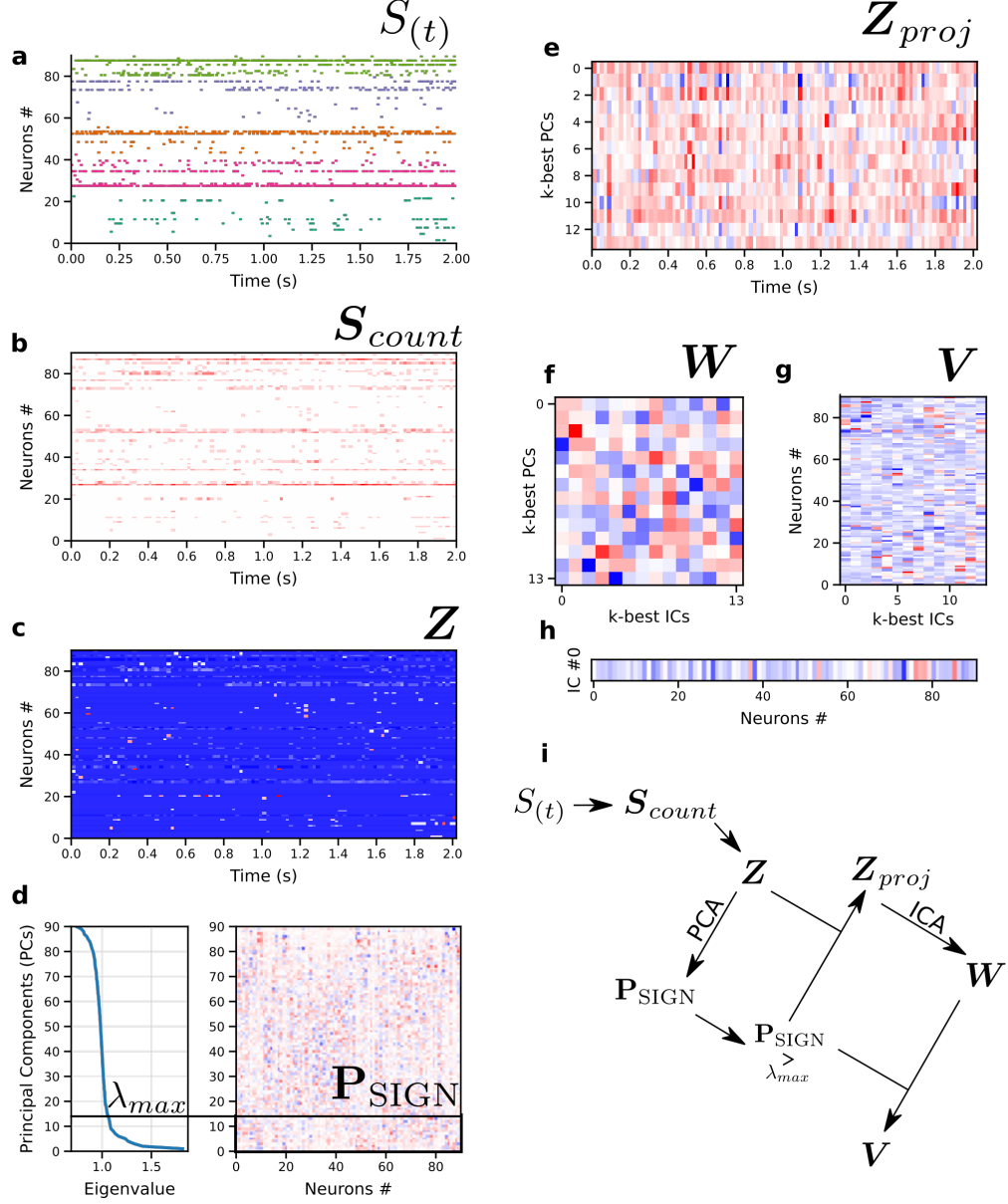


Figure 4.2: Illustration of assembly detection procedures corresponding to Equation 4.8 to Equation 4.11. **(a)** Raster plot of spike times for neurons from each region. Binned **(b)** and Z-scored spike counts **(c)** for the same time period. **(d)** Projection matrix for PC decomposition (right) of matrix Z with corresponding eigenvalues (left). **(e)** Principal components exceeding limit of the Marcenko Pastur distribution projected over time. **(f)** Demixing matrix that rotates the orthogonal principal components to be independent. **(g)** Weight vectors for neurons on the independent components. **(h)** Example weight vector i.e. one assembly. **(i)** Graphical description of assembly detection procedure.

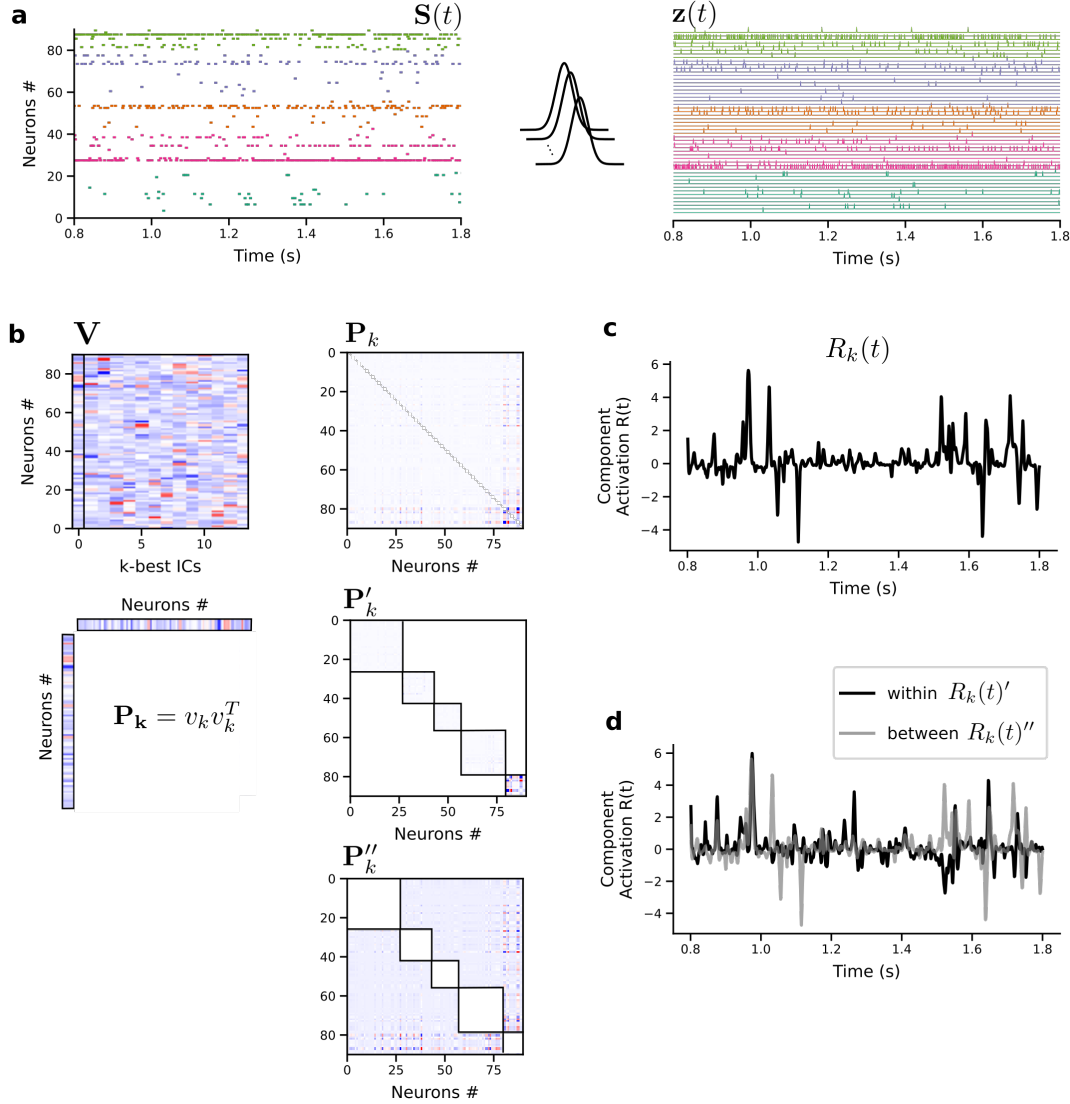


Figure 4.3: Illustration of procedures to track co-activity patterns with high temporal resolution corresponding to Equation 4.10. (a) Spike trains are convolved with a Gaussian kernel and then a z-score transform is applied. (b) We find each assembly's projection matrix by taking the outer product and then setting the diagonal to zero P_k . We can inspect specified interactions by masking this matrix. P'_k only contains within-region interactions, and P''_k only contains between-region interactions. (c) The activity of an assembly is the quadratic form of the projection matrix with the convolved spike trains. (d) shows the same as (c) except for the masked matrices.

Spike Shifting Procedure

All the spike times of all neurons were jittered by t seconds where t was chosen from a uniform distribution spanning from $-T_{max}$ to T_{max} . The expression of the original assemblies was then re-evaluated using these shifted spike times. Note that the assemblies are not re-detected only their projections onto the manipulated data are re-computed. This was repeated for 100 values of T_{max} logarithmically stepping from 0.1ms to 1 s to obtain a single length-100 vector for each assembly. To find the “critical timescale” a logistic function $p(x)$ was fitted to each vector and the midpoint (x_0) was taken.

$$p(x) = \frac{1}{1 + e^{k(x-x_0)}}$$

4.2.3 Assembly Classifiers

Behaviour was decoded using a naive Bayes classifier with null priors trained on the assembly responses to the tone. See Figure 4.11 for a schematic. Performance was evaluated using a 10-fold stratified cross-validation so that the number of observations for each label would be kept balanced for each train-test split.

4.3 Results

4.3.1 Differential engagement of multiple regions during successive trial stages

We observed that the transition between LED displays elicited rate responses in all five regions (Figure 4.4, first column; more example single neuron histograms and rasters in Figure A.2). The tone onset then triggered large rate changes in the Amygdala, VTA and PfC (Figure 4.4, second column), putatively indicating neuronal representations of the auditory stimulus. At the behavioural manifestation of the approach/avoidance response, estimated as the time-point prior to the sudden change in animal's velocity relative to the dispenser, rate responses were notably seen in the PfC and, to a lesser extent, the CA1 (Figure 4.4, third column). These responses may be related to the decision-making process. The arrival at the dispenser was marked by substantial rate changes in all five regions but most prominently in the VTA (Figure 4.4, final column). These results indicate that the events defining this task, from LED onset to outcome presentation, relate to changes in single-neuron spiking activity that span multiple brain regions. While there was some bias for each region to represent a particular dimension of the behavioural experience there was no selectivity of a given region for a particular stage at the level of population rate.

4.3.2 Distribution of single neuron responses does not suggest classes

The full distribution of the population responses revealed that they were long-tailed but unimodal, and so did not suggest that there were discrete classes of responding or non-responding cells for any of the regions or task stages (Figure 4.5 to Figure 4.7). This was the case for both responses defined purely by rate change (Figure 4.5) as well as those assessed using the wavelet decoder (Figure 4.6). We quantified whether any of the dips in the distributions were significant using a Hartigan's dip test (Hartigan et al. 1985) but neither of the sets of p-values indicated any evidence against the null hypothesis of unimodality (Figure 4.7; all uncorrected p-values > 0.01).

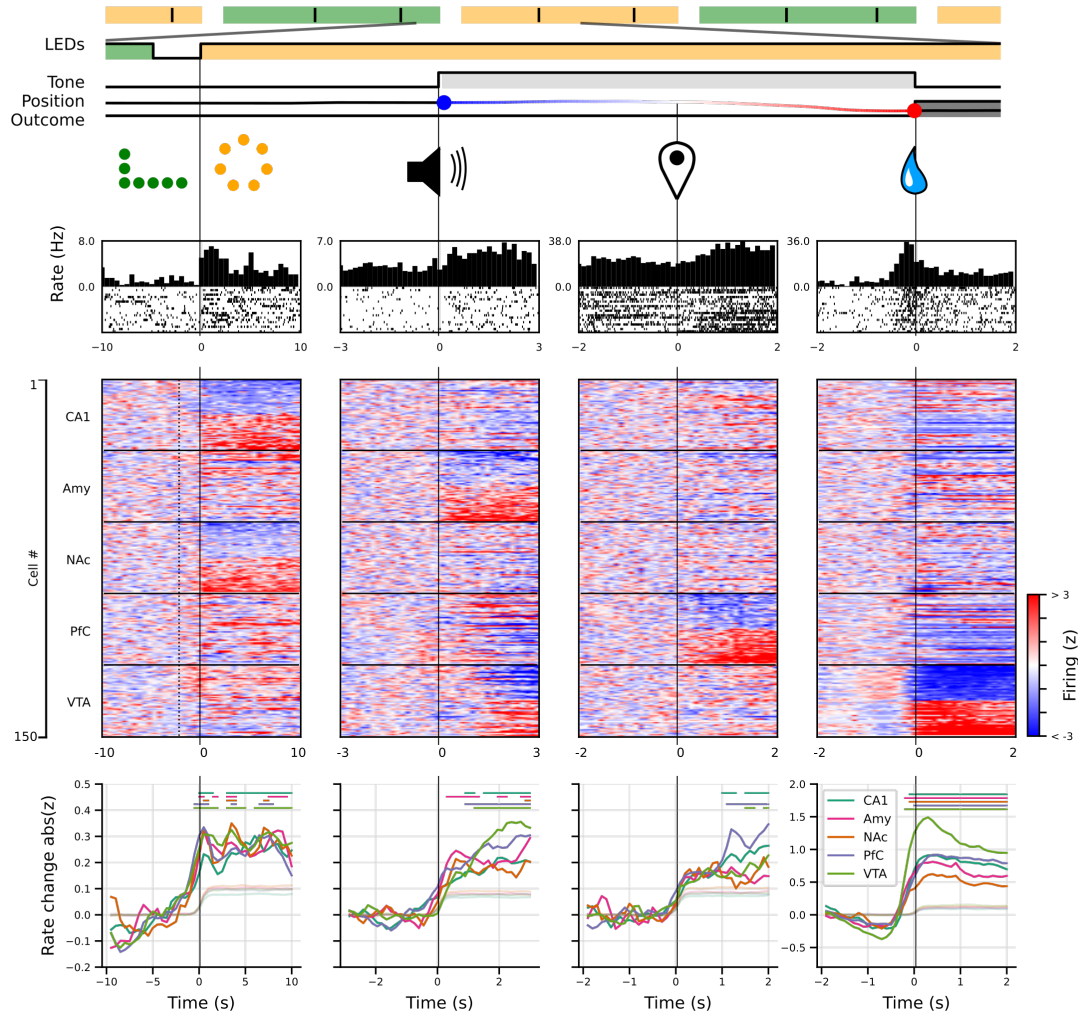


Figure 4.4: Differential engagement of multiple regions during successive trial stages. Top row: schematic of task timings. Second row: responses of 4 single neurons at each task stage denoted above. Namely LED switch, tone onset, decision point, drop delivery. Example neurons are from the amygdala, VTA, prefrontal cortex and the VTA respectively. Raster plots are shown for a representative subset of trials and histograms reflect spike counts over all trials. Third row: The z-axis in each row represents the mean z-scored firing rate over all trials (both sucrose and quinine). Neurons were ordered by the mean rate change during the time period in which their region showed the largest relative rate change. The neurons with the top 30 largest positive and negative rate changes were selected from across all mice. Bottom row: average absolute z-scored firing rates for all cells from each region. Solid lines denote periods where the responses were above 99% of shuffled responses. Washed-out lines are the mean of the shuffled responses.

4.3.3 Observing cross-structural assemblies

Next we investigated whether we could detect brain-wide expression of large-scale cell assemblies from the neuronal spiking observed in parallel across the five regions

we were recording from. To do so, we tested for the presence of firing patterns formed by groups of neurons exhibiting coincidental spike discharge using a combination of PCA and ICA (Lopes-dos-Santos et al. 2013). We conducted this assembly pattern detection during the off-trial epochs, when the tone was not being played, and when there was no solution at the dispenser. Consistent with previous studies, we found groups of repeatedly co-active neurons nested within a given brain region (e.g. column 5 of Figure 4.8b). In addition to these local cell assemblies, we also found firing patterns that engaged neurons spanning multiple brain regions, thus defining inter-regionally distributed cell assemblies (e.g. column 2 of Figure 4.8b). In total we found 260 assemblies (mean: 12.38, std: 3.40 per recording day), within these assemblies there were an average of 3.36 (std: 1.31) neurons who had weights greater than 2 standard deviations above zero.

4.3.4 Intrinsic properties between cross- vs within- region assemblies

Distribution of assembly contributions by region pairs

We quantified the contribution of each pair of regions by taking the sum of assembly activation time courses after applying a masking procedure (see Figure 4.3). The VTA and the PfC tended to contribute the most to assembly activations via within-region interactions, while the other regions tended to contribute equally via between- and within-region interactions (for single assembly examples see Figure A.3). The results were dependent on the behavioural state of the animal with, for example, the VTA decreasing its within-region contributions at higher speeds (Figure 4.9).

Next we used the same masking procedure to confirm, and quantify, the extent to which each detected cell assembly was driven by within-region (local) or between-region (distributed) interactions. We found that there was a bi-modal distribution of local vs. distributed contributions to each assembly (Hartigan’s dip-test $p < 0.01$; Figure 4.10). This implies that the co-activation patterns that occur between-regions tend to occur at different times than the ones within-regions (i.e. independently), rendering them separable by the demixing procedure during the

assembly detection. Thus the two patterns may reflect distinct modes of the underlying network activity. From here on for convenience we refer to the expression of the assemblies' within-region components as local assemblies and to the expression of their between-region components as distributed assemblies, although the same base assemblies are used in both cases.

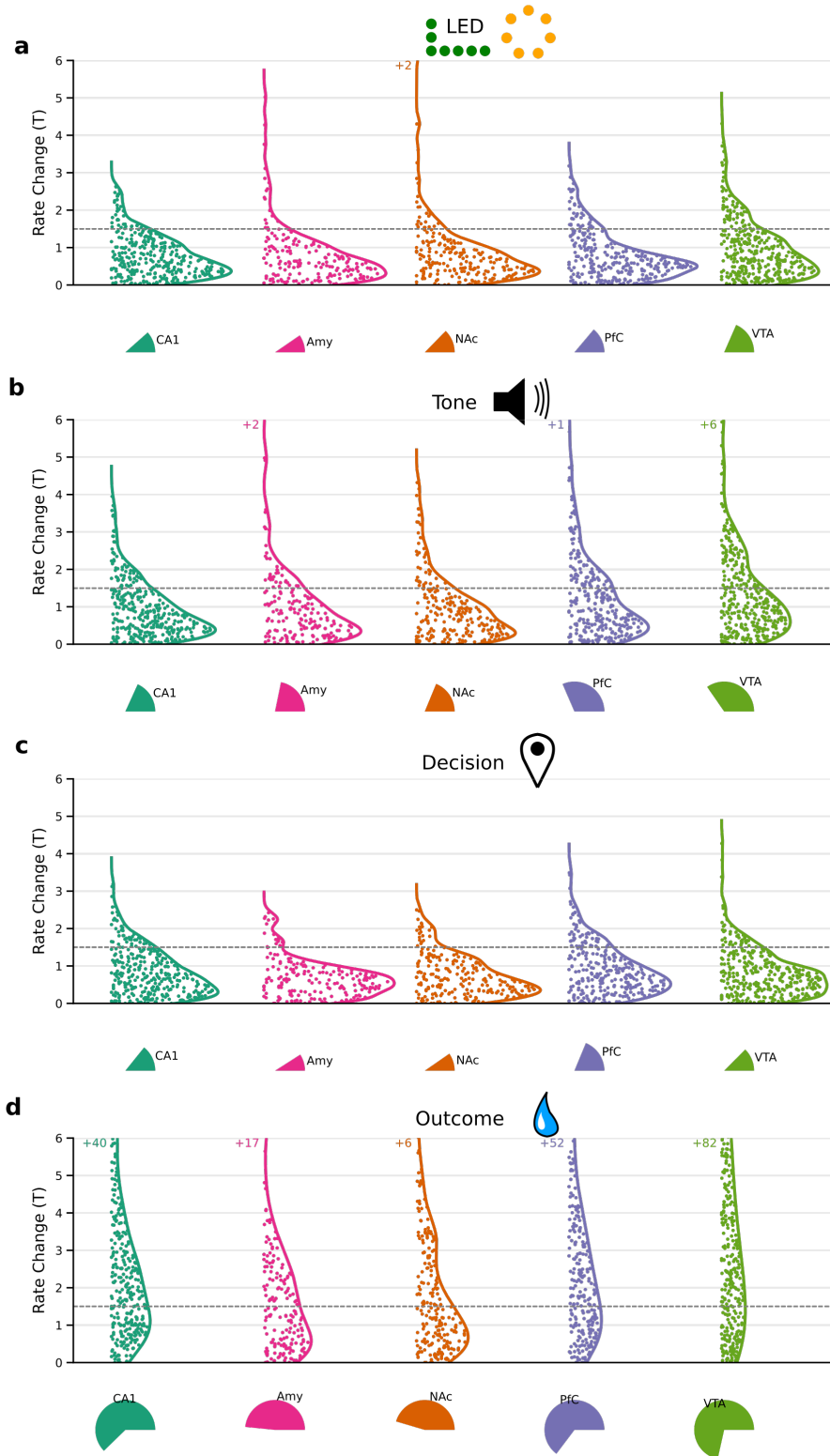


Figure 4.5: Rate Response at LED onsets (a), tone onsets (b), decision onsets (c) and outcome (d). Rate responses are assessed as t-statistic testing the mean rate change against zero. Pie charts show the fraction of neurons where $T > 1.5$. Time periods for the rate response are the same as in Figure 4.4.

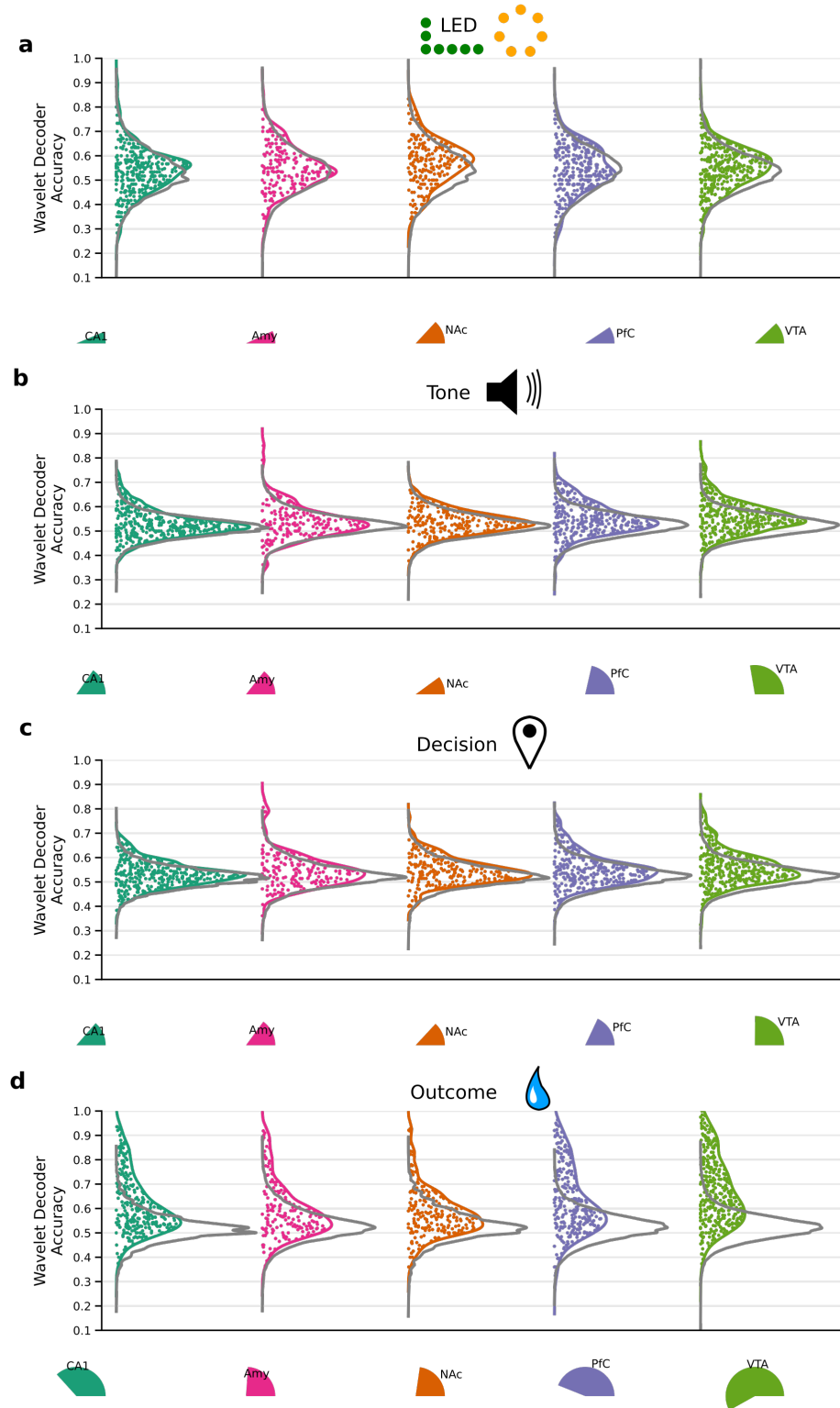


Figure 4.6: Modulation of neurons by LED onsets (a), tone onsets (b), decision onsets (c) and outcome (d). Modulation is assessed via the 10-fold cross-validated accuracy of a of a wavelet-based decoder. Pie charts show the fraction of neurons where performance exceeded the 95th percentile shuffled distribution (grey, computed for each cell independently). Time periods for the rate response are the same as in Figure 4.4.

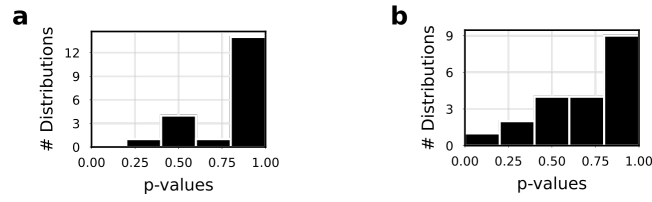


Figure 4.7: Distribution of p-values for Hartigan's dip tests applied to each of the distributions in Figure 4.5 (a) and Figure 4.6 (b).

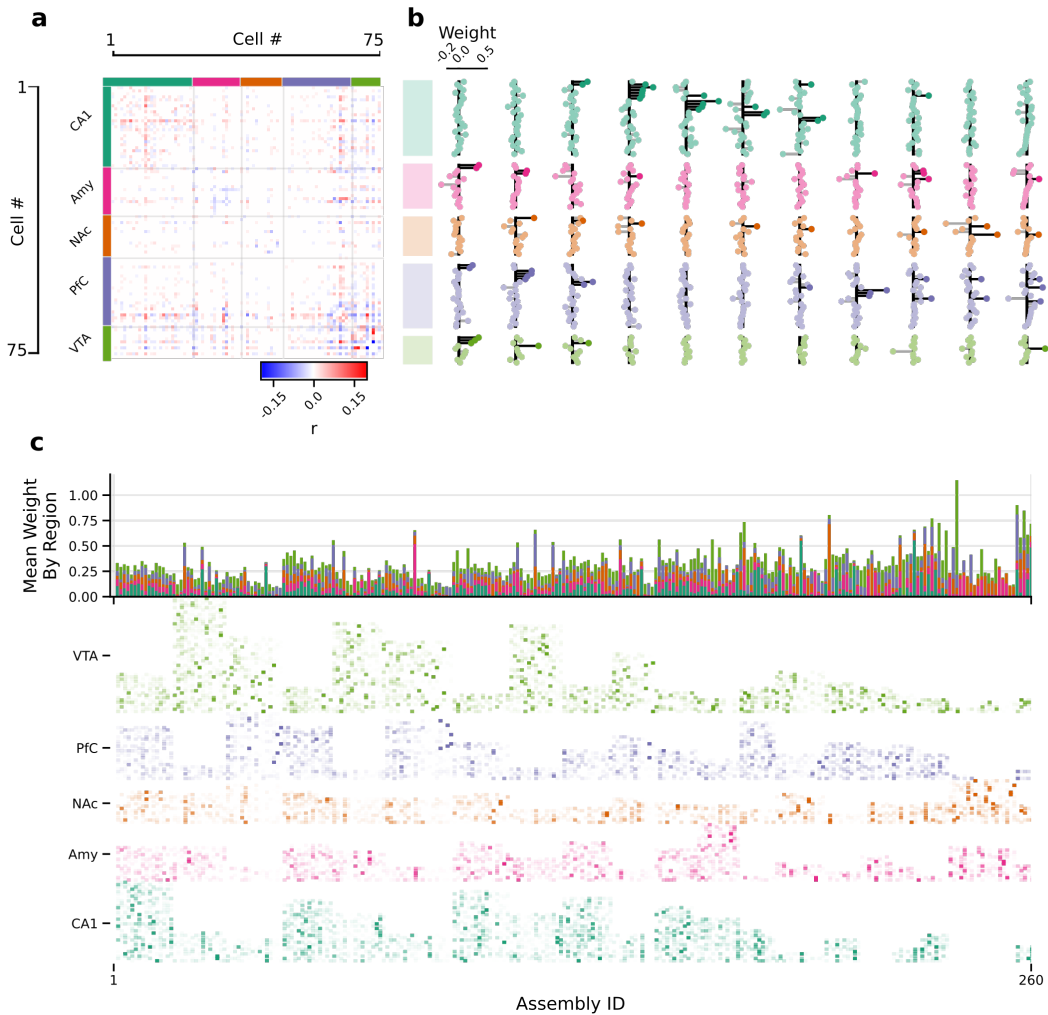


Figure 4.8: Assembly weight vectors span multiple regions. (a) Correlation matrix for all neurons in an example recording session. (b) Weight vectors for assemblies identified during the session. (c) All weight vectors identified. Transparency indicates the weight of each neuron to the particular assembly. The bar plot on top indicates the mean weight over neurons from each region for each assembly.

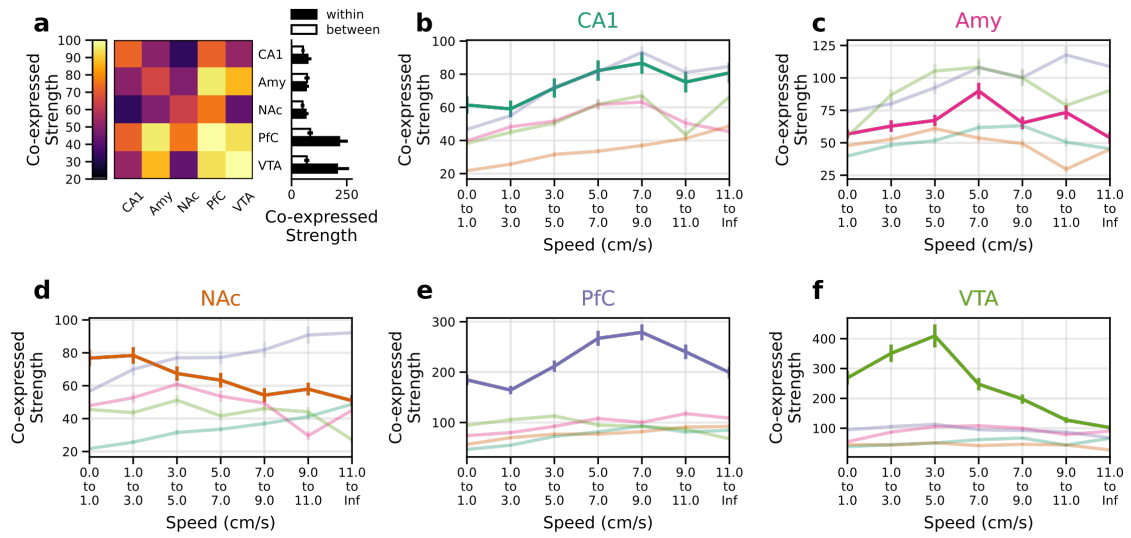


Figure 4.9: Contribution of cross-regional interactions to assembly activity. (a) Mean expression strength of all assemblies when only considering interactions between neurons from each pair of regions. For (b), (c), (d), (e), (f) the same data are shown but for each region separately across speed bins. Within-region strengths are shown in darker lines. Note that each between-region line appears in two panels. Error bars are ± 2 SEM.

Assessing the critical timescale for assembly formation.

Assemblies were detected using a bin size of 20 ms, a timescale concomitant with the timescale for synaptic integration in cortical principal cells (Koch et al. 1996; van de Ven et al. 2016). However the bin size only sets at what timescale we have the most statistical power to detect assemblies.⁵ To determine what timescale of interactions were captured by our detected assemblies we measured the strength of each assembly after manipulating *in silico* each spike train to remove the structure at a given timescale. From this spike-shifting procedure we derived two metrics for each assembly. The first was the threshold at which the assembly lost 10% of its expression strength. The second was the timescale reflecting the midpoint (x_0) in a logistic curve fitted to the expression strengths at each jitter.

We observed that most assemblies lost at least 10% of their expression strength after only 10ms of spike jitter and after 100ms most of the structure underlying their expression had been destroyed (Figure 4.10b). We hypothesised that those assemblies with greater contributions from between-region interactions (or less contribution from within region interactions) would show a slower stability degradation under the spike-shifting procedure because they may be interacting over longer timescales however we did not find any evidence for this (Figure 4.10c; x_0 $F=1.05$ $p=0.30$ $MI=0$, 10% $F=0.03$ $p=0.84$ $MI=0.12$).

4.3.5 Distributed assembly activity encodes performance information

Decoding of task contingencies using assembly activity

In this task, each tone presentation signalled the transient availability of either an appetite or an aversive outcome depending on the active LED. This sensorial ambiguity caused a conflict between reward-predicting approach and safety-optimising avoidance (see: Figure 3.6 and Figure 4.4). This uncertainty can be resolved by rapidly integrating task-related information and disentangling the associated

⁵Given infinite length recordings we could detect interactions at all timescales -from monosynaptic connections to slow population co-variations- using almost any bin size.

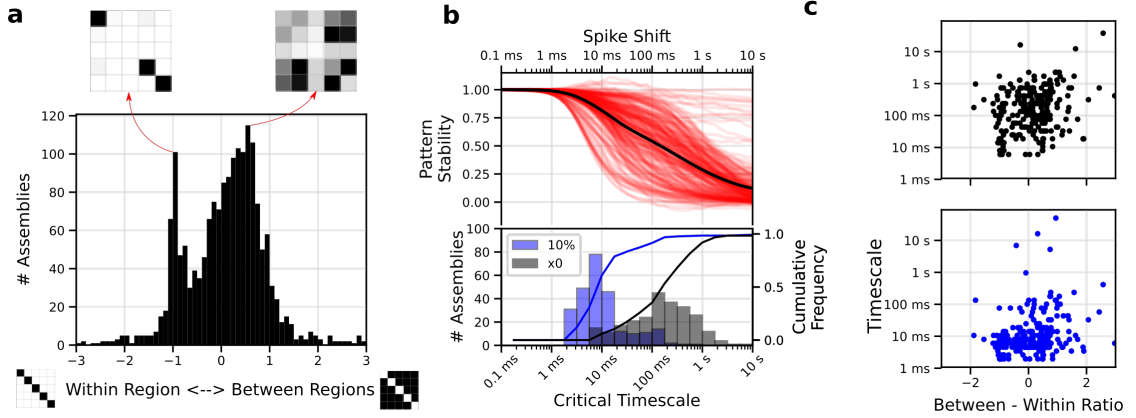


Figure 4.10: Intrinsic properties of assemblies in relation to analytical manipulations. (a) Relative contribution of interactions within-region versus between regions to each assemblies' total activity. Shown for all speed bins (see Figure 4.9). (b) Top: mean total activity of each assembly after shifting all spikes with a uniform distribution up to Spike Shift relative to its original total activity. Bottom: Critical timescale indicates the Spike Shift at which an assembly lost 10% of it's activation (blue, 10%) or the midpoint of a sigmoid fitted to it's pattern stability (grey, x0). (c) Relationship between the critical timescales from (b) and the contribution ratios from (a). Top, grey: using the midpoint criteria for the critical timescale. Bottom, blue: using the 10% critical timescale threshold.

contingencies while the corresponding outcome is available. We sought to determine whether local and distributed assemblies played a dissociable role in this process.

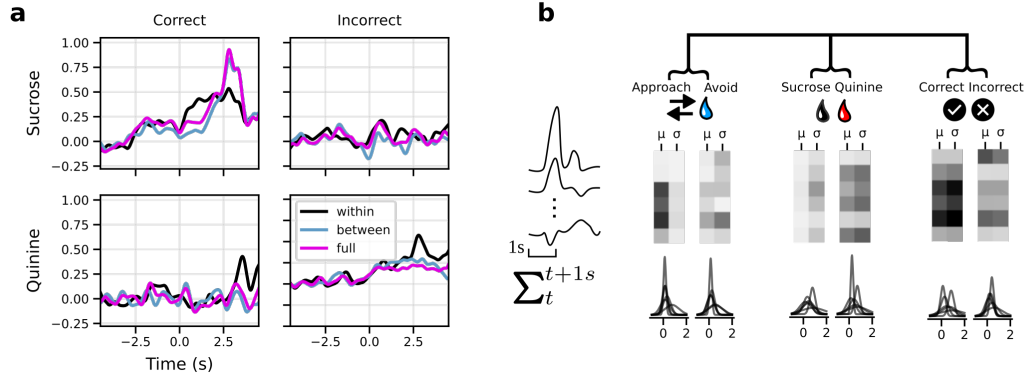


Figure 4.11: (a) Mean assembly traces contingent on the trial type and the performance of the mouse. (b) Schematic of fitting Gaussian Naive Bayes model to windows of assembly activation traces. Each feature is the sum of 1 s of activation for one assembly and is modelled by computing the mean and standard deviation separately for trials corresponding to each class.

We found that the strength of both local (within) and distributed (between) assemblies were increased following the tone during approach behaviour that would turn out to be successful (top-left, Figure 4.11a). However, the distributed assemblies

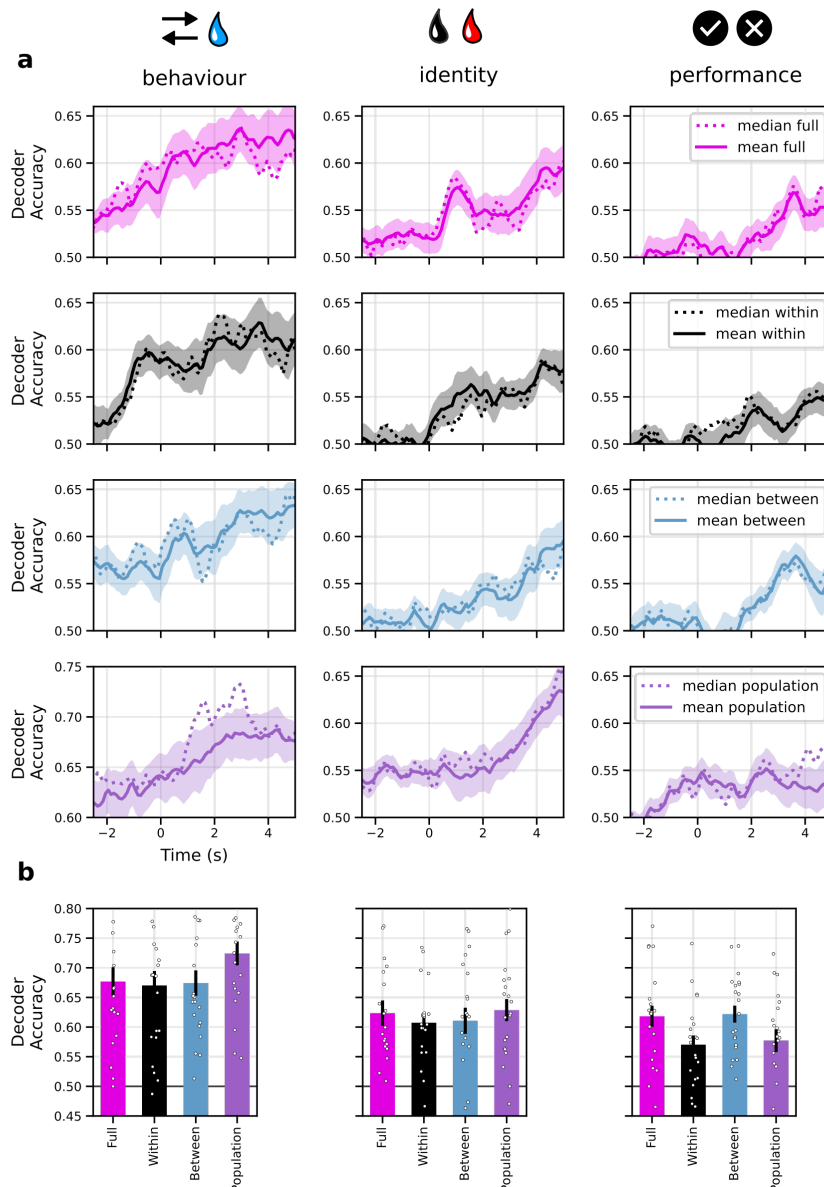


Figure 4.12: (a) Cross validated performance of the decoder applied to the responses with a sliding window starting from 2.5 seconds before the onset of the tone and stepping forwards in 200 ms increments. Error indicates 1 SEM, dashed line is the median across days, solid line is the mean. Note that the bottom left panel has a different scale to the others. (b) Comparison for the best prediction for each recording day for behaviour, identity and performance respectively.

did not show the same response during approach behaviour that would turn out to be incorrect (bottom-right, Figure 4.11a) while the local ones did.

In order to quantify this formally, we trained a Bayesian decoder on sliding windows of the assemblies' activity (Figure 4.11**b**) to predict which behaviour

the animal would perform (left-column Figure 4.12a), what the identity of the solution to be delivered was (centre-column), and whether or not the animal was going to make the correct choice (right-column). While decoding of behaviour and identity was similar between the two assembly types (Figure 4.12a,b), decoding whether the choice the animal made would be correct was better performed by the between-region interactions (Figure 4.12a,b paired t-test, $p=0.02$). To test whether this could be explained purely by the information contained within the rates of neurons across regions we also trained the decoder on the whole population of neurons. We found that the population decoder was not able to decode performance any better than the within-region assemblies (Figure 4.12a,b paired t-test, $p>0.5$) although interestingly it performed better than the other decoders in predicting the behaviour of the mouse (paired t-tests, all $p<0.05$).

Thus while we were able to decode the trial-type with only the within-region interactions, the representation of successful translation of the trial-type information into optimal behaviour was mostly found in the interactions between regions. Moreover, when the assemblies were tracked in full their predictive power equalled the best of the local or distributed interactions alone in each scenario. This implies that these two types of representation can co-exist within a larger one without interference.

Shifts in assembly decoding performance are not solely driven by behaviour

We controlled for the possibility that the differences in assembly-task predictions were just a readout of behaviour by finding periods outside of the task where the animal took a similar trajectory (see Figure 3.4). Using these surrogate periods we were unable to distinguish between local or distributed performance on any of the decoders. This suggests that these effects were not driven by e.g. locomotor feedback (all $p > 0.4$; Figure 4.13).

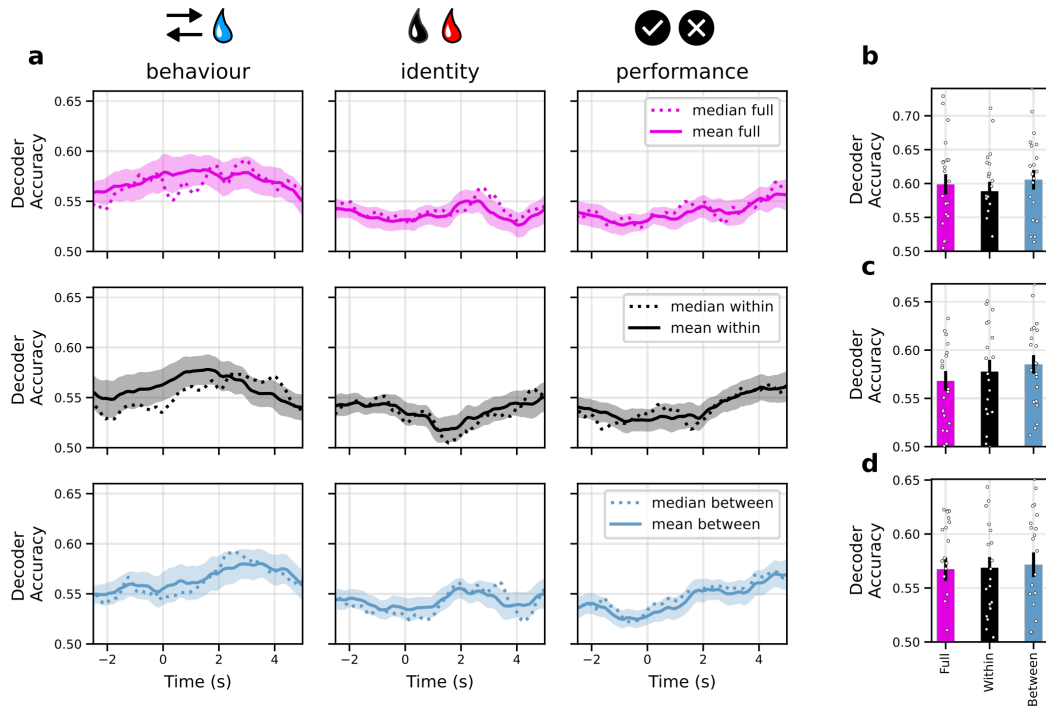


Figure 4.13: (a) as Figure 4.11c except during surrogate periods outside of the task where the animal had a similar trajectory profile. (b-d) as Figure 4.11d-f.

4.4 Discussion

In chapter 1 we outlined three possible simplified frameworks for understanding the passage from stimulus to response (Figure 1.2). Here we accompany those models with predictions about how their use in the brain might be reflected in this paradigm (Figure 4.14). In the sequential processing model we predict that the preponderance for the response to a stimulus be focused in a specific brain region would apply only for the areas that respond first, before the representation becomes more related to the behavioural output at each step. We found limited evidence for this in the observation that the amygdalar population had both the earliest response to the tone and the least indication of a response at the onset of behaviour. The equivalent representation of the task contingency (i.e. the meaning of the sensory information) between the local and distributed assemblies provided evidence for the divergent-convergent processing model. However we believe that our data most strongly reflect a processing model that supports multiple recurrent connectivity between areas. This is not only reflected in the fact that the information about the behavioural performance lies mostly in intra-area interactions, but also in that the time courses of these interactions reflect what would be seen on stimulation of a network with both high local and inter-area connectivity. This time course is presumed to reflect an initial linear increase as each region is area is engage and then a lagged, but larger increase as the combination of pairs of regions are engaged.

We believe the evidence in this chapter thus suggests a strong role for distributed representations in the translation of fragmented external stimuli into correct behaviour. Starting from this systems perspective, future work might aim to provide a more mechanistic account of these dynamics. We begin this work in chapter 6 by looking for oscillatory substrates of these representations, but other approaches such as computational modelling may also be fruitful. The goal of either of these would, in our eyes, be to find well-grounded targets for experimental manipulation and to provide causal evidence for the role of globally distributed representations through them.

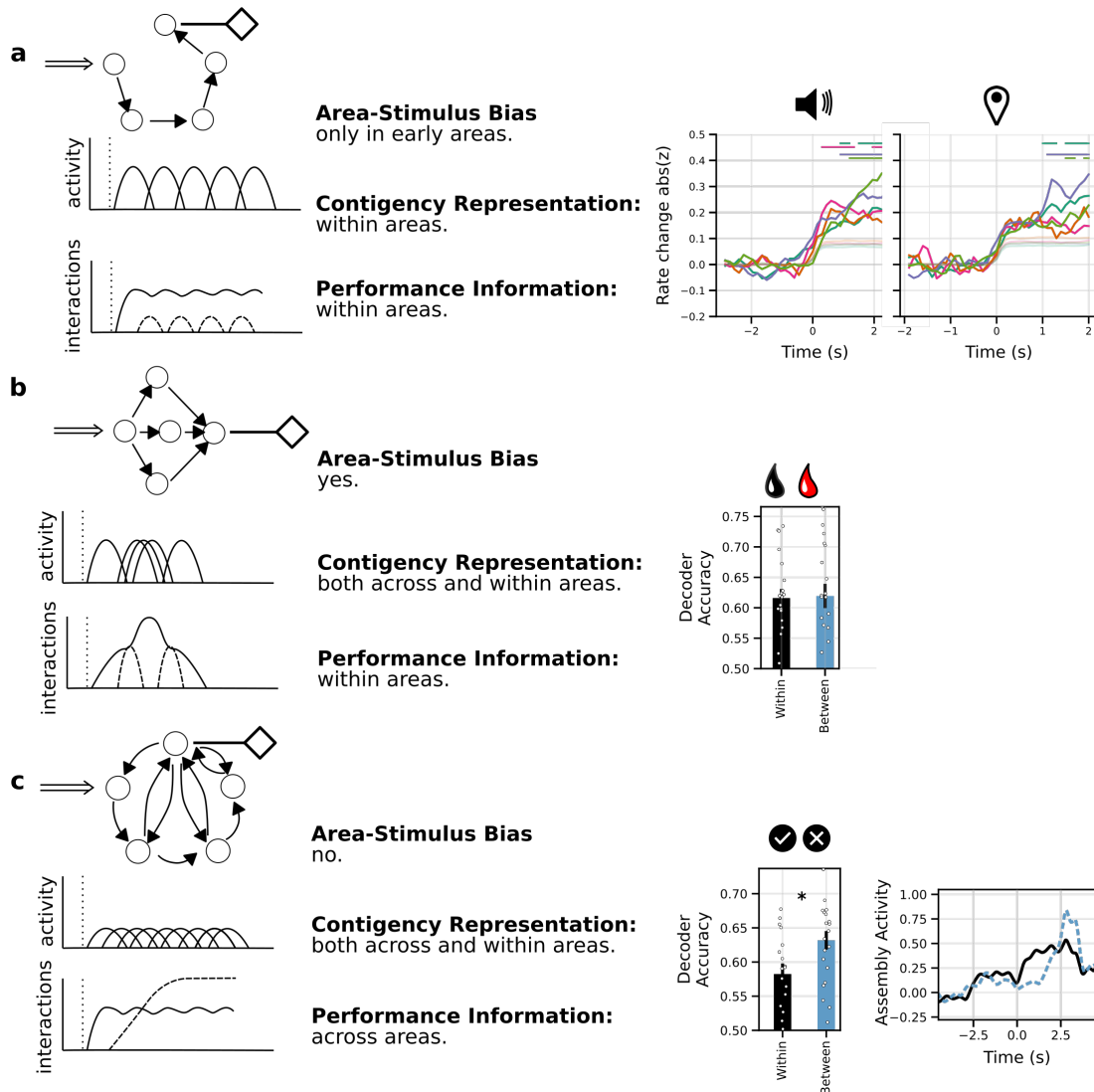


Figure 4.14: (a) The sequential model and Figure 4.4. (b) The diverge-converge model and Figure 4.12 identity bar plot. (c) The recurrent interaction model and the performance bar plot from Figure 4.12.

In the next chapter we consider in more detail the alternative approaches for analysing populations of interacting neurons. We introduce a new simple method for modelling patterns of interactions beyond co-activity and then try to build a framework around these population models in general. We use this framework to help guide decisions about when different approaches are most appropriate.

5

Population Dynamics – Comparison & development of methods for assessing population dynamics in multi-site, multi-unit recordings

Contents

5.1	Introduction	94
5.1.1	Motivations for assessing population dynamics	94
5.1.2	Defining interpretability in relation to models of population dynamics	95
5.2	Methods and Results	98
5.2.1	Overview of task-independent (single observation) methods	98
5.2.2	Overview of task-dependent (multiple observation) methods	104
5.2.3	Example results with models of population dynamics	105
5.2.4	Finding patterns of cross-correlations using ICA	106
5.3	Discussion	109
5.3.1	How interpretable are these models?	109
5.3.2	Choosing a Model	110

5.1 Introduction

There is a broad spectrum of methods for identifying and tracking groups of interacting neurons. Each of them have a different use-case according to the question being asked, as well as the experimental set-up.

5.1.1 Motivations for assessing population dynamics

Many methods for assessing population dynamics employ some form of dimensionality reduction, that is, they try to explain the data with fewer dimensions than there really are. Often this is applied over neurons, so instead of looking at how 100 neurons vary over the course of a task we can look at the k components that explain most of the variance in the activity of the neurons. That said, the goal of modelling population dynamics in general is not dimensionality reduction but finding a representation of the data that is in some way more interpretable. What exactly that means depends on the experiment and question that is being asked. Sometimes we aim to find the theoretical maximum amount of information a population contains about a stimulus. Sometimes we want to only take into account biologically informed measures. Sometimes we are using our models to generate hypotheses, and other times we are using them to test hypotheses.

Beyond the theoretical objectives there are also practical factors to consider such as the experimental design. For example, whether the data are structured into discrete (e.g. cued) trials, what non-neurophysiological data (e.g. behaviour and other physiological variables) are available, how many neurons can be recorded from and maybe even whether the model needs to be suitable to be included as part of a closed-loop system as in Deng et al. 2016 should all be taken into account.

In this chapter we will discuss a selection of previously developed neural population models and one novel one. First we try to define what can be considered an interpretable model in this context, as we often consider the term only intuitively. For illustrative purposes, we apply some of the models to a sample of the data set discussed in chapters 3 and 4. Finally, we try to synthesise this into a framework

that may serve as a useful starting point for deciding which model will be most appropriate under different circumstances.

5.1.2 Defining interpretability in relation to models of population dynamics

Here we try to codify some of our intuitions about what makes a given model interpretable. We use concepts of interpretability recently developed in the context of machine-learning where the use of complex models in public-facing and high-stake services (such as the legal system and healthcare) has resulted in less tolerance for approaches considered to be black boxes (Angwin et al. 2016; Lipton 2018; Lundberg et al. 2018).

What do we want from an interpretable model?

- **Trust:** It is desirable to be able to trust how the model will perform. That is given a particular data set you feel that you could predict approximately where the outputs of the model will lie. This is related to stability. For example linear regression is stable and with few enough coefficients it is easy enough to guess what the output is likely to be. For some stochastic models this may not be the case.
- **Causality:** Often the end goal is to find causal relationships, not just associations. While we cannot do this with analytical measures alone they can help guide experimental manipulations that might allow us to infer causality¹.
- **Informativeness:** A model might not just be able to provide information about neural activity via its outputs, but also by how it got there. For example any classifier can give a prediction but it is not always the case that the coefficients are meaningful in their own right.

¹Theoretically is not possible to analytically infer causality in a system where there are any unobserved dimensions. Practically it is sometimes possible if the number of unobserved dimensions is small. In the brain however, even if neural activity were the only dimensions that mattered 1000 neurons is still less than 0.001% of the total neurons in the mouse brain.

By what criteria can a model be evaluated to assess its interpretability?

We now try and break up these desirable but abstract characteristics of a model into the exact criteria that would define an interpretable model. First we address transparency which relates to how understandable the process of fitting the model is. We then consider post-hoc interpretability, which addresses how easy it is to interpret the results or output from a model independent of understanding how the model works.

Transparency

- **Simulatability:** We define simulatability as whether it is possible to comprehend the whole model at once. The class of model used is important here, it is usually easier to comprehend a linear combination of features than a highly non-linear one. However the size and sparsity of the model are important too. The smaller the number of features that affects each output, the easier understood they are. Under this definition a sparse model is more interpretable than a dense one because there are fewer features to consider. At the extreme a model whose output is driven by the activity of one neuron would be very simulatable, but potentially not very powerful.
- **Decomposability:** A decomposable model is one where each part affords a human-readable explanation. This can be a difficult criteria to achieve fully. For example even in the case of a linear regression we can understand the coefficients as the relationship of that feature to a target value only if we assume the features are independent.
- **Algorithmic transparency:** This corresponds to how easy it is to understand exactly what the model is learning, or how the features are extracted. One way to judge this would be to assess how well you can predict the output of the model qualitatively for simulated data. Another would be to assess how the outputs would change for a given transformation to data that has been previously fit.

Post-Hoc interpretability

- **Visualisation:** This refers to whether the model is easily visualised. We are specifically interested in whether those visualisations help us understand something about how the underlying neural activity is structured.
- **Explanation by example:** Being able to extract a valid prototypical example from the model can help us gain an intuition of how it is working. For example, for a given class label we might be able to extract representative feature sets for which that label might be predicted.
- **Principled input disturbance:** By modifying our inputs to the model (e.g. the spike times) we can make inferences about what the model required to perform well. For example in Figure 4.10 we shifted the spike times of each neuron relative to others by varying amounts to test which timescale of interaction the model relied on. Alternatively, we could add varying amounts of Poisson noise to test how stable the model is at varying SNRs, or add fake assembly patterns to check whether this impacts the detection of the real ones.

5.2 Methods and Results

5.2.1 Overview of task-independent (single observation) methods

Since the advent of multi-unit recordings in freely moving rodents, it has been common to record extended periods of spontaneous behaviour. This corresponds to periods of exploration, rest, and occasionally freezing behaviour. These types of recording are often used to ask questions of the effect of the context, and context-based learning. For example, assessing how the organisation of hippocampal CA1 neural activity changes in a familiar (learned) context versus a novel one (Wilson et al. 1993). One way to interrogate how the interactions between multiple simultaneous neurons change is to compare the correlation matrices of the binned spike counts (or equivalently as the co-variance matrices of the z-scored spike counts as in Equation 4.8). Moreover, it is possible to project these correlation matrices back into the time domain to track how they change over time. This can be equally well applied across regions as well: for example Girardeau et al. 2017 showed reactivations between the hippocampus and amygdala during non-REM sleep following aversive conditioning. However only using the correlation matrix restricts us to comparing how correlations change as a function of the global average. We cannot identify patterns of correlation that are strong and potentially meaningful if they only occur in time-limited bouts.

Dimensionality Reduction et al.

Applying dimensionality reduction to the spike counts allows us to separate out the patterns that might be important but could have been lost in the correlation matrix that represents the global average. All of the the following methods aim to describe our N dimensional neural activity in k dimensions where $k \ll N$. Where they differ is in what features they search for to generate these k components.

Dimensionality reduction can also be seen as a form of denoising: we reduce the number of, or eliminate, dimensions that are not informative for our purposes. Downstream that might be in order to better visualise changes in activity, to

train a better performing decoder, or to only use dimensions that we believe have some intrinsic biological meaning (such as patterns of co-activity representing Hebbian cell-assemblies).

PCA The most widely used method for dimensionality reduction, in all of its applications, is principal components analysis (PCA). PCA finds the linear combination of k dimensions that explain the most variance (Equation 4.9). Nicolelis et al. 1995 and Chapin et al. 1999 first used the technique to tease apart multi-dimensional representations in the somatosensory system. More recently, several advances were made in Peyrache et al. 2010 where they introduced an analytical method for computing the number of cell assemblies in a given recording by finding the number of eigenvalues above the Marchenko-Pasteur threshold (see "Assembly Detection" section of the methods in chapter 4).

Peyrache et al. 2010 also introduced a method for tracking the activity of the principal components (putative cell assemblies) at high temporal resolution, by projecting the outer product of each eigenvector back onto the spike matrix Z' , where Z' can be a spike matrix sampled at a higher rate than the one on which PCA was originally applied (or even from a different recording epoch). Further setting the diagonal of the projection matrices P to zero allows tracking changes in the correlation structure between neurons independently of their rate. In real multi-unit recordings, spikes may be mis-assigned to a different neuron which can lead to spurious assembly activity. In order to mitigate these effects, they suggest setting the values in the projection matrix P corresponding to neurons from a common tetrode to zero.

PCA + ICA While the PCA approach described above give us a reasonable estimate of the subspace in which the informative activity patterns might lie, it does not isolate them: multiple subsets of cells that are co-active together will be mixed together in single components (Lopes-dos-Santos et al. 2013). However, once we have the subspace which has the correct number of dimensions, we can employ demixing (or blind source separation) algorithms in order to isolate the different

activity patterns. In Lopes-dos-Santos et al. 2013 the authors demonstrate the use of ICA for this isolation procedure (see chapter 4).

FA Instead of PCA or ICA, another approach would be to use a generative model such as factor analysis (FA). Instead of preserving variance (like in PCA), FA preserves the correlations between the input features. However, like the other methods described above FA does not model any of the relationships between bins: the model does not take into account that the bin at timepoint 1 is next to the bin at timepoint 2. We can try and work around this by smoothing with a kernel, but we do not know what the width of said kernel should be a priori. Yu et al. 2009 introduce Gaussian Process Factor Analysis (GPFA) where each factor is constrained to have a smooth trajectory through time but the smoothness of each factor is modelled separately based on the reconstruction error. GPFA is often used in studies of motor outputs where the dynamics of the whole population is assumed to be in the service of the current behaviour.

The above tools are likely sufficient for the detection, and interrogation, of patterns of co-activation between populations of neurons. However, as stated above, our goal might instead be to find the set of dimensions that reflect specific aspects of the task in which the system is engaging. In this case we can use a supervised approach to search for the most informative dimensions.

LDA Using LDA we label the observations in one class or another and then explicitly find the linear projection which maximises the discrimination between those classes. This is especially useful in the case where the unsupervised principal components are likely to be dominated by variance that is shared between classes (e.g. Bartho et al. 2009).

Graph (Network Theory)-based methods

Models based on dimensionality reduction have some limitations. Firstly, by fitting models to the pairwise co-variance (or correlation) matrices, we assume some stationarity which is likely not fully justified (note this limitation does not fully apply to GPFA). Secondly, although some heuristics (e.g Marchenko Pasteur) or

surrogate measures can be used there are few well-defined null models to use with the above approaches. Finally, we have to make an assumption about how to find the meaningful dimensions in our data. In the case of PCA we choose those that explain the most variance, with ICA we force them to be statistically independent, and in LDA we use our task labels. These choices might be principled, but it is entirely possible that whichever method we choose might be discarding valuable information contained in higher-order correlations.

We mention these limitations here as they are all potential targets for another class of approaches to modelling populations based on graph theory (Humphries 2017). In this class of approaches we still generally aim to characterise a $N \times N$ matrix of pairwise interactions, but are free to choose from a wide range of interaction measures, not just those based on co-variance. For example, we could use information-based measures such as transfer entropy to construct a directed graph, or rate-corrected correlation measures for an un-directed one. Using graphical approaches we not only have access to a wide range of null models against which to test those fitted to our data, but also a wide range of statistics that characterise the graph itself. An example of a null model which could be powerful in the study of neural populations is the configuration model (Chung et al. 2002), which is a random model with fixed expected degree sequences allowing us to test whether the interaction between a given pair of neurons is expected given the number of interactions each of those neurons makes. The combination of null models, and well-validated graphical attributes allows us to rigorously test for changes in a given network. However it can be difficult to know exactly how to interpret these attributes in the context of a neural sub-population (Stumpf et al. 2005). Finally, in weighted graphical models we can keep each interaction between every pair and so are not forced to discard higher-order information that might be meaningful. However the toolbox of metrics that are easily available for unweighted networks is not as large for weighted ones. Nonetheless graphical models are an exciting area that we believe neuroscience, and specifically neurophysiology, has not yet taken full advantage of. While it is perhaps trickier to make qualitative statements

about the change in a network, there is a lot of potential for their use, especially in strongly hypothesis-driven research. The robust quantitative descriptions and null models make lighter work of trying to figure out how much things are changing even if it can be more difficult to understand what that means.

Temporal Warping - "Regressive" Models

Until now we have only considered zero-lag correlations and thus have ignored the possibility of meaningful sequences of activity, or even syn-fire chains (Abeles 1982; Bienenstock 1995; Izhikevich 2006).

One way to approach this problem is to extend any one of the dimensionality reduction techniques by concatenating the original matrices with shifted versions: for each feature (neuron) we could add another feature that contained the value of the original feature in the following bin. However, increasing the number of features (for each lag we need to add N features) in a fixed length recording results in a loss of statistical power for detecting assembly patterns and in practice this loss is too great for the method to be usable.

Two recent approaches combat some of these issues by using binary matrix theory (Russo et al. 2017) or by working directly with spike-trains and using an optimal-transport type measure similar to the Victor Purpura distance (Grossberger et al. 2018; Victor et al. 1996).

The crux of the issues with testing for interactions at multiple time-lags lie in keeping the algorithm computationally tractable while maintaining statistical power. Russo et al. 2017 apply three separate measures in order to achieve this. First in order to only deal with binary matrices which follow a specific (hyper-geometric) distribution, they split any vectors with spike-count $c > 1$ into c vectors that sum to make the original one. This allows them to use a parametric test without losing information by having to threshold the spike-counts. Next they test whether a pair of neurons A and B have more spikes in common than they would by chance using a difference statistic $\#_{ABBA,l} = \#_{AB,l} - \#_{AB,-l}$. The principle behind this is that non-stationarities in the data (e.g. a change in the base rate for all neurons) should

affect lags l and $-l$ equally, cancelling each other out. Note that not modelling non-stationarities such as the population rate is an opinionated choice. Coherent changes in the firing of two neurons for any reason, be it slow covariation or monosynaptic connectivity, may equally contribute to synaptic plasticity. One trade-off could be losing the ability to detect any pattern with a symmetric lag distribution potentially affecting "reverse replay" as in Foster et al. 2006 although perhaps those replay events would not be so perfectly symmetric for this to be an issue.

Finally they use a recursive merging procedure to detect higher-order patterns from pairs of neurons. Essentially, once a pair has been merged, their spikes are counted as if they were a single neuron in the above procedure. The procedure is then repeated until no more pairs remain. This is computationally efficient, but could result in missing any higher-order assemblies which do not have significant subsets.

Instead of using heuristics to be able to test for patterns at multiple lags Grossberger et al. 2018 compute the set of all cross-correlograms for a given epoch before clustering the epochs based on a (dis)similarity measure. The measure they use to compare the cross-correlograms is based on optimal transport theory, i.e. how far spikes need to be moved in order to make these cross-correlograms as similar as possible. Unlike in Victor et al. 1996 they do not allow the insertion or deletion of spikes, which removes some of the computational burden. Nonetheless, this approach is advantageous in the sense that it allows for jitter in the relative spike times and it allows matching complex shapes in the cross-correlogram.

A key hyperparameter in this model is the choice of epoch length over which each cross-correlogram is computed as this determines the timescale on which the interactions between neurons are assumed to change. It should be noted that in all of the simulations in this paper, the epochs used for generating the ground truth were aligned with the epochs used for detection. The authors assert that this provides no extra information to the clustering step, however they do not support this empirically. For real data, perhaps overlapping or sliding epochs would be a better choice if not too computationally demanding.

It is also worth noting that neither of these models allow for tracking of assemblies at high temporal resolution, or at least they do not suggest how it would be possible to do so. Indeed it is not precisely defined when exactly an assembly detected at multiple lags can be said to be active. Nevertheless, while this is not an explicit limitation of these models, it is something to be taken into consideration if one wishes to relate assembly patterns and fast oscillations in the LFP, for example.

Other methods not discussed here include those based on Hidden Markov Models (Maboudi et al. 2018) and convolutional NMF (Mackevicius et al. 2019).

5.2.2 Overview of task-dependent (multiple observation) methods

While often used on trial-structured (repeated measure) data none of the previous methods can be explicitly fitted to third order tensors $Neurons \times Time \times Trials$. As such, they cannot explicitly model the fast changes over time as well as the slower changes across trials and thus do not take full advantage of this experimental design. While there are supervised methods to do so (Vidaurre et al. 2019), we focus on an unsupervised method introduced in Williams et al. 2018 called Tensor Component Analysis. Broadly speaking this method is an extension of PCA to third order tensors² (Figure 5.1a). This method not only allows for much greater dimensionality reduction, but also allows us to visualise the "factors" of each component in each dimension. That is to say we can not only view the neuronal loadings of a given component, but also the progression of this component over time (i.e. over a "typical" trial), or over trials (i.e. how this component evolves over e.g. learning) using this method. Previous attempts to avoid averaging over trials involved concatenating the trials and time dimensions before applying PCA (Figure 5.1b).

While the authors suggest, and demonstrate, the utility of TCA in analysing data sets with externally-imposed trials, it could be equally useful for internally-driven periods that are well defined in time. For example it could be applied

²Specifically they introduce a non-negative version of TCA which introduces more sparsity into the components and helps to make them interpretable

to neural activity centred around sharp-wave-ripples or dentate spikes to identify how they evolve during bouts of sleep.

5.2.3 Example results with models of population dynamics

TCA is a good example of a model that has excellent post-hoc interpretability as the ability to visualise each set of factors independently is very powerful. We found that by applying the unsupervised model to a single recording day it uncovered one component who's temporal factors indicate that it was driven 700 ms after the onset of the tone and who's trial factors distinguish between the two trial types³ (Figure 5.1c; see chapter 4 for a description of the task). However it scores lower in terms of its simulatability, as although the definition of PCA applied to third order tensors is intuitive, the actual algorithm for fitting the model (canonical polyadic decomposition) is more complex.

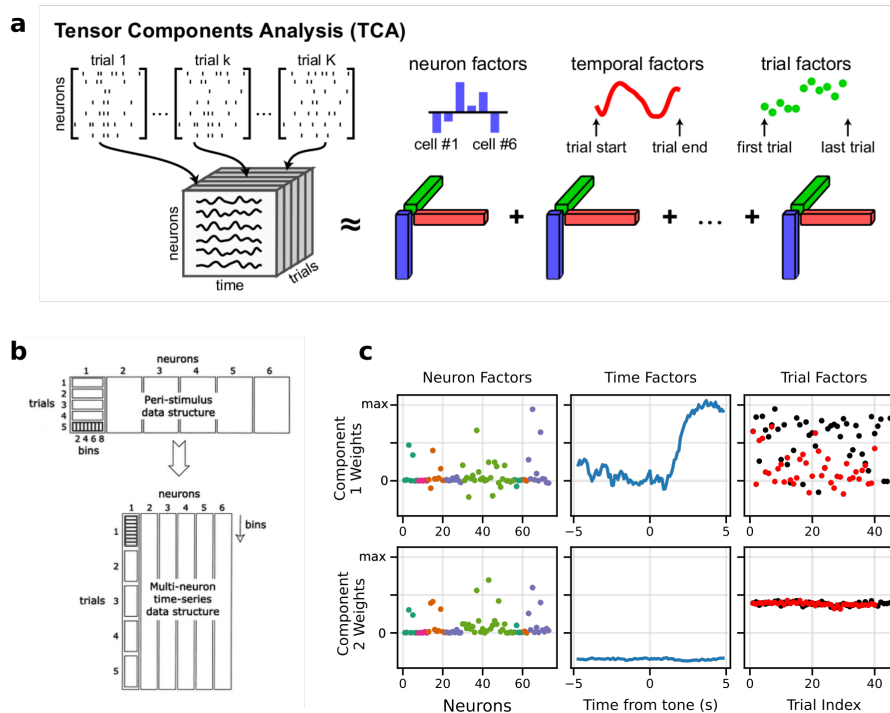


Figure 5.1: (a) The TCA approach. Reproduced with permission from Williams et al. 2018. (b) The PCA trial concatenation approach. Reproduced with permission from Laubach et al. 1999. (c) The TCA approach applied to one recording day from chapter 4.

³Remarkably they actually distinguish the contingencies better than suggested by the behaviour of the animal.

In order to illustrate the comparative utility of each, we applied three more of the approaches described here to the same recording day. Using PCA we found that the first PC of the trial averaged neuron firings rates the first was expressed more strongly in the sucrose versus the quinine condition (Figure 5.2a). Likewise we could see differences in the correlation structure during, but not before, the tone in the sucrose versus quinine conditions (Figure 5.2b). LDA also found a projection that discriminated between the two conditions after the tone (Figure 5.2c).

Thus in general the results from Figure 5.1 and Figure 5.2 are convergent. They both indicate components only discriminate between the two conditions after the onset of the tone, and not before. Indeed, for the PCA, LDA, and TCA approaches all suggest the components start to discriminate 1 second after the onset. They also all indicate that neurons from all regions appear to play a role in the activity of the discriminating component.

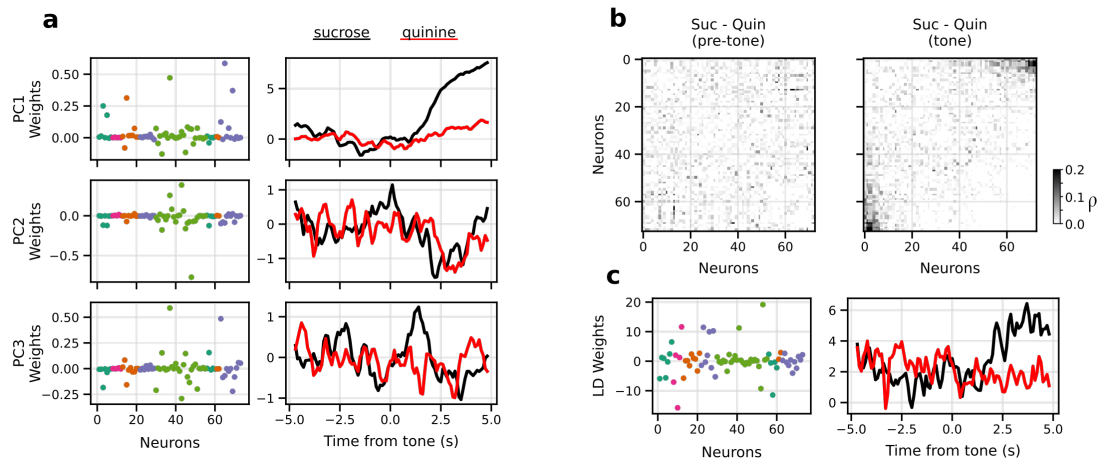


Figure 5.2: (a) The first 3 principal components of the trial averaged neuron firing rates. (b) The difference between the correlation matrix between all neurons in the sucrose versus quinine condition before (left) and during (right) the tone. (c) The component obtained via LDA with sucrose and quinine trials as labels.

5.2.4 Finding patterns of cross-correlations using ICA

Whilst we did not test the regressive models suggested by Russo et al. 2017 or Grossberger et al. 2018 (as the models are not trivially implemented) we did

investigate another approach along similar lines for finding patterns of co-activity at multiple lags by looking for significant patterns of cross-correlations.

To do this we first found the cross-correlation between each cell-pair j and k (Figure 5.3a,b):

$$C_{jk}(\tau) = \sum_{t=1}^T x_j(t)x_k(t + \tau) \quad (5.1)$$

We then stacked the resulting vectors to create a 2-dimensional $\#_{lags} \times N^2$ matrix where N is the total number of neurons (Figure 5.3c). This was repeated for all recording days and the resulting matrices concatenated along the first dimension. We then applied PCA to the concatenated matrices and used the Marchenko-Pasteur threshold (as in Equation 4.9) to compute the number of significant principal components. In the reduced subspace occupied by these components we found the un-mixing matrix W such that each pattern of cross-correlations would be as independent as possible (Figure 5.3d). Note both of the previous steps are analogous to the ones taken to find patterns of co-activity in chapter 4 and Lopes-dos-Santos et al. 2013.

We then projected each pattern back onto the cell-pairs and found the mean weight for each pattern amongst cell-pairs corresponding to each pair of regions (Figure 5.3e). Many of the patterns offered an easily interpretable explanation. For example the first pattern in panel e corresponds to short-latency (putatively monosynaptic) connections that occur more often within VTA cell pairs. The fifth pattern likely corresponds to theta coupling at zero lag, which has strong components within the CA1 and VTA, while the third pattern corresponds to theta coupling with a 0 ms lag, and is seen most strongly between the CA1 and NAc. These results highlight the potential utility of applying unsupervised pattern-detection algorithms to cross-correlations as a hypothesis generating tool.

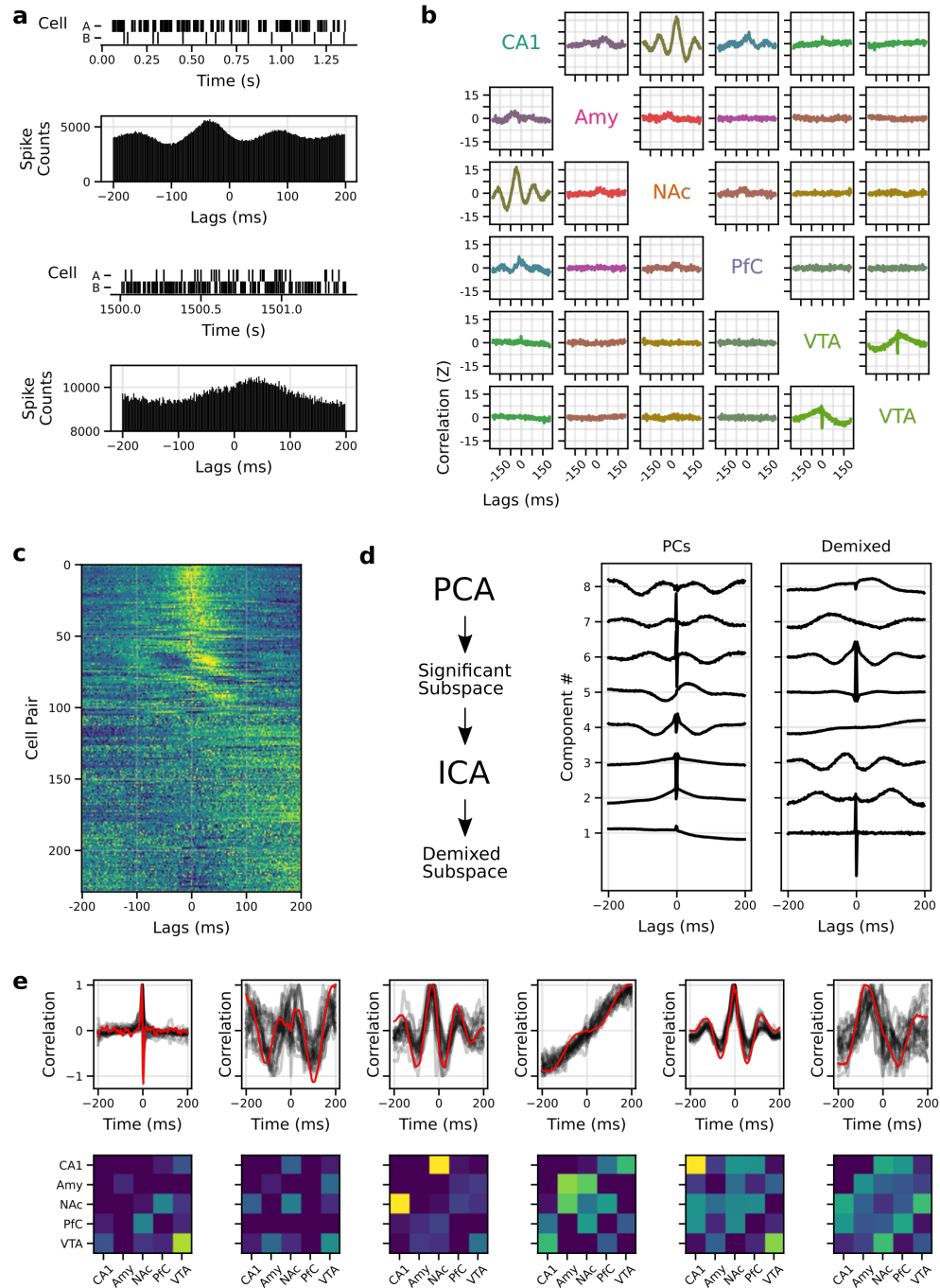


Figure 5.3: (a) Spike times and associated cross correlations for two pairs of units. (b) Cross-correlation matrix for 6 neurons from one recording day. (c) Stacked cross-correlations from all pairs with significant cross-correlations over multiple days. (d) Patterns of cross-correlations found using PCA or ICA. (e) For each some of the ICA patterns shown in (d) we show their mean projection against all the pairs for each pair of regions.

5.3 Discussion

5.3.1 How interpretable are these models?

According to our framework, PCA ranks somewhere in the middle for simulatability as the model is linear but dense (although it is possible to add sparsity constraints as in Zou et al. 2006). In terms of decomposability it is not ideal: as shown in Lopes-dos-Santos et al. 2013 the resulting components contain a mixture of assembly patterns. In this sense ICA performs better, although potentially at the cost of algorithmic transparency. In both cases the post-hoc interpretability of the models is good as they afford easy visualisation of the full model via their projection matrices.

GPFA maintains the decomposability advantages of ICA and additionally may have better post-hoc interpretability as a result of allowing the factors to vary smoothly over time. This comes at the cost of some algorithmic transparency as it requires the joint fitting of Gaussian processes and factor analysers.

Weighted network models are not simulatable as they contain every n th-order interaction between each neuron. Most of the measures computed on network models involve searching for the shortest paths between various pairs of nodes and the algorithms used to do so are transparent. The decomposability of the model depends on the measure used and in the context in which it is used. The global efficiency of a network has a clear interpretation in the case where we have full sampling of the network but less so in other circumstances (Stumpf et al. 2005). Graphical models afford a wide array of visualisations but it is not always clear how exactly these should be interpreted in the context of a neural population.

By including each feature at multiple lags regressive models will be larger than their non-regressive counterparts and so are less simulatable. Due to the statistical and computational constraints their algorithms are also less transparent. However, by allowing the visualisation of the lags and neurons that define a given pattern they have good post-hoc interpretability. Moreover, by including directional terms, they are the only class of model discussed here which are likely to be useful for guiding estimations about causal relationships between neurons.

5.3.2 Choosing a Model

Finally, we attempt to synthesise the above into a guide for the process of deciding which of the multitude of population models to select from under various circumstances. We propose that TCA and LDA are the most appropriate places to start for data that is trial structured, with LDA being only applicable for labelled data as it is a supervised method and TCA being better placed to assess changes over the course of trials. Graphical models seem well suited for studying slow changes in a quantitative fashion, and for testing an already well-defined hypothesis. For tracking sequences or patterns of activity, the methods by Grossberger et al. 2018 and Russo et al. 2017 are appropriate. Our suggested approach of applying ICA to the cross-correlation matrix may be useful for when maintaining simulatability is more important than statistical power. PCA and ICA methods can be used for tracking patterns of rate fluctuations and co-activity. They are especially useful in situations where the timescale of the patterns is to be determined a priori. For a model that allows flexibility in this timescale GPFA is more appropriate.

This guide is not intended to be prescriptive nor definitive but we believe that this highlights our underlying thesis that each of these methods cannot be usefully ranked outside of the specific context in which they are going to be applied.

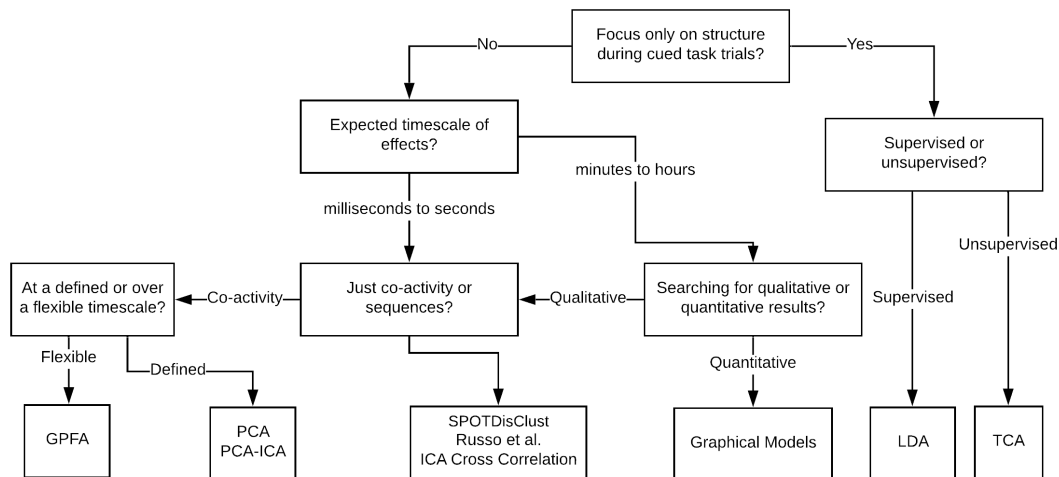


Figure 5.4: Flow chart to help guide decision about which model of population dynamics is appropriate for use in a given context.

6

Searching for global oscillatory states underlying long range coordination

Contents

6.1	Introduction	112
6.1.1	Long-range Coordination	112
6.1.2	Pairwise approaches for characterising multiple local field potentials	113
6.1.3	Multivariate approach for characterising the interaction between more than two local field potentials over time .	113
6.2	Methods	115
6.2.1	Preprocessing	115
6.2.2	Spectrograms	115
6.3	Results	117
6.3.1	Transient oscillatory states are driven by task-timings .	117
6.3.2	Transient oscillatory states support the emergence of brain-wide assemblies	117
6.4	Discussion	123

6.1 Introduction

6.1.1 Long-range Coordination

In chapter 4 we suggested that oscillatory states might support the long-range, short-timescale co-ordination between groups of neurons that we observed. This was motivated by several factors. First we know that oscillations can be coordinated across multiple regions (Buzsáki et al. 2004). Moreover, we know that synchrony between oscillating regions is driven by the extent of their recurrent connectivity (Singer 2018; Stettler et al. 2002). And recurrent processing is one of the key signatures that we posit underlie globally distributed representations.

Furthermore, coupling to specific phases of a given oscillatory rhythm could allow the formation of robust distributed assemblies by facilitating synaptic plasticity in response to common inputs (Huerta et al. 1996; Westat et al. 2004). However, the timescale required for facilitating selective synaptic plasticity would require coherence at frequencies in the gamma range for which reciprocal coupling may not be sufficient to maintain in the face of long conduction delays. Nonetheless, although the mechanisms are not yet clear, coupling at these frequency ranges has been reported in multiple distant regions (Canolty et al. 2010; Popescu et al. 2009) but there are fewer studies reporting similar observations in rodents (Buzsáki et al. 2013).

There is a wide body of work documenting interactions between local field potentials recorded from different regions at lower frequencies, primarily in the theta band (7-12 Hz) or around 4 Hz. Neuronal coupling to theta recorded in the hippocampus has been found in all other regions recorded here (Amygdala - Paré et al. 1996, Accumbens - Berke et al. 2004, Trouche et al. 2019, PFC - Siapas et al. 2005, VTA - Gomperts et al. 2015). Similar findings have also been presented for 4 Hz, a distinct oscillation likely driven by breathing-related activity (Fujisawa et al. 2011; Karalis et al. 2018; Karalis et al. 2016). Moreover coherence of both of these oscillations has been linked to performance in working memory tasks (Fujisawa et al. 2011; Jones et al. 2005). However as we found that shifting spikes by 10s of

milliseconds was enough to reduce assembly strength (Figure 4.10), we focused on oscillations above 20 Hz as candidates for coordinating their activity.

Importantly we are not constrained to mechanisms that allow multiple assemblies to be processed without interference such as locking to different phases of the same oscillation as in communication through coherence theory (Fries 2005). We assume the selection of neurons within a region can take place independently, and performance may only require one distributed representation to be active at a time. In other words, our hypothesis is that if there were an oscillatory mechanism for distributed synchrony it would act as a reporter for a switch between intra- versus inter-regional processing. This switch in processing types may be analogous to the sharp transitions to transient high-power attractor states predicted by modelling work (Lundqvist et al. 2011) and evidenced in a primate working memory task (Lundqvist et al. 2016).

6.1.2 Pairwise approaches for characterising multiple local field potentials

In prior studies where multiple (more than two) sites have been recorded simultaneously interactions between oscillations in the local field potential in rodents have almost exclusively used pairwise measures in order to characterise amplitude and phase coherence (Engel et al. 2001; Fujisawa et al. 2011; Herry et al. 2008; Karalis et al. 2016; Montgomery et al. 2007). This is likely as a result of pairwise interactions being better understood in general as well as the much smaller size of the methodological toolbox of methods available for multivariate analysis.

6.1.3 Multivariate approach for characterising the interaction between more than two local field potentials over time

The multivariate spectral analysis techniques that are available have been more broadly adopted by the cognitive neuroscience community, especially those using EEG and MEG (Allen et al. 2014). As such, we collaborated with Diego Viadurre and Mark Woolrich at the Oxford Centre for Human Brain Activity who recently

developed a method (referred to as the HMM-MAR) to describe multi-region time series as a series of states with high spectral and temporal resolution (Vidaurre et al. 2016). This contrasts with other commonly used (Allen et al. 2014) and even state-of-the-art (Hultman et al. 2018) methods that require the use of sliding windows.

The HMM-MAR decomposes a set of simultaneously recorded time series into a series of discrete states. Each state is characterised by a MAR (Multivariate Auto-Regressive) model which, in more familiar terms, relates to both the power spectral densities of each channel and the coherence between channels. Thus, in our application, the HMM-MAR allows us to identify repeating patterns of transient oscillatory states in an unsupervised manner.

Using a Hidden Markov Model as our structural model, we assume that the system which is generating the LFP recordings is moving through a set of states which we cannot directly observe. This allows us to consider each a state as a quasi-stationary process, making it easier to parameterise. In contrast to a windowed approach where the window length is fixed and chosen a priori, the length of each state is learned from the data. Therefore we need not have strong assumptions about how long the data remain approximately stationary¹. The auto-regressive nature of our observational model, a MAR, allows us to capture frequency information, and the multivariate nature of the MAR allows us to capture coherence between channels.

We applied it to our task-aligned data to see which, if any, states are up- or down-modulated during the execution of the task. Finally, we examined whether these states are preferentially coupled to the local or globally distributed activity of the previously described assembly patterns.

¹However there is a hyperparameter which sets our prior of the diagonal on the state-transition matrix i.e. our prior on whether given the state for the current sample the next sample should have the same state assignment. This will directly affect the lifetimes of the states but it is still possible to have orders of magnitude of difference between the median lifetimes of different states.

6.2 Methods

6.2.1 Preprocessing

All LFPs were taken from the first channel on the first tetrode for each region. The only selection criteria were that the channels were not visibly more noisy than the other channels on the same tetrode. If they were, then another channel was selected. The LFPs were first down-sampled from 20 kHz to 1250 Hz after spike sorting. Then LFPs for Figure 6.1 and Figure 6.2 were further down-sampled to 250 Hz for computational efficiency with the HMM-MAR.

6.2.2 Spectrograms

The spectrograms in Figure 6.1 were computed with a set of complex Morlet Wavelets with 10 scales and main frequencies from 1 Hz to 250 Hz at 1 Hz steps. The spectrograms were computed separately for each trial for all recording days and then averaged. To allow simultaneous display of a wide range of frequencies the spectrogram was normalised by the mean power of each.

Fitting the HMM-MAR

The full details of model-fitting, and other application of the HMM-MAR can be found in Vidaurre et al. 2016. We set the number of initial states $K = 6$, the MAR order $p = 5$ and used full unique covariance matrices for the noise of each state. The Dirichlet diagonal was set as $\frac{T}{4}$ where T is the number of samples. All other parameters were left at their default settings. We trained the model on a 20 second period around the tone for each trial. States were matched across days according to the similarity in their PSD profiles.

Lead - Lag Relationships

Lead-lag relationships were identified by taking snippets of the LFP during each state occurrence and computing the cross-correlation between each pair of channels. State occurrences with lifetimes shorter than double the number of lags we wished to estimate were not used. First a 6 pole elliptic (Cauer) filter with 0.5 dB ripple

in the passband and 30 dB attenuation in the stopband was applied to the LFP with a high-pass cut-off at 20 Hz to prevent peaks being dominated by theta oscillations. We then assigned each channel pair within each snippet a lead (1), lag (-1), or zero relationship if the peak in the cross-correlation was less-than, above or at the centre.

6.3 Results

6.3.1 Transient oscillatory states are driven by task-timings

To determine whether there were any gross changes in the local field potentials at the onset of the tone we first took the spectrogram for all trials and computed the grand average (Figure 6.1a-e, left column). We saw a robust increase in amplitude only in the VTA (8-20 Hz). This was accompanied by a deflection in the mean LFP over all trials showing that the timing of event was reliably locked to the onset of the tone. This deflection was present in trials in both the sucrose and quinine conditions (Figure 6.1).

We then applied the HMM-MAR to the same task epochs to see if there were transient oscillatory states that were hidden in the grand-average (Figure 6.2). Of the six states detected, we found that three were significantly modulated during the task. In the sucrose condition states 1 and 2 were up- and down- modulated respectively within the first 1s after the tone and then were slowly replaced with state 3 (Figure 6.2a). However, in the quinine condition neither state 1 nor state 2 were altered at any point (Figure 6.2b).

6.3.2 Transient oscillatory states support the emergence of brain-wide assemblies

State 1 rapidly disambiguate between the meaning of the tone in the two contexts and also had unique spectral properties (Figure 6.2c,e) characterised by peaks in power and coherence at 32-38 Hz between all five regions except the PFC. According to the peaks in the pairwise cross-correlations between each region during the occurrence of this state we determined that the NAc was leading the other regions at this frequency. This was most strongly the case for NAc's relationship with the amygdala and the VTA (Figure 6.2; Adhikari et al. 2010). This effect was either absent or attenuated in the other states Figure 6.3. We chose to avoid the use of more complex directionality-estimating approaches such as Granger causality in order to facilitate interpretation of the results and to avoid their propensity for fitting noise (Winterhalder et al. 2005).

Finally we considered whether the widespread gamma-band coherence observed during this state might reflect or facilitate the co-ordination between structures. We found that this gamma-related state (state 1) led to an increase in distributed assembly activity and a decrease in local activity (Figure 6.4**b,d**). State 2, on the other hand had no clear relationship to distributed assembly activity but was related to an increase in local activity (Figure 6.4**a,c**). However, we note that these conclusions were drawn from the mean profiles (Figure 6.4**c,d**) which do not fully reflect the variety of temporal relationships seen in individual state-assembly examples (Figure 6.4**a,b**). Future work should address this by describing the relationship between assembly and state time courses in a manner that can better characterise these variations.

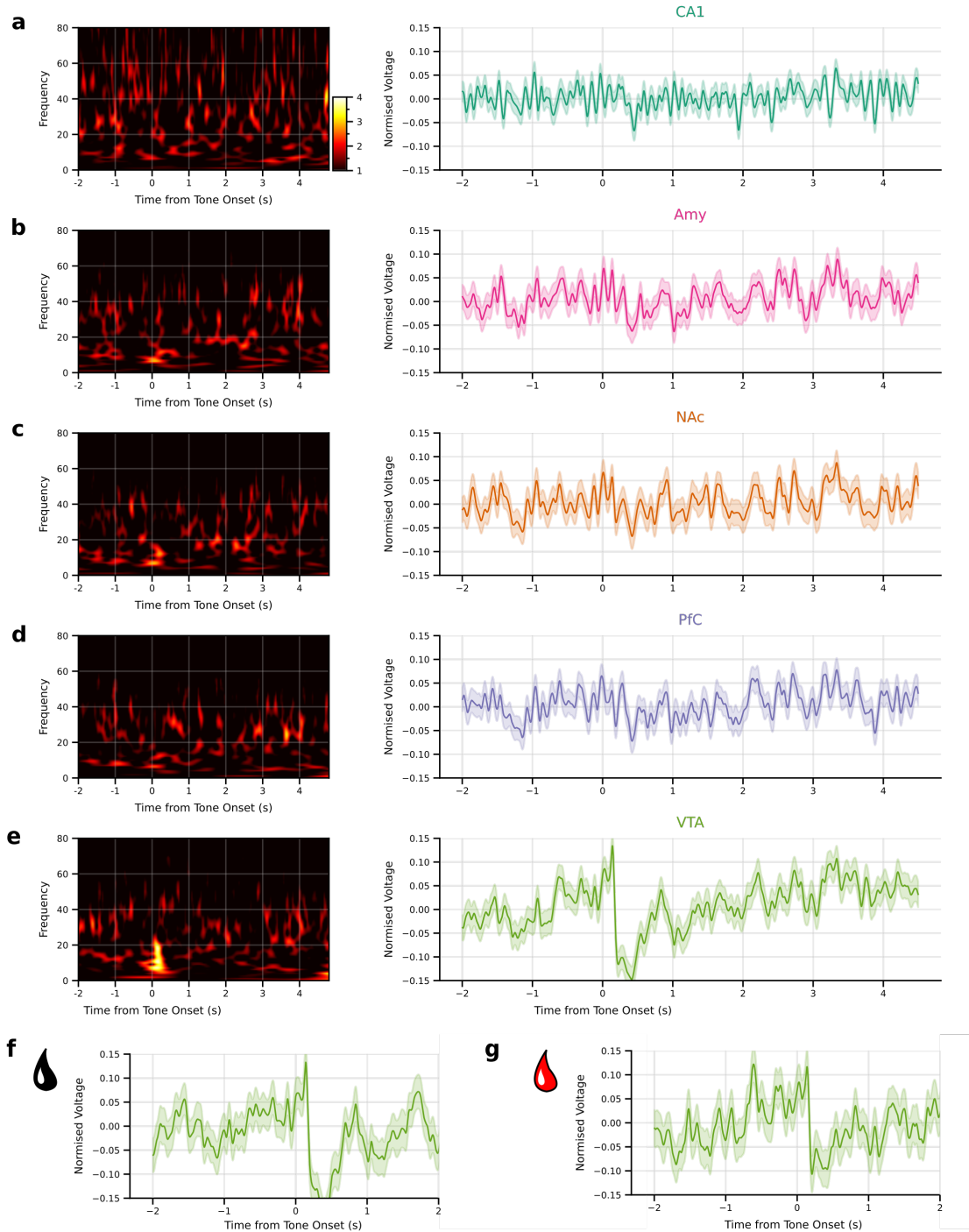


Figure 6.1: Spectrograms and LFPs from the CA1 (a), Amygdala (b), NAc (c), PFC (d) and the VTA (e) averaged over all task trials. The mean LFP is also shown for the VTA restricted to sucrose (f) or quinine (g) trials. Error is ± 1 SEM.

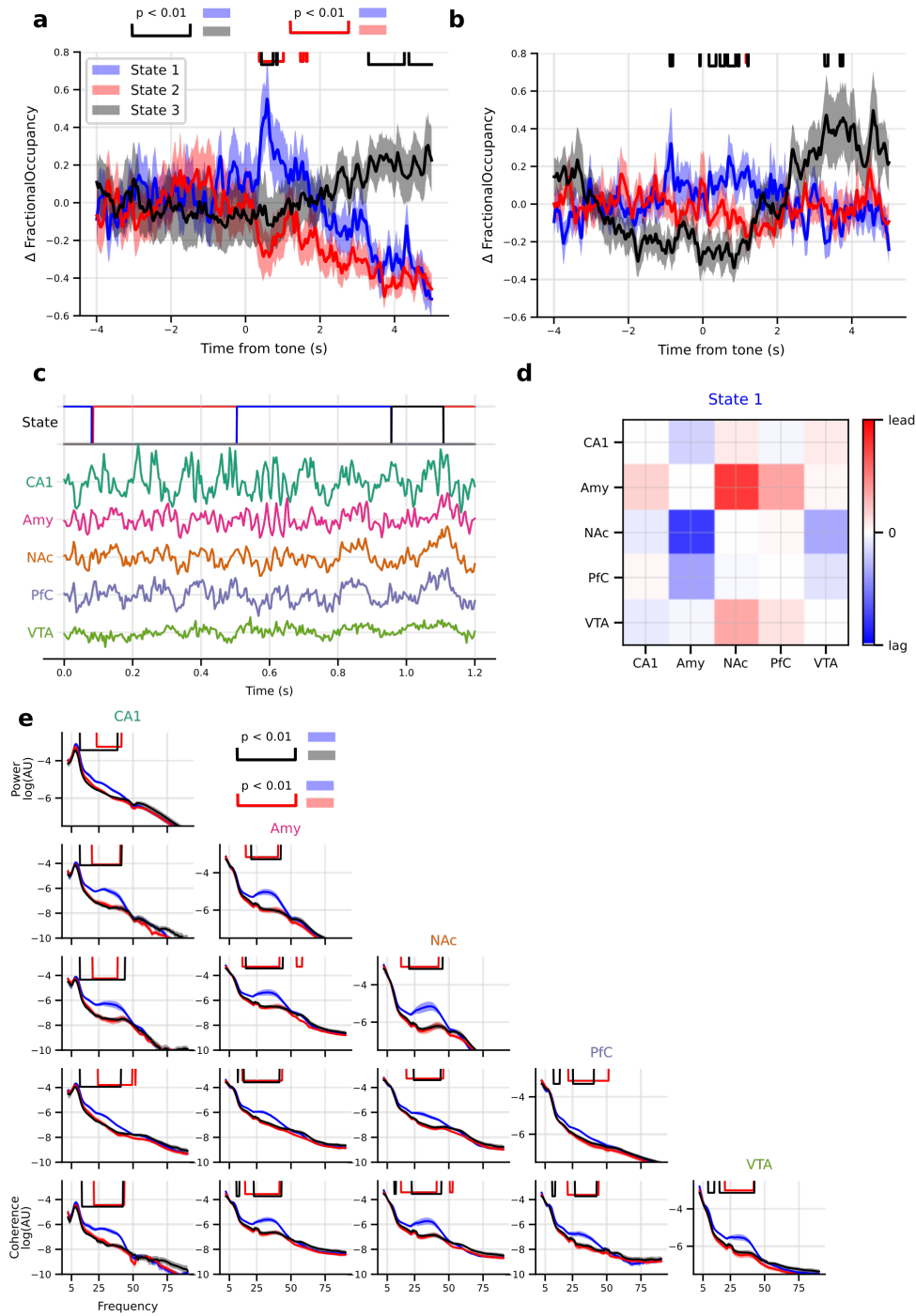


Figure 6.2: Change in fractional occupancy of each state evoked by tone in sucrose (a) and quinine (b) contexts. In all panels open rectangles indicate significance for paired differences between each state and state 1. (c) Example LFP from a single trial with the associated state assignments. (d) Average lead lag relationship between each pair of regions. (e) State-wise power spectral densities (diagonal) and coherences (off-diagonal) for each (pair of) region(s). Log y-scale.

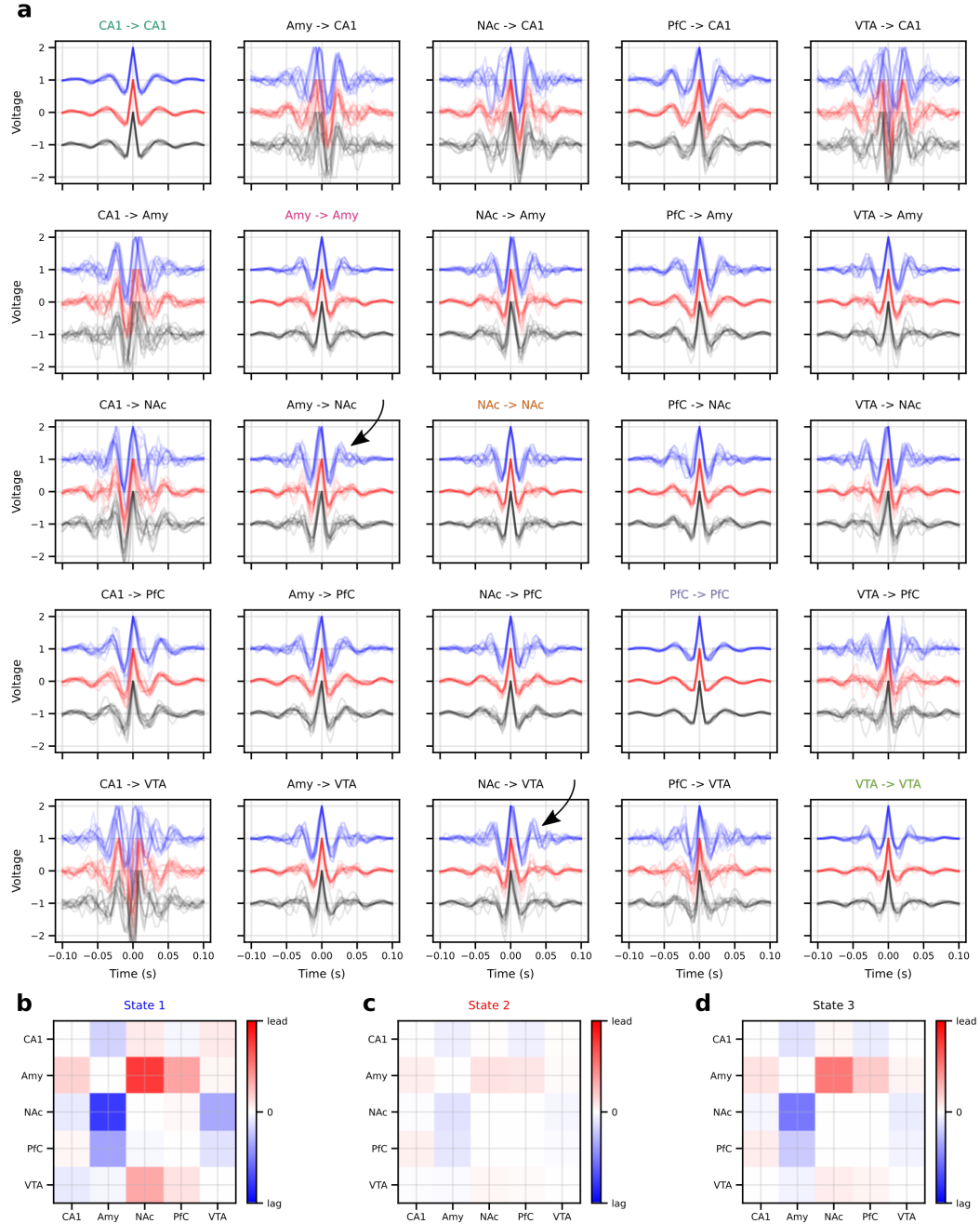


Figure 6.3: (a) Cross-correlation between the high-pass filtered local field potential for each pair of regions during each state. Each line represents the mean across all state occurrences during a single recording day. (b-d) Average lead lag relationship between each pair of regions.

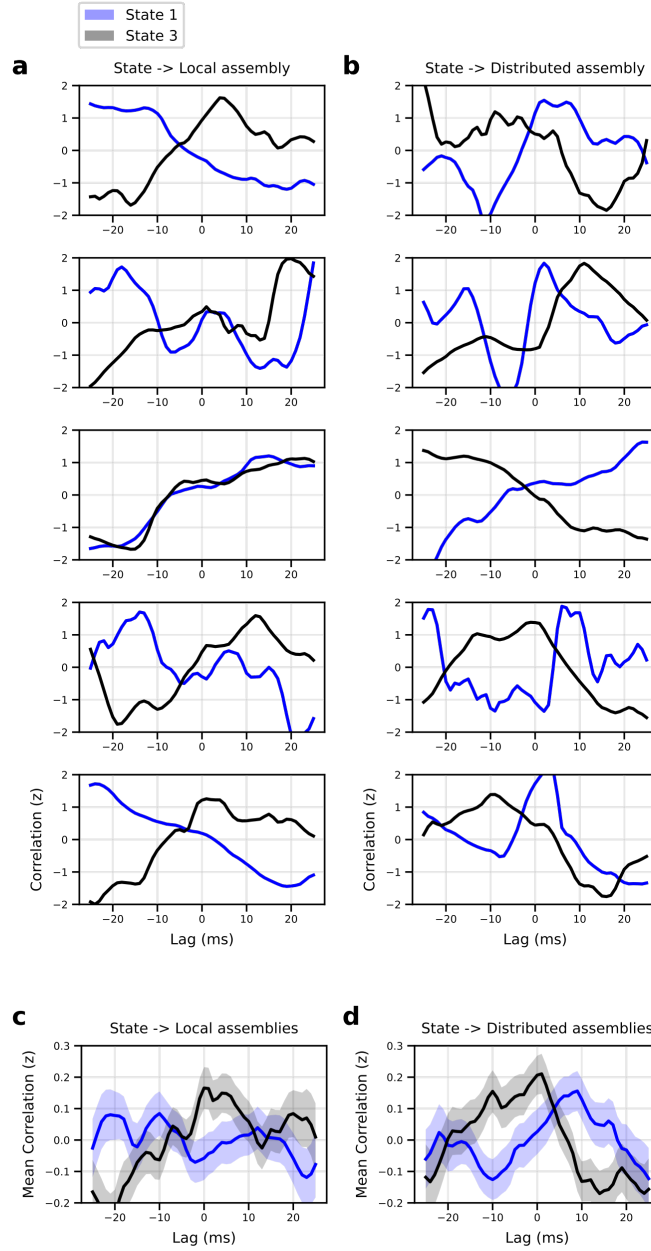


Figure 6.4: Cross-correlations between states and local (a,c) or distributed (b,d) assemblies. (a,b) are example state-assembly pairs. (c,d) show the mean across all days. Note that positive lags indicate the activity of the assembly follows the activity of the state.

6.4 Discussion

Thus we have found that as well as localised changes in the average LFP and its spectrogram the onset of the task is associated with a transient state of coherence in the low-gamma range. Moreover, the same state preceded an increase in distributed assembly activity versus local activity. Together these results provide preliminary evidence that the cross-circuit coordination required for effective behaviour is accompanied by widespread synchronous gamma oscillations.

In these analyses we did not consider lower frequency oscillations (below 20 Hz) as we were looking for mechanisms that would reflect coordination on a faster timescale. We also do not know whether the oscillatory state that we identified was itself correlated with, or carried upon, a lower frequency oscillation. There is a good chance that this is indeed the case as as many different gamma oscillations are modulated by both theta (Colgin 2015; Lopes-Dos-Santos et al. 2018) and 4 Hz (Fujisawa et al. 2011; Karalis et al. 2018).

In cats, coherence of these low-gamma oscillations between amygdala and striatum (Popescu et al. 2009) as well as amygdala and rhinal cortices (Bauer et al. 2007) has previously been associated with learning. Indeed Popescu et al. 2009 also found that the gamma-coupling increased only in response to a rewarded (not unrewarded) tone after learning and the absence of any coupling with (a different region of) cortex. While these oscillations are in the same frequency band as the ones we report, and seem to respond similarly during the execution of behaviour, we cannot be sure that the oscillations reported in cats have the same underlying mechanisms or function as in rodents. In either case, the exact origin of the oscillation is not yet clear (Bocchio et al. 2017). However, given the conduction delays between the regions it is likely that there are local oscillatory circuits within each region that are synchronised either by axon collaterals from pyramidal neurons or long-range interneurons (Buzsáki et al. 2012).

In the same vein, it is unclear if the slow-gamma described here is related to the slow-gamma reported in the CA1 region in the hippocampus and thought to

reflect input from CA3 and memory retrieval (Colgin 2015). In support of their relation the oscillation is seen immediately after the tone was played and before the mouse moves to the port, and thus potentially during recall or prospective coding with which slow-gamma in the CA3 is associated (Bieri et al. 2014). Moreover, Bass et al. 2015 found that electrical stimulation of the BLA at either 50 or 80 Hz drove coherence between CA1 and CA3 at 40 Hz, suggesting a link between amygdalar activity and slow-gamma events in the hippocampus. However it is dangerous to draw strong conclusions from the non-specific effects of electrical stimulation on oscillatory activity.

Further careful work is needed in order to more completely characterise this oscillatory state in relation to both assembly and spiking activity, as well as to prior work.

7

Closing Discussion

Contents

7.1	Overview	126
7.2	What is a cell assembly?	127
7.3	Future Directions	129

7.1 Overview

The existence of cell-assemblies distributed across multiple regions has been proposed previously (Buzsáki 2010; Buzsáki et al. 2012). Here we provide evidence for the existence of cross-structural patterns of co-activity which we believe reflects assembly activity in the Hebbian sense. These assemblies selectively evolve during the translation of a sensory cue, the meaning of which is contingent on the surrounding spatio-visual context, into successful behaviour. As such, these assemblies may reflect globally distributed representations of the adaptive mapping of sensory inputs into behaviour. The distributed activity of these assemblies was preceded by a global oscillatory state characterised by coherence in the low-gamma band. While further work is needed, we believe this may represent a widespread synchronisation signal used to recall multi-modal memories.

We have also contributed to two widely discussed methodological issues, namely spike sorting and modelling population dynamics. A comprehensive comparison of automated spike sorting techniques is almost intractable without first developing the underlying computational infrastructure and without seeking input from the community of both those developing new methods and those who rely on them. In relation to population dynamics, we have defined a semi-formal framework for assessing their interpretability and assert that the appropriate choice always depends on the context in which it will be used.

To conclude this thesis we will discuss the implications and justifications for our definition of a cell assembly. We will discuss some of the limitations of the work presented here and what we feel are the most exciting future directions.

7.2 What is a cell assembly?

The phrase “cell assembly”, coined by Donald Hebb in 1949, appears in a wide range of studies referring to a wide range of concepts.

For some, the focus is on the processes that resulted in the formation of cell assemblies. For example “Cells that fire together wire together” (Shatz 1992), refers to the fact that groups of neurons that are repeatedly activated by the same stimulus (be it externally- or internally-generated) are more like to undergo plastic changes that strengthen the synapses connecting them¹.

Cell assemblies are also often defined by their functional properties, such as a set of neurons that together represent a given concept or are related to the performance of a particular action (McKenzie et al. 2014; Tian et al. 2018). Thus some might refer to a group of neurons that respond to tone as “tone assemblies”. These representational definitions allow some overlap with the Hebbian one. It would be easy enough for the activity of a group of neurons to fulfil both the Hebbian co-activity and the functional criteria simultaneously but being precise in our definition is important for our experimental design, our analytical approach, and our interpretations.

A contrived hypothetical scenario that highlights the distinction would be one where two mice in the same environment are exposed to a pure tone. The tone will activate groups of neurons in both mice, but we may not wish to consider the "tone assembly" as one spanning both animals. However even with a co-activity based definition a similar issue arises if the co-activity between neurons was primarily a result of tone response, not of some intrinsic connectivity. In this case we might want to only consider a set of neurons to be an assembly if they fire together when the tone is not being played. Indeed, this "task-off" detection is what we used in this study. Even in this restricted case, though, it is hard to design an experiment where there would be no external inputs that could result in synchronous activity between neurons that are not connected. Nonetheless it may be fair to assert that co-activity

¹This phrase, while often mis-attributed to Hebb, was in fact popularised by Carla Shatz in her lectures in the 1990s.

during spontaneous behaviour is more likely to reflect a functionally meaningful organisation of activity than that driven by a highly salient discrete stimulus.

Aside from these considerations, there were specific reasons why, while useful on their own terms, none of the other methods described in chapter 5 were suitable to answer our question about globally distributed representations. Neither the "tone assembly" nor LDA or other supervised methods capture the dynamics of interactions between different groups of neurons. The strength of TCA lies in considering the per-trial and per-timepoint variability in a single model. However that would have restricted us from using it to detect assemblies outside of the task-period and then track them within. Finally the regressive models, including our own cross-correlation based method, would not have allowed us to test our predictions about the scenarios in Figure 1.2 because the lags are embedded within each assembly.

7.3 Future Directions

The most exciting next step related for our spike-sorting framework is its use for hyperparameter optimisation. With the SpikeForest framework in place the human overhead for systematically running the same algorithm with a range of parameters is very low. Without it, the possibility of finding the best hyperparameters is low and remains time-consuming. It will likely be most practical to use random search (Bergstra et al. 2012), or some form of Bayesian optimisation, as some sorters have many hyperparameters and the complexity of a standard grid search scales with the power of this number. There are two applications of this optimisation. The first is to allow a better comparison between sorters, as each has an more equal opportunity to perform well. The other application is for researchers to apply the optimisation using their curated results as the ground truth in order to find a sorter that best replicates their manual results.

Broadly, we feel that there are two (not mutually exclusive) long-term directions that could be taken in order to better delineate the roll for globally distributed representations as we describe them. The first would be to design causal manipulations to test for the existence and functional role for cross-regional assemblies. The second would be to use recent advances in recording technologies to increase the number of neurons we can sample across each region.

There are however some barriers in the design of such a causal manipulation. One direct way to test for their functional coordination would be to tag a distributed assembly across regions and then observe whether the optogenetic silencing of the assembly in one region lead to to the emergence of new neurons in that region that were co-active with the rest of the distributed assembly. However one would first need to resolve the issue of how to tag such an assembly. Standard IEG methods act on too long a timescale, and moreover while they are useful for tagging "representational" assemblies its unclear how you could tag an assembly based on co-activity. An alternative would be to use the spatial localisation of different units within a region to selectively silence those that are part of a selected

distributed assembly. This could potentially be achieved using with silicon probes with integrated μ LEDs (Wu et al. 2015). These probes could also be used to drive a closed-loop system where the pattern of stimulation is varied to either synchronise or desynchronise assembly members.

Leading edge technologies have recently enabled extracellular recordings from thousands of neurons. Early studies have taken advantage of this to provide a rich mathematical interrogation of the dynamics within cortical regions (Stringer et al. 2019a). Others have recorded across multiple regions and found brain-wide responses to individual task events similar to those we describe (Allen et al. 2019). However, they do not attempt to characterise how these brain-wide responses might be coordinated in relation to each other. And is not granted that current methods of assessing population dynamics will allow us to assess this coordination as they are. Regardless it is critical that those that are applied are as interpretable as possible, and we hope that the framework we have proposed will be useful in ensuring this. The characterisation of brain-wide dynamics at this scale could be a crucial next step in understanding the function of multiple regions as a system, rather than as a collection of subsystems.

Appendices

A

Additional Examples

Contents

A.1	Behaviour	134
A.2	Single-neuron responses	135
A.3	Assembly composition	136

A.1 Behaviour

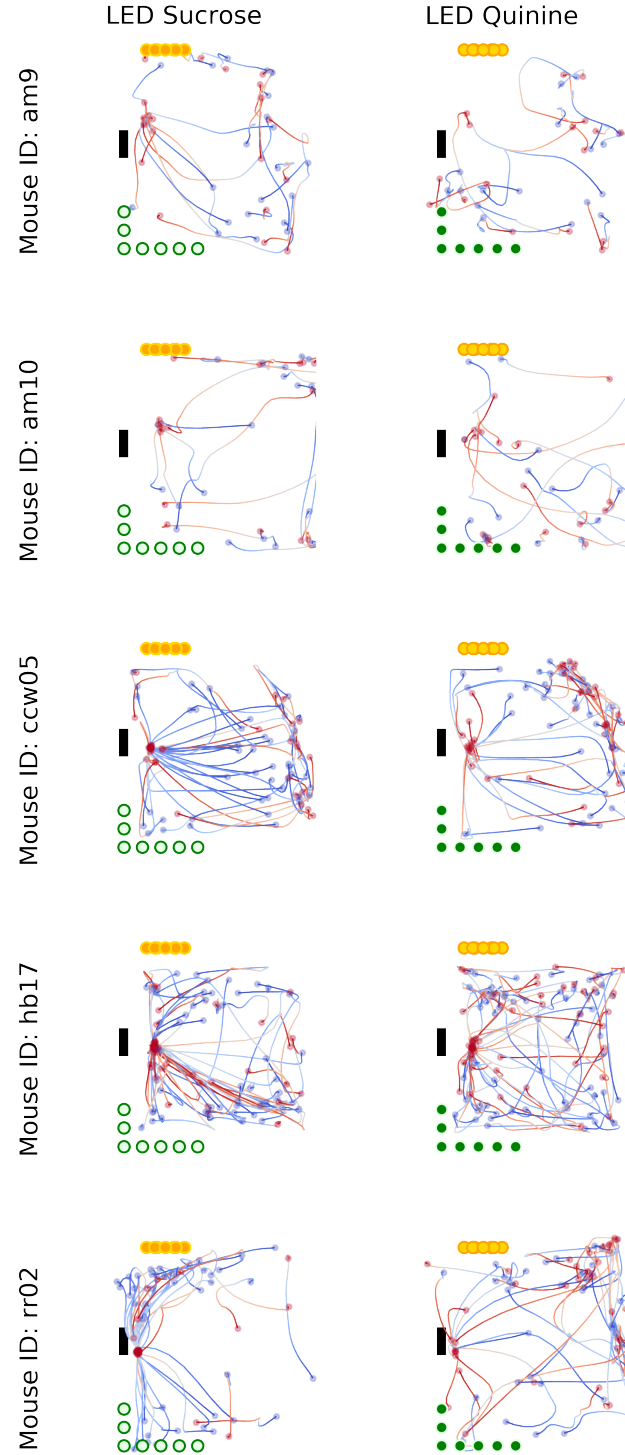


Figure A.1: Paths for one recording day for 5 different mice. The color-gradients of the paths represent the passage of time during the tone: they are blue at the onset and red at the offset.

A.2 Single-neuron responses

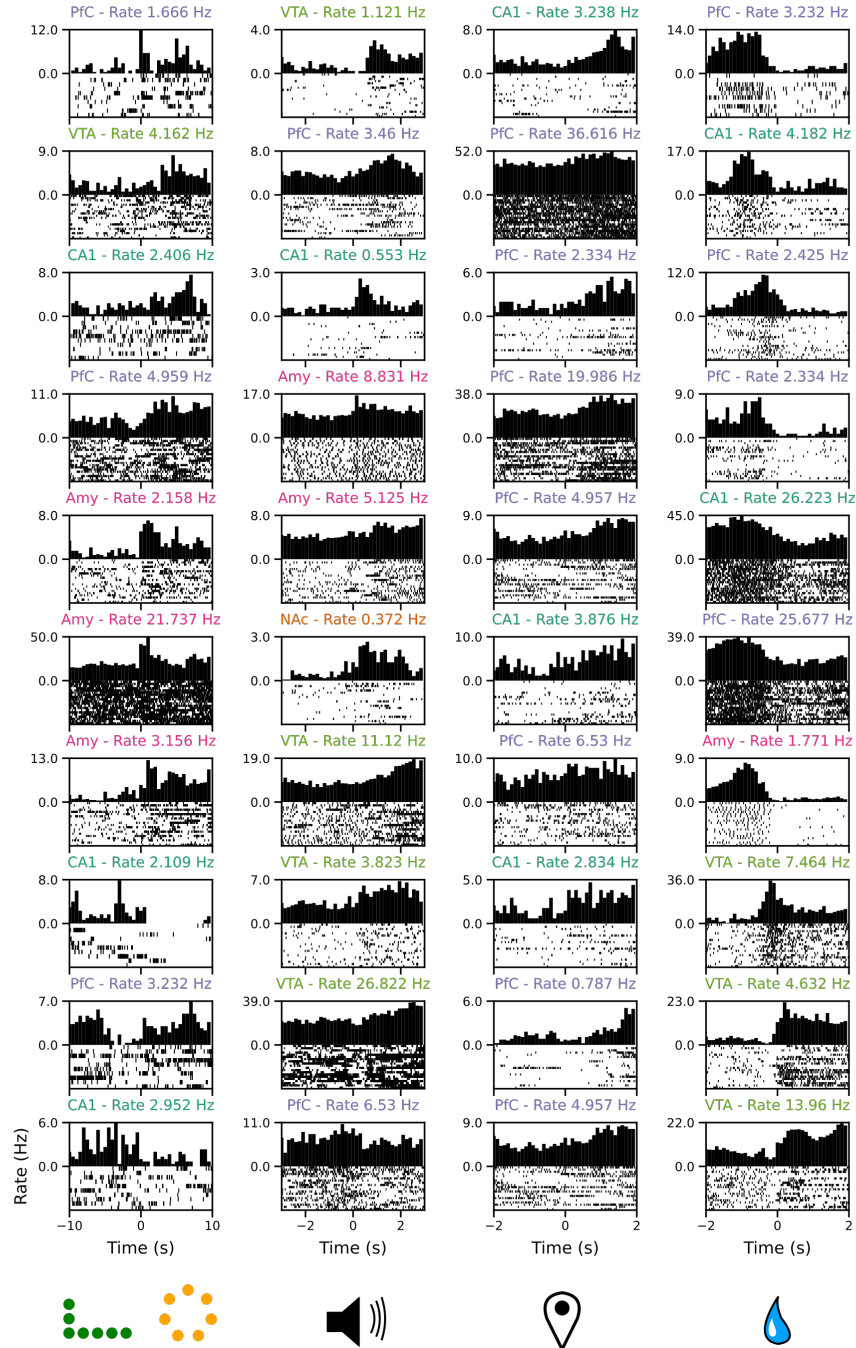


Figure A.2: Rate responses of neurons at various task stages shown as peri-stimulus histograms with raster plots showing the individual spikes for each trial. Maximum 20 trials shown for rasters.

A.3 Assembly composition



Figure A.3: Contribution of the interactions between each pair of regions in each recording day to the activity of each assembly.

B

Anatomy

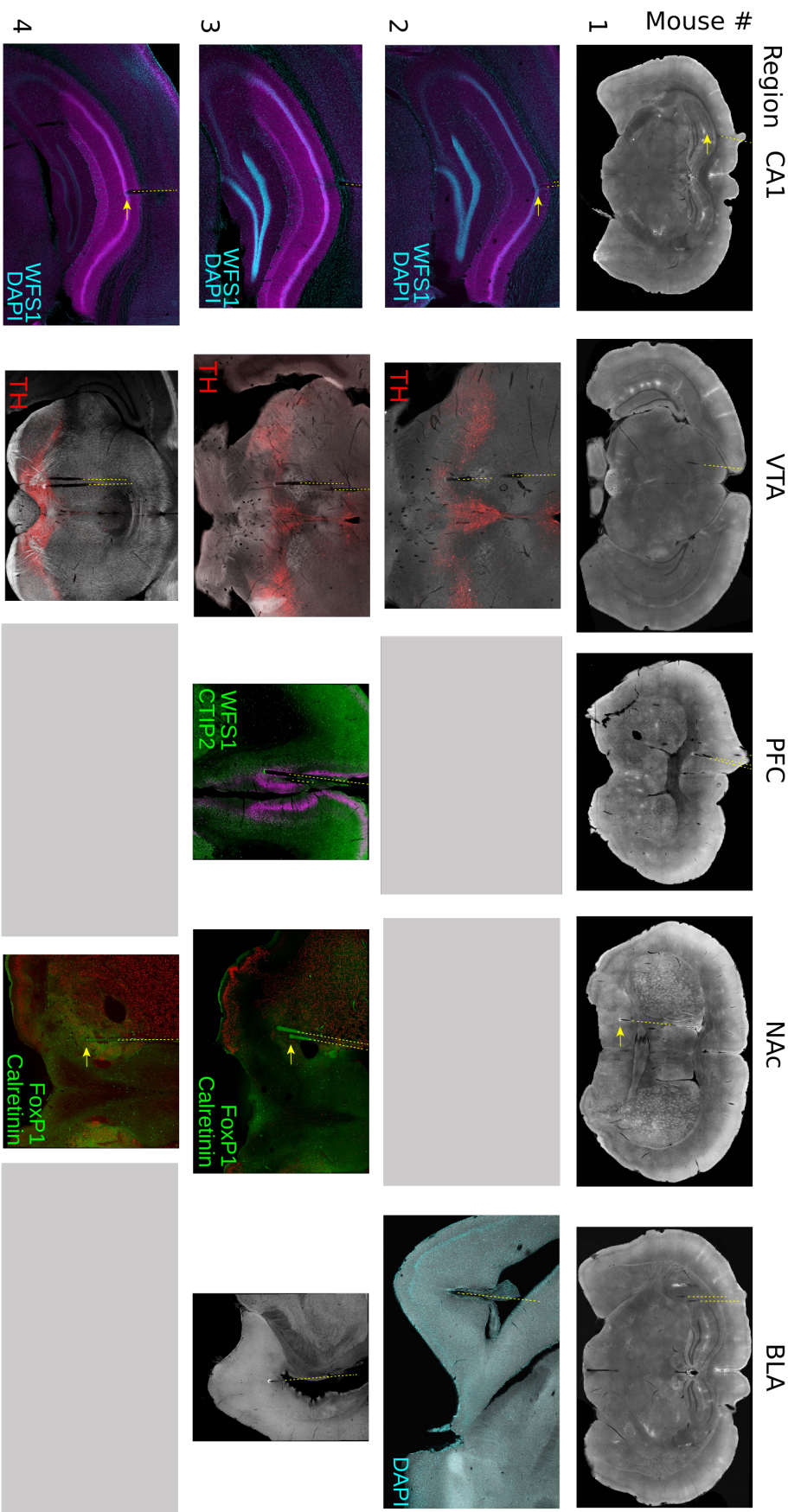


Figure B.1: Images of tetrode tracks for each mouse, supports Figure 3.5. Yellow dashed lines indicate the path of the track and yellow arrow indicates the bottom of the track if shown. For some tracks we could not trace them all of the way down to the target site and so we show them at a point where we could identify them clearly. The grey rectangles indicate where we were unable to find the tracks for this region.

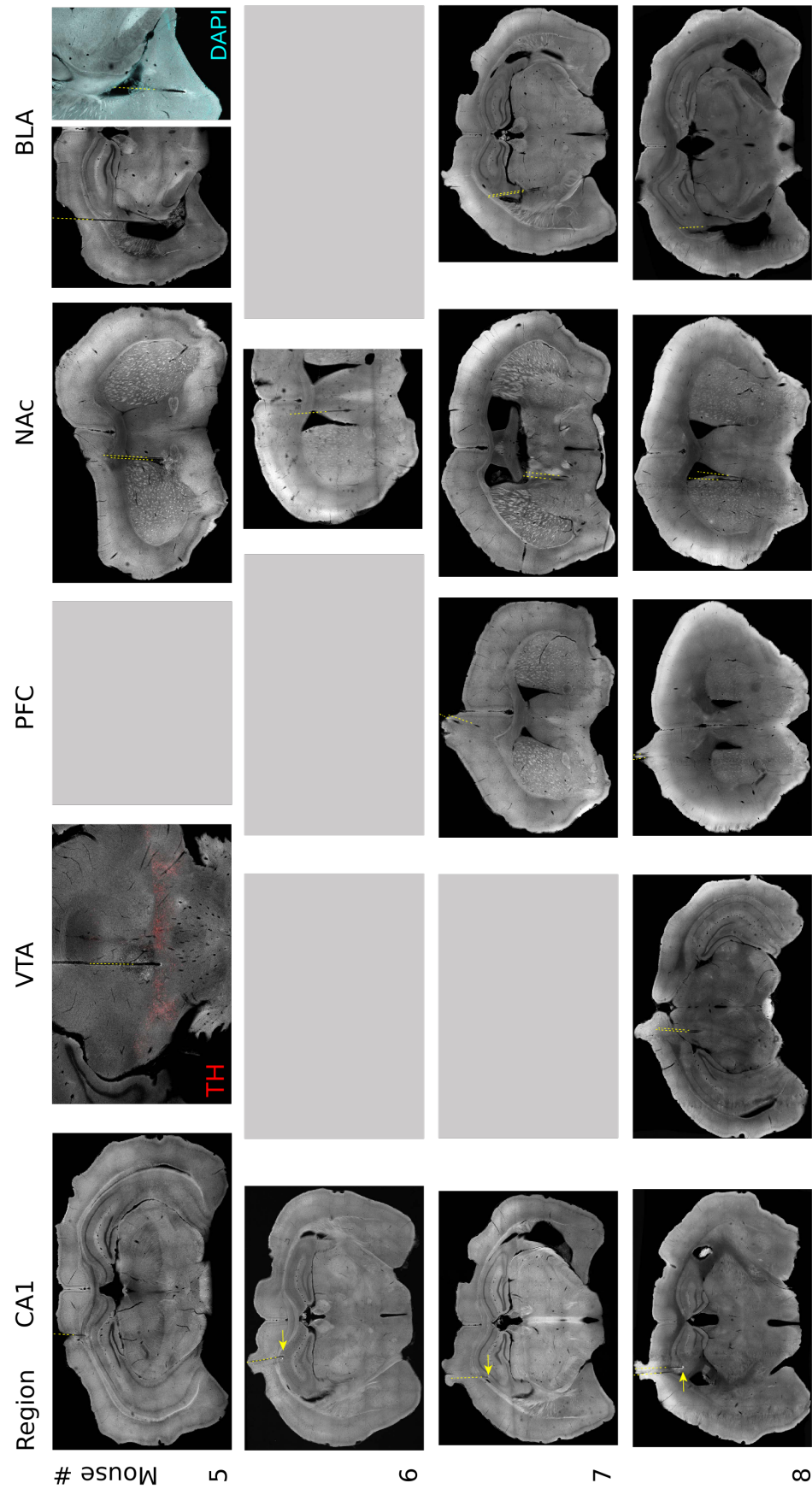


Figure B.2: Continued from Figure B.1.

Mouse ID	Tetrode Depth (mm)			Neurons											
	Date Recorded	CA1-1	CA1-2	CA1-3	BLA-1	BLA-2	BLA-3	NAC-1	NAC-2	NAC-3	PFC-1	PFC-2	PFC-3	VTA-1	VTA-2
1	07/03/16	0.25	0.25	X	7	0.25	0.25	X	8	0.25	0.25	0.25	8	0.25	0.25
6	24/05/16	0.25	0.00	X	29	0.25	0.25	X	16	0.25	0.25	0.25	14	0.25	0.25
	25/05/16	0.50	0.00	X	35	0.50	0.50	X	17	0.50	0.50	0.50	13	0.50	0.50
	27/05/16	0.50	0.00	X	32	0.75	0.75	X	17	0.75	0.75	0.75	7	0.75	0.75
	28/05/16	0.50	0.00	X	25	1.00	1.00	X	14	1.00	1.00	1.00	9	1.00	1.00
7	29/05/16	0.50	0.00	X	37	1.13	1.13	X	15	1.13	1.13	1.13	14	1.13	1.13
	13/02/17	1.05	0.73	X	4	0.88	0.75	X	8	3.38	2.75	2.13	13	0.88	1.13
8	14/02/17	1.53	1.13	X	6	1.13	1.00	X	6	3.63	3.00	2.25	9	1.50	1.13
	04/04/17	0.00	0.13	X	16	0.75	0.75	X	0	0.08	0.00	0.00	20	0.13	0.00
4	05/04/17	0.06	0.19	X	14	0.88	1.00	X	0	0.20	0.13	0.13	25	0.25	0.13
	12/12/17	0.91	0.00	X	13	0.19	0.13	X	15	0.31	0.25	0.28	13	0.31	0.25
	13/12/17	0.91	0.02	X	24	0.22	0.16	X	10	0.47	0.25	0.47	6	0.34	0.34
	14/12/17	0.91	0.02	X	23	0.22	0.16	X	13	0.56	0.25	0.50	15	0.41	0.34
2	17/04/18	0.19	0.12	X	12	0.78	0.78	X	8	0.87	0.87	1.28	11	0.19	0.16
	18/04/18	0.19	0.12	X	10	0.78	0.78	X	10	0.87	0.87	1.28	18	0.19	0.16
	19/04/18	0.19	0.12	X	12	0.78	0.78	X	6	0.87	0.87	1.28	12	0.19	0.16
3	09/05/18	0.25	0.28	X	15	0.45	0.62	X	0	-0.25	-0.06	0.01	12	0.69	0.69
	10/05/18	0.25	0.28	X	13	0.45	0.62	X	4	-0.25	-0.06	0.01	9	0.69	0.69
	13/05/18	0.25	0.15	X	14	0.45	0.81	X	3	-0.16	0.06	0.04	11	0.69	0.69
	15/05/18	0.06	-0.03	X	15	0.26	0.34	X	5	-0.35	-0.32	-0.14	11	0.50	0.50
5	24/10/18	0.56	0.09	0.28	20	0.10	0.56	0.50	25	0.64	0.64	0.38	14	0.34	0.22

Figure B.3: Depth of tetrodes in mm relative to implant depth. Mouse’s correspond to those in Figure B.1 and Figure B.2. The corresponding number of neurons for all tetrodes in the region is also indicated. Only neurons that were used for analysis (i.e. those with mean firing rates between 0.05 and 10 Hz) are counted.

C

Software Usage & Contribution

Contents

C.1	Software in science	142
C.2	Contributions to scientific software as part of this thesis	143

C.1 Software in science

As with most modern scientific enterprises this thesis relied heavily on openly available software. Specifically most of the analysis was done in the Julia language, a relatively recently developed language that combines the niceties of the MATLAB syntax with performance approaching that of compiled languages (Bezanson et al. 2014). All of the plots were made with matplotlib (Hunter 2007), via Julia’s python interface. And all of the spike sorting work was done using IPython (Perez et al. 2007) and its associated numerical libraries scipy and numpy (Millman et al. 2011; Oliphant 2007). We also make heavy use of the scikit-learn API (Buitinck et al. 2013; Pedregosa et al. 2011).

C.2 Contributions to scientific software as part of this thesis

Over the course of this thesis I authored, and contributed to, several open-source tools for data analysis. A brief description of each is listed below.

- **Co-author** of SpikeForest: a large-scale reproducible spike sorting validation platform with an interactive web interface.
- **Contributor** to ScikitLearn.jl which implements the popular scikit-learn interface and algorithms in Julia.
- **Contributor** to MultivariateStats.jl: A Julia package for multivariate statistics and data analysis.
- **Author** of WaveletInformation.jl: Implementations of wavelet-based classifiers and component analysers.
- **Author** of PopulationDynamics.jl: Implements a scikit-learn interface for neural population models.

All of these can be found on GitHub (github.com).

References

- Abeles, Moshe (1982). *Local Cortical Circuits: An Electrophysiological Study*. Studies of Brain Function. Berlin Heidelberg: Springer-Verlag.
- Adhikari, Avishek et al. (2010). “Cross-Correlation of Instantaneous Amplitudes of Field Potential Oscillations: A Straightforward Method to Estimate the Directionality and Lag between Brain Areas”. *Journal of neuroscience methods* 191.2, pp. 191–200.
- Allen, Elena A. et al. (2014). “Tracking Whole-Brain Connectivity Dynamics in the Resting State”. *Cerebral Cortex (New York, NY)* 24.3, pp. 663–676.
- Allen, William E. et al. (2019). “Thirst Regulates Motivated Behavior through Modulation of Brainwide Neural Population Dynamics”. *Science*, eaav3932.
- Ambroggi, Frederic et al. (2008). “Basolateral Amygdala Neurons Facilitate Reward-Seeking Behavior by Exciting Nucleus Accumbens Neurons”. *Neuron* 59.4, pp. 648–661.
- Amir, Alon et al. (2015). “Amygdala Signaling during Foraging in a Hazardous Environment”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 35.38, pp. 12994–13005.
- Angwin, Julia et al. (2016). *Machine Bias*. URL: <https://www.propublica.org/article/machine-bias-risk-assessments-in-criminal-sentencing> (visited on 06/26/2019).
- Barnes, C. A. et al. (1990). “Comparison of Spatial and Temporal Characteristics of Neuronal Activity in Sequential Stages of Hippocampal Processing”. *Progress in Brain Research* 83, pp. 287–300.
- Barnett, Alex H., Jeremy F. Magland, and Leslie F. Greengard (2015). “Validation of Neural Spike Sorting Algorithms without Ground-Truth Information”.
- Bartho, Peter et al. (2009). “Population Coding of Tone Stimuli in Auditory Cortex: Dynamic Rate Vector Analysis”. *The European journal of neuroscience* 30.9, pp. 1767–1778.
- Bartlett, Frederic Charles (1932). *Remembering: A Study in Experimental and Social Psychology*. Remembering: A Study in Experimental and Social Psychology. New York, NY, US: Cambridge University Press. xix, 317.
- Bass, David I. and Joseph R. Manns (2015). “Memory-Enhancing Amygdala Stimulation Elicits Gamma Synchrony in the Hippocampus”. *Behavioral neuroscience* 129.3, pp. 244–256.
- Bauer, Elizabeth P., Rony Paz, and Denis Paré (2007). “Gamma Oscillations Coordinate Amygdalo-Rhinal Interactions during Learning”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 27.35, pp. 9369–9379.
- Benchenane, Karim et al. (2010). “Coherent Theta Oscillations and Reorganization of Spike Timing in the Hippocampal- Prefrontal Network upon Learning”. *Neuron* 66.6, pp. 921–936.

- Bergel, Antoine et al. (2018). “Local Hippocampal Fast Gamma Rhythms Precede Brain-Wide Hyperemic Patterns during Spontaneous Rodent REM Sleep”. *Nature Communications* 9.1, p. 5364.
- Bergstra, James and Yoshua Bengio (2012). “Random Search for Hyper-Parameter Optimization”. *J. Mach. Learn. Res.* 13, pp. 281–305.
- Berke, Joshua D. et al. (2004). “Oscillatory Entrainment of Striatal Neurons in Freely Moving Rats”. *Neuron* 43.6, pp. 883–896.
- Beyeler, Anna et al. (2016). “Divergent Routing of Positive and Negative Information from the Amygdala during Memory Retrieval”. *Neuron* 90.2, pp. 348–361.
- Bezanson, Jeff et al. (2014). “Julia: A Fresh Approach to Numerical Computing”.
- Bienenstock, Elie (1995). “A Model of Neocortex”. *Network: Computation in Neural Systems* 6.2, pp. 179–224.
- Bieri, Kevin Wood, Katelyn N. Bobbitt, and Laura Lee Colgin (2014). “Slow and Fast Rhythms Coordinate Different Spatial Coding Modes in Hippocampal Place Cells”. *Neuron* 82.3, pp. 670–681.
- Bloodgood, Daniel W. et al. (2018). “Fear Extinction Requires Infralimbic Cortex Projections to the Basolateral Amygdala”. *Translational Psychiatry* 8.1, p. 60.
- Bocchio, Marco, Sadegh Nabavi, and Marco Capogna (2017). “Synaptic Plasticity, Engrams, and Network Oscillations in Amygdala Circuits for Storage and Retrieval of Emotional Memories”. *Neuron* 94.4, pp. 731–743.
- Buitinck, Lars et al. (2013). “API Design for Machine Learning Software: Experiences from the Scikit-Learn Project”.
- Burgos-Robles, Anthony et al. (2017). “Amygdala Inputs to Prefrontal Cortex Guide Behavior amid Conflicting Cues of Reward and Punishment”. *Nature Neuroscience* 20.6, pp. 824–835.
- Buzsáki, György (2010). “Neural Syntax: Cell Assemblies, Synapsembles and Readers”. *Neuron* 68.3, pp. 362–385.
- Buzsáki, György and Andreas Draguhn (2004). “Neuronal Oscillations in Cortical Networks”. *Science (New York, N.Y.)* 304.5679, pp. 1926–1929.
- Buzsáki, György, Nikos Logothetis, and Wolf Singer (2013). “Scaling Brain Size, Keeping Timing: Evolutionary Preservation of Brain Rhythms”. *Neuron* 80.3, pp. 751–764.
- Buzsáki, György and Xiao-Jing Wang (2012). “Mechanisms of Gamma Oscillations”. *Annual review of neuroscience* 35, pp. 203–225.
- Canolty, Ryan T. et al. (2010). “Oscillatory Phase Coupling Coordinates Anatomically Dispersed Functional Cell Assemblies”. *Proceedings of the National Academy of Sciences of the United States of America* 107.40, pp. 17356–17361.
- Cardinal, Rudolf N. et al. (2002). “Emotion and Motivation: The Role of the Amygdala, Ventral Striatum, and Prefrontal Cortex”. *Neuroscience & Biobehavioral Reviews* 26.3, pp. 321–352.
- Chapin, J. K. and M. A. Nicolelis (1999). “Principal Component Analysis of Neuronal Ensemble Activity Reveals Multidimensional Somatosensory Representations”. *Journal of Neuroscience Methods* 94.1, pp. 121–140.
- Chung, Fan and Linyuan Lu (2002). “Connected Components in Random Graphs with Given Expected Degree Sequences”. *Annals of Combinatorics* 6.2, pp. 125–145.
- Chung, Jason E. et al. (2017). “A Fully Automated Approach to Spike Sorting”. *Neuron* 95.6, 1381–1394.e6.

- Chung, Jason E. et al. (2019). “High-Density, Long-Lasting, and Multi-Region Electrophysiological Recordings Using Polymer Electrode Arrays”. *Neuron* 101.1, 21–31.e5.
- Colgin, Laura Lee (2015). “Do Slow and Fast Gamma Rhythms Correspond to Distinct Functional States in the Hippocampal Network?” *Brain research* 1621, pp. 309–315.
- Csicsvari, J. et al. (1999). “Oscillatory Coupling of Hippocampal Pyramidal Cells and Interneurons in the Behaving Rat”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 19.1, pp. 274–287.
- Day, Jeremy J. and Regina M. Carelli (2007). “The Nucleus Accumbens and Pavlovian Reward Learning”. *The Neuroscientist : a review journal bringing neurobiology, neurology and psychiatry* 13.2, pp. 148–159.
- Deng, Xinyi et al. (2016). “Rapid Classification of Hippocampal Replay Content for Real-Time Applications”. *Journal of Neurophysiology* 116.5, pp. 2221–2235.
- Dupret, David et al. (2010). “The Reorganization and Reactivation of Hippocampal Maps Predict Spatial Memory Performance.” *Nature neuroscience* 13.8, pp. 995–1002.
- Engel, Andreas K., Pascal Fries, and Wolf Singer (2001). “Dynamic Predictions: Oscillations and Synchrony in Top-down Processing”. *Nature Reviews Neuroscience* 2.10, p. 704.
- Engelhard, Ben et al. (2019). “Specialized Coding of Sensory, Motor and Cognitive Variables in VTA Dopamine Neurons”. *Nature* 570.7762, pp. 509–513.
- Farovik, Anja, Laura M. Dupont, and Howard Eichenbaum (2010). “Distinct Roles for Dorsal CA3 and CA1 in Memory for Sequential Nonspatial Events”. *Learning & Memory* 17.1, pp. 12–17.
- Foster, David J. and Matthew A. Wilson (2006). “Reverse Replay of Behavioural Sequences in Hippocampal Place Cells during the Awake State”. *Nature* 440.7084, pp. 680–683.
- Franklin, Keith B. J and George Paxinos (2008). *The Mouse Brain in Stereotaxic Coordinates*. OCLC: 845721906.
- Fries, Pascal (2005). “A Mechanism for Cognitive Dynamics: Neuronal Communication through Neuronal Coherence”. *Trends in Cognitive Sciences* 9.10, pp. 474–480.
- Fujisawa, Shigeyoshi and György Buzsáki (2011). “A 4-Hz Oscillation Adaptively Synchronizes Prefrontal, VTA and Hippocampal Activities”. *Neuron* 72.1, pp. 153–165.
- Garner, Aleena R. et al. (2012). “Generation of a Synthetic Memory Trace”. *Science (New York, N.Y.)* 335.6075, pp. 1513–1516.
- Geisler, Stefanie et al. (2007). “Glutamatergic Afferents of the Ventral Tegmental Area in the Rat”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 27.21, pp. 5730–5743.
- Gelman, Andrew and Hal Stern (2006). “The Difference Between “Significant” and “Not Significant” Is Not Itself Statistically Significant”. *The American Statistician* 60.4, pp. 328–331.
- Girardeau, Gabrielle, Ingrid Inema, and György Buzsáki (2017). “Reactivations of Emotional Memory in the Hippocampus-Amygdala System during Sleep”. *Nature Neuroscience* 20.11, pp. 1634–1642.
- Gomperts, Stephen N, Fabian Kloosterman, and Matthew A Wilson (2015). “VTA Neurons Coordinate with the Hippocampal Reactivation of Spatial Experience”. *eLife* 4. Ed. by Howard Eichenbaum, e05360.

- Gray, Jeffrey A. (1970). "The Psychophysiological Basis of Introversion-Extraversion". *Behaviour Research and Therapy* 8.3, pp. 249–266.
- Grossberger, Lukas, Francesco P Battaglia, and Martin Vinck (2018). "Unsupervised Clustering of Temporal Patterns in High-Dimensional Neuronal Ensembles Using a Novel Dissimilarity Measure".
- Gupta, Anoopum S. et al. (2012). "Segmentation of Spatial Experience by Hippocampal Sequences". *Nature Neuroscience* 15.7, pp. 1032–1039.
- Harris, Kenneth D et al. (2000). "Accuracy of Tetrode Spike Separation as Determined by Simultaneous Intracellular and Extracellular Measurements". *Journal of Neurophysiology* 84.1, pp. 401–414.
- Harris, Kenneth D et al. (2001). "Temporal Interaction between Single Spikes and Complex Spike Bursts in Hippocampal Pyramidal Cells". *Neuron* 32.1, pp. 141–149.
- Harris, Kenneth D et al. (2003). "Organization of Cell Assemblies in the Hippocampus". *Nature* 424.6948, pp. 552–556.
- Hartigan, J. A. and P. M. Hartigan (1985). "The Dip Test of Unimodality". *The Annals of Statistics* 13.1, pp. 70–84.
- Hasselmo, M. E., E. Schnell, and E. Barkai (1995). "Dynamics of Learning and Recall at Excitatory Recurrent Synapses and Cholinergic Modulation in Rat Hippocampal Region CA3". *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 15 (7 Pt 2), pp. 5249–5262.
- Herry, Cyril et al. (2008). "Switching on and off Fear by Distinct Neuronal Circuits". *Nature* 454.7204, pp. 600–606.
- Hirase, H. et al. (1999). "Firing Rate and Theta-Phase Coding by Hippocampal Pyramidal Neurons during 'Space Clamping'". *The European Journal of Neuroscience* 11.12, pp. 4373–4380.
- Horvitz, J. C. (2000). "Mesolimbocortical and Nigrostriatal Dopamine Responses to Salient Non-Reward Events". *Neuroscience* 96.4, pp. 651–656.
- Huerta, P. T. and J. E. Lisman (1996). "Synaptic Plasticity during the Cholinergic Theta-Frequency Oscillation in Vitro". *Hippocampus* 6.1, pp. 58–61.
- Huerta, P. T. et al. (2000). "Formation of Temporal Memory Requires NMDA Receptors within CA1 Pyramidal Neurons". *Neuron* 25.2, pp. 473–480.
- Hultman, Rainbo et al. (2018). "Brain-Wide Electrical Spatiotemporal Dynamics Encode Depression Vulnerability". *Cell* 173.1, 166–180.e14.
- Humphries, Mark D. (2017). "Dynamical Networks: Finding, Measuring, and Tracking Neural Population Activity Using Network Science". *Network Neuroscience* 1.4, pp. 324–338.
- Humphries, Mark D. and Kevin Gurney (2008). "Network 'Small-World-Ness': A Quantitative Method for Determining Canonical Network Equivalence". *PLoS ONE* 3.4.
- Hunter, John D. (2007). "Matplotlib: A 2D Graphics Environment". *Computing in Science & Engineering* 9.3, pp. 90–95.
- Hyvarinen, A. (1999). "Fast and Robust Fixed-Point Algorithms for Independent Component Analysis". *IEEE Transactions on Neural Networks* 10.3, pp. 626–634.
- Isbister James B. et al. (2018). "A New Approach to Solving the Feature-Binding Problem in Primate Vision". *Interface Focus* 8.4, p. 20180021.
- Ito, Rutsuko et al. (2008). "Functional Interaction between the Hippocampus and Nucleus Accumbens Shell Is Necessary for the Acquisition of Appetitive Spatial Context Conditioning". *The Journal of Neuroscience* 28.27, pp. 6950–6959.

- Izhikevich, Eugene M. (2006). “Polychronization: Computation with Spikes”. *Neural Computation* 18.2, pp. 245–282.
- Jadhav, Shantanu P. et al. (2016). “Coordinated Excitation and Inhibition of Prefrontal Ensembles during Awake Hippocampal Sharp-Wave Ripple Events”. *Neuron* 90.1, pp. 113–127.
- Janak, Patricia H and Kay M Tye (2015). “From Circuits to Behaviour in the Amygdala.” *Nature* 517.7534, pp. 284–292.
- Jezek, Karel et al. (2011). “Theta-Paced Flickering between Place-Cell Maps in the Hippocampus”. *Nature* 478.7368, pp. 246–249.
- Jones, Matthew W. and Matthew A. Wilson (2005). “Theta Rhythms Coordinate Hippocampal–Prefrontal Interactions in a Spatial Memory Task”. *PLOS Biology* 3.12, e402.
- JuliaStatistics (2019). *MultivariateStats.Jl: A Julia Package for Multivariate Statistics and Data Analysis*. Julia Statistics.
- Jun, James J. et al. (2017a). “Fully Integrated Silicon Probes for High-Density Recording of Neural Activity”. *Nature* 551.7679, pp. 232–236.
- Jun, James Jaeyoon et al. (2017b). “Real-Time Spike Sorting Platform for High-Density Extracellular Probes with Ground-Truth Validation and Drift Correction”. *bioRxiv*, p. 101030.
- Kaiser, Marcus and Claus C Hilgetag (2006). “Nonoptimal Component Placement, but Short Processing Paths, Due to Long-Distance Projections in Neural Systems”. *PLoS Computational Biology* 2.7.
- Karalis, Nikolaos and Anton Sirota (2018). “Breathing Coordinates Limbic Network Dynamics Underlying Memory Consolidation”. *bioRxiv*, p. 392530.
- Karalis, Nikolaos et al. (2016). “4-Hz Oscillations Synchronize Prefrontal-Amygdala Circuits during Fear Behavior”. *Nature Neuroscience* 19.4, pp. 605–612.
- Koch, C., M. Rapp, and I. Segev (1996). “A Brief History of Time (Constants)”. *Cerebral Cortex (New York, N.Y.: 1991)* 6.2, pp. 93–101.
- Kreiter, A. K. and W. Singer (1996). “Stimulus-Dependent Synchronization of Neuronal Responses in the Visual Cortex of the Awake Macaque Monkey”. *Journal of Neuroscience* 16.7, pp. 2381–2396.
- Kumar, Sunil et al. (2013). “Cortical Control of Affective Networks”. *The Journal of neuroscience : the official journal of the Society for Neuroscience* 33.3, pp. 1116–1129.
- Laubach, M., M. Shuler, and M. A. Nicolelis (1999). “Independent Component Analyses for Quantifying Neuronal Ensemble Interactions”. *Journal of Neuroscience Methods* 94.1, pp. 141–154.
- LeDoux, Joseph (1998). “Fear and the Brain: Where Have We Been, and Where Are We Going?” *Biological Psychiatry* 44.12, pp. 1229–1238.
- Leutgeb, Jill K. et al. (2007). “Pattern Separation in the Dentate Gyrus and CA3 of the Hippocampus”. *Science (New York, N.Y.)* 315.5814, pp. 961–966.
- Lipton, Zachary C (2018). “The Mythos of Model Interpretability”. *machine learning*, p. 28.
- Liu, Xu et al. (2012). “Optogenetic Stimulation of a Hippocampal Engram Activates Fear Memory Recall”. *Nature* 484.7394, pp. 381–385.
- Long, Lauren L. et al. (2014). “Theta Dynamics in Rat: Speed and Acceleration across the Septotemporal Axis”. *PloS One* 9.5, e97987.

- Lopes-dos-Santos, Vítor, Sidarta Ribeiro, and Adriano B. L. Tort (2013). “Detecting Cell Assemblies in Large Neuronal Populations”. *Journal of Neuroscience Methods* 220.2, pp. 149–166.
- Lopes-Dos-Santos, Vítor et al. (2018). “Parsing Hippocampal Theta Oscillations by Nested Spectral Components during Spatial Exploration and Memory-Guided Behavior”. *Neuron* 100.4, 940–952.e7.
- Lopes-dos-Santos, Vítor et al. (2011). “Neuronal Assembly Detection and Cell Membership Specification by Principal Component Analysis”. *PLOS ONE* 6.6, e20996.
- Lopes-dos-Santos, Vítor et al. (2014). “Extracting Information in Spike Time Patterns with Wavelets and Information Theory.” *Journal of Neurophysiology*, jn.00380.2014.
- Lundberg, Scott M. et al. (2018). “Explainable Machine-Learning Predictions for the Prevention of Hypoxaemia during Surgery”. *Nature Biomedical Engineering* 2.10, p. 749.
- Lundqvist, Mikael, Pawel Herman, and Anders Lansner (2011). “Theta and Gamma Power Increases and Alpha/Beta Power Decreases with Memory Load in an Attractor Network Model”. *Journal of Cognitive Neuroscience* 23.10, pp. 3008–3020.
- Lundqvist, Mikael et al. (2016). “Gamma and Beta Bursts Underlie Working Memory”. *Neuron* 90.1, pp. 152–164.
- Maboudi, Kourosh et al. (2018). “Uncovering Temporal Structure in Hippocampal Output Patterns”. *eLife* 7. Ed. by Laura Colgin, e34467.
- Mackevicius, Emily L et al. (2019). “Unsupervised Discovery of Temporal Sequences in High-Dimensional Datasets, with Applications to Neuroscience”. *eLife* 8. Ed. by Laura Colgin and Timothy E Behrens, e38471.
- Marčenko, V A and L A Pastur (1967). “Distribution of Eigenvalues for Some Sets of Random Matrices”. *Mathematics of the USSR-Sbornik* 1.4, pp. 457–483.
- Maren, Stephen and Gregory J. Quirk (2004). “Neuronal Signalling of Fear Memory”. *Nature Reviews Neuroscience* 5.11, p. 844.
- McClelland, James L., Bruce L. McNaughton, and Randall C. O’Reilly (1995). “Why There Are Complementary Learning Systems in the Hippocampus and Neocortex: Insights from the Successes and Failures of Connectionist Models of Learning and Memory”. *Psychological Review* 102.3, pp. 419–457.
- McDonald, A. J. (1991). “Organization of Amygdaloid Projections to the Prefrontal Cortex and Associated Striatum in the Rat”. *Neuroscience* 44.1, pp. 1–14.
- McKenzie, Sam et al. (2014). “Hippocampal Representation of Related and Opposing Memories Develop within Distinct, Hierarchically Organized Neural Schemas”. *Neuron* 83.1, pp. 202–215.
- McNamara, Colin G. et al. (2014). “Dopaminergic Neurons Promote Hippocampal Reactivation and Spatial Memory Persistence”. *Nature Neuroscience* 17.12, pp. 1658–1660.
- McNaughton N., Neil and Philip J Corr (2004). “A Two-Dimensional Neuropsychology of Defense: Fear/Anxiety and Defensive Distance”. *Neuroscience & Biobehavioral Reviews* 28.3, pp. 285–305.
- McNaughton, B. L., C. A. Barnes, and J. O’Keefe (1983). “The Contributions of Position, Direction, and Velocity to Single Unit Activity in the Hippocampus of Freely-Moving Rats”. *Experimental Brain Research* 52.1, pp. 41–49.
- Millan, Mark J. (2003). “The Neurobiology and Control of Anxious States”. *Progress in Neurobiology* 70.2, pp. 83–244.

- Miller, Earl K. (2000). “The Prefrontal Cortex and Cognitive Control”. *Nature Reviews Neuroscience* 1.1, p. 59.
- Millman, K. Jarrod and Michael Aivazis (2011). “Python for Scientists and Engineers”. *Computing in Science & Engineering* 13.2, pp. 9–12.
- Montgomery, Sean M. and György Buzsáki (2007). “Gamma Oscillations Dynamically Couple Hippocampal CA3 and CA1 Regions during Memory Task Performance”. *Proceedings of the National Academy of Sciences of the United States of America* 104.36, pp. 14495–14500.
- Morales, Marisela and Elyssa B. Margolis (2017). “Ventral Tegmental Area: Cellular Heterogeneity, Connectivity and Behaviour”. *Nature Reviews Neuroscience* 18.2, pp. 73–85.
- Morris, R. G. et al. (1986). “Selective Impairment of Learning and Blockade of Long-Term Potentiation by an N-Methyl-D-Aspartate Receptor Antagonist, AP5”. *Nature* 319.6056, pp. 774–776.
- Moser, Edvard I., May-Britt Moser, and Bruce L. McNaughton (2017). “Spatial Representation in the Hippocampal Formation: A History”. *Nature Neuroscience* 20.11, pp. 1448–1464.
- Nakazawa, Kazu et al. (2002). “Requirement for Hippocampal CA3 NMDA Receptors in Associative Memory Recall”. *Science (New York, N.Y.)* 297.5579, pp. 211–218.
- Nicolelis, M. A. et al. (1995). “Sensorimotor Encoding by Synchronous Neural Ensemble Activity at Multiple Levels of the Somatosensory System”. *Science (New York, N.Y.)* 268.5215, pp. 1353–1358.
- O’Keefe, J and . Dostrovsky (1971). “The Hippocampus as a Spatial Map. Preliminary Evidence from Unit Activity in the Freely-Moving Rat”. *Brain Research* 34.1, pp. 171–175.
- O’Neill, Joseph et al. (2008). “Reactivation of Experience-Dependent Cell Assembly Patterns in the Hippocampus”. *Nature Neuroscience* 11.2, pp. 209–215.
- Oliphant, Travis E. (2007). “Python for Scientific Computing”. *Computing in Science & Engineering* 9.3, pp. 10–20.
- Pachitariu, Marius et al. (2016). “Kilosort: Realtime Spike-Sorting for Extracellular Electrophysiology with Hundreds of Channels”. *bioRxiv*, p. 061481.
- Packard, M. G. and J. L. McGaugh (1992). “Double Dissociation of Fornix and Caudate Nucleus Lesions on Acquisition of Two Water Maze Tasks: Further Evidence for Multiple Memory Systems”. *Behavioral Neuroscience* 106.3, pp. 439–446.
- Paré, Denis and Hélène Gaudreau (1996). “Projection Cells and Interneurons of the Lateral and Basolateral Amygdala: Distinct Firing Patterns and Differential Relation to Theta and Delta Rhythms in Conscious Cats”. *Journal of Neuroscience* 16.10, pp. 3334–3350.
- Paton, Joseph J. et al. (2006). “The Primate Amygdala Represents the Positive and Negative Value of Visual Stimuli during Learning”. *Nature* 439.7078, pp. 865–870.
- Pedregosa, Fabian et al. (2011). “Scikit-Learn: Machine Learning in Python”. *Journal of Machine Learning Research* 12, 28252830.
- Pennartz, Cyriel M. A., Henk J. Groenewegen, and Fernando H. Lopes da Silva (1994). “The Nucleus Accumbens as a Complex of Functionally Distinct Neuronal Ensembles: An Integration of Behavioural, Electrophysiological and Anatomical Data”. *Progress in Neurobiology* 42.6, pp. 719–761.
- Perez, Fernando and Brian E. Granger (2007). “IPython: A System for Interactive Scientific Computing”. *Computing in Science & Engineering* 9.3, pp. 21–29.

- Perreault, Charles (2010). “A Note on Reconstructing Animal Social Networks from Independent Small-Group Observations”. *Animal Behaviour* 80.3, pp. 551–562.
- Perrett, D. I., E. T. Rolls, and W. Caan (1982). “Visual Neurones Responsive to Faces in the Monkey Temporal Cortex”. *Experimental Brain Research* 47.3, pp. 329–342.
- Peters, Jamie, Peter W. Kalivas, and Gregory J. Quirk (2009). “Extinction Circuits for Fear and Addiction Overlap in Prefrontal Cortex”. *Learning & Memory* 16.5, pp. 279–288.
- Peyrache, Adrien et al. (2009). “Replay of Rule-Learning Related Neural Patterns in the Prefrontal Cortex during Sleep”. *Nature Neuroscience* 12.7, pp. 919–926.
- Peyrache, Adrien et al. (2010). “Principal Component Analysis of Ensemble Recordings Reveals Cell Assemblies at High Temporal Resolution”. *Journal of Computational Neuroscience* 29.1-2, pp. 309–325.
- Popescu, Andrei T., Daniela Popa, and Denis Paré (2009). “Coherent Gamma Oscillations Couple the Amygdala and Striatum during Learning”. *Nature Neuroscience* 12.6, pp. 801–807.
- Popescu, Andrei T., Michael R. Zhou, and Mu-ming Poo (2016). “Phasic Dopamine Release in the Medial Prefrontal Cortex Enhances Stimulus Discrimination”. *Proceedings of the National Academy of Sciences* 113.22, E3169–E3176.
- Quian Quiroga, Rodrigo and Stefano Panzeri (2009). “Extracting Information from Neuronal Populations: Information Theory and Decoding Approaches”. *Nature Reviews. Neuroscience* 10.3, pp. 173–185.
- Quiroga, R. Quian, Z. Nadasdy, and Y. Ben-Shaul (2004). “Unsupervised Spike Detection and Sorting with Wavelets and Superparamagnetic Clustering”. *Neural Computation* 16.8, pp. 1661–1687.
- Reardon, Sara (2017). “A Giant Neuron Found Wrapped around Entire Mouse Brain”. *Nature News* 543.7643, p. 14.
- Redgrave, P., T. J. Prescott, and K. Gurney (1999). “Is the Short-Latency Dopamine Response Too Short to Signal Reward Error?” *Trends in Neurosciences* 22.4, pp. 146–151.
- Redondo, Roger L. et al. (2014). “Bidirectional Switch of the Valence Associated with a Hippocampal Contextual Memory Engram”. *Nature* 513.7518, pp. 426–430.
- Rolls, Edmund T. (2012). “Invariant Visual Object and Face Recognition: Neural and Computational Bases, and a Model, VisNet”. *Frontiers in Computational Neuroscience* 6.
- Rondi-Reig, Laure et al. (2006). “Impaired Sequential Egocentric and Allocentric Memories in Forebrain-Specific-NMDA Receptor Knock-out Mice during a New Task Dissociating Strategies of Navigation”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 26.15, pp. 4071–4081.
- Rosen, Zev B., Stephanie Cheung, and Steven A. Siegelbaum (2015). “Midbrain Dopamine Neurons Bidirectionally Regulate CA3-CA1 Synaptic Drive”. *Nature Neuroscience* 18.12, pp. 1763–1771.
- Rothschild, Gideon, Elad Eban, and Loren M Frank (2017). “A Cortical–hippocampal–cortical Loop of Information Processing during Memory Consolidation”. *Nature neuroscience* 20.2, pp. 251–259.
- Rumelhart, David E., James L. McClelland, and CORPORATE PDP Research Group, eds. (1986). *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*. Cambridge, MA, USA: MIT Press.

- Russo, Eleonora and Daniel Durstewitz (2017). “Cell Assemblies at Multiple Time Scales with Arbitrary Lag Constellations”. *eLife* 6. Ed. by Marc Howard, e19428.
- Salgado, Sanjay and Michael G. Kaplitt (2015). “The Nucleus Accumbens: A Comprehensive Review”. *Stereotactic and Functional Neurosurgery* 93.2, pp. 75–93.
- Scalia, F. and S. S. Winans (1975). “The Differential Projections of the Olfactory Bulb and Accessory Olfactory Bulb in Mammals”. *The Journal of Comparative Neurology* 161.1, pp. 31–55.
- Schoenbaum, G., A. A. Chiba, and M. Gallagher (1999). “Neural Encoding in Orbitofrontal Cortex and Basolateral Amygdala during Olfactory Discrimination Learning”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 19.5, pp. 1876–1884.
- Schultz, W. (1998). “Predictive Reward Signal of Dopamine Neurons”. *Journal of Neurophysiology* 80.1, pp. 1–27.
- Schultz, W., P. Dayan, and P. R. Montague (1997). “A Neural Substrate of Prediction and Reward”. *Science (New York, N.Y.)* 275.5306, pp. 1593–1599.
- SEMON, Richard Wolfgang (1909). *Die Mnemischen Empfindungen in Ihren Beziehungen Zu Den Originalempfindungen ... Erste Fortsetzung Der "Mneme."* OCLC: 504684059. Pp. xv. 392. Leipzig.
- Semon, Richard Wolfgang (1904). *Die Mneme als erhaltendes Prinzip im Wechsel des organischen Geschehens*. OCLC: 707160345. Leipzig: Engelmann.
- (1921). *The mneme*. OCLC: 717164. London: Allen & Unwin.
- (1923). *Mnemic Psychology*; OCLC: 4027830. London: Allen & Unwin.
- Shannon, C. E. (1948). “A Mathematical Theory of Communication”. *The Bell System Technical Journal* 27.3, pp. 379–423.
- Shatz, C. J. (1992). “The Developing Brain”. *Scientific American* 267.3, pp. 60–67.
- Shi, C. J. and M. D. Cassell (1998). “Cortical, Thalamic, and Amygdaloid Connections of the Anterior and Posterior Insular Cortices”. *The Journal of Comparative Neurology* 399.4, pp. 440–468.
- Shi, C. and M. Davis (2001). “Visual Pathways Involved in Fear Conditioning Measured with Fear-Potentiated Startle: Behavioral and Anatomic Studies”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 21.24, pp. 9844–9855.
- Siapas, Athanassios G., Evgueniy V. Lubenov, and Matthew A. Wilson (2005). “Prefrontal Phase Locking to Hippocampal Theta Oscillations”. *Neuron* 46.1, pp. 141–151.
- Singer, W. (1999). “Neuronal Synchrony: A Versatile Code for the Definition of Relations?” *Neuron* 24.1, pp. 49–65, 111–125.
- Singer, Wolf (2018). “Neuronal Oscillations: Unavoidable and Useful?” *European Journal of Neuroscience* 48.7, pp. 2389–2398.
- Spellman, Timothy et al. (2015). “Hippocampal-Prefrontal Input Supports Spatial Encoding in Working Memory”. *Nature* 522.7556, pp. 309–314.
- Stettler, Dan D. et al. (2002). “Lateral Connectivity and Contextual Interactions in Macaque Primary Visual Cortex”. *Neuron* 36.4, pp. 739–750.
- Stringer, Carsen et al. (2019a). “High-Dimensional Geometry of Population Responses in Visual Cortex”. *Nature*, p. 1.
- Stringer, Carsen et al. (2019b). “Spontaneous Behaviors Drive Multidimensional, Brainwide Activity”. *Science* 364.6437, eaav7893.

- Stumpf, Michael P. H. and Carsten Wiuf (2005). “Sampling Properties of Random Graphs: The Degree Distribution”. *Physical Review E* 72.3, p. 036118.
- Sutton, Richard S. (1998). *Reinforcement Learning: An Introduction*. In collab. with Andrew G. Barto. Adaptive Computation and Machine Learning. Cambridge, Mass. ; London: MIT Press. xviii+322.
- Swanson, L. W. (1981). “A Direct Projection from Ammon’s Horn to Prefrontal Cortex in the Rat”. *Brain Research* 217.1, pp. 150–154.
- Takahashi, Yuji K. et al. (2009). “The Orbitofrontal Cortex and Ventral Tegmental Area Are Necessary for Learning from Unexpected Outcomes”. *Neuron* 62.2, pp. 269–280.
- Tanke, Nouk, J. Gerard G. Borst, and Arthur R. Houweling (2018). “Single-Cell Stimulation in Barrel Cortex Influences Psychophysical Detection Performance”. *Journal of Neuroscience* 38.8, pp. 2057–2068.
- Tian, Yonglu et al. (2018). “An Excitatory Neural Assembly Encodes Short-Term Memory in the Prefrontal Cortex”. *Cell Reports* 22.7, pp. 1734–1744.
- Tonegawa, Susumu, Mark D. Morrissey, and Takashi Kitamura (2018). “The Role of Engram Cells in the Systems Consolidation of Memory”. *Nature Reviews Neuroscience* 19.8, p. 485.
- Treves, A. and E. T. Rolls (1994). “Computational Analysis of the Role of the Hippocampus in Memory”. *Hippocampus* 4.3, pp. 374–391.
- Trouche, Stéphanie et al. (2013). “Fear Extinction Causes Target-Specific Remodeling of Perisomatic Inhibitory Synapses”. *Neuron* 80.4, pp. 1054–1065.
- Trouche, Stéphanie et al. (2016). “Recoding a Cocaine-Place Memory Engram to a Neutral Engram in the Hippocampus”. *Nature Neuroscience* 19.4, pp. 564–567.
- Trouche, Stéphanie et al. (2019). “A Hippocampus-Accumbens Tripartite Neuronal Motif Guides Appetitive Memory in Space”. *Cell* 176.6, 1393–1406.e16.
- Tsien, J. Z., P. T. Huerta, and S. Tonegawa (1996). “The Essential Role of Hippocampal CA1 NMDA Receptor-Dependent Synaptic Plasticity in Spatial Memory”. *Cell* 87.7, pp. 1327–1338.
- Ungless, Mark A. (2004). “Dopamine: The Salient Issue”. *Trends in Neurosciences* 27.12, pp. 702–706.
- Van de Ven, Gido M. et al. (2016). “Hippocampal Offline Reactivation Consolidates Recently Formed Cell Assembly Patterns during Sharp Wave-Ripples”. *Neuron* 92.5, pp. 968–974.
- Victor, J. D. and K. P. Purpura (1996). “Nature and Precision of Temporal Coding in Visual Cortex: A Metric-Space Analysis”. *Journal of Neurophysiology* 76.2, pp. 1310–1326.
- Vidaurre, Diego et al. (2016). “Spectrally Resolved Fast Transient Brain States in Electrophysiological Data”. *NeuroImage* 126, pp. 81–95.
- Vidaurre, Diego et al. (2019). “Temporally Unconstrained Decoding Reveals Consistent but Time-Varying Stages of Stimulus Processing”. *Cerebral Cortex* 29.2, pp. 863–874.
- Vogel, J. R., B. Beer, and D. E. Clody (1971). “A Simple and Reliable Conflict Procedure for Testing Anti-Anxiety Agents”. *Psychopharmacologia* 21.1, pp. 1–7.
- Watts, D. J. and S. H. Strogatz (1998). “Collective Dynamics of ‘small-World’ Networks”. *Nature* 393.6684, pp. 440–442.
- Weiskrantz, Lawrence (1956). “Behavioral Changes Associated with Ablation of the Amygdaloid Complex in Monkeys”. *Journal of Comparative and Physiological Psychology* 49.4, pp. 381–391.

- Wespatat, Valérie, Frank Tennigkeit, and Wolf Singer (2004). “Phase Sensitivity of Synaptic Modifications in Oscillating Cells of Rat Visual Cortex”. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* 24.41, pp. 9067–9075.
- Williams, Alex H. et al. (2018). “Unsupervised Discovery of Demixed, Low-Dimensional Neural Dynamics across Multiple Timescales through Tensor Component Analysis”. *Neuron* 98.6, 1099–1115.e8.
- Williams, Graham V. and Patricia S. Goldman-Rakic (1995). “Modulation of Memory Fields by Dopamine D1 Receptors in Prefrontal Cortex”. *Nature* 376.6541, pp. 572–575.
- Wilson, M. A. and B. L. McNaughton (1993). “Dynamics of the Hippocampal Ensemble Code for Space”. *Science (New York, N.Y.)* 261.5124, pp. 1055–1058.
- Winterhalder, Matthias et al. (2005). “Comparison of Linear Signal Processing Techniques to Infer Directed Interactions in Multivariate Neural Systems”. *Signal Processing. Neuronal Coordination in the Brain: A Signal Processing Perspective* 85.11, pp. 2137–2160.
- Wise, Roy A. (2004). “Dopamine, Learning and Motivation”. *Nature Reviews Neuroscience* 5.6, p. 483.
- Wohrer, Adrien, Mark D. Humphries, and Christian K. Machens (2013). “Population-Wide Distributions of Neural Activity during Perceptual Decision-Making”. *Progress in Neurobiology* 103, pp. 156–193.
- Wu, Fan et al. (2015). “Monolithically Integrated pLEDs on Silicon Neural Probes for High-Resolution Optogenetic Studies in Behaving Animals”. *Neuron* 88.6, pp. 1136–1148.
- Yger, Pierre 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. Ed. by David Kleinfeld, e34518.
- Yu, Byron M. et al. (2009). “Gaussian-Process Factor Analysis for Low-Dimensional Single-Trial Analysis of Neural Population Activity”. *Journal of Neurophysiology* 102.1, pp. 614–635.
- Zippo, Antonio G. et al. (2013). “Neuronal Functional Connection Graphs among Multiple Areas of the Rat Somatosensory System during Spontaneous and Evoked Activities”. *PLoS computational biology* 9.6, e1003104.
- Zou, Hui, Trevor Hastie, and Robert Tibshirani (2006). “Sparse Principal Component Analysis”. *Journal of Computational and Graphical Statistics* 15.2, pp. 265–286.