Neural population activity in the visual cortex: Statistical methods and application

Benjamin R. Cowley

MAY 2018
CMU-ML-18-102

Machine Learning Department
School of Computer Science
Carnegie Mellon University
Pittsburgh, PA 15213

Thesis Committee:
Byron Yu, Co-chair
Matthew A. Smith, Co-chair (University of Pittsburgh)
Geoff Gordon (Carnegie Mellon University)
Adam Kohn (Albert Einstein College of Medicine)
Maneesh Sahani (Gatsby Computational Neuroscience Unit)

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy.

This research was sponsored by a National Science Foundation Graduate Research Fellowship, a National Defense Science and Engineering Graduate Fellowship, a Richard K. Mellon BrainHub Fellowship, a NSF-NCS BCS grant (1734901/1734916), a National Eye Institute grant (R01EY022928), and the Simons Foundation (364994).

Keywords: Dimensionality reduction, macaque visual cortex, adaptive stimulus selection

Think cortically, act neuronally.

Abstract

The traditional approach to understand the visual cortex is to relate the responses of a visual cortical neuron to the visual stimulus. However, the response of a neuron is not only related to the stimulus but also on the responses of other neurons. One approach to identify the interactions across neurons is dimensionality reduction, which identifies latent variables that are shared among neurons. The focus of this thesis is to apply dimensionality reduction to activity recorded from visual cortical neurons to gain insight into the underlying neural mechanisms that govern the interactions among neurons. In the first part of this thesis, we use dimensionality reduction to ask if the complexity of the visual stimulus varies with the complexity of neural activity in monkey primary visual cortex (V1). We systematically vary the number of neurons, trials, and stimuli, and find that dimensionality reduction returns sensible and interpretable outputs. This motivates us to use dimensionality reduction for visual brain areas beyond primary visual cortex that are farther from the sensory periphery.

In the second part of this thesis, we focus on understanding the activity of neurons in monkey V4, a visual brain area known for mid-level visual processing. Because V4 neurons selectively respond to complex features of visual stimuli and interact with higher-cortical brain areas, new statistical methods are needed. First, we develop an adaptive stimulus selection algorithm that efficiently chooses natural images that elicit diverse responses from V4 neurons. Next, to relate the recorded V4 activity to activity recorded from other brain areas, we develop a novel dimensionality reduction method that identifies linear and nonlinear interactions among brain areas. Finally, we use these methods to characterize a latent variable that slowly drifts in V4 activity on a long time scale (30 minutes). We find that the slow drift is present in both V4 and PFC, and is related to slow changes in behavior, suggesting that the slow drift is an arousal signal. Overall, this thesis advances a new way to analyze population activity recorded from the visual cortex that can better elucidate how visual cortical neurons transform visual input into behavior.

Acknowledgments

Thanks to my co-advisor Byron Yu, who always made time for me and asked insightful questions. Thanks to my co-advisor Matthew Smith, who supported us to run experiments and now explains dimensionality reduction better than myself. Thanks to my many friends, family, collaborators, and colleagues who have supported me throughout my PhD.

Contents

List of Figures

The following publications are represented in this thesis:

1. Cowley, Benjamin R., Matthew A. Smith, Adam Kohn, and Byron M. Yu. "Stimulus-driven population activity patterns in macaque primary visual cortex." *PLoS computational biology* 12, no. 12 (2016): e1005185.
2. Cowley, Benjamin R., Ryan C. Williamson, Katerina Acar, Matthew A. Smith, and Byron M. Yu. "Adaptive stimulus selection for optimizing neural population responses." In *Advances in Neural Information Processing Systems*, pp. 1395-1405. 2017.
3. Cowley, Benjamin R., Joao D. Semedo, Amin Zandvakili, Matthew A. Smith, Adam Kohn, and Byron M. Yu. "Distance Covariance Analysis." In *Artificial Intelligence and Statistics*, pp. 242-251. 2017.
4. Williamson, Ryan C., Benjamin R. Cowley, Ashok Litwin-Kumar, Brent Doiron, Adam Kohn, Matthew A. Smith, and Byron M. Yu. "Scaling properties of dimensionality reduction for neural populations and network models." *PLoS computational biology* 12, no. 12 (2016): e1005141.
5. Cowley, Benjamin R., Matthew T. Kaufman, Zachary S. Butler, Mark M. Churchland, Stephen I. Ryu, Krishna V. Shenoy, and Byron M. Yu. "DataHigh: graphical user interface for visualizing and interacting with high-dimensional neural activity." *Journal of neural engineering* 10, no. 6 (2013): 066012.

Chapter 1

Introduction

The brain comprises roughly 100 billion neurons, making it one of the most complex systems to study. One approach to understanding this system is to record from neurons while measuring behavior. The idea is that if we can relate neural activity to behavioral variables, we can begin to understand the computational mechanisms that generate the behavior. To increase the likelihood of finding a relationship between neural activity and behavior, we desire to record from as many neurons as possible during an experiment. In current experiments, we can typically record from hundreds of neurons, but advances in recording technologies may increase this number to tens and hundreds of thousands within years (Stevenson and Kording, 2011). This poses an intriguing question: While we certainly desire to record from all neurons that comprise a neural circuit, how do we begin to understand the activity of hundreds to thousands of recorded neurons?

To understand the activity of a population of neurons (termed *population activity*), one might display the raster plot for each trial, where a tick mark represents a neuron's action potential (Fig. 1.1*A*). As the number of neurons and trials grows, it can be difficult to pick out key features in the raster plots that differentiate one trial from another (Churchland and Shenoy, 2007). In addition, one may seek to understand how population activity differs across experimental conditions. A common approach is to average the spike trains across trials to create a peri-stimulus time histogram (PSTH) for each neuron and experimental condition (Fig. 1.1*B*). As the numbers of neurons and conditions increase, the task of comparing population dynamics across different conditions can be challenging due to the hetereogeneity of the PSTHs (Churchland et al., 2007; Machens et al., 2010; Mante et al., 2013; Rigotti et al., 2013)

To overcome these difficulties, we can extract a smaller number of *latent variables* that succinctly summarize the population activity for each experimental trial (Fig. 1.1*C*) or for each experimental condition (Fig. 1.1*D*). There are two complementary ways of understanding the relationship between the latent variables and the recorded neural activity. First, the latent variables can be viewed as "readouts" of the population activity, where each latent variable captures a prominent co-fluctuation shared among the recorded neurons. The latent variables can be obtained by simply adding and subtracting the activity of different neurons, while possibly incorporating smoothing in time. Second, because these latent variables capture the most prominent co-fluctuations, the population activity can be "reconstructed" by adding and subtracting the patterns in different ways for different trials or conditions.

To extract these latent variables, one can apply a dimensionality reduction method to the

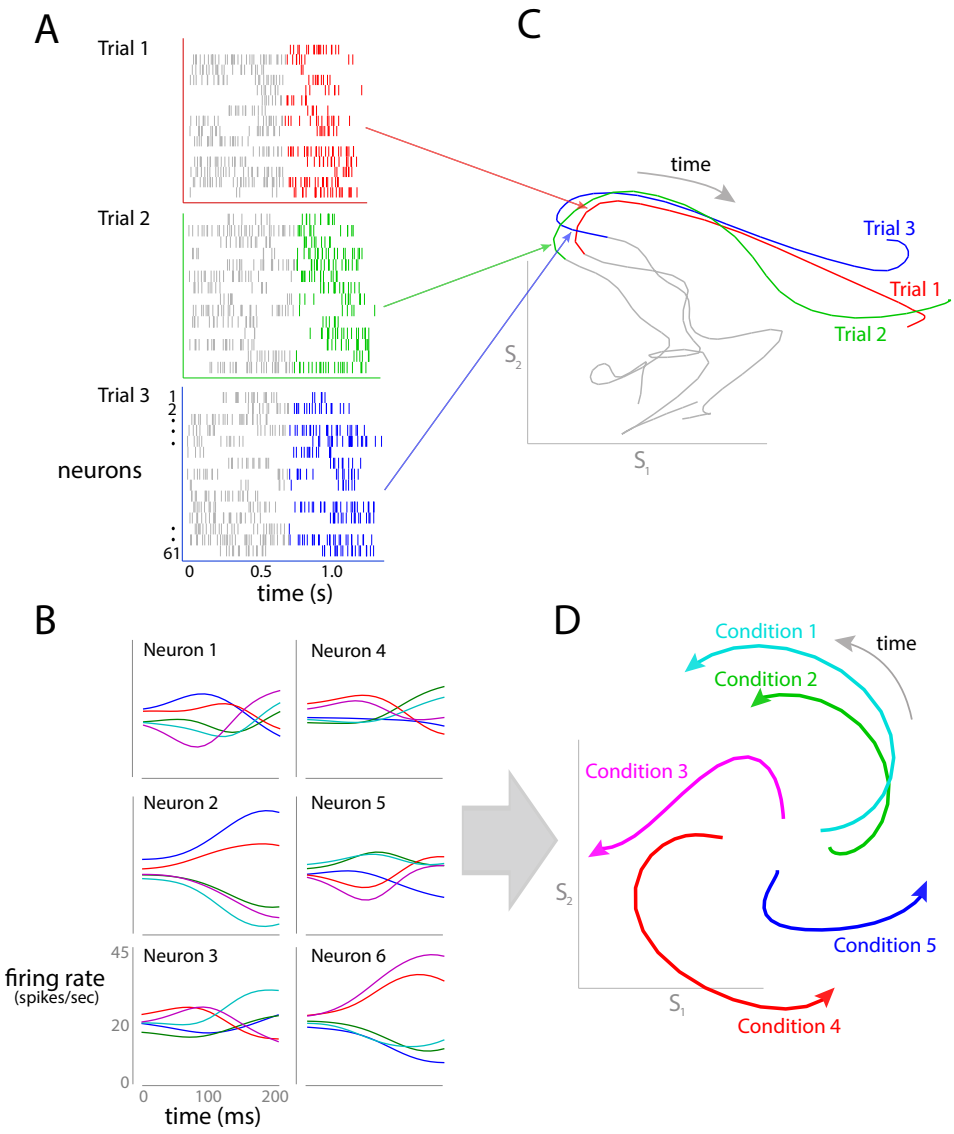


Figure 1.1: Conceptual illustration of applying dimensionality reduction to neural population activity. *A*. Comparing population activity across repeated trials of the same experimental condition. Each raster plot corresponds to an individual experimental trial. *B*. Comparing trial-averaged population activity across different experimental conditions. The peri-stimulus time histograms (PSTHs) of six neurons with five different experimental conditions are shown. *C*. A dimensionality reduction method (GPFA) was applied to single-trial population activity (three trials are shown in panel A) to extract 15-d single-trial neural trajectories. Each trajectory corresponds to a different experimental trial. S_1 and S_2 define a 2-d projection of the extracted 15-d latent space. *D*. A dimensionality reduction method (PCA) was applied to trial-averaged population activity (five experimental conditions are shown in panel B) to extract 6-d trial-averaged neural trajectories. Each trajectory corresponds to a different experimental condition. S_1 and S_2 define a 2-d projection of the extracted 6-d latent space.

population activity. Importantly, the number of latent variables is typically much smaller than the number of neurons. This enables us to analyze a small number of latent variables that represent the activity of the many neurons. To date, dimensionality reduction has been largely used to understand population activity from motor (Churchland et al., 2010, 2012; Cunningham and Yu, 2014; Santhanam et al., 2009), olfactory (Mazor and Laurent, 2005), and prefrontal (Mante et al., 2013; Rigotti et al., 2013) cortices. However, few studies have applied these dimensionality reduction methods to population activity from the visual cortex. This is surprising, as visual cortical neurons have rich and complicated responses that depend on the dynamics and features of the stimulus (Kohn, 2007), on the context of the task (Cohen and Maunsell, 2010, 2009; Ruff and Cohen, 2014), and on the responses of nearby and faraway neurons (Bondy et al., 2018; Coen-Cagli et al., 2015). Because encoding models may make strong assumptions about the response properties of visual cortical neurons (Carandini et al., 2005; Olshausen and Field, 2005), dimensionality reduction is a complementary approach that makes little to no assumptions about the neural activity and thus can find salient features of the neural responses that modelling may miss.

In this thesis, we empirically test if dimensionality reduction is an appropriate tool to study population activity from the visual cortex. As a first test, we applied a commonly-used dimensionality reduction method, principal components analysis, to activity recorded from neurons in the primary visual cortex, whose response properties are well-known (Carandini et al., 2005; Olshausen and Field, 2005). We showed that the outputs of dimensionality reduction are interpretable and can be related to the well-known response properties. Encouraged by these results, we then focused on applying dimensionality reduction on population activity from V4, a brain area known for mid-level visual processing but whose neurons have response properties that are not well-defined (Roe et al., 2012; Yamane et al., 2008). Because V4 neurons sparsely respond to complex features of visual stimuli (e.g., curvature and color) and receive feedback from higher cortical areas (e.g., attention), new statistical methods were needed beyond commonly-used dimensionality reduction methods. We developed and employed an adaptive stimulus selection algorithm that efficiently chooses natural images that elicit diverse responses from V4 neurons. This algorithm provided a principled approach to ensure that the neurons were driven strongly by our chosen set of stimuli. Next, we developed a novel dimensionality reduction method, called distance covariance analysis (DCA), that combines the interpretability of linear dimensions with the ability to detect both linear and nonlinear interactions. Importantly, DCA can detect interactions among brain areas, as well as take stimulus and behavioral variables into account.

We used both adaptive stimulus selection and DCA to understand interactions between V4 and prefrontal cortex. We found that the population activity in V4 and PFC co-varied on a slow time scale (~30 minutes), much longer than those of previously reported fluctuations (Kohn et al., 2016; McGinley et al., 2015b). This slow drift was correlated in time with slow changes in behavior, suggesting the slow drift is related to arousal. In addition, we found evidence that the slow drift does not necessarily corrupt the ability of V4 neurons to encode stimulus information. This is because a downstream readout may access the slow drift and remove it from the readout. This work showcases how applying the dimensionality reduction framework to populations of visual cortical neurons can lead to new scientific discoveries.

This thesis makes advances in both machine learning and neuroscience. For machine learning, we develop new statistical methods motivated by neuroscientific questions in the areas of

dimensionality reduction and active learning. We compare these methods to other state-of-the-art machine learning methods. For neuroscience, we validate the use of dimensionality reduction to study high-dimensional population activity from the visual cortex. We then use dimensionality reduction to uncover a previously-unknown internal signal of the brain, and link that signal to behavior. Overall, this thesis furthers our ability to analyze and understand the activity of many neurons.

The thesis is structured as follows. Chapter 2 covers background and related work on dimensionality reduction, adaptive stimulus selection, and global fluctuations of the brain. In Chapter 3, we validate dimensionality reduction by applying it to population activity from the macaque primary visual cortex and then assessing if its outputs are sensible and interpretable. In Chapter 5, we propose a dimensionality reduction method, called *distance covariance analysis* (DCA), which combines the interpretability of a linear mapping from neural activity to latent variables with the ability to detect both linear and nonlinear interactions. Chapter 4 explores an adaptive stimulus selection method, called *Adept*, that chooses the next stimulus to present such that Adept optimizes the responses of a population of neurons. Finally, in Chapter 6, we use our dimensionality reduction framework to understand interactions between two brain areas. We then consider our related work in Chapter 7, and provide an outlook on future areas of work relevant to this thesis in Chapter 8.

Chapter 2

Background

Before presenting the contributions of this thesis, we first provide a brief review of the anatomical and physiological properties of the brain areas considered in this thesis. We then survey recent approaches in understanding the activity of many neurons. Many of these approaches have taken the form of dimensionality reduction, although the types of dimensionality reduction methods vary greatly across studies. We discuss these different methods and for what types of data they have been successful. Then, we change the focus onto previous adaptive stimulus selection methods that hold promise for settings in which we can record from many neurons but have limited recording time. Finally, we end the chapter with a review of recent work that relates the latent variables extracted from dimensionality reduction to neural mechanisms that globally fluctuate the activity of many neurons.

2.1 Review of the anatomy and physiology of the brain

Here, we give a basic overview of the anatomy and physiology of specific brain areas of the macaque monkey that will be useful to know for this thesis. The brain of the macaque monkey is smaller and less wrinkled than a human's brain, but the anatomy of the macaque brain largely matches that of a human's (Fig. 2.1*A*). We consider three prominent brain areas of the macaque: V1, V4, and PFC (prefrontal cortex). One can think of these areas as part of a chain of brain areas that lead to a decision (e.g., if two stimuli are the same or different) (Fig. 2.1*B*). The first part of the chain is the lateral geniculate nucleus (LGN), which relays its input of the visual scene from retinal cells to neurons in the primary visual cortex (V1). In turn, the V1 neurons are thought to extract edges of the visual scene, and pass this information along to other higher-level visual areas, such as V4. As a mid-level visual processing area, V4 extracts color and contour information from the visual scene. After many stages of processing, the brain's representation of the visual scene can be read off from a decision-making brain area, such as PFC. PFC can then use this representation to make a decisions, and sends this signal to motor output areas. While this is a much oversimplified characterization of these brain areas, it will be useful to think about the different parts of this chain throughout this thesis. We now give more details about the anatomy and physiology of these brain areas, followed by a description of the recording devices we use.

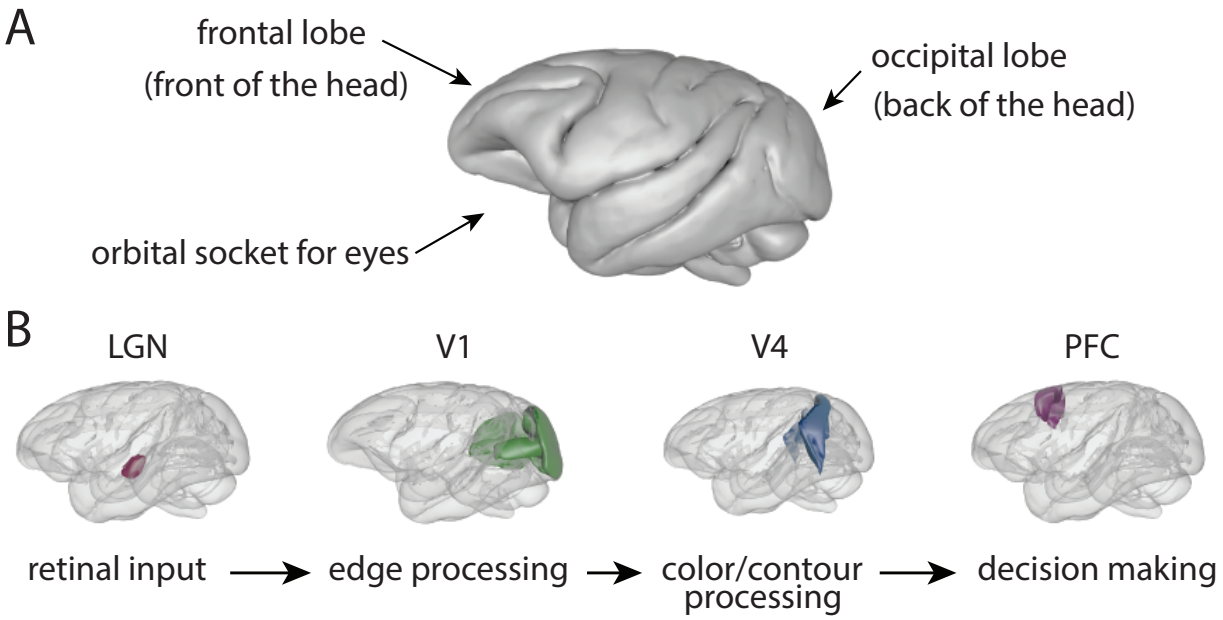


Figure 2.1: Brain of macaque monkey. *A*. The brain is presented along a side view of the head, where the eyes are to the viewer's left. The frontal lobe is the brain region to the front of the head, and the occipital lobe is the brain region to the back of the head. The "orbital socket for eyes" denotes where the eyes of the monkey reside (not shown). *B*. Four different brain structures and their simplified function are highlighted. Images captured from the Scalable Brain Atlas (Bakker et al., 2015).

2.1.1 Primary visual cortex, V1

The primary visual cortex (V1) is located at the very back of the head in the occipital lobe (Fig. 2.1*B*), and is primarily known for edge detection (Carandini et al., 2005). V1 receives feedforward inputs from the LGN. These inputs are organized in such a way as to structure the spatial location of neurons in V1 to the spatial visual area (Casagrande and Kaas, 1994; Olshausen and Field, 2005). V1 also receives a large amount of feedback from higher-order visual areas (Rockland and Van Hoesen, 1994), whose purpose has been theorized to enable Bayesian inference of the visual scene (Lee et al., 1998). V1 projects its output to higher-order visual areas, such as V2, V4, and MT, as well as feedback to LGN (Casagrande and Kaas, 1994). These results, along with lesion studies, indicate that V1 is critical for visual perception (Carandini et al., 2005; Olshausen and Field, 2005).

The standard stimulus-response encoding model of V1 is a Gabor-like filter (Fig. 2.2) (Olshausen and Field, 2005). The model applies a linear kernel to the image and applies two nonlinearites (normalization and a pointwise nonlinearity) to predict a V1 neuron's response ($r(t)$). The oriented linear filter allows V1 neurons to detect edges, while the nonlinearities have been shown to capture important deviations of V1 responses from a linear filter's prediction (Carandini et al., 2005; Olshausen and Field, 2005). Still, this model is not a perfect description, and it tends to fail at predicting V1 responses to natural images (Olshausen and Field, 2005; Vinje and Gallant, 2000). Recent studies employing deep convolutional neural networks have achieved

better prediction performance, but not substantially high (~50) (Cadena et al., 2017; Kindel et al., 2017).

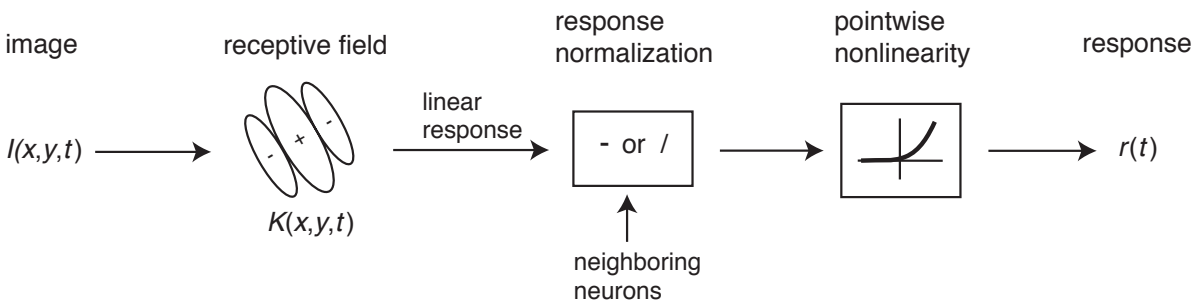


Figure 2.2: Standard encoding model of V1 neuron. A dot product is taken between an image $I(x, y, t)$ and a linear filter $K(x, y, t)$, which can be thought of as a 'receptive field', to produce a linear response. This response is then normalized by surrounding responses and then passed through a pointwise nonlinearity to predict the response $r(t)$ of a V1 neuron. Diagram taken from Olshausen and Field (2005).

2.1.2 Visual brain area V4

Brain area V4 is a mid-level visual processing area known for its selectivity to curvature and color, as well as its role in feature and spatial attention (Roe et al., 2012). V4 is located in the back of the brain, but closer to the center of the brain than V1 (Fig. 2.1*B*). V4 is a highly connected area, receiving input from the primary visual cortex (V1), other lower-level visual areas (e.g., V2, V3), higher-level visual areas (e.g., IT), motion-processing areas (e.g., MT), as well as higher-cortical areas in the frontal cortex (e.g., FEF) (Ungerleider et al., 2007). V4 outputs its activity as feedback to V2 and V3, as feedfoward input to IT, and as reciprocal input to higher-cortical areas, such as FEF (Ungerleider et al., 2007). All of these connections place V4 as a hub for object recognition and attention.

Unlike the response properties of V1 neurons, the response properties of V4 neurons are not fully understood. This is in part due to the a V4 neuron's complex receptive field preference for surface properties (e.g., color, brightness, texture), shape (e.g., orientation, curvature), motion, and depth (Roe et al., 2012). For example, consider the responses of two V4 neurons to different natural images (Fig. 2.3*A*). One V4 neuron shows a preference for images of teddy bears (i.e., images of teddy bears maximally drives the V4 neuron, Fig. 2.3*A*, top), while the other shows a preference for arranged fruit (Fig. 2.3*A*, bottom). When plotting the responses of both neurons together, we observe that both neurons are moderately activated by images of animals (Fig. 2.3*B*). As evident from these responses, deciphering which features a V4 neuron prefers can be difficult, and is one reason why no standard stimulus-response encoding model exists for V4 neurons. To date, the encoding model that achieves state-of-the-art prediction of V4 responses is a linear combination of units in the middle layer of a deep convolutional neural network (Yamins et al., 2014).

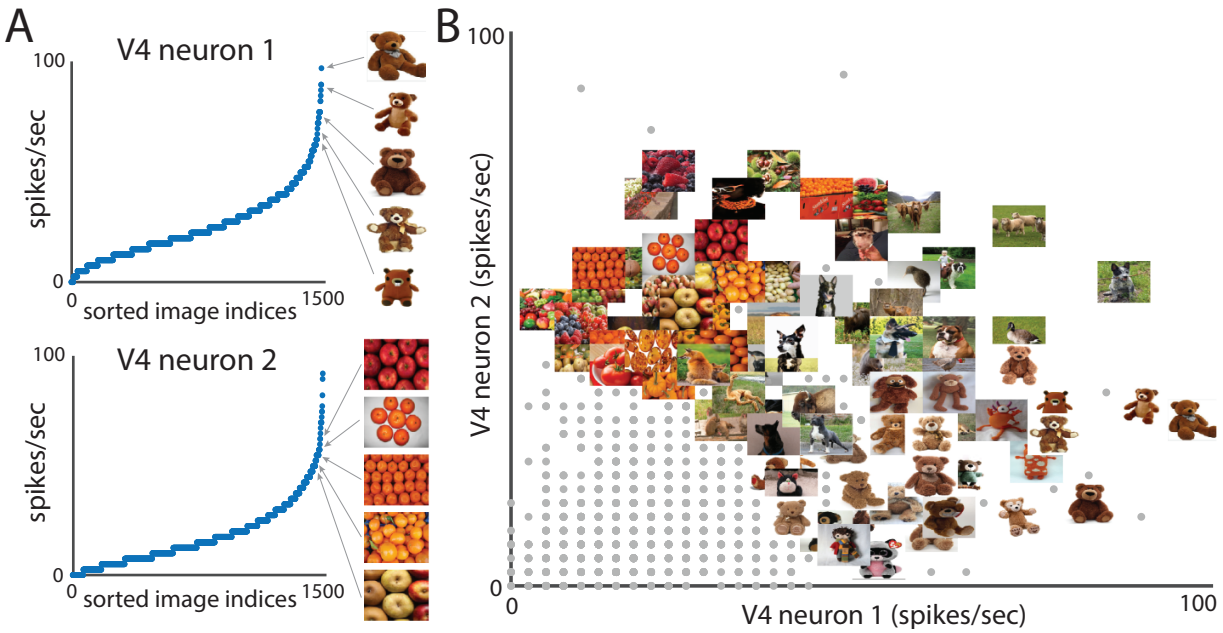


Figure 2.3: Responses of two macaque V4 neurons. *A*. Different neurons prefer different stimuli. Displayed images evoked 5 of top 25 largest responses. *B*. Images placed according to their responses. Gray dots represent responses to other images. Same neurons as in *A*.

2.1.3 Higher cortical brain area PFC

Briefly, the prefrontal cortex is in the portion of the brain near the front of the head and is located in the frontal lobe (Fig. 2.1*B*). PFC is densely connected, receiving input and projecting output to many different brain areas (Cavada et al., 2000). PFC is thought to control executive functions, such as decision making (Alvarez and Emory, 2006), contribute to working memory (Miller et al., 1996), and integrate sensory information (Mante et al., 2013). Thus, PFC likely plays a role in sensory feedback (i.e., attention) as well as arousal. However, much is still unknown about the neural properties of PFC (Alvarez and Emory, 2006). In this work, we consider dorsolateral PFC, also known as 8ar.

2.1.4 Recording devices

In this thesis, we simultaneously record from a population of neurons using a multi-electrode array (Fig. 2.4). Each array contains a 10×10 grid of 100 micro-electrodes spaced out equally with 400 μm intervals (Kelly et al., 2007). The arrays are chronically implanted into the monkey by first removing a small portion of the skull to reveal the underlying brain tissue. The electrodes of the array are inserted into the brain with a pneumatically driven "punch." A wire bundle connected to the electrodes of the array leads to a connector that is attached to the skull, allowing chronic recordings. Array recordings last typically several months to years before the neural signals are lost, likely due to scarring tissue built up around the electrodes (Polikov et al., 2005). A single electrode typically records a voltage traces containing the action potentials (or "spikes") of 0-3 neurons. The voltage trace is "spike-sorted" to remove any recording artifacts and recover

each neuron's action potentials. The spike counts for each neuron are then taken within some time bin (e.g., 1 second bins).

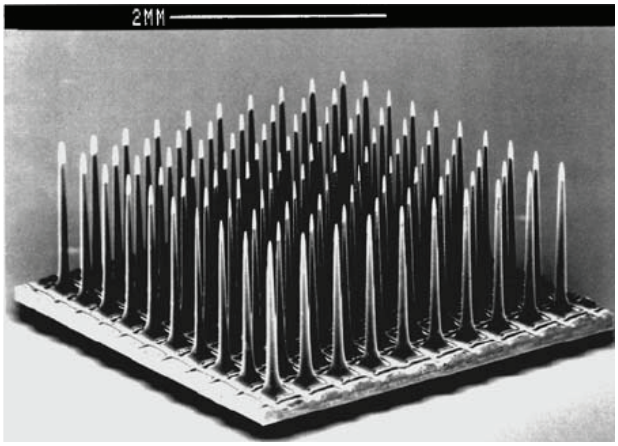


Figure 2.4: An image of the multi-electrode array used to record populations of neurons. The array is roughly one fifth the size of a penny. Image taken from (Kim et al., 2006).

2.2 Dimensionality reduction methods used in neuroscience

Dimensionality reduction has been applied to neural population activity to study decision making (Harvey et al., 2012; Mante et al., 2013), motor control (Churchland et al., 2012; Sadtler et al., 2014), olfaction (Mazor and Laurent, 2005), working memory (Daie et al., 2015; Rigotti et al., 2013), audition (Luczak et al., 2009), rule learning (Durstewitz et al., 2010), speech (Bouchard et al., 2013), and more (for a review, see Cunningham and Yu (2014)). Many of these studies adopted out-of-the-box dimensionality reduction techniques from machine learning that are applicable to many different types of data and make little assumptions about the joint distribution of the observed variables. Such methods include principal components analysis and factor analysis, which seek to identify latent variables that are linear combinations of the observed variables (Bishop, 2006). Other methods, such as local-linear embedding (Roweis and Saul, 2000), isomap (Tenenbaum et al., 2000), and tSNE (Maaten and Hinton, 2008), seek to identify latent variables that capture nonlinear relationships between the observed variables. These methods have been fruitfully applied in neuroscience (Cunningham and Yu, 2014; Stopfer et al., 2003; Ziemba et al., 2016), but have primarily been used as a means of visualizing neural activity rather than testing scientific hypotheses (Sadtler et al., 2014, but see). Further work is needed to validate the interpretability of the output of dimensionality reduction methods for neuroscience. A promising brain area in which these methods may be used to foster new scientific discoveries is the visual cortex. Because the responses of visual cortical neurons are largely dependent on presented visual stimuli, latent variables identified by dimensionality reduction can be attributed to the stimulus parameters. This is not usually the case for activity recorded from other brain areas, where it is unclear how a latent variable relates to stimulus parameters or behavioral variables. Our initial work on applying dimensionality reduction to visual cortical data is presented in Chapter 2.

2.2.1 Dimensionality reduction methods tailored for neuroscience

Building off the initial success of out-of-the-box dimensionality reduction methods, many new dimensionality reduction methods have been developed to be tailored to neural data. These methods address a challenging aspect of neural data in which a neuron's observed output (i.e., its action potentials) is binary rather than continuous, the latter of which is assumed by many out-of-the-box methods. To overcome this challenge, new methods assume that while the observed output of a neuron is binary, the underlying firing rate of the neuron (a latent variable) smoothly changes over time. By leveraging simultaneous recordings of neurons and assuming a prior that latent variables smoothly change over time, these new methods, including Gaussian-process factor analysis (Yu et al., 2009), time-delay Gaussian-process factor analysis (Lakshmanan et al., 2015), variational latent Gaussian process (Zhao and Park, 2017), and Poisson Gaussian-process latent variable model (Wu et al., 2017), yield smooth, more interpretable latent variables for single trials and a better reconstruction error than that of out-of-the-box methods. Another method (fLDS) assumes that the latent variables follow a linear dynamical system (i.e., each latent state depends on its previous latent state), and uses a neural network to map the latent state to an estimate of the underlying firing rate of an observed neuron (Gao et al., 2016). In a similar spirit, latent factor analysis via dynamical systems (LFADS) uses a recurrent neural network (RNN) to model the dynamics of the latent variables, and then applies factor analysis to the activity of the RNN neurons to estimate the underlying firing rates of the observed neurons (Sussillo et al., 2016). These methods provide a wide selection of choices when analyzing and interpreting high-dimensional population activity. The most appropriate method for a particular neural data set will depend on the number of neurons and the number of trials. In addition, the choice of method will also involve a trade-off among the desired amount of interpretability, computation time, and cross-validated reconstuction error. We provide an in-depth perspective on choosing between different dimensionality reduction methods based on the neural data set in Chapter 7.

2.2.2 Targeted dimensionality reduction methods

Another challenge for dimensionality reduction is that the identified latent variables are often difficult to attribute to experimenter-defined stimulus parameters or measured behavioral variables. This can make interpreting the latent variables difficult (Cunningham and Yu, 2014). To make latent variables more interpretable, targeted dimensionality reduction methods have been developed to identify latent variables of the population activity that reflect the stimulus and behavioral variables. Some of these methods come from machine learning, including linear regression and kernel dimensionality reduction (Fukumizu et al., 2004b). Other methods were developed specifically for neuroscience, including jPCA, which identifies latent variables that capture rotational structure in the population activity (Churchland et al., 2012), as well as demixed PCA, which identifies latent variables that distinctly encode different experimentally-defined variables (Brendel et al., 2011; Kobak et al., 2016). Additionally, a long line of research has been pursued about Poisson generalized linear models (GLMs), whose aim is to model a neuron's response based on its previous responses as well as stimulus input and other relevant features (Paninski, 2004; Pillow et al., 2008; Truccolo et al., 2005; Vidne et al., 2012). Targeted dimensionality reduction methods can provide a more interpretable perspective of neural data by allowing one

to analyze the extracted latent variables, in comparison to predictive methods, such as support vector machines and neural networks, that focus on fitting the mapping between neural activity and stimulus or behavioral variables and output prediction accuracies. The former approach is conducive for exploratory data analysis, and may be a valuable pre-processing step for the latter approach.

2.2.3 Dimensionality reduction to identify interactions among brain areas

An exciting line of research in neuroscience is to understand the interactions among two or more brain areas by simultaneously recording a population of neurons in each brain area. These interactions may be nonlinear, occur on multiple time scales, and vary in strength. To begin to understand such interactions, one may make the weak assumption that these interactions can be explained by a number of latent variables smaller than the number of recorded neurons. After dimensionality reduction, one can simply plot an extracted latent variable of one brain area against the extracted latent variable of another brain area to observe the interaction. For example, consider the population activity of two brain areas that are recorded simultaneously (Fig. 2.5*A*, illustrative example). We can identify two axes ($\mathbf{a}_1$ and $\mathbf{a}_2$), one for each brain area, that capture the interaction between the brain areas. When we project the population activity onto each axis and plot the projections against one another, we find a nonlinear interaction between the brain areas (Fig. 2.5*B*). This interaction can be missed or difficult to interpret by only analyzing the high-dimensional population activity (Fig. 2.5*A*).

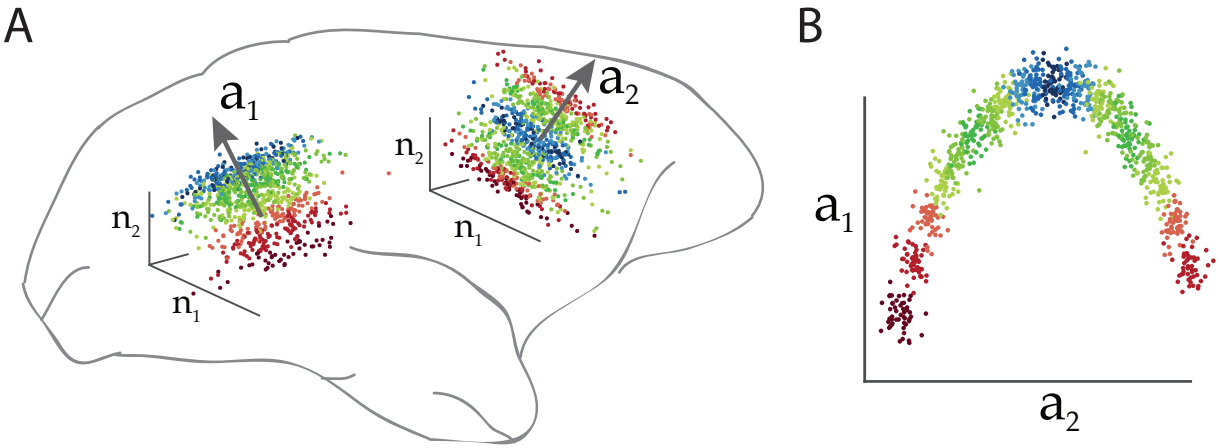


Figure 2.5: Illustrative example of an interaction between two brain areas. *A*. Consider the population activity of two brain areas, where n_i corresponds to the firing rate of one recorded neuron. Neurons from both brain areas are recorded simultaneously, such that each dot in the left plot (corresponding to the binned spike counts for one trial) has a corresponding dot in the right plot with the same color. We can identify axes $\mathbf{a}_1$ and $\mathbf{a}_2$ in the firing rate spaces whose projected activity is most related between the brain areas. *B*. The projected activity of $\mathbf{a}_1$ and $\mathbf{a}_2$, which are the latent variables of the population activity in *A*, are plotted against one another. By plotting the latent variables in this way, interactions between the brain areas become salient that may otherwise be missed by analyzing the high-dimensional population activity directly.

Some out-of-the-box machine learning methods have already been applied to these multiple-brain-area recordings, including canonical correlation analysis (Hotelling, 1936), partial least squares (Helland, 1988; Höskuldsson, 1988), and reduced-rank regression (Izenman, 1975). Few methods have been developed that are tailored to these type of data. One example of these methods is group latent auto-regressive analysis, which identifies time-varying latent variables that capture interactions within each brain area as well as across brain areas (Semedo et al., 2014). More work is needed in developing dimensionality reduction methods to understand inter-area interactions. In Chapter 5, we propose a dimensionality reduction method to understand interactions among two or more brain areas that can take into account stimulus and behavioral variables.

2.3 Interpreting latent variables: Global fluctuations in population activity

One of the most fruitful demonstrations of dimensionality reduction for neuroscience is the identification of global fluctuations in population activity. One of the earliest studies observed that a global fluctuation, seemingly independent of the stimulus drive, also drove the recorded neurons up and down together (Arieli et al., 1996). Further studies have used dimensionality reduction to quantify such global fluctuations (Arandia-Romero et al., 2016; Kohn et al., 2016; Lin et al., 2015), and have linked them to internal brain signals, such as attention (Ecker et al., 2016; Rabinowitz et al., 2015), as well as related them to anatomical connectivity (Okun et al., 2015). Changes in global fluctuations reflect changes in noise correlations (i.e., the correlations between neurons during repeats of the same stimulus) (Cohen and Kohn, 2011; Kohn et al., 2016; Okun et al., 2015), which have been a signature of cognitive processes, like attention and motivation (Cohen and Kohn, 2011; Cohen and Maunsell, 2009; Ecker et al., 2016; Ruff and Cohen, 2014).

The time scale of these global fluctuations varies widely across studies, from 200 ms (Arieli et al., 1996; Ecker et al., 2014; Okun et al., 2015; Rabinowitz et al., 2015) to minutes (Goris et al., 2014). This is likely due to different animal models, experimental tasks, amount and types of anesthesia (when used), and recording technology. Another likely factor is that global fluctuations arise from different neural mechanisms, including top-down feedback and common input that have varied time scales themselves. For these reasons, categorizing global fluctuations to their respective mechanisms and relating identified global fluctuations across studies has been difficult (McGinley et al., 2015b). This highlights the importance of having a unified approach in assessing the time scales and linking the global fluctuations to behavioral variables for the most accurate comparisons.

The presence of global fluctuations of population activity in sensory areas poses a natural question: Do these global fluctuations affect the fidelity of decoding stimulus information by a downstream readout? One's first reasoning may be yes, simply because adding any noise to a system will likely hurt the system's fidelity, especially if that noise is correlated across neurons and cannot be averaged out by pooling across neurons (Shadlen and Newsome, 1998). However, careful theoretical studies have laid the groundwork in understanding when global fluctuations can be harmful to downstream readout (Abbott and Dayan, 1999; Ecker et al., 2011; Moreno-Bote et al., 2014). In particular, a type of noise correlation called differential correlations is the only

correlation that can hurt downstream decoding (Moreno-Bote et al., 2014). Through simulations, these studies predict that differential correlations are likely small, and that the vast majority of noise correlations are harmless, unable to corrupt any stimulus information that a sensory population may be encoding. Few experimental studies have been able to confirm this prediction, as the number of neurons and number of trials required to test this hypothesis is large (Moreno-Bote et al., 2014). However, some experimental studies have suggested that changes in a particular global fluctuation from 'up' to 'down' states (or 'synchronized' and 'desynchronized' states) tend to increase information transfer (Beaman et al., 2017; McGinley et al., 2015a; Pachitariu et al., 2015; Schölvinck et al., 2015). Other studies have questioned whether downstream readouts are optimal decoders for two specific stimuli (e.g., two orientation gratings whose angles differ by a few degrees) or instead are general decoders for many types of stimuli (e.g., orientation gratings with a range of angles) (Ni et al., 2018). A key signature of these general decoders is that they should be more related to behavior than optimal decoders.

Alongside attention and adaptation, another key internal signal of interest is arousal or alertness, whose effects on neural activity are unclear (McGinley et al., 2015b). Previous studies suggest that arousal can change noise correlations (Ruff and Cohen, 2014; Vinck et al., 2015), and have increased signal-to-noise ratios for single neurons (McGinley et al., 2015a; Vinck et al., 2015). These arousal effects have also been linked to changes in pupil diameter (McGinley et al., 2015b; Reimer et al., 2014, 2016; Vinck et al., 2015) and locomotion (McGinley et al., 2015a; Reimer et al., 2014). Studies have also found a link between pupil diameter and the activity of neurons in the locus coeruleus, a small nucleus in the brain stem (Joshi et al., 2016), suggesting that the locus coeruleus plays a role in arousal (Aston-Jones and Cohen, 2005). Given the observed changes in neural activity and a mechanistic link to behavior, using dimensionality reduction to understand how arousal plays a role in the global fluctuation of populations of neurons seems promising, but has not been pursued. In Chapter 6, we use dimensionality reduction to identify a slowly-drifting latent variable present in macaque monkey V4 population activity. We then link this slow drift to behavior, and provide implications on how the slow drift affects the fidelity of downstream readout.

Chapter 3

Relationship between the dimensionality of population activity in V1 and the dimensionality of the visual stimulus

Dimensionality reduction has been applied to neural population activity to study decision making (Harvey et al., 2012; Mante et al., 2013), motor control (Churchland et al., 2012; Sadtler et al., 2014), olfaction (Mazor and Laurent, 2005), working memory (Daie et al., 2015; Rigotti et al., 2013), audition (Luczak et al., 2009), rule learning (Durstewitz et al., 2010), speech (Bouchard et al., 2013), temporal encoding (Richmond and Optican, 1990), and more (for a review, see Cunningham and Yu (2014)). In many cases, dimensionality reduction is applied in brain areas for which the relationship between neural activity and external variables, such as the sensory stimulus or behavior, is not well characterized. This is indeed the setting in which dimensionality reduction may be most beneficial because it allows one to relate the activity of a neuron to the activity of other recorded neurons, without needing to assume a moment-by-moment relationship with external variables. However, it is also the setting in which the outputs of dimensionality reduction can be the most difficult to interpret.

To aid in interpreting the outputs of dimensionality reduction in such settings, it is important to vary the inputs to a brain area and ask whether the outputs of dimensionality reduction change in a sensible way. This is most readily done for a brain area close to the sensory periphery, such as the primary visual cortex (V1). Here, we apply dimensionality reduction to V1 and ask two fundamental, population-level questions. First, how is neural dimensionality related to stimulus or task dimensionality? Previous studies have used dimensionality reduction to analyze population activity in a reduced space (e.g., (Churchland et al., 2012; Mazor and Laurent, 2005; Sadtler et al., 2014)). Implicit in these studies is the appropriate dimensionality of the reduced space. It is currently unknown how neural dimensionality scales with stimulus or task dimensionality for a given population of neurons (Gao et al., 2017). Second, how does a neural circuit flexibly encode (or "multiplex") the representation of the vast number of stimuli encountered in the natural world? Recent studies suggest that it may be possible to take advantage of the multi-dimensional properties of the population activity space (DiCarlo et al., 2012; Kaufman et al., 2014; Lehky et al., 2014; Mante et al., 2013; Rigotti et al., 2013). In particular, the population activity representing different stimuli might occupy similar dimensions of the population activity space (Luczak et al.,

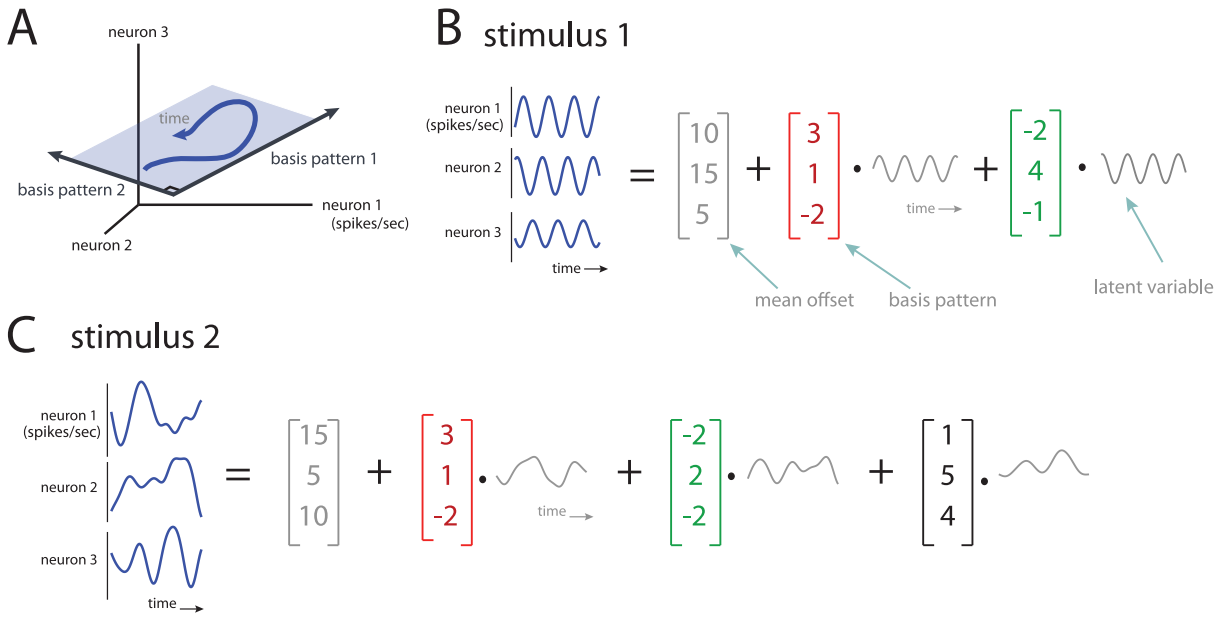


Figure 3.1: Conceptual illustration of dimensionality and basis patterns. *A*: The activity of three neurons can be plotted in a 3-d population firing rate space, where each axis represents the firing rate of one neuron. The population activity evolves over time (blue trace), and occupies a 2-d plane (blue shade). The basis patterns are orthogonal axes that define this 2-d plane. *B*: The activity of three neurons can also be represented as time-varying firing rates or peri-stimulus time histograms, PSTHs. The activity can be decomposed into a weighted sum of basis patterns (red and green) and a mean offset (gray). Each basis pattern is weighted by a time-varying latent variable. Note that basis patterns are mutually orthogonal, by definition. *C*: The activity of the same three neurons as in *B*, but for a different stimulus. Same conventions as in *B*.

2009; Sadtler et al., 2014). It is currently unknown how the similarity of the dimensions being occupied by the population activity changes with the similarity of the stimuli.

The concept of dimensionality is illustrated in Fig. 3.1. Consider a high-dimensional space (termed the *population firing rate space*) in which each axis represents the firing rate of a recorded neuron (Fig. 3.1*A*). The goal of dimensionality reduction is to identify i) how many dimensions are occupied by the neural population activity, i.e., the *dimensionality* of the population activity, and ii) how these dimensions are oriented within the population firing rate space. In this three-neuron example, the population activity is two-dimensional, where the dimensions are defined by the orthogonal basis patterns (Fig. 3.1*A*, basis patterns 1 and 2). Equivalently, we can think of dimensionality reduction in terms of decomposing the population activity into a weighted sum of basis patterns and a mean offset (Fig. 3.1*B*). A basis pattern describes a characteristic way in which activity of the neurons covaries. Each basis pattern is fixed and is weighted by a time-varying latent variable, which represents the contribution of the basis pattern at each point in time.

We can compare the outputs of dimensionality reduction for two different stimuli (Fig. 3.1*B* and 3.1*C*). The population activity for stimulus 1 is two-dimensional because it can be described

by two basis patterns (Fig. 3.1*B*), whereas that for stimulus 2 is three-dimensional (Fig. 3.1*C*). Thus, the population response to stimulus 2 would be deemed to have a larger dimensionality than the population response to stimulus 1. In addition, we can ask whether the population responses to different stimuli occupy similar dimensions within the population firing rate space. This is assessed by comparing the basis patterns across stimuli. In this example, there is one basis pattern that is shared by both stimuli (red), one basis pattern that is similar between stimuli (green), and one basis pattern (black) that is employed only by stimulus 2.

In this chapter, we characterize how neural dimensionality varies with stimulus dimensionality in macaque V1 by applying principal components analysis (PCA) to the trial-averaged neural responses to different classes of visual stimuli, including sinusoidal gratings, a natural movie, and white noise. In addition, we develop a new method (termed the pattern aggregation method) to measure how the basis patterns extracted from the population responses to each stimulus relate to each other. This method allows one to characterize how the similarity of the dimensions being occupied by the population activity changes with the similarity of the stimuli. A key advantage of studying these questions in V1 is that there are well-established receptive field (RF) models. By applying the same dimensionality reduction methods to the outputs of an RF model, we can more deeply understand the relationship between the outputs of dimensionality reduction and known properties of V1 neurons. The results described in this chapter show that the outputs of dimensionality reduction, when applied to V1 population activity, are sensible, and thus may be fruitfully applied to other brain areas.

3.1 Results

3.1.1 Dimensionality of population responses to gratings

We investigated how changing the dimensionality of the visual stimulus changes the dimensionality of trial-averaged population responses evoked by drifting sinusoidal gratings. To change the stimulus dimensionality, we included different numbers of consecutive grating orientations in the analysis (ranging from 1 to 12 orientations). For example, the stimulus with the smallest dimensionality included a single orientation, and stimuli with larger dimensionalities included two or five consecutive orientations (Fig. 3.2*A*).

We asked how quickly the dimensionality of the population activity grows as the number of orientations increases. At one extreme, it may be that the population response to each orientation uses an entirely different set of basis patterns (i.e., dimensions). In this case, the dimensionality for two orientations would be two times the dimensionality for one orientation. At the other extreme, it may be that the population response to each orientation resides in the same set of dimensions. In other words, the population response is formed using the same basis patterns, but linearly combined using different weights for different orientations. In this case, the dimensionality for two orientations would be the same as the dimensionality for one orientation.

We first computed the basis patterns of each orientation individually by applying PCA to the trial-averaged population response (taken in 20 ms bins), and identifying the patterns explaining the greatest variance in the population response (up to a chosen cumulative variance threshold, e.g., 90%). To assess the dimensionality of the population response to multiple orientations,

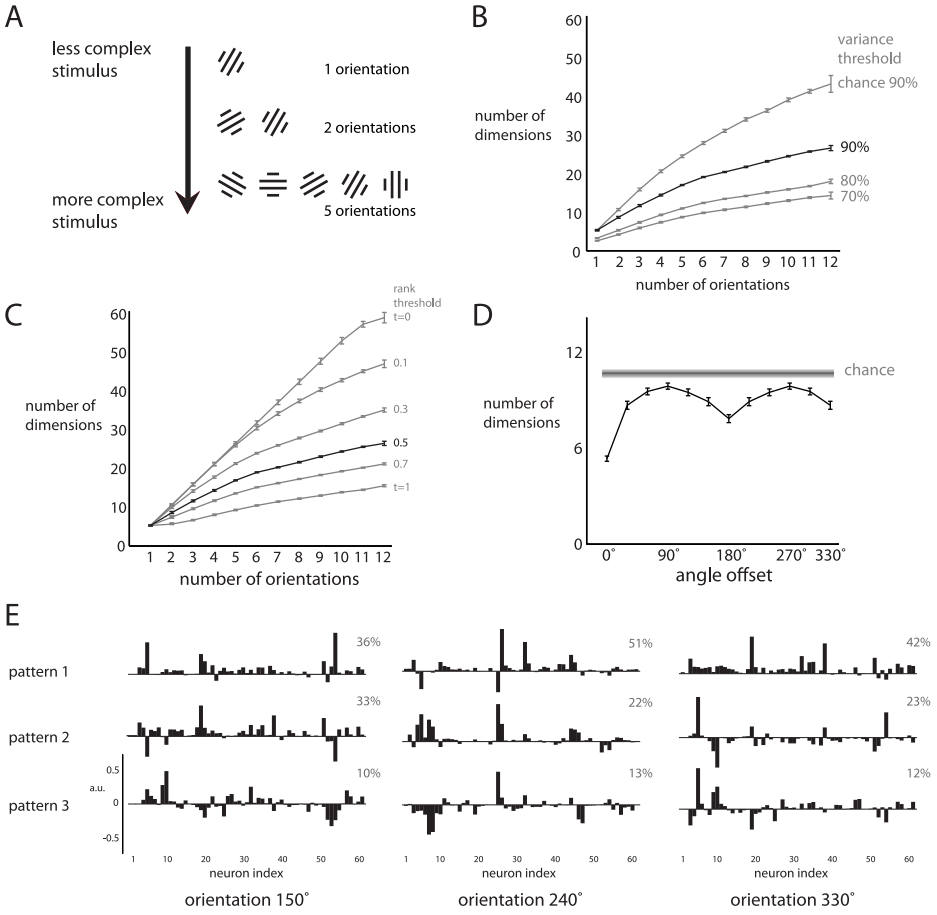


Figure 3.2: Dimensionality of population responses to individual gratings. *A*: The dimensionality of the stimulus was varied by combining a different number of consecutive orientations. The stimulus with the smallest dimensionality consisted of a single orientation, and stimuli with larger dimensionalities included two or five orientations. *B*: Dimensionality of population activity versus number of orientations. Bottom three curves correspond to the number of dimensions needed to explain 70%, 80%, and 90% of the variance. Top gray curve corresponds to the number of dimensions expected by chance for the 90% variance threshold. Error bars represent the standard error across monkeys and all possible combinations of consecutive orientations. *C*: Varying the rank threshold for a fixed variance threshold (90%). Same data as in *B*. Each curve represents the dimensionality of the population response as the number of consecutive orientations varies, for a particular rank threshold t. Error bars are computed as in *B*. *D*: Dimensionality of population activity versus angle offset between two orientations (bottom black curve). Chance dimensionality (top gray curve) and error bars are computed as in *B*. *E*: Basis patterns describing the largest percentage of variance for the population responses to three example orientations (90° apart) for one monkey. Each pattern is a unit vector with a norm of 1. Percentages denote the percent variance explained by each pattern.

we developed the pattern aggregation method (see Methods), which first aggregates the basis patterns for different orientations as column vectors in a matrix, and then computes the number of linearly independent columns of the aggregated matrix (i.e., the effective rank). This value is the dimensionality for multiple orientations. Using a 90% variance threshold, we found that the dimensionality for two orientations was 1.62 times the dimensionality for one orientation (Fig. 3.2*B*, '90%' curve), and significantly smaller than what would be expected had the basis patterns been randomly chosen (Fig. 3.2*B*, 'chance 90%' curve, $p < 10^{-5}$). In other words, for consecutive orientations separated by 30°, the population responses share about half of their basis patterns. As more orientations were included, the dimensionality of population responses remained lower than expected by chance (Fig. 3.2*B*), indicating that population responses to oriented gratings separated by angles larger than 30° also tend to use similar basis patterns. Similar trends were found using different variance thresholds (Fig. 3.2*B*, '70%', '80%' curves), so we use a 90% threshold in the rest of this work.

The pattern aggregation method requires a choice for the rank threshold t to determine how different basis patterns need to be before they define separate dimensions. We repeated the above analysis for different choices of t and a fixed variance threshold of 90% (Fig. 3.2*C*). We found that the dimensionality trends are similar for rank thresholds t near 0.5, so we use $t = 0.5$ in the rest of this work. Because the absolute dimensionality depends on the variance and rank thresholds, we make no claims about absolute dimensionality. Rather, we focus on relative comparisons of dimensionality for fixed variance and rank thresholds.

We observed a change in the rate of increase of dimensionality after six orientations (Fig. 3.2*B*, black curve). Because consecutive orientations were separated by 30°, the first and seventh orientations were 180° apart and differed only in their drift direction. Thus, the seventh to twelfth orientations were identical to the first to sixth orientations respectively, but drifted in opposite directions. A small proportion of V1 neurons are direction selective (Hawken et al., 1988; Movshon and Newsome, 1996), so the change in slope of the dimensionality curve might be due to the population activity using similar basis patterns for opposite drift directions.

To test this possibility, we performed two analyses. First, we assessed the direction selectivity of each neuron (direction index = 1 - null response / preferred response), and found that 16 of 183 neurons had a direction index greater than 0.5, consistent with (Movshon and Newsome, 1996). If none of the neurons encoded direction selectivity, we would expect the dimensionality curve to be flat beyond 6 orientations in Fig. 3.2*B*. The increase in dimensionality after 6 orientations is consistent with the finding that at least some neurons show direction selectivity. A potential concern is that the increase in dimensionality beyond 6 orientations arises from the fact that a larger number of patterns are being aggregated for a larger number of orientations. To address this, we performed a control analysis which equalized the number of patterns being aggregated across different numbers of conditions by including patterns from many subsamples of the data. We found that the dimensionalities for 7 or more orientations were significantly greater than the dimensionality for 6 orientations ($p < 0.001$), following the trend as shown in Fig. 3.2*B*.

Second, we assessed how the dimensionality of the population activity for two orientations varies with the angular offset between the orientations (Fig. 3.2*D*). This indicates how the similarity of the dimensions being occupied by the population activity changes with the similarity of the stimuli. We found that the dimensionality increases with angular offset up to 90°, indicating that the population activity differs the most for two orientations with 90° offset. Then, the

dimensionality decreases as the angular offset increases, reaching a minimum at a 180° offset, where gratings drift in opposite directions. Thus, the population activity uses more similar basis patterns for gratings drifting in opposite directions (180° offset) than to gratings of different orientations (angular offsets other than 180°). The dimensionality for 180° offset was higher than that for 0° offset ($p < 10^{-5}$), indicating that the population activity does not use identical basis patterns for opposite drift directions. Because these dimensionalities were computed differently (the pattern aggregation method for 180° offset and a variance threshold for 0° offset), we aggregated an equal number of patterns (identified over many subsampled runs) for 0° offset as that for 180° offset, and still found a higher dimensionality for 180° offset than for 0° offset ($p < 10^{-5}$).

This result, combined with the change in slope of the dimensionality curve (Fig. 3.2*B*), indicates that the population activity tends to use similar (but not identical) basis patterns for opposite drift directions. Taken together, the analyses in Fig. 3.2*B* and 3.2*D* characterize how the outputs of dimensionality reduction vary with the sensory input to V1.

We also visualized the basis patterns describing the largest percentage of variance for three different orientations (Fig. 3.2*E*). These basis patterns extracted from the trial-averaged population activity are akin to the hypothetical basis patterns shown in Figure 3.1 (red, green, black). For a given basis pattern, the absolute height of each bar indicates the degree to which each neuron contributes to that basis pattern. The following are two salient properties of the identified basis patterns. First, most of the neurons in the recorded population contribute to each basis pattern to some extent. Thus, the basis patterns capture changes in firing rates that are shared broadly across the population, rather than reflecting the activity of only a small number of neurons. Second, the basis patterns capture both positive and negative signal correlations between neurons, where the signal is the phase of the grating at different time points. A basis pattern describes positive signal correlation between a pair of neurons if the neurons have coefficients of the same sign. Conversely, a basis pattern describes negative signal correlation for coefficients of opposite sign. We can relate these basis patterns to the results shown in Fig. 3.2*B-D* by asking how similar are the linear combinations of each set of basis patterns across different stimulus orientations. This is difficult to assess by eye, and so we rely on the quantifications shown in Fig. 3.2*B-D* to determine how similar are the basis patterns across stimulus orientations.

3.1.2 Dimensionality of population responses to different classes of visual stimuli

We next sought to determine how the dimensionality of the trial-averaged population activity varies with the dimensionality of the visual stimulus for a wider range of stimuli. We presented three movie stimuli (Fig. 3.3*A*): a sequence of sinusoidal gratings (‘gratings movie’), contiguous natural scenes (‘natural movie’), and white noise (‘noise movie’). In contrast to Fig. 3.2*A* where the order of stimulus dimensionality is clear (i.e., a larger number of orientations has a larger dimensionality), here we needed first to assess the relative dimensionality of the three movie stimuli. By applying PCA to the pixel intensities, we found that 40 dimensions could explain nearly 100% of the variance of the pixel intensities for the gratings movie (Fig. 3.3*B*). For the natural movie, the top few dimensions explained a large percentage of the variance due to global

luminance changes caused by zooming and panning the camera, and a large number of additional dimensions were needed to explain the remaining variance. For the noise movie, each dimension explained only a small percentage of the total variance. We summarized these cumulative percent variance curves by finding the number of dimensions (gratings movie: 24, natural movie: 64, noise movie: 459) needed to explain 90% of the variance (Fig. 3.3*B*, dashed line). Based on these values, the gratings movie had the smallest dimensionality, followed by the natural movie, then the noise movie. Similar results were obtained when first transforming the pixel intensities using V1 receptive field models, then applying PCA (see "Comparing to V1 receptive field models"). We further asked how much the pixel intensities varied for each movie stimulus, and found that the noise movie had a smaller variance than the other two movie stimuli (Fig. 3.3*B*, inset). Together, this indicates that the distribution of pixel intensities for the gratings movie and natural movie is akin to an elongated ellipsoid (low dimensionality, high variance), whereas that for the noise movie is akin to a small ball (high dimensionality, low variance).

Having measured the relative dimensionality of the movie stimuli, we then asked how the dimensionality of the population responses to the movie stimuli varies with stimulus dimensionality. We found that for a 90% variance threshold, the dimensionality of the trial-averaged population responses (20 ms bins) was ordered in the same way as the stimulus dimensionality (Fig. 3.3*C*); namely, the population responses to the gratings movie had the lowest dimensionality, followed by the natural movie, then the noise movie. This ordering did not simply follow from the ordering of the mean population firing rates for the different movies (monkey 1: 4.2, 6.4, 5.4 spikes/sec, monkey 2: 6.6, 8.2, 6.7 spikes/sec for gratings, natural, and noise movies, respectively), and was consistent for a wide range of neuron counts for both monkeys (Supp. Fig. B.2). We also assessed how much the firing rates varied in response to each movie stimulus—that is, we measured how much the mean firing rate (averaged across experimental trials) varies over time. As with pixel intensities, we found that the population response to the noise movie had the smallest variance, followed by the gratings movie, then the natural movie (Fig. 3.3*C*, inset). Overall, the dimensionality and variance ordering in the visual stimuli and the population responses were similar, indicating that the population activity in V1 retains the dimensionality of the ordering of the visual stimuli themselves.

3.1.3 Basis patterns of population responses

Having compared the dimensionality of the population activity across stimuli, we next asked how the basis patterns of the population activity (corresponding to the dimensions being occupied by the population activity) compare across stimuli. Previous studies have found that the ability of a RF model to predict a neuron's response can depend on the stimulus class on which the model was trained (David et al., 2004; Smyth et al., 2003; Talebi and Baker, 2012), suggesting that the population activity might use somewhat distinct basis patterns for different stimulus classes. On the other hand, if basis patterns are influenced by the shared underlying network structure (Luczak et al., 2009; Sadtler et al., 2014), then we would expect them to be shared across responses to different stimuli.

We first asked whether there are qualitative differences in the coefficients of the basis patterns for population responses to the stimulus movies (Fig. 3.4*A*). As in Fig. 3.2*E*, we found that most of the basis patterns represented activity across a large number of neurons and described both

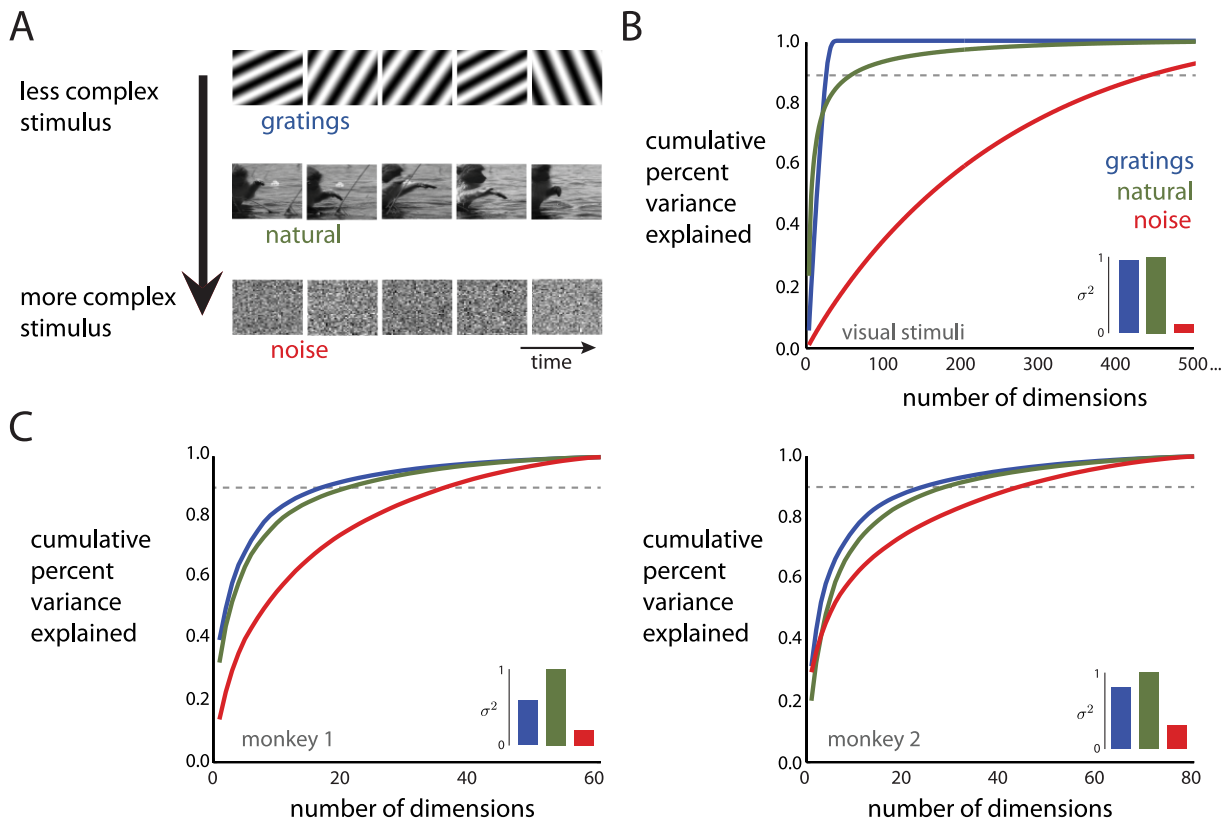


Figure 3.3: Dimensionality of movie stimuli and population responses to those movies. *A*: Example frames of the movies for the gratings movie, natural movie, and noise movie. *B*: For each stimulus, cumulative percent variance of the stimulus pixel intensities explained by different numbers of dimensions. Dashed line corresponds to 90% of the variance explained. Inset: summed variance of the pixel intensities, normalized by the maximum variance across movies. *C*: For each stimulus, cumulative percent variance of the population activity explained by different numbers of dimensions. Dashed lines correspond to 90% of variance explained. Left panel: monkey 1, 61 neurons. Right panel: monkey 2, 81 neurons. Insets: summed variance of each neuron's activity, normalized by the maximum variance across movies.

positive and negative signal correlations. However, there were two notable exceptions. First, the basis pattern describing the most variance for the population response to the gratings and noise movies involved primarily two neurons (Fig. 3.4*A*, right and left panels, pattern 1, neuron indices 26 and 27). For these movies, the two neurons had the highest firing rate modulation (maximum - minimum firing rate) across the recorded population. The weights for the gratings movie appeared to be sparser than those for individual gratings (Fig. 3.2*E*), likely because the gratings movie contained different orientations that strongly co-modulated these similarly-tuned neurons, whereas individual gratings with a single orientation co-modulated many neurons together with phase. Second, the basis pattern describing the most variance for the population response to the natural movie had coefficients of the same sign (Fig. 3.4*A*, middle panel, pattern 1). This is due to the entire population increasing or decreasing its firing rates together in response to global luminance changes prevalent in natural movies. Other than the similarity of the top basis pattern for the gratings and noise movies, it was difficult to determine by eye whether the basis patterns were being shared across stimuli. Thus, we used the pattern aggregation method to quantitatively assess the similarity of the identified basis patterns.

3.1.4 Assessing similarity of basis patterns

As a baseline, we assessed the extent to which the visual stimuli themselves reside in the same dimensions in pixel space. We compared the dimensionalities of the individual movies (Fig. 3.4*B*, teal dots, consistent with Fig. 3.3*B*) to those of combinations of movies (Fig. 3.4*B*, orange and purple dots). If the stimuli reside in overlapping dimensions, then the resulting dimensionality would be the maximum of the dimensionalities for the individual movies. However, if the stimuli reside in completely non-overlapping (i.e., orthogonal) dimensions, then the resulting dimensionality would be the sum of the dimensionalities for the individual movies. To measure the extent of overlap, we computed a similarity index s, for which $s > 0$ indicates that the patterns are more similar (i.e., more overlapping) than expected by chance and $s < 0$ indicates that the patterns are less similar (i.e., closer to orthogonal) than expected by chance (see Methods). There are two main observations. First, the dimensions occupied by the gratings movie overlap with those for the natural movie. To see this, note that the dimensionality for the gratings and natural movies together (80 dimensions) was less than the sum of the individual dimensionalities for the two movies (88 dimensions). To ensure that the overlap in patterns was meaningful, we confirmed that the aggregated dimensionality of 80 was less than the dimensionality (88 dimensions) that would be expected by combining randomly-chosen dimensions (Fig. 3.4*B*, gray, $s = 0.33, p < 10^{-5}$). Second, the dimensions occupied by the noise movie include many of the dimensions for the other two stimuli. This is indicated by the fact that the dimensionality corresponding to any combination of stimuli that included the noise movie (gratings + noise: 472, natural + noise: 495, and gratings + natural + noise: 500 dimensions) was less than the dimensionality that would be expected by chance ($s > 0.4, p < 10^{-5}$ for all cases). Note that, in all cases, the chance dimensionality was near the maximum dimensionality (indicating orthogonality between the two sets of randomly-chosen patterns) because the dimensionality of the pixel space (1,600 dimensions) was much larger than the dimensionalities of the individual movies.

We used the same approach to analyze the population responses as we did the visual stimuli. We compared the dimensionality of the population responses to individual movies (Fig. 3.4*C*,

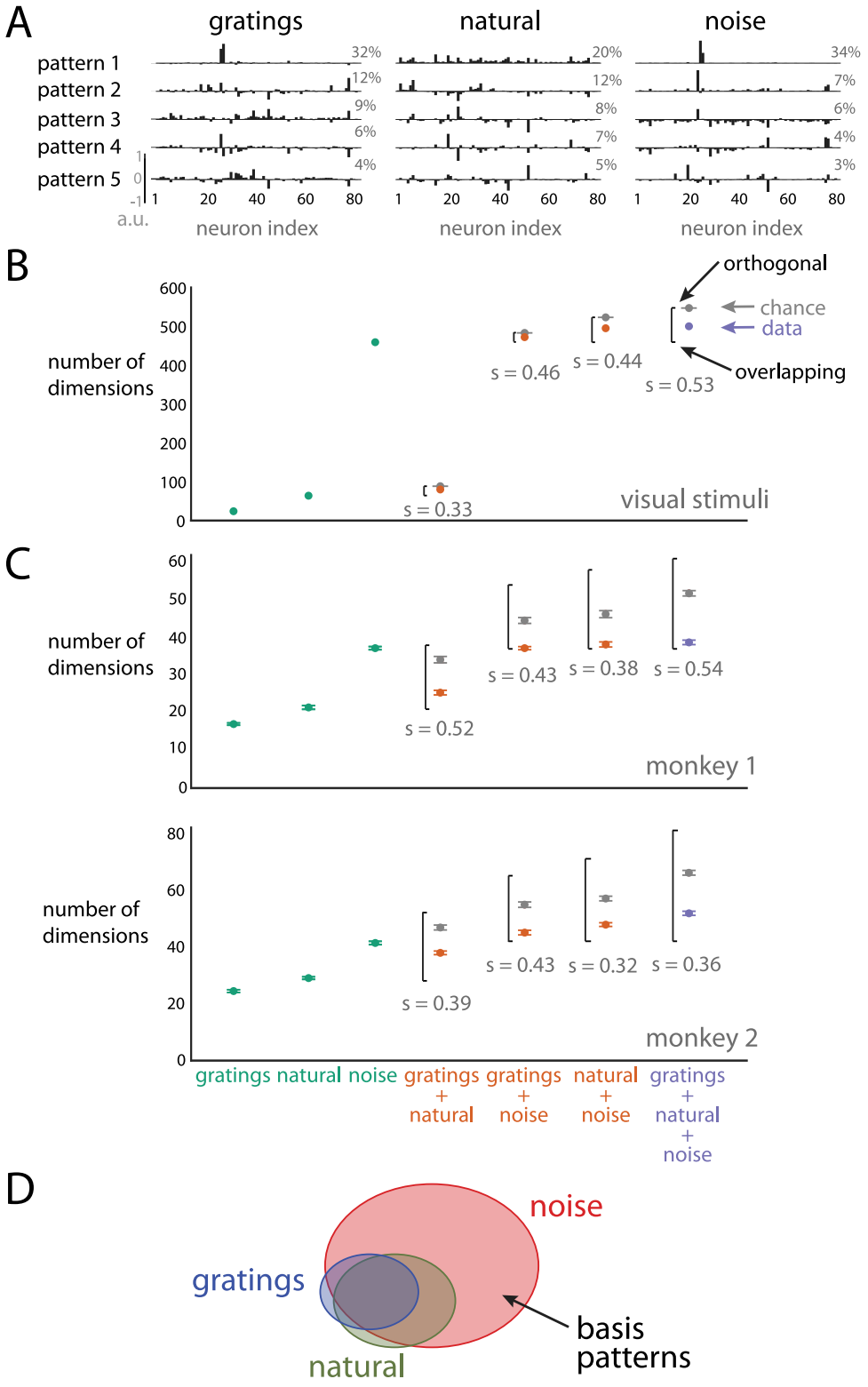


Figure 3.4: Caption on the following page.

Figure 3.4: Similarity of basis patterns across stimuli. *A*: Basis patterns describing the largest percentage of variance for the population responses to the gratings, natural, and noise movies for monkey 2. Each pattern is a unit vector with a norm of 1. Percentages denote percent variance explained by each pattern. *B*: Dimensionality of visual stimuli for individual movies (teal dots), and combinations of two (orange dots) or three (purple dots) movies. The teal dots correspond to where the curves intersect the dashed lines in Fig. 3.3*B*. Black brackets denote the range of possible dimensionalities (bottom of the black bracket corresponds to overlapping patterns; top of the black bracket corresponds to orthogonal patterns). Gray dots indicate dimensionalities expected by chance, and error bars represent the standard deviation of 100 random samples. The similarity index s indicates if patterns overlap more than expected by chance ($s > 0$) or are closer to orthogonal than expected by chance ($s < 0$). *C*: Dimensionality of population activity, for individual and combinations of movies. Same conventions as in *B*. Error bars represent standard deviations of the subsampled estimates. *D*: Venn diagram that summarizes the similarity of basis patterns across stimuli. The size of each ellipse indicates the number of basis patterns, and the overlap indicates the extent to which basis patterns are shared by different stimuli.

teal dots, consistent with Fig. 3.3*C*) to that of population responses to combinations of movies (Fig. 3.4*C*, orange and purple dots). We found that the relationship of the basis patterns employed by the population activity across stimuli (Fig. 3.4*C*) was similar to the relationship between the stimuli themselves (Fig. 3.4*B*). First, the dimensions occupied by the population responses to the gratings movie are overlapping with those for the natural movie. This is indicated by the fact that the aggregated dimensionality for the population responses to the gratings and natural movies (monkey 1: 25, monkey 2: 38) was less than the dimensionality if the patterns were orthogonal (top of the black brackets) and the dimensionality expected by chance ($s > 0.39, p < 10^{-5}$). Second, the dimensions occupied by the population response to the noise movie include most of the dimensions for the other two stimuli. This is indicated by the fact that the dimensionality corresponding to any combination of stimuli that included the noise movie was less than the dimensionality expected by chance ($s > 0.32, p < 10^{-5}$ in all cases), and close to entirely overlapping.

The chance dimensionality in Fig. 3.4*C* is computed by assuming that any population activity pattern within the N-dimensional population firing rate space can be achieved (where N=61 for monkey 1 and N=81 for monkey 2). Previous studies indicate that the population activity may only be able to occupy a subset of dimensions due to underlying network constraints (Luczak et al., 2009; Sadtler et al., 2014). Although we had no way of identifying exactly how many dimensions could have been occupied by the population of recorded neurons, we computed a lower bound by determining M, the largest value of dimensionality observed in response to any combination of stimuli (Fig. 3.4*C*, $M = 39$ for monkey 1 and $M = 52$ for monkey 2). If the chance level is computed instead by drawing random patterns from this M-dimensional space, we still find that the population activity tends to occupy more similar dimensions than expected by chance for all pairs of movies ($p < 0.05$). This is a conservative assessment because larger values of M would only make the comparison more statistically significant. We note that even if the population responses to different stimuli occupy similar dimensions, this does not imply that the responses occupy the same regions of the subspace defined by those dimensions. In other

words, the activity may covary along the same dimensions but be centered at different locations in the population activity space. We found that the population responses to the three movies indeed were centered in different locations of the population activity space (Supp. Fig. B.3).

Our analysis of the similarity of basis patterns for the population activity is summarized by the schematic in Fig. 3.4*D*, where the size of each ellipse represents the dimensionality (i.e., number of basis patterns) of the population response to the corresponding stimulus and the overlap between ellipses represents the extent to which the population responses share basis patterns. We found that the basis patterns for the gratings movie were overlapping with those of the natural movie, and that the basis patterns for the noise movie largely contain the basis patterns for the other two stimuli. However, there were a small number of basis patterns that were unique to each stimulus, shown by the small areas of non-overlap among the ellipses. Overall, this suggests that a neural circuit is capable of expressing a limited repertoire of basis patterns, and that a subset of those basis patterns is employed for any given stimulus.

3.1.5 Time-resolved dimensionality of population responses to movie stimuli

In the preceding sections, the analyses of the movie stimuli and the corresponding neural activity used the entire 30-second movie (i.e., 750 time points) together. Here, we consider time-resolved measurements of dimensionality using one second windows, each comprising 25 time points. This allows us to assess how dimensionality changes over time, and compare the basis patterns employed by the trial-averaged population activity during different parts of the 30-second movies.

For the visual stimuli (Fig. 3.5*A*, left panel) and the population responses (Fig. 3.5*A*, center and right panels), we found that the dimensionality corresponding to the noise movie was higher than the dimensionality corresponding to the gratings and natural movies in each one-second window, consistent with the results of analyzing each 30-second movie in its entirety (Fig. 3.3). However, the dimensionality of the natural movie was not greater than that of the gratings movie (cf. colored triangles in Fig. 3.5*A*, which indicate average dimensionality over time), in contrast to Fig. 3.3. We hypothesized that this was due to temporal correlations in the natural movie, in which frames within a one-second window tend to be self-similar. In contrast, for the gratings and noise movies, there are at least three different grating orientations and 25 different frames of white noise within each one-second window.

To reconcile the results for short and long time windows, we performed three analyses. First, we tested the hypothesis that temporal correlations in the natural movie result in lower dimensionalities (relative to the other movies) for short time windows. To break the temporal correlations, we shuffled the time points (20 ms resolution) across each 30 second period and performed the same analysis as in Fig. 3.5*A*. We found that, in a one-second window, the mean dimensionality corresponding to the natural movie was higher than that corresponding to the gratings movie for the shuffled data. This was true for the visual stimuli ($p < 0.01$) and for the population responses (monkey 1: $p < 10^{-3}$, monkey 2: $p < 0.05$). This indicates that the range of basis patterns expressed by the population responses to the natural movie within a short time window is limited by the temporal correlations in the natural movie itself.

Second, we asked how the dimensionality grows when increasing the window size progres-

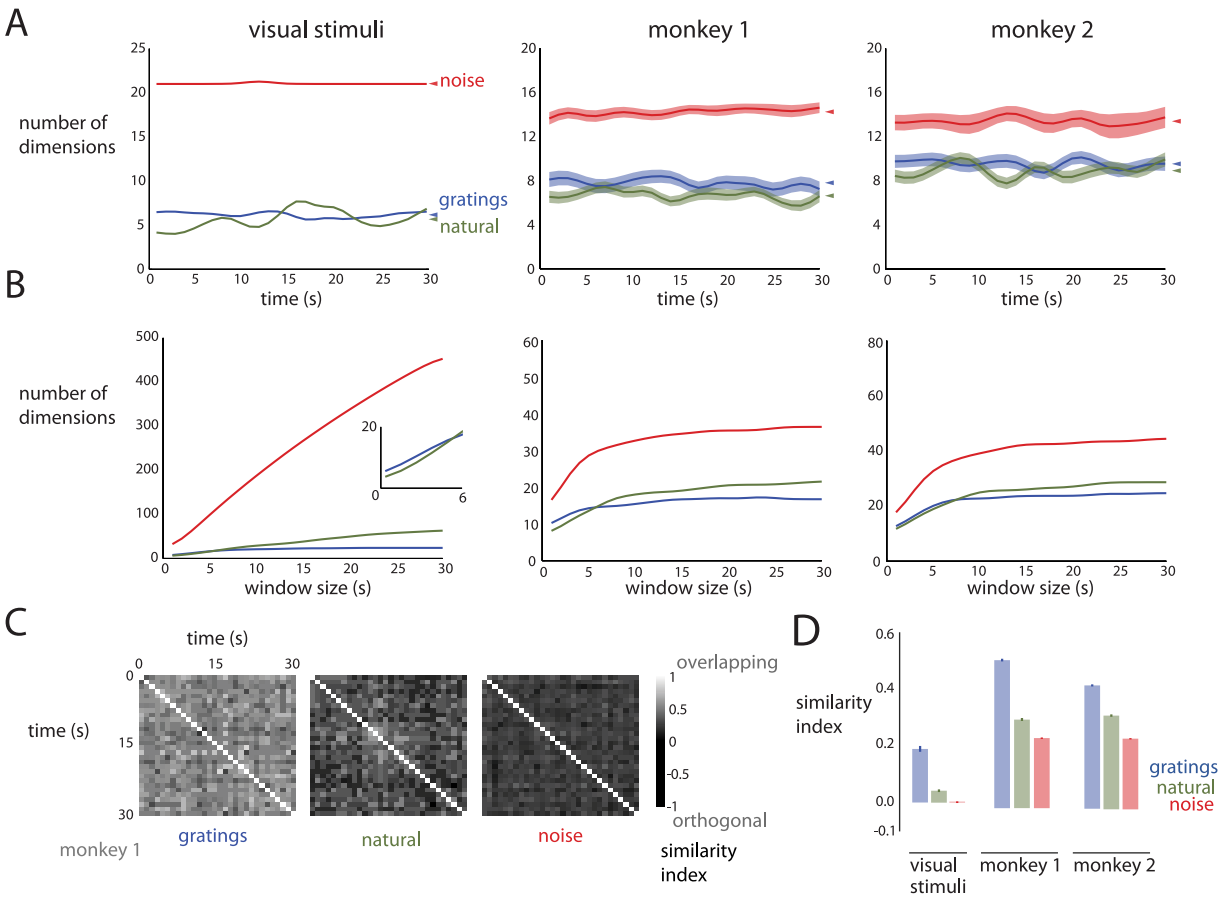


Figure 3.5: Time-resolved dimensionality of movie stimuli and population responses. *A*: Dimensionality versus time. Left panel: visual stimuli. Center panel: monkey 1 (61 neurons). Right panel: monkey 2 (81 neurons). Error bars are standard deviations of subsampled estimates. Triangles denote mean dimensionality across time for each movie. Curves were smoothed with a Gaussian kernel with a width of 1.5 s. *B*: Dimensionality with growing time windows starting at the beginning of each movie. Left panel: visual stimuli. Inset: zoomed portion of the bottom-left of the plot. Center panel: monkey 1. Right panel: monkey 2. Curves were smoothed as in *A*. Due to the smoothing, the dimensionalities for the 1 second window do not exactly match the leftmost point in *A*, and those for the 30 second window do not exactly match the dimensionalities in Fig. 3.3. *C*: Similarity of the basis patterns employed by the population responses across time (monkey 1). Each element in the similarity matrix corresponds to the similarity index between the sets of patterns for a pair of time points. *D*: Mean similarity index for visual stimuli and population responses. Error bars denote standard error over similarity indices.

sively from one second to 30 seconds, where each window starts at the beginning of the movie (Fig. 3.5*B*). If the dimensionality increases with window size, this would indicate that new patterns (of pixels or of population activity) are being used throughout the length of the movie. However, if the dimensionality plateaus, then the patterns are being reused and no new patterns are being expressed. The leftmost points on these curves (one-second window) correspond to the leftmost points on the corresponding curves in Fig. 3.5*A*. The rightmost points on these curves (30-second window) correspond to the dimensionalities shown in Fig. 3.3. Although the dimensionality corresponding to the natural movie (green) is lower than that corresponding to the gratings movie (blue) for short time windows, the dimensionality corresponding to the natural movie grows more quickly and surpasses that corresponding to the gratings movie for longer time windows. We found this to be true for both the visual stimuli (Fig. 3.5*B*, left panel) and the population responses (Fig. 3.5*B*, center and right panels). This indicates that the patterns for the natural movie tend to be self-similar within a short time window, and new patterns are expressed over longer time windows. In contrast, the dimensionality corresponding to the gratings movie does not grow as quickly with window size. This is because once a few different grating orientations are included, patterns corresponding to additional grating orientations can be represented approximately as linear combinations of patterns corresponding to existing grating orientations (both for the visual stimuli and the population responses), and therefore do not increase the dimensionality appreciably.

Third, we directly measured the similarity of the patterns between different one-second windows using the pattern aggregation method. Fig. 3.5*C* shows the matrix of similarity indices for the population responses to each of the three movies (monkey 1). By averaging the values of the off-diagonal similarity indices across the matrix, we found that the patterns corresponding to the gratings movie tended to be more similar across time than those for the natural and noise movies (Fig. 3.5*D*). This was true for the visual stimuli (left panel), as well as for the population responses (middle and right panels). This result indicates that new patterns tend to be expressed (for both the visual stimuli and the population responses) as the movies play out over time more for the natural and noise movies than for the gratings movie. This result also supports the finding in Fig. 3.5*B* that the dimensionality corresponding to the natural movie grows more quickly than that corresponding to the gratings movie.

Taken together, these results indicate that the noise movie drives a large number of basis patterns in the population activity in a short time window, relative to the gratings and natural movies. As the movies play out over time, the noise and natural movies tend to drive new basis patterns, whereas the grating movie tends to recruit the same patterns.

We also asked whether the second-by-second fluctuations of the dimensionality of the visual stimulus during the 30-second movie (Fig. 3.5*A*, left panel) were related with that of the population responses (Fig. 3.5*A*, middle and right panels) and found weak correlations (mean across movies $\rho = 0.20$). Although the dimensionality fluctuations were larger for the natural movie than the other two movies (Fig. 3.5*A*, left panel), these fluctuations were less salient in the population activity (Fig. 3.5*A*, middle and right panels). These two observations underscore that, although there are similarities in the statistical properties of the visual stimuli and population responses, there are also differences that remain to be understood.

3.1.6 Comparing to a V1 receptive field model

One of the key advantages of performing this study in V1 is that much is known about the stimulus-response relationship of individual neurons, as described by the many RF models that have been proposed in the literature (Carandini et al., 2005). Although the RF models do not capture every aspect of V1 neuronal activity (David et al., 2004; Olshausen and Field, 2005; Smyth et al., 2003; Talebi and Baker, 2012), we can apply the same dimensionality reduction methods to the activity generated by an RF model to help interpret the outputs of dimensionality reduction. We consider it a strength that, in many cases (described below), the outputs of dimensionality reduction applied to population activity show the same trends as when applied to activity from an RF model. This similarity indicates that single-neuron properties are reflected in population metrics, such as dimensionality. In cases where there are discrepancies between the outputs of dimensionality reduction for population activity and RF models, our results can provide guidance necessary to improve RF models.

Although a complete study of the many V1 RF models available is beyond the scope of this work, here we focus on one recently-proposed model (Goris et al., 2015). The model has four components: a Gabor filter, whose output is half-rectified; an untuned suppressive filter, whose output is also half-rectified; a normalization signal; and an exponentiating output nonlinearity. These components are shared with many other models of V1 (Carandini et al., 2005), and thus make it well-suited for studying how dimensionality changes as the input image is transformed by each component. The parameter values for 100 model neurons were drawn from parameter distributions reported in (Goris et al., 2015), since we did not present the stimuli appropriate for fitting the model parameters to data. For this reason, we only compare trends between the dimensionality of the outputs from the model and that of population activity.

We first assessed the dimensionality of each component's responses to the same individual gratings as presented to the monkeys. We observed similar trends between the population activity (Fig. 3.2*B* and Fig. 3.2*D*) and the activity of the RF model (Fig. 3.6*B* and 3.6*C*). As expected, the model captures key aspects of the population response to gratings, including direction selectivity. However, the last component of the model ("pointwise nonlinearity") showed substantially smaller dimensionalities than the other components (Fig. 3.6*B* and 3.6*C*, rightmost panels). This decrease in dimensionality is due to the exponential function exaggerating the anisotropy of the distribution of the model activity. For example, if the distribution of the responses across images resembles an ellipsoid in the 100-dimensional firing rate space, the exponential function would expand the variances of the major axes considerably more than the variances of the minor axes. The major axes would explain a greater proportion of the overall variance, resulting in fewer dimensions. Because we selected the values of the exponents independently from other model parameters, the discrepancy in dimensionality for the last component might not be present in the original model in (Goris et al., 2015), whose parameters were fit together. Still, our results indicate dimensionality can be sensitive to some nonlinear transformations.

Next, we assessed the dimensionality of each component's responses to the same movie stimuli as presented to the monkeys. The ordering of dimensionality for each component of the model (Fig. 3.6*D*) followed that of the population activity (Fig. 3.3*C*). As for the individual gratings, the pointwise nonlinearity substantially reduced the dimensionality of the model activity. A discrepancy between the model and the population activity was the ordering of variance: each

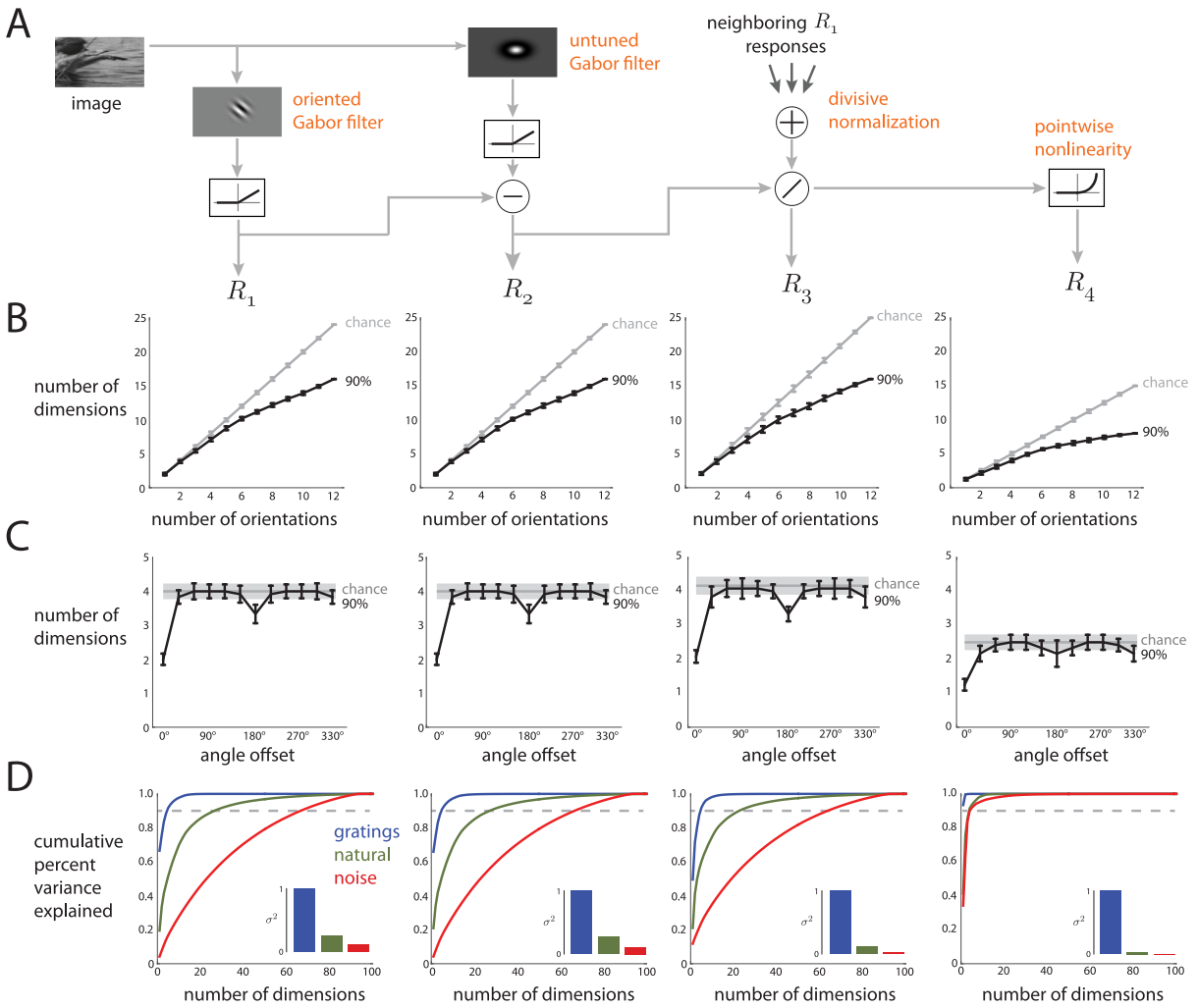


Figure 3.6: Dimensionality of model responses to individual gratings and movies. *A*: Block diagram of the RF model. We considered the activity in the model at four different components (R_1, R_2, R_3, R_4). *B*: Dimensionality of model activity versus number of orientations, computed in the same manner as in Fig. 3.2*B*. *C*: Dimensionality of model activity versus angle offset between two orientations, computed in the same manner as in Fig. 3.2*D*. *D*: Dimensionality and variance of model responses to movies, computed in the same manner as in Fig. 3.3*C*. Results are based on 100 model neurons.

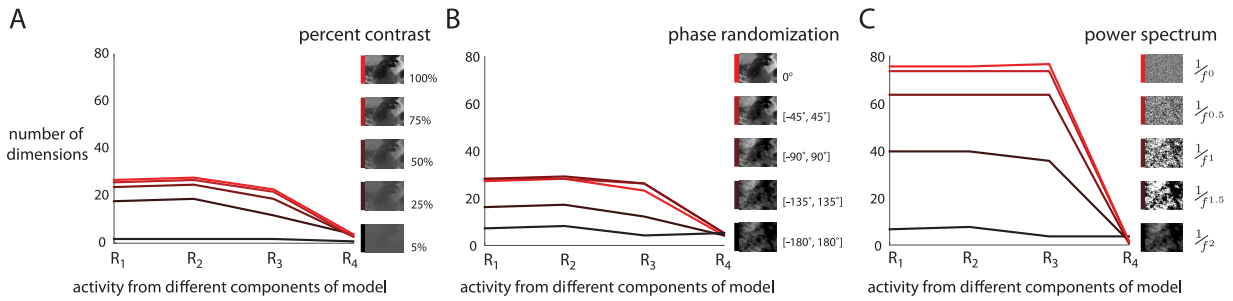


Figure 3.7: Dimensionality of model responses to parametrically-altered versions of the images in the natural movie. *A*: Images were generated by gradually decreasing contrast of the original images. *B*: Images were generated by adding a random offset to the phases of the original images, which transformed the natural images to pink noise. Each random offset was drawn from a uniform distribution over the specified range. *C*: Images were generated by raising the power spectrum of the same pink noise images generated in *B* to a fractional exponent. This gradually transformed the pink noise images to white noise.

component of the model exhibited greater variance for the gratings movie than for the natural movie (Fig. 3.6*D*, insets), but this was not the case for population activity (Fig. 3.3*C*, insets). This discrepancy stems from the oriented Gabor filters in the first component of the model. Because the temporal frequency of these filters was matched to that of the drifting sinuosoid gratings, the filters were modulated strongly by the gratings movie.

3.1.7 Dimensionality of model outputs to parametrically-varied stimuli

One advantage of working with an RF model is that we can assess the dimensionality of the responses to images that were not shown to the monkeys. With the caveat that the RF model might not capture aspects of the responses of real neurons, we assessed how the dimensionality of each component's outputs varies as we parametrically alter the visual stimuli.

We first considered varying the contrast of the natural movie images. Initially, one might think that contrast would have no effect on dimensionality because each dimension in pixel space is scaled equally. However, this is not the case for two reasons. First, before scaling, each image was re-centered by subtracting out the image's mean pixel intensity — not the mean pixel intensity across all images. Thus, changing the contrast is not a simple scaling across all images. The second reason is that there are nonlinearities in the model (e.g., the linear rectifying functions), which can "warp" the scatter of data points (where each point corresponds to an image), thereby changing the dimensionality. When we varied the contrast of the natural movie images, we found that dimensionality decreased with decreasing contrast (different colors in Fig. 3.7*A*). This decrease in dimensionality occurs because as contrast decreases, the mean luminance dominates each image. At a contrast of 0%, the images lie along a line in pixel space (i.e., the $[1, 1, \ldots, 1]$ direction).

Next, we randomized the phases of each natural image by adding a random offset to each phase value. The offsets were drawn from uniform distributions of varying extent. We considered small offsets drawn from the range $[-45°, 45°]$ to large offsets drawn from the range $[-180°, 180°]$.

When the phases were completely random (i.e., a range of $[-180°, 180°]$), the statistics of the images were akin to pink noise (Wichmann et al., 2006). We observed that as the extent of the phase randomization increased, the dimensionality decreased (Fig. 3.7*B*). Although randomizing the phases removed higher-order spatial correlations (e.g., edges and textures), it did not remove the spatial correlations brought about by the low frequencies of the power spectrum, which are strongly represented in natural images (Simoncelli and Olshausen, 2001). Only a small number of dimensions are needed to capture these low-frequency spatial correlations, because most of the pixels inside the region covered by the RFs of the model's filters have pixel intensities that co-vary strongly.

Finally, to remove these low-frequency spatial correlations, we gradually transformed each phase-randomized image (i.e., pink noise image) to a white noise image by raising the power spectrum of the pink noise images to a fractional exponent. As the power spectrum of the pink noise images became more similar to white noise, the dimensionality increased (Fig. 3.7*C*, black to red). This increase is not because the model neurons were more responsive to white noise images than pink noise images. Instead, the outputs of the model expressed more activity patterns for the white noise images due to the uncorrelated pixel intensities. This is consistent with the dimensionality trends of the population activity, where we observed that the population response to white noise had the highest dimensionality of the three movies but the lowest amount of variance (Fig. 3.3*C*).

Based on these results, the model predicts that lowering contrast and randomizing the phases of a set of natural images will decrease the dimensionality of the population response. We also observed that for all manipulations of the visual stimuli, the outputs of the last component of the model (pointwise nonlinearity) showed substantially lower dimensionality than outputs at other model components, consistent with the results in Fig. 3.6. This suggests that certain nonlinearities (e.g., pointwise exponentiation) affect dimensionality more than others (e.g., divisive normalization).

3.1.8 Dimensionality in different layers of a deep neural network

To build intuition about how the dimensionality of population activity might change at different stages of visual processing, we examined a deep convolutional neural network that was previously trained with over 1 million natural images and showed high accuracy on a well-known image recognition challenge (Szegedy et al., 2015). The deep network takes an image as input, processes the image through layers of filtering and nonlinear operations, including convolution, pooling, and normalization (Fig. 3.8*A*). The earlier layers capture low-level image statistics, such as oriented edges, while the deeper layers capture high-level image statistics, such as features that distinguish a car from a building (Szegedy et al., 2015). This hierarchical processing resembles the processing found in the visual system, and indeed, the progressive layers of deep networks appear to mimic the progressive processing stages of the ventral stream (Yamins and DiCarlo, 2016). We used this deep network to assess how dimensionality changes from one layer to the next, providing a prediction of how dimensionality might change in different visual areas along the ventral stream.

We input the same movie stimuli that we presented to the monkeys into the deep network, and examined the dimensionality of the filter outputs in different layers of the deep network (Fig. 3.8*B*). For each layer, we analyzed the 100 filters that had RFs closest to the center of the image. We found

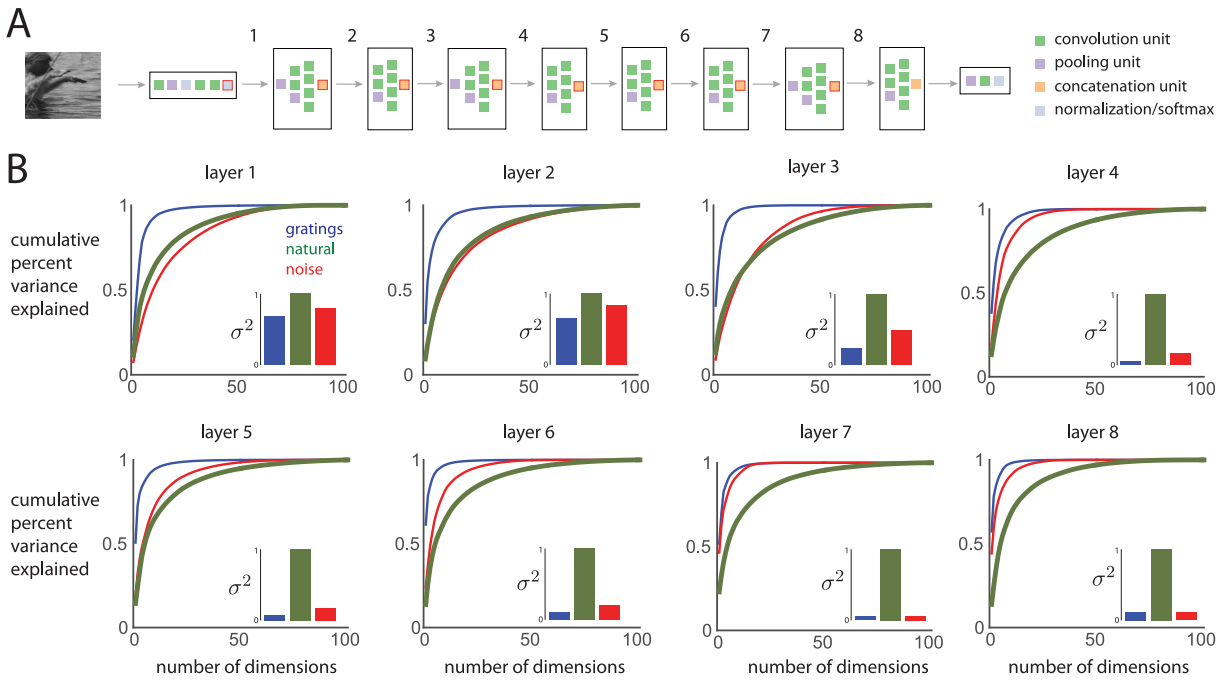


Figure 3.8: Dimensionality of activity in different layers of a deep convolutional neural network (CNN). *A*: The CNN comprised different layers (black outline boxes), where each layer comprised a group of units that performed specific operations, such as convolution, pooling, concatenation, and normalization. For each of the red-outlined units, we assessed the dimensionality of the activity of 100 filters. *B*: Dimensionality and variance of model responses in each layer to movies, computed in the same manner as Fig. 3.3*C*.

that the earliest layer (Fig. 3.8*B*, layer 1) showed dimensionality and response variance trends that matched the V1 population activity (Fig. 3.3*C*). In progressively deeper layers (Fig. 3.8*B*, going from layers 1 to 8), the responses to the gratings and noise movies decreased in dimensionality and relative variance, whereas the responses to the natural movie increased in dimensionality and relative variance. These findings are consistent with our understanding of the ventral stream: in progressive stages of visual processing, neurons become more sensitive to features in natural images and less sensitive to artificial images (Freeman et al., 2013). These results thus provide a prediction of how the dimensionality and variance of trial-averaged population responses to natural and artificial images should change along the ventral stream, which can be tested in future experiments (Lehky et al., 2014).

3.2 Discussion

To aid in understanding the outputs of dimensionality reduction, we chose to study a brain area close to the sensory periphery (V1). This allowed us to vary the sensory inputs and ask whether the outputs of dimensionality reduction change in a sensible way. By applying PCA to trial-averaged population responses to different classes of visual stimuli, including sinusoidal gratings, a natural movie, and white noise, we found that the dimensionality of the population responses grows with the stimulus dimensionality. In addition, we assessed whether the population responses to different stimuli occupy similar dimensions of the population firing rate space using a novel statistical method (the pattern aggregation method). We found that the population responses to stimuli as different as gratings and natural movies tended to occupy similar dimensions. For comparison, we applied the same analyses to the activity of a recently-proposed V1 receptive field model and a deep convolutional neural network, both of which showed trends similar as the real data. We further used these models to predict the dimensionality trends of the population responses to visual stimuli not shown to the monkeys, as well as the dimensionality trends of population responses in brain areas other than V1.

Many previous studies have compared visual cortical responses to natural and artificial stimuli on a single-neuron level (David et al., 2004; Felsen et al., 2005a; Smyth et al., 2003; Talebi and Baker, 2012; Touryan et al., 2005, e.g.,). Their predominant approach was to define a parameterized RF model to relate a neuron's activity to the visual stimulus. These studies found that, although RF models derived from natural stimuli share properties with those derived from artificial stimuli (Smyth et al., 2003; Touryan et al., 2005), there can be important differences (David et al., 2004; Smyth et al., 2003; Talebi and Baker, 2012). Here, rather than relating each neuron's activity to the stimulus, we relate the activity of the recorded neurons to each other. We can then ask how this relationship (i.e., the covariation of trial-averaged activity among neurons) changes for different classes of visual stimuli. This approach has been adopted for pairs of neurons (i.e., signal correlation) (Martin and Schröder, 2013), and we extend this work to characterize the signal correlation among all pairs of neurons at once. We found that the gratings, natural, and noise stimuli elicit many common basis patterns, consistent with previous studies showing similarities between RFs measured with gratings and natural stimuli (Smyth et al., 2003) and those measured with natural and noise stimuli (Touryan et al., 2005). Finally, our finding of some unique basis patterns for each stimulus class suggests that estimates of RFs will best capture the responses to the same type of stimulus used in estimating the RFs, as reported in previous studies (David et al., 2004; Talebi and Baker, 2012).

The quality of most RF models has been evaluated based on their ability to predict the activity of individual neurons (e.g., (Carandini et al., 2005; David et al., 2004; Goris et al., 2015; Smyth et al., 2003; Talebi and Baker, 2012; Touryan et al., 2005)). Given that there can be, in some cases, a substantial difference between the predictions of RF models and the recorded neural activity (David et al., 2004; Talebi and Baker, 2012; Touryan et al., 2005), especially for natural scenes, we need to quantify how they are different in an effort to improve the RF models. This is often quantified by computing the percent variance of the recorded activity explained by the model for each neuron individually (Carandini et al., 2005; David et al., 2004; Talebi and Baker, 2012). Our work provides a complementary way to compare RF models and recorded activity by examining the entire population together. We can compare the many V1 models that have been proposed by

assessing which ones best reproduce the relative dimensionalities across stimuli and the similarity of basis patterns observed in recorded activity. The model that we tested does reproduce the dimensionality trend of the population activity across stimuli, but does not reproduce the response variance trend (Fig. 3.6*D*). We speculate that a different spatiotemporal filter in the first component of the model can help to increase the response variance to natural images (David et al., 2004), thereby better matching the response variance trend of the model to that of the recorded activity.

Different basis patterns may be used by the population activity during different task epochs, suggesting that certain basis patterns drive downstream areas more effectively than others (Kaufman et al., 2014; Raposo et al., 2014). Thus, the identification of which basis patterns are used may be critical for understanding how different brain areas interact on a population level (Semedo et al., 2014). Furthermore, the activity patterns across a neural population have been used to study normalization (Busse et al., 2009), decision making (Machens et al., 2010; Mante et al., 2013), learning (Sadtler et al., 2014), and motor planning (Li et al., 2016). In the present work, we have developed a statistical framework (the pattern aggregation method) to measure the similarity of basis patterns across any number of stimulus conditions or time points. We validated this framework using recordings in V1, and the framework can be applied broadly to other brain areas.

The measurement of dimensionality depends on many factors, including the choice of dimensionality reduction method, the number of neurons (cf. Supp. Fig. B.2), and the number of data points. In principle, one should use a nonlinear dimensionality reduction method (e.g., (Camastra and Vinciarelli, 2002; Roweis and Saul, 2000; Tenenbaum et al., 2000)) because the underlying manifold of the population activity is likely to be nonlinear. For example, divisive normalization nonlinearly maps the tuning curves of a population of neurons onto a high-dimensional sphere (Ringach, 2010), and a nonlinear dimensionality reduction method may be able to extract the lower-dimensional embedding. However, most studies using dimensionality reduction in systems neuroscience have focused on linear methods (Bouchard et al., 2013; Churchland et al., 2012; Durstewitz et al., 2010; Harvey et al., 2012; Kaufman et al., 2014; Mante et al., 2013; Mazor and Laurent, 2005; Richmond and Optican, 1990; Sadtler et al., 2014). The reasons are that 1) most nonlinear methods rely on a dense sampling of the population activity space, in contrast to experimental data which tend to sparsely sample the space, and 2) it is usually difficult to assess the contribution of each neuron to a low-dimensional space identified by nonlinear methods. For the latter reason, we would not be able to compare how similar are the patterns for different stimuli, as we do in this study. Despite these caveats, we applied a nonlinear method, fractal dimensionality (Camastra and Vinciarelli, 2002; Lehky et al., 2014), to the three movies and their population responses (Supp. Fig. B.4). The ordering of fractal dimensionality across stimuli was consistent with that of PCA dimensionality (Fig. 3.3). Together with the results showing how the dimensionality ordering of the population activity depends on stimulus dimensionality (Figs. 3.2*B* and 3.3*C*) and neuron count (Supp. Fig. B.2), this finding indicates that a linear method can provide useful insights, even if the underlying manifold is indeed nonlinear.

There are several other factors that can affect the dimensionality of population responses. Dimensionality can depend on the properties of the particular neurons being sampled. In V1, these properties include the size and scatter of the receptive fields, as well as their preferred phases, orientations, and spatial frequencies. Another factor that can affect dimensionality in V1 is the size of the visual stimulus. We presented large visual stimuli that extended outside of the classical RF of most neurons. Previous studies have shown that stimulation outside of the classical RF

tends to increase the sparseness of V1 responses (Coen-Cagli et al., 2015; Pecka et al., 2014; Vinje and Gallant, 2000, 2002), which may affect the dimensionality of the population response. Sparseness leads to independence in the responses between neurons (Olshausen et al., 1996; Vinje and Gallant, 2000, 2002), and may lead to increased discriminability of the population activity (Froudarakis et al., 2014). Independence implies that each basis pattern only captures modulations of a single neuron (i.e., only one element of each pattern is non-zero), and the dimensionality (as assessed by PCA) depends on the relative variances captured by the basis patterns (in this case, the relative variances of the neurons). To our knowledge, there is no general relationship between sparseness and dimensionality. For all these reasons, it is not possible to make absolute statements about the dimensionality of V1. Instead, we made relative comparisons where all of the factors affecting dimensionality are fixed, except for the stimulus content.

The ideal characterization of population activity is a model with the fewest number of parameters that reproduces the data. The number of parameters of this model is called the *complexity* of the population activity (Gell-Mann and Lloyd, 1996). For example, trial-averaged responses of V1 neurons to sinusoidal gratings with the same spatial and temporal frequencies but all possible orientation angles would have a complexity of 1 parameter (i.e., orientation angle). Although dimensionality is often used as a proxy for complexity (Gao et al., 2017; Lehky et al., 2014), the relationship between dimensionality and complexity is difficult to define. For the trial-averaged responses to sinusoidal gratings with a complexity of 1 parameter, applying PCA to these responses would yield a dimensionality of 2. This is because the 1-d nonlinear manifold of the responses (i.e., a circle) resides in a 2-dimensional linear subspace in firing rate space. In this case, the dimensionality is an upper bound of complexity. However, for real data, dimensionality is not necessarily an upperbound for complexity. Dimensionality is typically defined as the number of top basis patterns that capture some fraction of the total variance, and assume the remaining basis patterns capture measurement noise. It may be the case that the assumed measurement noise is in fact also crucial to characterize the underlying computations of the neurons. Here, the dimensionality underestimates the complexity of population activity because the measured dimensionality is smaller than the necessary number of dimensions to characterize the computations. This work suggests that dimensionality can be used as an approximate measure of complexity of population activity and as a guide to build models whose number of parameters reach the true complexity of the data.

Although we believe that the results shown here are representative of a wide range of gratings, natural, or noise stimuli, they should be interpreted in the context of the particular visual stimuli used. For example, the dimensionality of the gratings movie and its population response could be increased by including more than one spatial and temporal frequency. Similarly, the dimensionality of the natural movie and its population response likely depends on the particular movie clip shown. If the scenes in the movie change more quickly (or slowly) over time, then we would expect the dimensionality over a 30-second time window to be larger (or smaller). For the noise movie, our results in Fig. 3.5*B* indicate that showing more instances of white noise is not likely to further increase the dimensionality of the population response. However, changing the statistics of the noise in the pixels could change the dimensionality of the population response.

At a population level, several studies have compared visual cortical activity evoked by natural and artificial stimuli to spontaneous activity (Berkes et al., 2011; Fiser et al., 2004; Okun et al., 2012). These studies focused on the raw neural activity, which includes both the trial-averaged

component (i.e., the PSTHs) and trial-to-trial variability. Here, we focused on the trial-averaged component. Because trial-to-trial variability can be substantial relative to the trial-averaged component (Arieli et al., 1996; Tolhurst et al., 1983), it is difficult to directly compare results of these previous studies to those reported here. The current study can be extended to study the population structure of trial-to-trial variability using a dimensionality reduction method such as factor analysis rather than PCA (Cunningham and Yu, 2014) in tandem with the pattern aggregation method.

For the stimuli that we tested and the recordings we made, we found that the dimensionality trends were consistent between the visual stimuli and the population responses. However, this need not be the case for other visual stimuli and other brain areas. In fact, a dimensionality trend that is inconsistent between the stimuli and responses may yield important insight into how the stimuli are encoded by the neurons under study. Part of resolving this potential discrepancy may relate to the way in which stimulus dimensionality is measured. PCA dimensionality captures only a particular aspect of the stimulus, namely the anisotropy of the distribution of pixel intensities in an Euclidian space. Other aspects of the stimuli may influence the neural responses more strongly, and alternate measures of stimulus dimensionality can be used, for example fractal dimensionality (Supp. Fig. B.4) (Camastra and Vinciarelli, 2002; Lehky et al., 2014) or a method based on image features extracted by a deep neural network (Fig. 3.8). Future studies employing additional stimuli and brain areas can elucidate whether dimensionality trends remain consistent between sensory stimuli and population responses.

Our work lays a solid foundation of to assess the dimensionality and similarily of basis patterns of neural population activity. Because V1 is a well-studied brain area and is close to the sensory input, our results can be compared with expectations based on our intuition and well-established RF models. Moving forward, these methods can be applied broadly to other brain areas and behavioral tasks to examine how the dimensionality of the population response changes due to conditions such as attentional state, learning, and contextual modulation.

Chapter 4

Adaptive stimulus selection for optimizing neural population responses

In the previous chapter, we applied dimensionality reduction to population activity from a brain area close to the sensory periphery and whose neural response properties are well-understood (primary visual cortex, V1). We now move our focus onto population activity from brain areas farther from the sensory periphery and whose neural response properties are less understood. In particular, we focus on the visual brain area V4. A key choice when recording from a population of V4 neurons is to determine which stimuli to present.

Often, it is unknown *a priori* which stimuli will drive a to-be-recorded neuron, especially in brain areas far from the sensory periphery like V4. Most studies either choose from a class of parameterized stimuli (e.g., sinusoidal gratings or pure tones) or present many randomized stimuli (e.g., white noise) to find the stimulus that maximizes the response of a neuron (i.e., the preferred stimulus) Ringach and Shapley (2004); Rust and Movshon (2005). However, the first approach limits the range of stimuli explored, and the second approach may not converge in a finite amount of recording time Schwartz et al. (2006). To efficiently find a preferred stimulus, studies have employed adaptive stimulus selection (also known as "adaptive sampling" or "optimal experimental design") to determine the next stimulus to show given the responses to previous stimuli in a closed-loop experiment Benda et al. (2007); DiMattina and Zhang (2014). Many adaptive methods have been developed to find the smallest number of stimuli needed to fit parameters of a model that predicts the recorded neuron's activity from the stimulus Lewi et al. (2009); Machens (2002); Machens et al. (2005); Paninski (2005); Park et al. (2014); Pillow and Park (2016). When no encoding model exists for a neuron (e.g., neurons in higher visual cortical areas), adaptive methods rely on maximizing the neuron's firing rate via genetic algorithms Carlson et al. (2011); Hung et al. (2012); Yamane et al. (2008) or gradient ascent Földiák (2001); O'Connor et al. (2005) to home in on the neuron's preferred stimulus. To our knowledge, all current adaptive stimulus selection methods focus solely on optimizing the firing rate of a *single* neuron.

Developments in neural recording technologies now enable the simultaneous recordings of tens to hundreds of neurons Stevenson and Kording (2011), each of which has its own preferred stimulus. For example, consider two neurons recorded in V4, a mid-level visual cortical area (Fig. 4.1*A*). Whereas neuron 1 responds most strongly to teddy bears, neuron 2 responds most

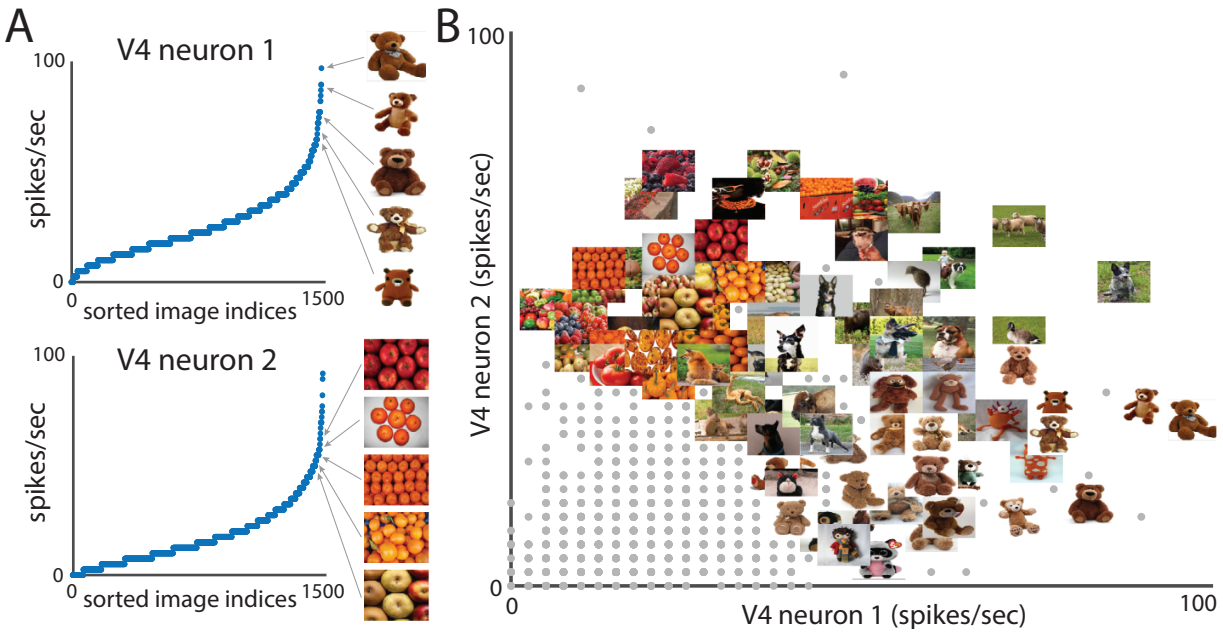


Figure 4.1: Responses of two macaque V4 neurons. *A*. Different neurons prefer different stimuli. Displayed images evoked 5 of top 25 largest responses. *B*. Images placed according to their responses. Gray dots represent responses to other images. Same neurons as in *A*.

strongly to arranged circular fruit. Both neurons moderately respond to images of animals (Fig. 4.1*B*). Given that different neurons have different preferred stimuli, how do we select which stimuli to present when simultaneously recording from multiple neurons? This necessitates defining objective functions for adaptive stimulus selection that are based on a population of neurons rather than any single neuron. Importantly, these objective functions can go beyond simply maximizing the firing rates of neurons and instead can be optimized for other attributes of the population response, such as maximizing the scatter of the responses in a multi-neuronal response space (Fig. 4.1*B*).

We propose Adept, an adaptive stimulus selection method that "adeptly" chooses the next stimulus to show based on a population objective function. Because the neural responses to candidate stimuli are unknown, Adept utilizes feature embeddings of the stimuli to predict to-be-recorded responses. In this work, we use the feature embeddings of a deep convolutional neural network (CNN) for prediction. We first confirmed with simulations that Adept, using a population objective function, elicited larger mean responses and a larger diversity of responses than optimizing the response of each neuron separately. Then, we ran Adept on V4 population activity recorded during a closed-loop electrophysiological experiment. Images chosen by Adept elicited higher mean firing rates and more diverse population responses compared to randomly-chosen images. This demonstrates that Adept is effective at finding stimuli to drive a population of neurons in brain areas far from the sensory periphery.

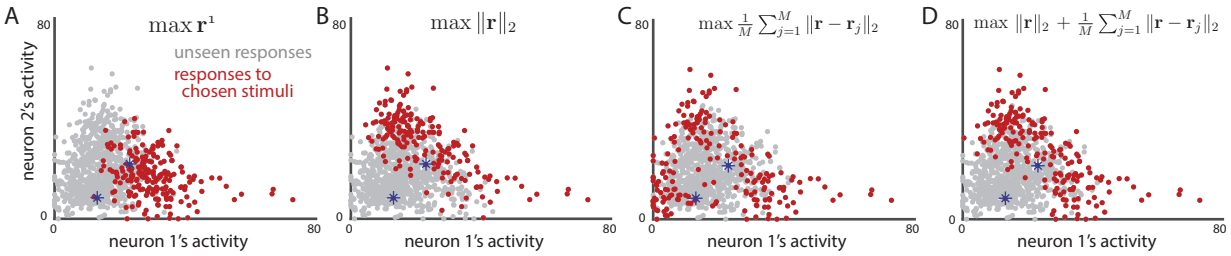


Figure 4.2: Different objective functions for adaptive stimulus selection yield different observed population responses (red dots). Blue * denote responses to stimuli used to initialize the adaptive method (the same for each panel).

4.1 Population objective functions

Depending on the desired outcomes of an experiment, one may favor one objective function over another. Here we discuss different objection functions for adaptive stimulus selection and the resulting responses $\mathbf{r} \in \mathbb{R}^p$, where the ith element $\mathbf{r}^i$ is the response of the ith neuron ($i = 1, \ldots, p$) and p is the number of neurons recorded simultaneously. To illustrate the effects of different objective functions, we ran an adaptive stimulus selection method on the activity of two simulated neurons (see details in Section 4.4.1). We first consider a single-neuron objective function employed by many adaptive methods Carlson et al. (2011); Hung et al. (2012); Yamane et al. (2008). Using this objective function $f(\mathbf{r}) = \mathbf{r}^i$, which maximizes the response for the ith neuron of the population, the adaptive method for $i = 1$ chose stimuli that maximized neuron 1's response (Fig. 4.2*A*, red dots). However, images that produced large responses for neuron 2 were not chosen (Fig. 4.2*A*, top left gray dots).

A natural population-level extension to this objective function is to maximize the responses of all neurons by defining the objective function to be $f(\mathbf{r}) = \|\mathbf{r}\|_2$. This objective function led to choosing stimuli that maximized responses for neurons 1 and 2 individually, as well as large responses for both neurons together (Fig. 4.2*B*). Another possible objective function is to maximize the scatter of the responses. In particular, we would like to choose the next stimulus such that the response vector $\mathbf{r}$ is far away from the previously-seen response vectors $\mathbf{r}_1, \ldots, \mathbf{r}_M$ after M chosen stimuli. One way to achieve this is to maximize the average Euclidean distance between $\mathbf{r}$ and $\mathbf{r}_1, \ldots, \mathbf{r}_M$, which leads to the objective function $f(\mathbf{r}, \mathbf{r}_1, \ldots, \mathbf{r}_M) = \frac{1}{M}\sum_{j=1}^{M}\|\mathbf{r} - \mathbf{r}_j\|_2$. This objective function led to a large scatter in responses for neurons 1 and 2 (Fig. 4.2*C*, red dots near and far from origin). This is because choosing stimuli that yield small and large responses produces the largest distances between responses.

Finally, we considered an objective function that favored large responses that are far away from one another. To achieve this, we summed the objectives in Fig. 4.2*B* and 4.2*C*. The objective function $f(\mathbf{r}, \mathbf{r}_1, \ldots, \mathbf{r}_M) = \|\mathbf{r}\|_2 + \frac{1}{M}\sum_{j=1}^{M}\|\mathbf{r} - \mathbf{r}_j\|_2$ was able to uncover large responses for both neurons (Fig. 4.2*D*, red dots far from origin). It also led to a larger scatter than maximizing the norm of $\mathbf{r}$ alone (e.g., compare red dots in bottom right of Fig. 4.2*B* and Fig. 4.2*D*). For these reasons, we use this objection function in the remainder of this work. However, the Adept framework is general and can be used with many different objective functions, including all presented in this section.

4.2 Using feature embeddings to predict norms and distances

We now formulate the optimization problem using the last objective function in Section 4.1. Consider a pool of N candidate stimuli $\mathbf{s}_1, \ldots, \mathbf{s}_N$. After showing $(t-1)$ stimuli, we are given previously-recorded response vectors $\mathbf{r}_{n_1}, \ldots, \mathbf{r}_{n_{t-1}} \in \mathbb{R}^p$, where $n_1, \ldots, n_{t-1} \in \{1, \ldots, N\}$. In other words, $\mathbf{r}_{n_j}$ is the vector of responses to the stimulus $\mathbf{s}_{n_j}$. At the tth iteration of adaptive stimulus selection, we choose the index n_t of the next stimulus to show by the following:

$$n_t = \underset{s \in \{1,\ldots,N\} \setminus \{n_1,\ldots,n_{t-1}\}}{\arg\max} \|\mathbf{r}_s\|_2 + \frac{1}{t-1} \sum_{j=1}^{t-1} \|\mathbf{r}_s - \mathbf{r}_{n_j}\|_2 \tag{4.1}$$

where $\mathbf{r}_s$ is the unseen population response vector to stimulus $\mathbf{s}_s$.

If the $\mathbf{r}_s$ were known, we could directly optimize Eqn. 4.1. However, in an online setting, we do not have access to the $\mathbf{r}_s$. Instead, we can directly predict the norm and average distance terms in Eqn. 4.1 by relating distances in neural response space to distances in a feature embedding space. The key idea is that if two stimuli have similar feature embeddings, then the corresponding neural responses will have similar norms and average distances. Concretely, consider feature embedding vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^q$ corresponding to candidate stimuli $\mathbf{s}_1, \ldots, \mathbf{s}_N$. For example, we can use the activity of q neurons from a CNN as a feature embedding vector for natural images Szegedy et al. (2015). To predict the norm of unseen response vector $\mathbf{r}_s \in \mathbb{R}^p$, we use kernel regression with the previously-recorded response vectors $\mathbf{r}_{n_1}, \ldots, \mathbf{r}_{n_{t-1}}$ as training data Watson (1964). To predict the distance between $\mathbf{r}_s$ and a previously-recorded response vector $\mathbf{r}_{n_j}$, we extend kernel regression to account for the paired nature of distances. Thus, the norm and average distance in Eqn. 4.1 for the unseen response vector $\mathbf{r}_s$ to the sth candidate stimulus are predicted by the following:

$$\widehat{\|\mathbf{r}_s\|_2} = \sum_k \frac{K(\mathbf{x}_s, \mathbf{x}_{n_k})}{\sum_\ell K(\mathbf{x}_s, \mathbf{x}_{n_\ell})} \|\mathbf{r}_{n_k}\|_2, \qquad \widehat{\|\mathbf{r}_s - \mathbf{r}_{n_j}\|_2} = \sum_k \frac{K(\mathbf{x}_s, \mathbf{x}_{n_k})}{\sum_\ell K(\mathbf{x}_s, \mathbf{x}_{n_\ell})} \|\mathbf{r}_{n_k} - \mathbf{r}_{n_j}\|_2 \tag{4.2}$$

where $k, \ell \in \{1, \ldots, t-1\}$. Here we use the radial basis function kernel $K(\mathbf{x}_j, \mathbf{x}_k) = \exp(-\|\mathbf{x}_j - \mathbf{x}_k\|_2^2 / h^2)$ with kernel bandwidth h, although other kernels can be used.

We tested the performance of this approach versus three other possible prediction approaches. The first two approaches use linear ridge regression and kernel regression, respectively, to predict $\mathbf{r}_s$. Their prediction $\hat{\mathbf{r}}_s$ is then used to evaluate the objective in place of $\mathbf{r}_s$. The third approach is a linear ridge regression version of Eqn. 4.2 to directly predict $\|\mathbf{r}_s\|_2$ and $\|\mathbf{r}_s - \mathbf{r}_{n_j}\|_2$. To compare the performance of these approaches, we developed a testbed in which we sampled two distinct populations of neurons from the same CNN, and asked how well one population can predict the responses of the other population using the different approaches described above. Formally, we let $\mathbf{x}_1, \ldots, \mathbf{x}_N$ be feature embedding vectors of $q = 500$ CNN neurons, and response vectors $\mathbf{r}_{n_1}, \ldots, \mathbf{r}_{n_{800}}$ be the responses of $p = 200$ different CNN neurons to 800 natural images. CNN neurons were from the same GoogLeNet CNN Szegedy et al. (2015) (see CNN details in Results). To compute performance, we took the Pearson's correlation ρ between the predicted and actual objective values on a held out set of responses not used for training. We also tracked the computation time τ (computed on an Intel Xeon 2.3GHz CPU with 36GB RAM) because these

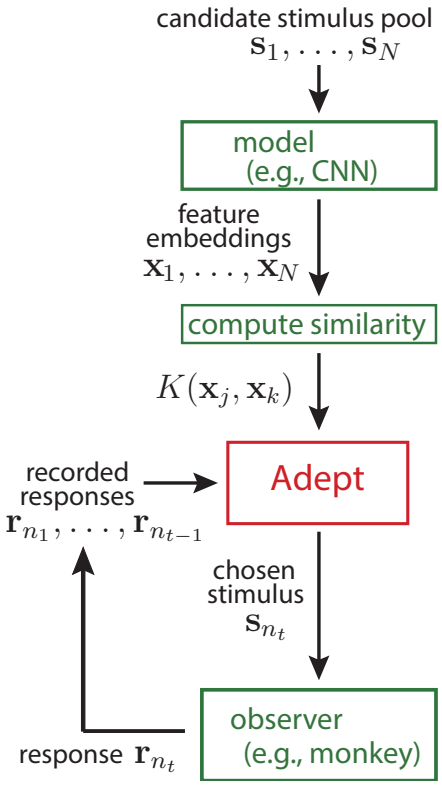
computations need to occur between stimulus presentations in an electrophysiological experiment. The approach in Eqn. 4.2 performed the best ($\rho = 0.64$) and was the fastest ($\tau = 0.2$ s) compared to the other prediction approaches ($\rho = 0.39, 0.41, 0.23$ and $\tau = 12.9$ s, 1.5 s, 48.4 s, for the three other approaches, respectively). The remarkably faster speed of Eqn. 4.2 over other approaches comes from the evaluation of the objective function (fast matrix operations), the fact that no training of linear regression weight vectors is needed, and the fact that distances are directly predicted (unlike the approaches that first predict $\hat{\mathbf{r}}_s$ and then must re-compute distances between $\hat{\mathbf{r}}_s$ and $\mathbf{r}_{n_1}, \ldots, \mathbf{r}_{n_{t-1}}$ for each candidate stimulus s). Due to its performance and fast computation time, we use the prediction approach in Eqn. 4.2 for the remainder of this work.

4.3 Adept algorithm

We now combine the optimization problem in Eqn. 4.1 and prediction approach in Eqn. 4.2 to formulate the Adept algorithm. We first discuss the adaptive stimulus selection paradigm (Fig. 7.8, left) and then the Adept algorithm (Fig. 7.8, right).

For the adaptive stimulus selection paradigm (Fig. 7.8, left), the experimenter first selects a candidate stimulus pool $\mathbf{s}_1, \ldots, \mathbf{s}_N$ from which Adept chooses, where N is large. For a vision experiment, the candidate stimulus pool could comprise natural images, textures, or sinusoidal gratings. For an auditory experiment, the stimulus pool could comprise natural sounds or pure tones. Next, feature embedding vectors $\mathbf{x}_1, \ldots, \mathbf{x}_N \in \mathbb{R}^q$ are computed for each candidate stimulus, and the pre-computed $N \times N$ kernel matrix $K(\mathbf{x}_j, \mathbf{x}_k)$ (i.e., similarity matrix) is input into Adept. For visual neurons, the feature embeddings could come from a bank of Gabor-like filters with different orientations and spatial frequencies Simoncelli and Freeman (1995), or from a more expressive model, such as CNN neurons in a middle layer of a pre-trained CNN. Because Adept only takes as input the kernel matrix $K(\mathbf{x}_j, \mathbf{x}_k)$ and not the feature embeddings $\mathbf{x}_1, \ldots, \mathbf{x}_N$, one could alternatively use a similarity matrix computed from psychophysical data to define the similarity between stimuli if no model exists. The previously-recorded response vectors $\mathbf{r}_{n_1}, \ldots, \mathbf{r}_{n_{t-1}}$ are also input into Adept, which then outputs the next chosen stimulus $\mathbf{s}_{n_t}$ to show. While the observer views $\mathbf{s}_{n_t}$, the response vector $\mathbf{r}_{n_t}$ is recorded and appended to the previously-recorded response vectors. This procedure is iteratively repeated until the end of the recording session. To show as many stimuli as possible, Adept does not choose the same stimulus more than once.

For the Adept algorithm (Fig. 7.8, right), we initialize by randomly choosing a small number of stimuli (e.g., $N_{\text{init}} = 5$) from the large pool of N candidate stimuli and presenting them to the observer. Using the responses to these stimuli $\mathbf{R}(:, 1{:}N_{\text{init}})$, Adept then adaptively chooses a new stimulus by finding the candidate stimulus that yields the largest objective (in this case, using the objective defined by Eqns. 4.1 and 4.2). This search is carried out by evaluating the objective for every candidate stimulus. There are three primary reasons why Adept is computationally fast enough to consider all candidate stimuli. First, the kernel matrix $K_{\mathbf{X}}$ is pre-computed, which is then easily indexed. Second, the prediction of the norm and average distance is computed with fast matrix operations. Third, Adept updates the distance matrix $D_{\mathbf{R}}$, which contains the pairwise distances between recorded response vectors, instead of re-computing $D_{\mathbf{R}}$ at each iteration.



Algorithm 1: Adept algorithm

Input: N candidate stimuli, feature embeddings $\mathbf{X}(q \times N)$, kernel bandwidth h (hyperparameter)

Initialization:

$K_{\mathbf{X}}(j,k) = \exp(-\|\mathbf{X}(:,j) - \mathbf{X}(:,k)\|_2^2/h^2)$ for all j,k

$\mathbf{R}(:, 1{:}N_{\text{init}}) \leftarrow$ responses to N_{init} initial stimuli

$D_{\mathbf{R}}(j,k) = \|\mathbf{R}(:,j) - \mathbf{R}(:,k)\|_2$ for $j,k = 1,\ldots,N_{\text{init}}$

ind_obs $\leftarrow$ indices of N_{init} observed stimuli

Online algorithm:

for tth stimulus to show **do**

  for sth candidate stimulus **do**

    $k_{\mathbf{X}} = K_{\mathbf{X}}(\text{ind\_obs}, s) / \sum_{\ell \in \text{ind\_obs}} K_{\mathbf{X}}(\ell, s)$

    % predict norm from recorded responses

    norms$(s) \leftarrow \widehat{\|\mathbf{r}_s\|_2} = {k_{\mathbf{X}}}^T \text{diag}(\sqrt{\mathbf{R}^T\mathbf{R}})$

    % predict average distance from recorded responses

    avgdists$(s) \leftarrow \frac{1}{t-1} \sum_\ell \widehat{\|\mathbf{r}_s - \mathbf{r}_{n_\ell}\|_2} = \text{mean}({k_{\mathbf{X}}}^T D_{\mathbf{R}})$

  end

  ind_obs$(N_{\text{init}} + t) \leftarrow$ argmax(norms + avgdists)

  $\mathbf{R}(:, N_{\text{init}} + t) \leftarrow$ recorded responses to chosen stimulus

  update $D_{\mathbf{R}}$ with $\|\mathbf{R}(:, N_{\text{init}} + t) - \mathbf{R}(:,\ell)\|_2$ for all ℓ

end

Figure 4.3: Flowchart of the adaptive sampling paradigm (left) and the Adept algorithm (right).

4.4 Results

We tested Adept in two settings. First, we tested Adept on a surrogate for the brain—a pre-trained CNN. This allowed us to perform comparisons between methods with a noiseless system. Second, in a closed-loop electrophysiological experiment, we performed Adept on population activity recorded in macaque V4. In both settings, we used the same candidate image pool of $N \approx 10{,}000$ natural images from the McGill natural image dataset Olmos and Kingdom (2004) and Google image search goo. For the predictive feature embeddings in both settings, we used responses from a pre-trained CNN different from the CNN used as a surrogate for the brain in the first setting. The motivation to use CNNs was inspired by the recent successes of CNNs to predict neural activity in V4 Yamins and DiCarlo (2016).

4.4.1 Testing Adept on CNN neurons

The testbed for Adept involved two different CNNs. One CNN is the surrogate for the brain. For this CNN, we took responses of $p = 200$ neurons in a middle layer of the pre-trained ResNet CNN He et al. (2016) (layer 25 of 50, named 'res3dx'). A second CNN is used for feature embeddings to predict responses of the first CNN. For this CNN, we took responses of $q = 750$ neurons in a middle layer of the pre-trained GoogLeNet CNN Szegedy et al. (2015) (layer 5 of 10, named 'icp4_out'). Both CNNs were trained for image classification but had substantially different architectures. Pre-trained CNNs were downloaded from MatConvNet Vedaldi and Lenc (2015), with the PVT version of GoogLeNet Xiao (2013). We ran Adept for 2,000 out of the 10,000 candidate images (with $N_{\text{init}} = 5$ and kernel bandwidth $h = 200$—similar results were obtained for different h), and compared the CNN responses to those of 2,000 randomly-chosen images. We asked two questions pertaining to the two terms in the objective function in Eqn. 4.1. First, are responses larger for Adept than for randomly-chosen images? Second, to what extent does Adept produce larger scatter of responses than if we had chosen images at random? A larger scatter implies a greater diversity in evoked population responses (Fig. 4.1*B*).

To address the first question, we computed the mean response across all 2,000 images for each CNN neuron. The mean responses using Adept were on average 15.5% larger than the mean responses to randomly chosen images (Fig. 4.4*A*, difference in means was significantly greater than zero, $p < 10^{-4}$). For the second question, we assessed the amount of response scatter by computing the amount of variance captured by each dimension. We applied PCA separately to the responses to images chosen by Adept and those to images selected randomly. For each dimension, we computed the ratio between the Adept eigenvalue divided by the randomly-chosen-image eigenvalue. In this way, we compared the dimensions of greatest variance, followed by the dimensions of the second-most variance, and so on. Ratios above 1 indicate that Adept explored a dimension more than the corresponding ordered dimension of random selection. We found that Adept produced larger response scatter compared to randomly-chosen images for many dimensions (Fig. 4.4*B*). Ratios for dimensions of lesser variance (e.g., dimensions 10 to 75) are nearly as meaningful as those of the dimensions of greatest variance (i.e., dimensions 1 to 10), as the top 10 dimensions explained only 16.8% of the total variance (Fig. 4.4*B*, inset).

Next, we asked to what extent does optimizing a population objective function perform better than optimizing a single-neuron objective function. For the single-neuron case, we implemented

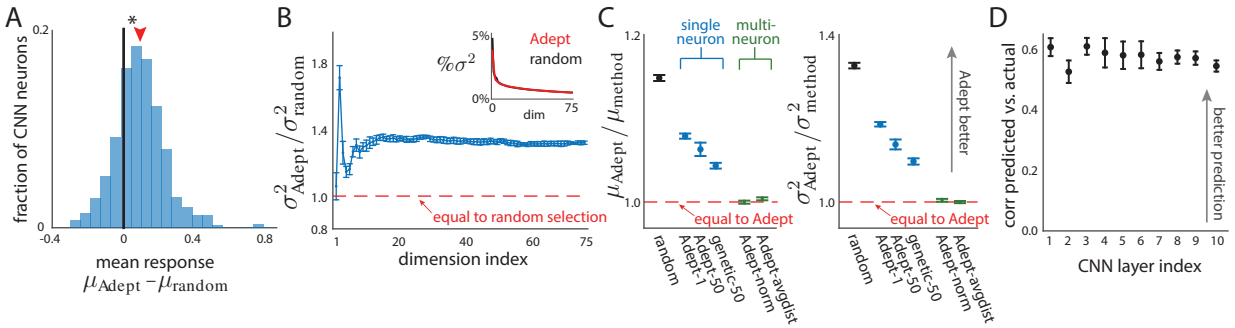


Figure 4.4: CNN testbed for Adept. *A*. Mean responses (arbitrary units) to images chosen by Adept were greater than to randomly-chosen images. *B*. Adept produced higher response variance for each PC dimension than when randomly choosing images. Inset: Percent variance explained. *C*. Relative to the full objective function in Eqn. 4.1, population objective functions (green) yielded higher response mean and variance than those of single-neuron objective functions (blue). *D*. Feature embeddings for all CNN layers were predictive. Error bars are $\pm$ s.d. across 10 runs.

three different methods. First, we ran Adept to optimize the response of a single CNN neuron with the largest mean response ('Adept-1'). Second, we applied Adept in a sequential manner to optimize the response of 50 randomly-chosen CNN neurons individually. After optimizing a CNN neuron for 40 images, optimization switched to the next CNN neuron ('Adept-50'). Third, we sequentially optimized 50 randomly-chosen CNN neurons individually using a genetic algorithm ('genetic-50'), similar to the ones proposed in previous studies Carlson et al. (2011); Hung et al. (2012); Yamane et al. (2008). We found that Adept produced higher mean responses than the three single-neuron methods (Fig. 4.4*C*, blue points in left panel), likely because Adept chose images that evoked large responses across neurons together. All methods produced higher mean responses than randomly choosing images (Fig. 4.4*C*, black point above blue points in left panel). Adept also produced higher mean eigenvalue ratios across the top 75 PCA dimensions than the three single-neuron methods (Fig. 4.4*C*, blue points in right panel). This indicates that Adept, using a population objective, is better able to optimize population responses than using a single-neuron objective to optimize the response of each neuron in the population.

We then modified the Adept objective function to include only the norm term ('Adept-norm', Fig. 4.2*B*) and only the average distance term ('Adept-avgdist', Fig. 4.2*C*). Both of these population methods performed better than single-neuron methods (Fig. 4.4*C*, green points below blue points). While their performance was comparable to Adept using the full objective function, upon closer inspection, we observed differences in performance that matched our intuition about the objective functions. The mean response ratio for Adept using the full objection function and Adept-norm was close to 1 (Fig. 4.4*C*, left panel, Adept-norm on red-dashed line, $p = 0.65$), but the eigenvalue ratio was greater than 1 (Fig. 4.4*C*, right panel, Adept-norm above red-dashed line, $p < 0.005$). Thus, Adept-norm maximizes mean responses at the expense of less scatter. On the other hand, Adept-avgdist produced a lower mean response than that of Adept using the full objective function (Fig. 4.4*C*, left panel, Adept-avgdist above red-dashed line, $p < 10^{-4}$), but an eigenvalue ratio of 1 (Fig. 4.4*C*, right panel, Adept-avgdist on red-dashed line, $p = 0.62$). Thus, Adept-avgdist increases the response scatter at the expense of a lower mean response.

The results in this section were based on middle layer neurons in the GoogLeNet CNN predicting middle layer neurons in the ResNet CNN. However, it is possible that CNN neurons in other layers may be better predictors than those in a middle layer. To test for this, we asked which layers of the GoogLeNet CNN were most predictive of the objective values of the middle layer of the ResNet CNN. For each layer of increasing depth, we computed the correlation between the predicted objective (using 750 CNN neurons from that layer) and the actual objective of the ResNet responses (200 CNN neurons) (Fig. 4.4*D*). We found that all layers were predictive ($\rho \approx 0.6$), although there was variation across layers. Middle layers were slightly more predictive than deeper layers, likely because deeper layers of GoogLeNet have a different embedding of natural images than the middle layer of the ResNet CNN.

4.4.2 Testing Adept on V4 population recordings

Next, we tested Adept in a closed-loop neurophysiological experiment. We implanted a 96-electrode array in macaque V4, whose neurons respond differently to a wide range of image features, including orientation, spatial frequency, color, shape, texture, and curvature, among others Roe et al. (2012). Currently, no existing parametric encoding model fully captures the stimulus-response relationship of V4 neurons. The current state-of-the-art model for predicting the activity of V4 neurons uses the output of middle layer neurons in a CNN previously trained without any information about the responses of V4 neurons Yamins and DiCarlo (2016). Thus, we used a pre-trained CNN (GoogLeNet) to obtain the predictive feature embeddings.

The experimental task flow proceeded as follows. On each trial, a monkey fixated on a central dot while an image flashed four times in the aggregate receptive fields of the recorded V4 neurons. After the fourth flash, the monkey made a saccade to a target dot (whose location was unrelated to the shown image), for which he received a juice reward. During this task, we recorded threshold crossings on each electrode (referred to as "spikes"), where the threshold was defined as a multiple of the RMS voltage set independently for each channel. This yielded 87 to 96 neural units in each session. The spike counts for each neural unit were averaged across the four 100 ms flashes to obtain mean responses. The mean response vector for the p neural units was then appended to the previously-recorded responses and input into Adept. Adept then output an image to show on the next trial. For the predictive feature embeddings, we used $q = 500$ CNN neurons in the fifth layer of GoogLeNet CNN (kernel bandwidth $h = 200$). In each recording session, the monkey typically performed 2,000 trials (i.e., 2,000 of the N =10,000 natural images would be sampled). Each Adept run started with $N_{\text{init}} = 5$ randomly-chosen images.

We first recorded a session in which we used Adept during one block of trials and randomly chose images in another block of trials. To qualitatively compare Adept and randomly selecting images, we first applied PCA to the response vectors of both blocks, and plotted the top two PCs (Fig. 4.5*A*, left panel). Adept uncovers more responses that are far away from the origin (Fig. 4.5*A*, left panel, red dots farther from black * than black dots). For visual clarity, we also computed kernel density estimates for the Adept responses (p_{Adept}) and responses to randomly-chosen images (p_{random}), and plotted the difference $p_{\text{Adept}} - p_{\text{random}}$ (Fig. 4.5*A*, right panel). Responses for Adept were denser than for randomly-chosen images further from the origin, whereas the opposite was true closer to the origin (Fig. 4.5*A*, right panel, red region further from origin than black region). These plots suggest that Adept uncovers large responses that are far from one another.

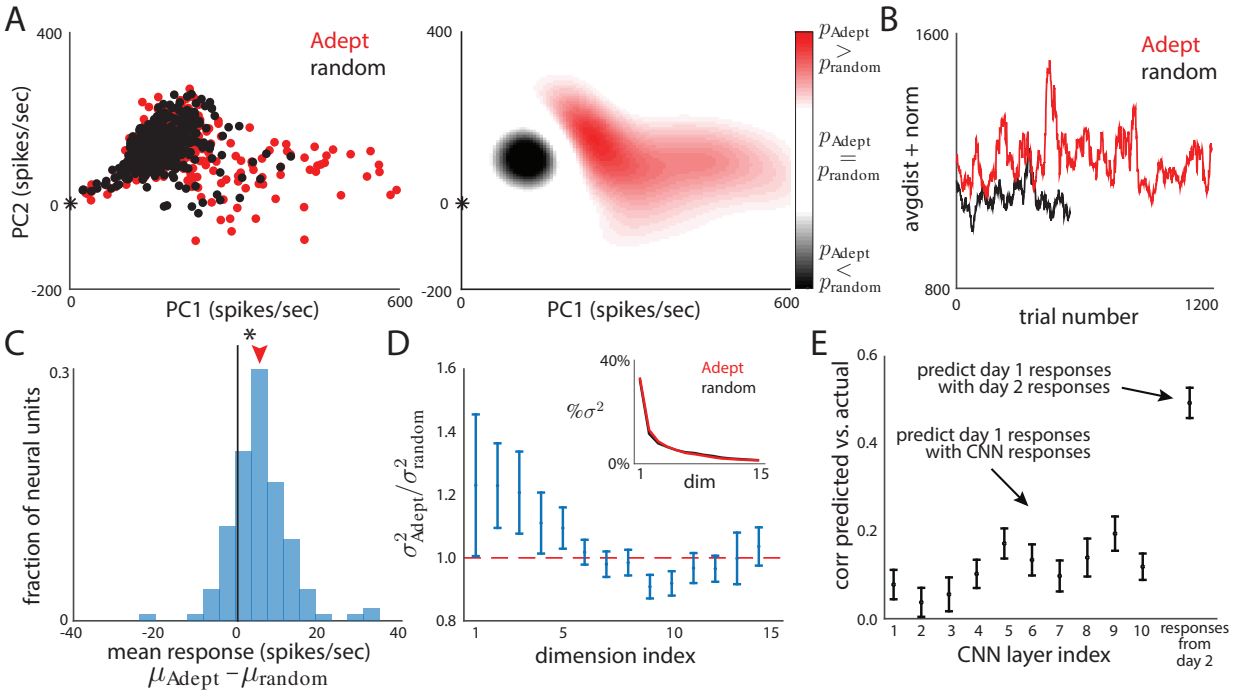


Figure 4.5: Closed-loop experiments in V4. *A*. Top 2 PCs of V4 responses to stimuli chosen by Adept and random selection (500 trials each). Left: scatter plot, where each dot represents the population response to one stimulus. Right: difference of kernel densities, $p_{\text{Adept}} - p_{\text{random}}$. Black * denotes a zero response for all neural units. *B*. Objective function evaluated across trials (one stimulus per trial) using V4 responses. Same data as in *A*. *C*. Difference in mean responses across neural units from 7 sessions. *D*. Ratio of eigenvalues for different PC dimensions. Error bars: $\pm$ s.e.m. *E*. Ability of different CNN layers to predict V4 responses. For comparison, we also used V4 responses from a different day to predict the same V4 responses. Error bars: $\pm$ s.d. across 100 runs.

Quantitatively, we verified that Adept chose images with larger objective values in Eqn. 4.1 than randomly-chosen images (Fig. 4.5*B*). This result is not trivial because it relies on the ability of the CNN to predict V4 population responses. If the CNN predicted V4 responses poorly, the objective evaluated on the V4 responses to images chosen by Adept could be lower than that evaluated on random images.

We then compared Adept and random stimulus selection across 7 recording sessions, including the above session (450 trials per block, with three sessions with the Adept block before the random selection block, three sessions with the opposite ordering, and one session with interleaved trials). We found that the images chosen by Adept produced on average 19.5% higher mean responses than randomly-chosen images (Fig. 4.5*C*, difference in mean responses were significantly greater than zero, $p < 10^{-4}$). We also found that images chosen by Adept produced greater response scatter than for randomly-chosen images, as the mean ratios of eigenvalues were greater than 1 (Fig. 4.5*D*, dimensions 1 to 5). Yet, there were dimensions for which the mean ratios of eigenvalues were less than 1 (Fig. 4.5*D*, dimensions 9 and 10). These dimensions explained little overall variance ($< 5\%$ of the total response variance).

Finally, we asked to what extent do the different CNN layers predict the objective of V4 responses, as in Fig. 4.4*D*. We found that, using 500 CNN neurons for each layer, all layers had some predictive ability (Fig. 4.5*E*, $\rho > 0$). Deeper layers (5 to 10) tended to have better prediction than superficial layers (1 to 4). To establish a noise level for the V4 responses, we also predicted the norm and average distance for one session (day 1) with the V4 responses of another session (day 2), where the same images were shown each day. In other words, we used the V4 responses of day 2 as feature embeddings to predict V4 responses of day 1. The correlation of prediction was much higher ($\rho \approx 0.5$) than that of any CNN layer ($\rho < 0.25$). This discrepancy indicates that finding feature embeddings that are more predictive of V4 responses is a way to improve Adept's performance.

4.4.3 Testing Adept for robustness to neural noise and overfitting

A potential concern for an adaptive method is that stimulus responses are susceptible to neural noise. Specifically, spike counts are subject to Poisson-like variability, which might not be entirely averaged away based on a finite number of stimulus repeats. Moreover, adaptation to stimuli and changes in attention or motivation may cause a gain factor to scale responses dynamically across a session Lewi et al. (2009). To examine how Adept performs in the presence of noise, we first recorded a "ground-truth", spike-sorted dataset in which 2,000 natural images were presented (100 ms flashes, 5 to 30 repeats per image randomly presented throughout the session). We then re-ran Adept on simulated responses under three different noise models (whose parameters were fit to the ground truth data): a Poisson model ('Poisson noise'), a model that scales each response by a gain factor that varies independently from trial to trial Lin et al. (2015) ('trial-to-trial gain'), and the same gain model but where the gain varies smoothly across trials ('slowly-drifting gain'). Because the drift in gain was randomly generated and may not match the actual drift in the recorded dataset, we also considered responses in which the drift was estimated across the recording session and added to the mean responses as their corresponding images were chosen ('recorded drift'). For reference, we also ran Adept on responses with no noise ('no noise'). To compare performance across the different settings, we computed the mean response and variance ratios between responses based on Adept and random selection (Fig. 4.6*A*). All settings showed better performance using Adept than random selection (Fig. 4.6*A*, all points above red-dashed line), and Adept performed best with no noise (Fig. 4.6, 'no noise' point at or above others). For a fair comparison, ratios were computed with the ground truth responses, where only the chosen images could differ across settings. These results indicate that, although Adept would benefit from removing neural noise, Adept continues to outperform random selection in the presence of noise.

Another concern for an adaptive method is overfitting. For example, when no relationship exists between the CNN feature embeddings and neural responses, Adept may overfit to a spurious stimulus-response mapping and perform worse than random selection. To address this concern, we performed two analyses using the same ground truth dataset as in Fig. 4.6*A*. For the first analysis, we ran Adept on the ground truth responses (choosing 500 of the 2,000 candidate images) to yield on average a 6% larger mean response and a 21% larger response scatter (average over top 5 PCs) than random selection (Fig. 4.6*B*, unshuffled responses). Next, to break any stimulus-response relationship, we shuffled all of the ground truth responses across images, and re-ran Adept. Adept

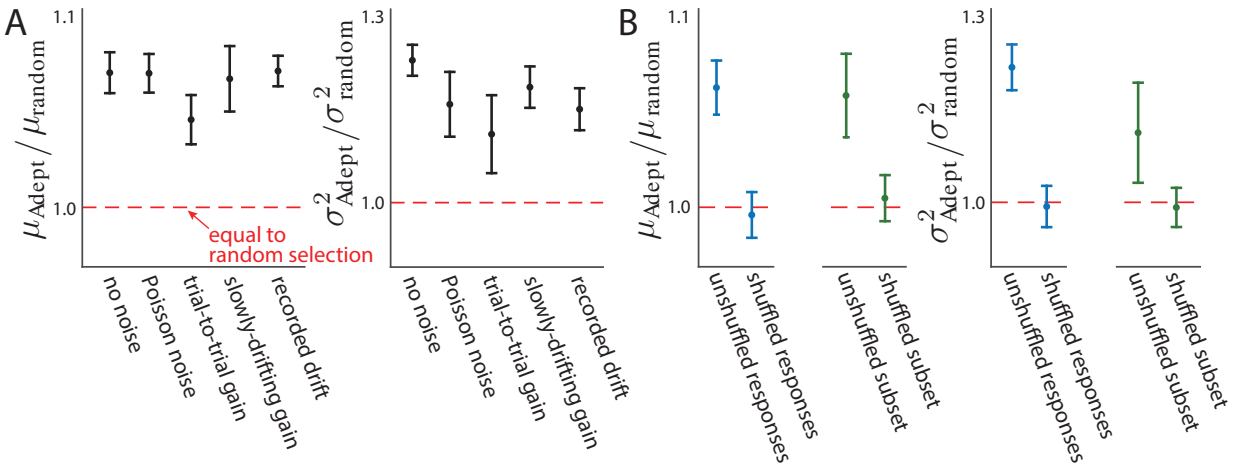


Figure 4.6: *A*. Adept is robust to neural noise. *B*. Adept shows no overfitting when responses are shuffled across images. Error bars: ± s.d. across 10 runs.

performed no worse than random selection (Fig. 4.6*B*, shuffled responses, blue points on red-dashed line). For the second analysis, we asked if Adept focuses on the most predictable neurons to the detriment of other neurons. We shuffled all of the ground truth responses across images for half of the neurons, and ran Adept on the full population. Adept performed better than random selection for the subset of neurons with unshuffled responses (Fig. 4.6*B*, unshuffled subset), but no worse than random selection for the subset with shuffled responses (Fig. 4.6*B*, shuffled subset, green points on red-dashed line). Adept showed no overfitting in either scenario, likely because Adept cannot choose exceedingly similar images (i.e., differing by a few pixels) from its discrete candidate pool.

4.5 Discussion

Here we proposed Adept, an adaptive method for selecting stimuli to optimize neural population responses. To our knowledge, this is the first adaptive method to consider a population of neurons together. We found that Adept, using a population objective, is better able to optimize population responses than using a single-neuron objective to optimize the response of each neuron in the population (Fig. 4.4*C*). While Adept can flexibly incorporate different feature embeddings, we take advantage of the recent breakthroughs in deep learning and apply them to adaptive stimulus selection. Adept does not try to predict the response of each V4 neuron, but rather uses the similarity of CNN feature embeddings to different images to predict the similarity of the V4 population responses to those images.

Widely studied neural phenomena such as changes in responses due to attention Cohen and Maunsell (2009) and trial-to-trial variability Kohn et al. (2016); Okun et al. (2015) likely depend on mean response levels Cohen and Kohn (2011). When recording from a single neuron, one can optimize to produce large mean responses in a straightforward manner. For example, one can optimize the orientation and spatial frequency of a sinusoidal grating to maximize a neuron's firing rate Lewi et al. (2009). However, when recording from a population of neurons, identifying

stimuli that optimize the firing rate of each neuron can be infeasible due to limited recording time. Moreover, neurons far from the sensory periphery tend to be more responsive to natural stimuli Felsen et al. (2005b), and the search space for natural stimuli is vast. Adept is a principled way to efficiently search through a space of natural stimuli to optimize the responses of a population of neurons. Experimenters can run Adept for a recording session, and then present the Adept-chosen stimuli in subsequent sessions when probing neural phenomena.

A future direction is to deveop theory to obtain performance guarantees for Adept. For example, what is the optimal objective function one should use for desired properties of the responses? In this work, we proposed intuitive objective functions to maximize response levels and the diversity of responses. However, other objective functions that optimally sample the response manifold based on a set of assumptions may be considered. Another avenue of theory for Adept is to study for which conditions will Adept choose an optimal sequence of stimuli to show, assuming that the objective prediction is errorless. This optimal sequence maximizes the sum of values of the objective function for the first M chosen stimuli (out of N candidate stimuli). Adept iteratively chooses the next stimulus to show in a greedy manner. If the objectives for different iterations were conditionally independent given response vector $\mathbf{r}$ (e.g., for objective function $\|\mathbf{r}\|_2$, Fig. 4.2*B*), this greedy manner is optimal. However, the greedy manner is not optimal when the objective is conditionally dependent given $\mathbf{r}$ across iterations (e.g., for objective function $\|\mathbf{r}\|_2 + \frac{1}{M}\sum_{j=1}^{M}\|\mathbf{r}-\mathbf{r}_j\|_2$, Fig. 4.2*D*). Identifying the optimal sequence is computationally intractable, as the number of combinations is equal to $N!/(N-M)!$, which exponentially increases with increasing M and N. New theory would provide estimates for how many greedily-chosen stimuli M_{greedy} are needed such that the set of M_{greedy} greedily-chosen stimuli contains the subset of M optimally-chosen stimuli. This theory can also incorporate prediction errors of the objective function to estimate the number of stimuli to adaptively choose to obtain a guarantee that the M_{greedy} shown stimuli contain the subset of M optimally-chosen stimuli. Experimenters can then use this number of stimuli to plan for enough recording time in their experiments.

A future challenge for adaptive stimulus selection is to generate natural images rather than selecting from a pre-existing pool of candidate images. For Adept, one could use a parametric model to generate natural images, such as a generative adversarial network (Radford et al., 2015), and optimize Eqn. 4.1 with gradient-based or Bayesian optimization.

Chapter 5

Dimensionality reduction for multiple brain areas

In the previous chapter, we considered an adaptive stimulus selection algorithm to choose a set of stimuli that drives a recorded population of V4 neurons. Another important aspect of V4 neurons is that they receive input from many different brain areas (Roe et al., 2012). The interactions between V4 and other brain areas likely reflect important computations carried out by the neural circuit. Thus, we would like to characterize these multi-area interactions to gain insight into the underlying computations.

Multi-area interactions are likely nonlinear (e.g., a gating mechanism between brain areas), and can be captured by a nonlinear dimensionality reduction method, such as kernel canonical correlation analysis (KCCA) (Bach and Jordan, 2002; Hardoon et al., 2004). However, nonlinear dimensionality reduction methods have two important limitations. First, the amount of data that is collected in real-world experiments is often insufficient to sample the high-dimensional space densely enough for many of these methods (Cunningham and Yu, 2014; Van Der Maaten et al., 2009). Second, most nonlinear methods provide only a low-dimensional embedding, but do not provide a direct mapping from the low-dimensional embedding to the high-dimensional data space. As a result, it is difficult to compare the topology of different low-dimensional spaces. For these reasons, many scientific (e.g., neuroscience or genetics) studies rely on linear dimensionality reduction methods, such as principal component analysis (PCA) and canonical correlation analysis (CCA) (Cowley et al., 2016; Cunningham and Yu, 2014; Kobak et al., 2016; Witten and Tibshirani, 2009).

Recently, methods that identify linear projections have been developed that can detect both linear and nonlinear interactions for multiple sets of variables. For example, hsic-CCA (Chang et al., 2013) maximizes a kernel-based correlational statistic called the Hilbert-Schmidt independence criterion (HSIC) (Gretton et al., 2007), and is a hybrid between CCA, which identifies linear projections but can only detect linear interactions, and KCCA, which can detect both linear and nonlinear interactions but identifies nonlinear projections. Thus, hsic-CCA has the interpretability of CCA as well as KCCA's ability to detect nonlinear relationships. Still, hsic-CCA is limited to identifying dimensions for two sets of variables, and cannot be used to identify dimensions for three or more sets of variables.

In this work, we propose distance covariance analysis (DCA), a dimensionality reduction

method to identify linear projections that maximize the Euclidean-based correlational statistic distance covariance (Székely and Rizzo, 2009). As with HSIC, distance covariance can detect linear and nonlinear relationships (Sejdinovic et al., 2013). DCA has several important advantages over existing linear methods that can detect both linear and nonlinear relationships, such as hsic-CCA. First, DCA can identify dimensions for more than two sets of variables. Second, DCA can take into account dependent variables for which dimensions are not identified. Finally, DCA is computationally fast—in some cases, orders of magnitude faster than competing methods—without sacrificing performance. DCA can be applied to continuous and categorical variables, order the identified dimensions based on the strength of interaction, and scale to many variables and samples. Using simulated data for one, two, and multiple sets of variables, we found that DCA performed better than or comparable to existing methods, while being one of the fastest methods. We then applied DCA to real data in three different neuroscientific contexts.

5.1 Distance covariance

Distance covariance is a statistic that tests for independence between paired random vectors $X \in \mathbb{R}^p$ and $Y \in \mathbb{R}^q$, and can detect linear and nonlinear relationships between X and Y (Székely and Rizzo, 2009). The intuition is that if there exists a relationship between X and Y, then for two similar samples $X_i, X_j \in \mathbb{R}^p$, the two corresponding samples $Y_i, Y_j \in \mathbb{R}^q$ should also be similar. In other words, the Euclidean distance between X_i and X_j covaries with that of Y_i and Y_j. To compute the sample distance covariance $\nu(X, Y)$ for N samples, one first computes the $N \times N$ distance matrices D^X and D^Y for X and Y, respectively, where $D^X_{ij} = \|X_i - X_j\|_2$ for $i, j = 1, \ldots, N$. D^Y is computed in a similar manner. To take the covariance between distance matrices D^X and D^Y, the row, column, and matrix means must all be zero. This is achieved by computing the re-centered distance matrix R^X, where $R^X_{ij} = D^X_{ij} - \bar{D}^X_{\cdot j} - \bar{D}^X_{i\cdot} + \bar{D}^X_{\cdot\cdot}$ and the $\bar{D}^X$ terms are the row, column, and matrix means. R^Y is defined in a similar manner. The (squared) distance covariance $\nu^2(X, Y)$, a scalar, is then computed as:

$$\nu^2(X, Y) = \frac{1}{N^2} \sum_{i,j=1}^{N} R^X_{ij} R^Y_{ij} \tag{5.1}$$

If $\nu = 0$, then X and Y are independent (Székely and Rizzo, 2009). The formulation of distance covariance utilizes both small and large Euclidean distances, in contrast to the locality assumption of many nonlinear dimensionality reduction methods (Van Der Maaten et al., 2009). Whereas nonlinear dimensionality reduction methods seek to identify a nonlinear manifold, distance covariance seeks to detect relationships between sets of variables.

5.2 Optimization framework for DCA

In this section, we first formulate the DCA optimization problem for identifying a dimension[1] of X that has the greatest distance covariance with Y. We then extend this formulation for multiple sets of variables by identifying a dimension for each set that has the greatest distance covariance with all other sets. In Section 5.3, we propose the DCA algorithm to identify orthonormal linear dimensions ordered by decreasing distance covariance for each set of variables.

5.2.1 Identifying a DCA dimension for one set of variables

Consider maximizing the distance covariance between $\mathbf{u}^T X$ and Y with respect to dimension $\mathbf{u} \in \mathbb{R}^p$. We compute the (squared) distance covariance $\nu^2(\mathbf{u}^T X, Y)$ defined in (5.1), with re-centered distance matrix $R^X(\mathbf{u})$ for $\mathbf{u}^T X$. The optimization problem is:

$$\max_{\substack{\mathbf{u} \\ \|\mathbf{u}\| \leq 1}} \frac{1}{N^2} \sum_{i,j=1}^{N} R^Y_{ij} R^X_{ij}(\mathbf{u}) \tag{5.2}$$

This problem was proposed in Sheng and Yin (2013), where it was proven that the optimal solution $\mathbf{u}^*$ is a consistent estimator of $\beta \in \mathbb{R}^p$ such that X is independent of Y given $\beta^T X$. Similar guarantees exist for HSIC-related methods, such as kernel dimensionality reduction (KDR) (Fukumizu et al., 2004a). However, Sheng and Yin (2013) only considered the case of identifying one dimension for one set of variables, and optimized with an approximate-gradient method (Matlab's `fmincon`).

Instead, we optimize this problem using projected gradient descent with backtracking line search. The gradient of the objective function with respect to $\mathbf{u}$ is:

$$\frac{\partial \nu^2}{d\mathbf{u}} = \frac{1}{N^2} \sum_{i,j=1}^{N} R^Y_{ij} (\delta_{ij}(\mathbf{u}) - \bar{\delta}_{\cdot j}(\mathbf{u}) - \bar{\delta}_{i\cdot}(\mathbf{u}) + \bar{\delta}_{\cdot\cdot}(\mathbf{u})) \tag{5.3}$$

where $\delta_{ij}(\mathbf{u}) = (X_i - X_j)\,\text{sign}(\mathbf{u}^T(X_i - X_j))$ and the $\bar{\delta}$ terms are the derivatives of the row, column, and matrix means that are used to re-center the distance matrix. For large numbers of samples, we can also make use of the fact that each gradient step is computationally inexpensive to employ stochastic projected gradient descent (with a momentum term (Hu et al., 2009) and a decaying learning rate $\tau = 0.9$).

We can intuit how DCA optimizes its objective function from its gradient update. First, the $(X_i - X_j)$ vectors are reflected onto the positive half-space of the hyperplane with normal vector $\mathbf{u}$, which avoids $(X_i - X_j)$ canceling $(X_j - X_i)$ when recentering. We then recenter the reflected $(X_i - X_j)$ vectors to consider deviations from the mean. These deviations are then weighted by the extent to which $\|Y_i - Y_j\|_2$ also deviates from its mean, and the resulting vectors are averaged. Thus, an initial $\mathbf{u}$ will be re-oriented to align in a direction in which the $(X_i - X_j)$ vectors covary

[1] In this work, we use the phrase "linear dimensions" or "dimensions" of a random vector $X \in \mathbb{R}^p$ to refer to either a set of orthonormal basis vectors that define a subspace in $\mathbb{R}^p$, or the projection of X onto those vectors, depending on the context.

with $(Y_i - Y_j)$. Because the gradient considers all possible directions in $\mathbb{R}^p$, the final solution does not heavily depend on the initialization of $\mathbf{u}$ for a strong relationship between X and Y (see Results).

We found that projected gradient descent performed better and was faster than other optimization approaches, such as Stiefel manifold optimization (Cunningham and Ghahramani, 2015). This is likely the case because we only optimize one dimension at a time, and do not optimize directly for multiple dimensions (see Section 5.3).

5.2.2 Identifying DCA dimensions for multiple sets of variables

Consider identifying dimensions $\mathbf{u}_1 \in \mathbb{R}^{p_1}$ and $\mathbf{u}_2 \in \mathbb{R}^{p_2}$ for two sets of variables $X^1 \in \mathbb{R}^{p_1}$ and $X^2 \in \mathbb{R}^{p_2}$, where p_1 need not equal p_2, that maximize the distance covariance by extending (5.2):

$$\max_{\substack{\mathbf{u}^1,\mathbf{u}^2 \\ \|\mathbf{u}^1\|,\|\mathbf{u}^2\|\leq 1}} \frac{1}{N^2}\sum_{i,j=1}^{N} R^1_{ij}(\mathbf{u}^1)R^2_{ij}(\mathbf{u}^2) \tag{5.4}$$

To optimize, we alternate optimizing $\mathbf{u}^1$ and $\mathbf{u}^2$, whereby on each iteration we first fix $\mathbf{u}^2$ and optimize for $\mathbf{u}^1$, then fix $\mathbf{u}^1$ and optimize for $\mathbf{u}^2$. Because of the symmetry of the objective function, each alternating optimization reduces to solving (5.2).

To identify dimensions for multiple sets of variables, we extend the definition of distance covariance in (5.1) to capture pairwise dataset interactions across M sets of variables $X^1, \ldots, X^M$ (with $X^m \in \mathbb{R}^{p_m}$), where each set may contain a different number of variables:

$$\nu(X^1, \ldots, X^M) = \frac{1}{\binom{M}{2}} \sum_{1\leq m<n\leq M} \nu(X^m, X^n) \tag{5.5}$$

Using (5.5), we extend the optimization problem of (5.4) to multiple sets of variables, where we desire one dimension for each set of variables that maximizes the distance covariance. We can also include information from Q sets of dependent variables $Y^1, \ldots, Y^Q$ for which we are not interested in identifying dimensions but are interested in detecting their relationship with dimensions of each X^m. An example of this is a neuroscientific experiment where we identify dimensions of the recorded activity of neurons that are related across M subjects $(X^1, \ldots, X^M)$ *and* related to both stimulus (Y^1) and behavioral (Y^2) information.

We seek to identify dimensions $\mathbf{u}^1, \ldots, \mathbf{u}^M$, where $\mathbf{u}^m \in \mathbb{R}^{p_m}$, that maximize the distance covariance $\nu({\mathbf{u}^1}^T X^1, \ldots, {\mathbf{u}^M}^T X^M, Y^1, \ldots, Y^Q)$. Using (5.5), the optimization problem is:

$$\max_{\substack{\mathbf{u}^1,\ldots,\mathbf{u}^M \\ \|\mathbf{u}^m\|\leq 1}} \frac{1}{\binom{M}{2}}\frac{1}{N^2} \sum_{1\leq m<n\leq M} \langle R^m(\mathbf{u}^m), R^n(\mathbf{u}^n)\rangle + \frac{1}{M}\frac{1}{N^2}\sum_{m=1}^{M}\langle R^m(\mathbf{u}^m), R^D\rangle \tag{5.6}$$

where $R^m(\mathbf{u}^m)$ is the re-centered distance matrix of ${\mathbf{u}^m}^T X^m$, $\langle R^m, R^n\rangle = \sum_{ij} R^m_{ij} R^n_{ij}$, and $R^D = \frac{1}{Q}\sum_{q=1}^{Q} R^q$ (i.e., the average of the re-centered distance matrices $R^1, \ldots, R^Q$ of the sets of

dependent variables $Y^1, \ldots, Y^Q$). The first term is the distance covariance for multiple sets of variables as defined in (5.5), and the second term is the distance covariance between each set of variables and the sets of dependent variables. Similar to optimizing for two sets of variables, we optimize each $\mathbf{u}^m$ in an alternating manner which reduces solving (5.6) to solving (5.2) because we only consider terms that include $\mathbf{u}^m$.

5.3 DCA algorithm

For the optimization problem (5.6), we identify only one dimension for each set of variables. We now present the DCA algorithm (Algorithm 2), which identifies a set of DCA dimensions ordered by decreasing distance covariance for each set of variables. Given K desired DCA dimensions, DCA identifies the kth DCA dimension $\mathbf{u}_k^m$ for each of the M datasets by iteratively optimizing $\mathbf{u}_k^1, \ldots, \mathbf{u}_k^M$ in (5.6) until some criterion is reached (e.g., the fraction of change in the objective function for two consecutive iterations is less than some ϵ). Then, the data are projected onto the orthogonal subspace of the k previously-identified dimensions before optimizing for the $(k+1)$th DCA dimension. This ensures that all subsequently-identified dimensions are orthogonal to the previously-identified dimensions. DCA returns the identified dimensions as columns in the matrices $U^1, \ldots, U^M$, where $U^m \in \mathbb{R}^{p_m \times K}$, and the corresponding ordered distance covariances $d_1, \ldots, d_K$.

Algorithm 2: DCA algorithm

Input: $\{X^1, \ldots, X^M\}$, $\{Y^1, \ldots, Y^Q\}$, K desired dims
Output: $\{U^1, \ldots, U^M\}$, $\{d_1, \ldots, d_K\}$
initialize $\{U^1, \ldots, U^M\}$ randomly;
for $k = 1, \ldots, K$ **do**
  while *criterion not reached* **do**
    for $m = 1, \ldots, M$ **do**
      $\mathbf{u}_k^m \leftarrow \max \nu(\mathbf{u}_k^{1T} X^1, \ldots, \mathbf{u}_k^{MT} X^M, Y^1, \ldots, Y^Q)$ w.r.t. $\mathbf{u}_k^m$ s.t. $\|\mathbf{u}_k^m\|_2 \leq 1$
    end
  end
  $d_k \leftarrow \nu(\mathbf{u}_k^{1T} X^1, \ldots, \mathbf{u}_k^{MT} X^M, Y^1, \ldots, Y^Q)$;
  for each of the M datasets, $U^m(:, k) \leftarrow \mathbf{u}_k^m / \|\mathbf{u}_k^m\|_2$;
  for each of the M datasets, $X^m \leftarrow$ project X^m onto orthogonal space of $U^m(:, 1{:}k)$;
end

To determine the number of DCA dimensions needed, one can test if the distance covariance of the kth dimension is significant by a permutation test. Samples are first projected onto the orthogonal space of the previously-identified $(k-1)$ dimensions, because those dimensions are not considered when optimizing the kth dimension. Then, the samples are shuffled within datasets to break any relationships across datasets. A dimension is statistically significant if its distance covariance is greater than a large percentage of the distance covariances for many shuffled runs (e.g., 95% for significance level $p = 0.05$)

5.4 Simulated example: Gating mechanism among three brain areas

To illustrate the usefulness of DCA in a neuroscientific context, we simulate a realistic gating mechanism among three brain areas and test if DCA can recover the interactions of the mechanism. Consider three brain areas X, Y, and Z, where we simultaneously record 20 neurons from each brain area (i.e., 60 recorded neurons in total) (Fig. 5.1*A*). We simulated neural activity following a gating mechanism (Fig. 5.1*B*). When the activity of neurons in brain area G is large, the activity of X and Y neurons is nonlinearly related. When the activity of G neurons is low, the activity of X neurons is independent of the activity of Y neurons. Mathematically, we generated the activity of X neurons from a Poisson distribution with mean equal to 15 spikes/sec. We generated the activity of G neurons following a Poisson distribution whose mean parameter changed for all neurons between either 5 or 15 spikes/sec. If the population mean firing rate $\bar{G}$ across the G neurons was larger than 10 spikes/sec (Fig. 5.1*C*, red dots), then we generated the activity of the ith Y neuron as $_i = \sin(\beta^T \mathbf{X}) + \epsilon$, where $\mathbf{X}$ is the (20×1) vector of spike counts for brain area X, β is the vector of ground truth projection vector weights, and ϵ is a small amount of added Gaussian noise. If the population rate $\bar{G} < 10$ spikes/sec (Fig. 5.1*C*, blue dots), then the activity of the Y neurons was generated from a Poisson distribution with a mean of 15 spikes/sec. By defining the interactions in this way, we guaranteed that a single 3-d projection captured all interactions among brain areas (Fig. 5.1*C*).

We next asked if commonly-used dimensionality reduction methods could identify the multi-area interactions. Applying PCA separately to the population activity of each brain area recovered the neural state change in G, but did not identify the interaction between X and Y (Fig. 5.1*D*, red dots). PCA failed to identify the interaction because it has no notion of relationships between areas. We also applied multi-CCA or generalized CCA, an extension of CCA for multiple sets of variables (Kettenring, 1971) (Fig. 5.1*E*). Multi-CCA failed to capture the nonlinear relationship between X and Y (Fig. 5.1*E*, red dots), because multi-CCA can only detect linear interactions. Finally, we tested DCA on the same simulated data (Fig 5.1*F*), and found that DCA faithfully recovered the ground truth projection (compare Fig. 5.1*C* and 5.1*F*). This is because DCA can identify both linear and nonlinear interactions.

5.5 Performance on previous testbeds

We compared the performance of DCA to existing methods on testbeds used in previous work. We first considered the setting of identifying dimensions for X that are related to Y. We replicated the testbed used for KDR (Fukumizu and Leng, 2014), which included five different relationships between X and Y, ranging from sinusoidal to a 4th-degree polynomial (Fig. 5.2*A*). The five simulations are labeled as "A", "B", "C-a", "C-b", and "D", matching the labeling in Fukumizu and Leng (2014). Each simulation had 10 or 50 variables, 1,000 samples, and a ground truth β whose columns determined which dimensions of X related to Y.

We then measured performance by computing the mean prinicipal angle between β and the identified $\hat{\beta}$. Existing methods included the HSIC-based methods KDR (Fukumizu and Leng, 2014) and supervised PCA (SPCA) (Barshan et al., 2011), the distance-based method supervised

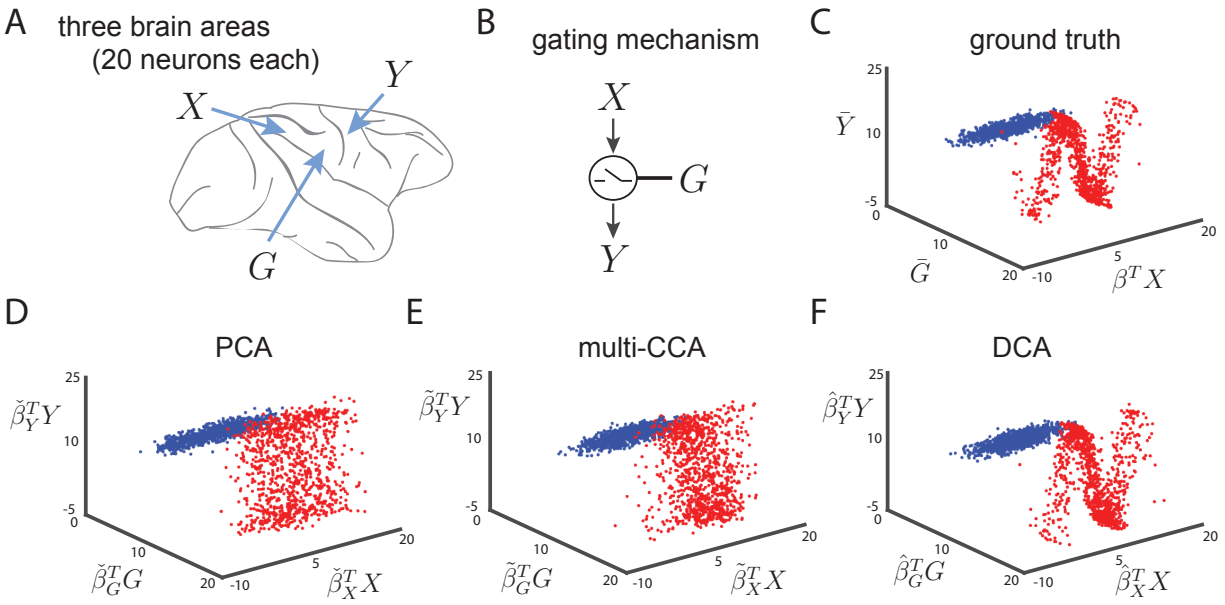


Figure 5.1: *A*. Simulated neural activity among three brain areas X, Y, and G, where 20 neurons were simultaneously recorded from each brain area (60 total neurons simultaneously recorded). *B*. Data were simulated from a gating mechanism (see text for details). *C*. Ground truth projection that captures all of the interactions among the three brain areas. Each axis represents a linear combination of simulated neurons. We applied PCA (*D*), multi-CCA (*E*) and DCA (*F*) to the simulated activity, where each method identified different projection vectors ($\check{\beta}$, $\tilde{\beta}$, $\hat{\beta}$, respectively).

distance preserving projections (SDPP) (Zhu et al., 2013), the distance covariance-based method DCOV (Sheng and Yin, 2016), and as a control, PCA. Note that DCA, which optimizes dimensions sequentially, is a statistically different method than DCOV, which optimizes for all dimensions at once. For existing methods across all testbeds in this work, we used publicly available code cited by the methods' corresponding papers, as well as their suggested procedures to fit hyperparameters, such as kernel bandwidths. We found that DCA was among the best performing methods for each simulation (Fig. 5.2*A*, red). Simulation D showed worse performance for all methods compared to the other simulations because it required identifying 10 dimensions for 50 variables, while the other methods required identifying only a few dimensions for 10 variables.

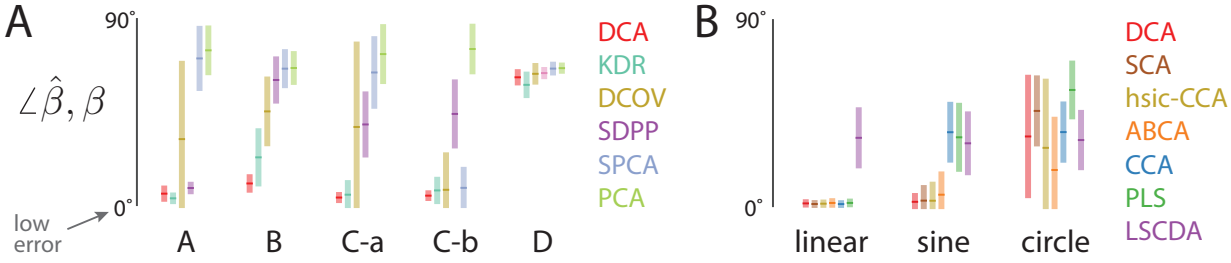


Figure 5.2: (A) KDR testbeds. (B) hsic-CCA testbeds. Error bars: standard deviation over 100 runs.

Next, we considered the setting of identifying dimensions for both X and Y. We replicated the testbed used for hsic-CCA (Chang et al., 2013), which comprised three different relationships between variables in X (5 variables) and Y (4 variables) with 1,000 samples (Fig. 5.2*B*). We measured performance by computing the principal angles between the ground truth dimensions β and the identified dimensions $\hat{\beta}$ for both X and Y, and taking the average. Existing methods included those that can detect only linear interactions, such as CCA (Hotelling, 1936) and partial least squares (PLS) (Helland, 1988; Höskuldsson, 1988), as well as methods that can detect both linear and nonlinear interactions by maximizing a correlational statistic. These methods include least squares canonical dependency analysis (LSCDA, minimizing the squared-loss mutual information) (Karasuyama and Sugiyama, 2012), hsic-CCA (maximing the kernel-based HSIC statistic) (Chang et al., 2013), semiparametric canonical analysis (SCA, minimizing the prediction error of a local polynomial smoother) (Xia, 2008), and AB-canonical analysis (ABCA, maximizing the alpha-beta divergence) (Mandal and Cichocki, 2013).

Similar to the results in Fig. 5.2*A*, we found that DCA was among the best performing methods for the "linear" and "sine" simulations. The "circle" simulation proved challenging for all methods, with ABCA and hsic-CCA having the best mean performance (but within the error margin for DCA). The results of the two testbeds in Fig. 5.2 demonstrate that DCA is highly competitive with existing methods at detecting both linear and nonlinear interactions. However, these simulations tested only a small handful of linear and nonlinear relationships, and it is unclear how well these results generalize to other types of nonlinearities.

5.6 Performance on novel testbeds

Because the previous testbeds probed a small number of nonlinearities, we designed a testbed that allowed us to systematically vary the relationship between datasets from linear to highly nonlinear. Because existing methods are typically only applicable to one setting (identifying dimensions for one, two, or multiple sets of variables), we tested DCA separately on the three different settings for comparison. None of the existing methods can be applied to all three settings, which highlights the versatility of DCA.

5.6.1 Identifying dimensions for one set of variables

To systematically vary the relationship between X and Y, we generated the data (1,000 samples for each of 10 runs) as follows. Let $X = [x_1, \ldots, x_{50}]^T$, where $x_i \sim \mathcal{N}(0, 1)$, and let $\beta \in \mathbb{R}^{50\times 5}$, where each element is drawn from a standard Gaussian. The columns $\beta_1, \ldots, \beta_5$ are then orthonormalized. Define each element of $Y = [y_1, \ldots, y_5]^T$ as $y_i = \sin(\frac{2\pi}{\alpha} f {\beta_i}^T X)$. We chose the sine function because for $f = 1$, it is approximately linear (i.e., $\sin(x) \approx x$ for $[-\frac{\pi}{4}, \frac{\pi}{4}]$), and increases in nonlinearity with increasing f. To ensure that $\beta_i^T X$ did not exceed the domain of $[-\frac{\pi}{4}, \frac{\pi}{4}]$, we included a normalization constant $\alpha = 8\sqrt{50} \cdot \|[X_1 \ \ldots \ X_{1000}]\|_\infty$. The 45 dimensions in X not related to Y represent noise, although further noise could be added to Y. We measure the performance of a method by comparing the mean principal angle between identified dimensions $\hat{\beta}$ and the ground truth β.

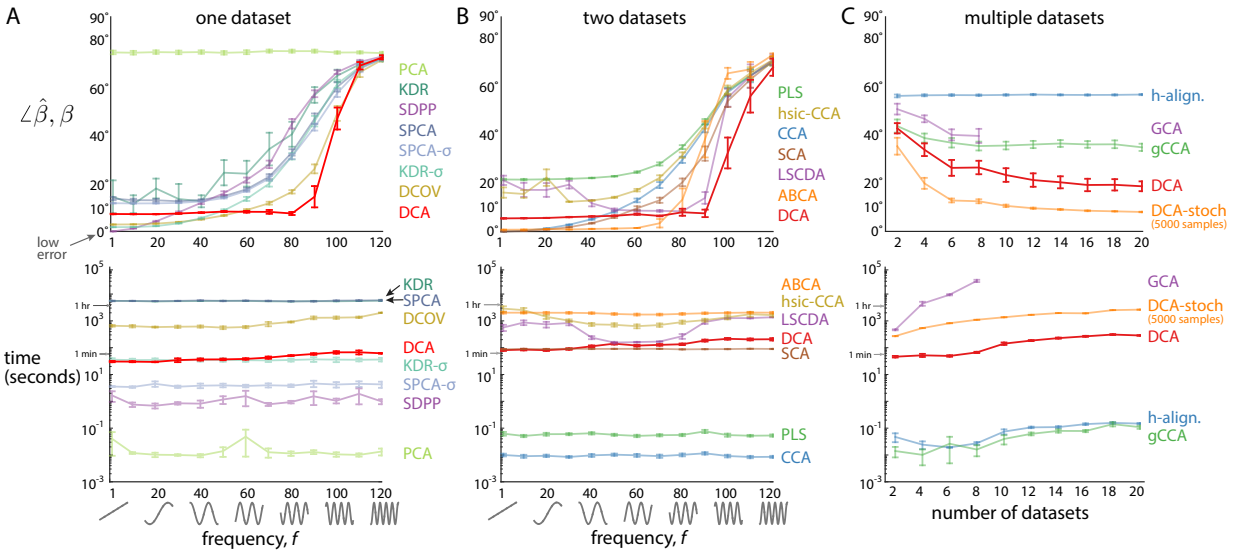


Figure 5.3: Top panels show error (measured by angle of overlap) for identifying dimensions for (A) one set, (B) two sets, and (C) multiple sets of variables. We varied the degree of nonlinearity in A and B, and the number of datasets in C. Bottom panels show running time in log scale. Error bars: standard deviations over 10 runs.

We tested DCA against existing methods that identified dimensions for X related to Y. We found that DCA performed well (error $< 10°$) for low frequencies, and outperformed the other methods for $60 < f < 100$ (Fig. C.1*A*, top panel). DCA also ran remarkably fast— orders of magnitude faster than KDR and SPCA, which require fitting kernel bandwidths, as well as DCOV, which relies on an approximate gradient descent method (Fig. C.1*A*, bottom panel). We confirmed that fitting kernel bandwidths to the data was the cause of the large computation time for KDR and SPCA by selecting a kernel bandwidth σ a priori (equal to the median of the Euclidean distances between data points). In this case, KDR-σ and SPCA-σ required similar running times as DCA (Fig. C.1*A*, bottom panel). We note that for $f \leq 40$, SDPP and KDR-σ performed better than DCA with comparable running times. Thus, these methods are more appropriate for detecting linear interactions between X and Y, while DCA is more appropriate for detecting nonlinear interactions. Since DCA is a nonconvex optimization problem, we confirmed that for 100 random starts for the same X and Y (at $f = 30$), the solutions were consistent, with mean principal angle $7.78° \pm 0.27°$.

5.6.2 Identifying dimensions for two sets of variables

In the case of two sets of variables, we sought to identify dimensions for both X and Y. We used the same simulated data as in Section 5.6.1, and assessed how well a method recovered β, the dimensions of X related to Y (Fig. C.1*B*). There are three main findings. First, DCA performed well (error $< 10°$) for low frequencies and outperformed the other methods for $80 < f < 110$. CCA, SCA, and ABCA were better able to capture linear interactions than DCA, although ABCA was much slower. We also confirmed that DCA performed well for non-orthonormalized β's

and added Gaussian noise to Y (Supp. Fig. 1). Second, DCA performed better when it removed previously-identified dimensions of Y that could mask other relevant dimensions of Y (Fig. C.1*B*, red curve) than when DCA did not identify dimensions of Y (Fig. C.1*A*, red curve, e.g., compare with Fig. C.1*B* for $f = 90$). This is an advantage shared by SCA, ABCA, and hsic-CCA (Chang et al., 2013). Finally, DCA was fast—an order of magnitude faster than some competing methods (Fig. C.1*B*, bottom panel).

5.6.3 Identifying dimensions for multiple sets of variables

In the case of multiple sets of variables, we aimed to identify dimensions for up to $M = 20$ datasets. To test performance, we extended the previous testbed in the following way. First, we generated 500 samples of $Z \in \mathbb{R}^5$, where each element was drawn from a standard Gaussian. Then, for each set of variables $X^1, \ldots, X^M \in \mathbb{R}^{10}$, we generated a random orthonormal basis $\beta^m = [\beta_1^m, \ldots, \beta_5^m]$, where $\beta_i^m \in \mathbb{R}^5$. The first five of ten variables in the mth dataset $X^m = [x_1^m, \ldots, x_{10}^m]^T$ were $x_i^m = \sin(\frac{2\pi}{\alpha} f \beta_i^{mT} Z)$ for $i = 1, \ldots, 5$, $m = 1, \ldots, M$, $f = 30$, and $\alpha = 8\sqrt{5} \cdot \|[Z_1 \ \ldots \ Z_{500}]\|_\infty$. The remaining five variables of X^m were shuffled versions of the first five variables (shuffled across samples). By generating the data in this way, we ensured that only the first five variables in each dataset were related across datasets, and we defined ground truth to be any 5-d subspace spanned by the first five variables. To measure performance, we computed the mean principal angle between the top five identified dimensions and the first five standard basis dimensions $\{e_1, \ldots, e_5\}$, and averaged over the M datasets.

We found that as the number of datasets increased, the performance of most methods improved (Fig. C.1*C*, top panel). This is because the methods had access to more samples to better detect interactions between datasets. DCA showed better performance than hyperalignment ("h-align.") (Haxby et al., 2011), whose PCA step returns random dimensions because all variables in X^m have equal variance. We also tested generalized CCA (gCCA) (Kettenring, 1971), which can only detect linear interactions between datasets. gCCA performed better than hyperalignment, presumably because it detected weak linear interactions between datasets, but performed worse than DCA. Finally, we tested generalized canonical analysis (GCA), a method that can detect nonlinear interactions between multiple datasets (Iaci et al., 2010), but this method performed worse and was orders of magnitude slower than DCA (Fig. C.1*C*, bottom panel). We also tested the scalability of DCA by increasing the number of samples from 500 to 5,000. As expected, DCA with stochastic projected gradient descent outperformed the other methods at detecting the nonlinear relationships between datasets (Fig. C.1*C*, "DCA-stoch").

5.7 Applications

To demonstrate how DCA can be applied to real-world data, we apply DCA to three datasets comprising recordings from tens of neurons in primary visual cortex that represent the three different settings (identifying dimensions for one, two, and multiple sets of variables). Because we do not know ground truth for these datasets, we do not compare DCA with all existing methods. Here, the purpose is to highlight how DCA returns sensible results in all three settings.

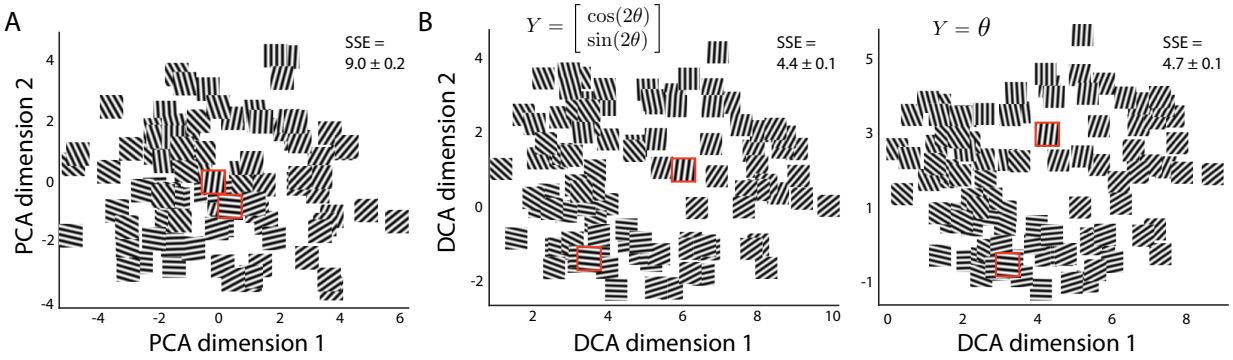


Figure 5.4: (A) PCA projection of neural responses overlaid with corresponding grating images. Red-outlined gratings have angles 90° apart. SSE: sum of squared error. (B) DCA projections for different Y. Same data and red-outlined gratings as in (A).

5.7.1 Identifying stimulus-related dimensions of neural population activity

When recording the activity of neurons in response to a sensory stimulus, there are typically aspects of the neural responses that covary with the stimulus and those that do not covary with the stimulus. Having recorded from a population of neurons, we can define a multi-dimensional activity space, where each axis represents the activity level of each neuron (Cunningham and Yu, 2014). It is of interest to identify dimensions in this space in which the population activity covaries with the stimulus (Kobak et al., 2016; Mante et al., 2013). For example, one can record from neurons in primary visual cortex (V1) and seek dimensions in which the population activity covaries with the orientation of moving bars. We analyzed a dataset in which we recorded the activity of 61 V1 neurons in response to drifting sinusoidal gratings presented for 300 ms each with the same spatial frequency and 49 different orientation angles (equally spaced between 0° and 180°) (Kelly et al., 2010). We computed the mean spike counts taken in 300 ms bins and averaged over 120 trials.

If there were a strong relationship between the neural activity and grating orientation, we would expect to see nearby neural responses encode similar grating orientations. However, when we applied PCA to the trial-averaged population activity (Fig. 5.4*A*), we did not observe this similarity for all nearby responses (Fig. 5.4*A*, red-outlined gratings). To provide supervision, we sought to identify dimensions of the trial-averaged population activity X (61 neurons $\times$ 49 orientations) that were most related to Y, the representation of grating orientation. Because we did not seek to identify dimensions in Y, this example is in the setting of identifying dimensions for one dataset.

The mean response of a V1 neuron $f(\theta)$ to different orientations θ can be described by a cosine tuning model with a preferred orientation θ_{pref} (Shriki et al., 2012). If V1 neurons were truly cosine-tuned, then $f(\theta) \propto \cos(2(\theta - \theta_{\text{pref}})) = \alpha_1 \cos(2\theta) + \alpha_2 \sin(2\theta)$, where α_1 and α_2 depend on θ_{pref}. This motivates letting $Y = [\cos(2\theta), \sin(2\theta)]^T$ to define a linear relationship between X and Y. DCA identified two dimensions in firing rate space that strongly capture orientation (Fig. 5.4*B*, left panel). However, if instead we represented orientation directly as $Y = \theta$ (therefore not utilizing domain knowledge), X and Y would have a nonlinear relationship. For this representation of orientation, DCA was still able to identify two dimensions that strongly

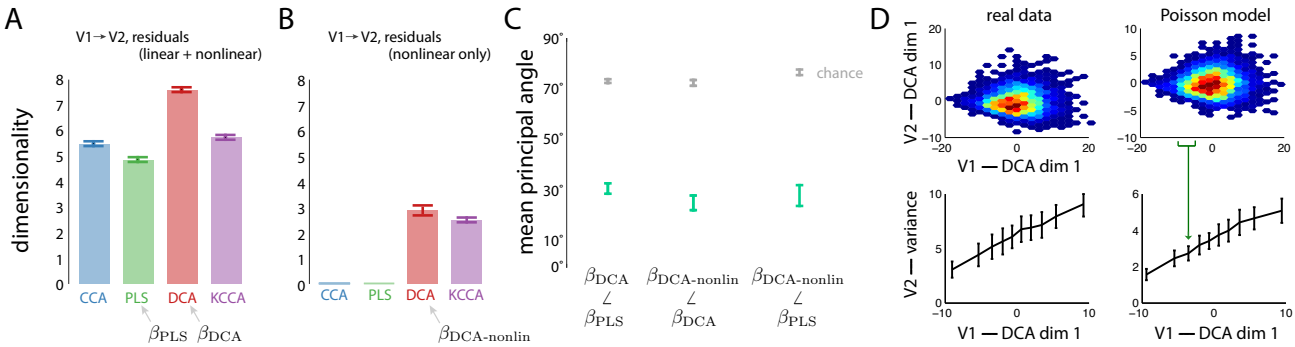


Figure 5.5: Dimensionality (A) between activity of V1 and V2 and (B) between the same activity, minus linear relationships. (C) Similarity between identified dimensions. (D) Top panels: Density plots of projections of the top DCA dimension of V1 and V2 activity for one grating (left, same data as in B) and for data generated from a linear-Poisson model (right). Red (blue) areas denote a high (low) density of datapoints. Bottom panels: Variance of projected V2 activity was computed in bins with equal number of datapoints; green arrow shows an example bin.

capture orientation (Fig. 5.4*B*, right panel). For both cases, the two DCA dimensions had nearly half of the cross-validated sum of squared error (SSE), computed with linear ridge regression, than that of the two PCA dimensions. This highlights the ability of DCA to detect both linear and nonlinear relationships and return sensible results.

5.7.2 Identifying nonlinear relationships between neural population activity recorded in two distinct brain areas

An open question in neuroscience is how populations of neurons interact across different brain areas. Previous studies have examined linear interactions between brain areas V1 and V2 (Semedo et al., 2014). Here, we attempt to identify nonlinear interactions between V1 and V2. We analyzed a dataset in which we presented drifting sinusoidal gratings (8 orientations, each with 400 trials; 1 sec stimulus presentation for each trial) while simultaneously recording population activity from 75 V1 neurons and 22 V2 neurons (Zandvakili and Kohn, 2015). To focus on the moment-by-moment interactions, we computed the residuals of the activity by subtracting the mean spike counts from the raw spike counts (100 ms bins) for each orientation.

We asked whether DCA could identify a relationship between V1 activity and V2 activity after the linear relationship between them was removed. If so, this would imply that a nonlinear relationship exists between V1 and V2, and could be identified by DCA. We first applied CCA, PLS, DCA, and KCCA to the activity for each orientation. We chose CCA and PLS because they can only detect linear interactions, and KCCA because it can detect nonlinear interactions. Dimensionality was determined as described in Section 5.3 with a significance of $p < 0.05$. We found that CCA, PLS, DCA, and KCCA all returned a dimensionality greater than zero (Fig. 5.5*A*), indicating that interactions exist between V1 and V2. We then subtracted the linear contribution of V1 from the V2 activity (identified with linear ridge regression). We confirmed that CCA and PLS, both linear methods, identified zero dimensions for the V2 activity that had no linear contribution from V1 (Fig. 5.5*B*). However, DCA identified 2 to 3 dimensions, suggesting that

nonlinear interactions exist between V1 and V2. KCCA also identified a non-zero dimensionality (Fig. 5.5*B*), consistent with DCA.

A key advantage of methods that identify linear projections is that they can address certain types of scientific questions more readily than nonlinear methods. For example, we asked if the linear and nonlinear interactions between V1 and V2 occur along similar dimensions in the V1 activity space. It is difficult for KCCA to address this question because it does not return a mapping between the low-dimensional embedding and the high-dimensional activity space. In contrast, DCA, which identifies linear dimensions but can still detect nonlinear interactions, can readily be used to address this question. We first computed the mean principal angle between the dimensions identified by PLS and DCA in Fig. 5.5*A*, and found that the dimensions overlapped more than expected by chance (Fig. 5.5*C*, $\beta_{DCA} \angle \beta_{PLS}$, green bar lower than gray bar). This suggests that some DCA dimensions represent similar linear interactions as those identified by PLS. Next, we asked if DCA returned similar dimensions when considering the full activity (β_{DCA}, Fig. 5.5*A*) versus the activity minus any linear contributions ($\beta_{\text{DCA-nonlin}}$, Fig. 5.5*B*). As expected, the mean principal angle was small compared to chance (Fig. 5.5*C*, $\beta_{\text{DCA-nonlin}} \angle \beta_{\text{DCA}}$), confirming that DCA detects similar nonlinear interactions in both cases. Finally, we asked if the linear and nonlinear interactions occur along similar dimensions. We found that the mean principal angle between the PLS dimensions and the DCA-nonlinear dimensions was smaller than chance (Fig. 5.5*C*, $\beta_{\text{DCA-nonlin}} \angle \beta_{\text{PLS}}$). This suggests that linear and nonlinear interactions do occur along similar dimensions. Similar results hold for CCA (albeit with larger principal angles, ~60°).

To gain intuition about the nonlinear interactions identified by DCA, we plotted the top DCA dimension for V1 versus the top DCA dimension for V2 (Fig. 5.5*D*, top left, one representative grating). We noticed that the variance of the V2 activity increased as the V1 activity increased—a nonlinear, heteroskedastic interaction (Fig. 5.5*D*, bottom left). We hypothesized that a linear-nonlinear-Poisson model, where V2 activity is generated by a Poisson process whose rate is a linear projection of V1 activity passed through a hinge function, could explain this relationship. Indeed, when applying DCA to data generated from the linear-nonlinear-Poisson model, we found a similar trend as that of the real data (Fig. 5.5*D*, right panels).

5.7.3 Aligning neural population activity recorded from different subjects

The recording time (i.e., number of trials) in a given experimental session is typically limited by factors such as the subject's satiety or neural recording stability. To increase the number of trials, one can consider combining many individual datasets into one large dataset for analysis. The question is how to combine the different datasets given that possibly different neurons are recorded in each dataset. One way is to align population activity recorded from different subjects, provided that the neurons are recorded in similar brain locations. This is similar in spirit to methods that align fMRI voxels across subjects (Haxby et al., 2011). DCA is well-suited for alignment because of its ability to identify dimensions that are similar across subjects *and* related to stimulus information, and its ability to detect nonlinear relationships across subjects. To showcase DCA as an alignment method, we analyzed a dataset in which V1 population activity was recorded from 4 different monkeys (111, 118, 109, and 97 neurons, respectively) while drifting sinusoidal gratings (8 orientations, each with 300 trials; 1 sec stimulus presentation for each trial) were presented

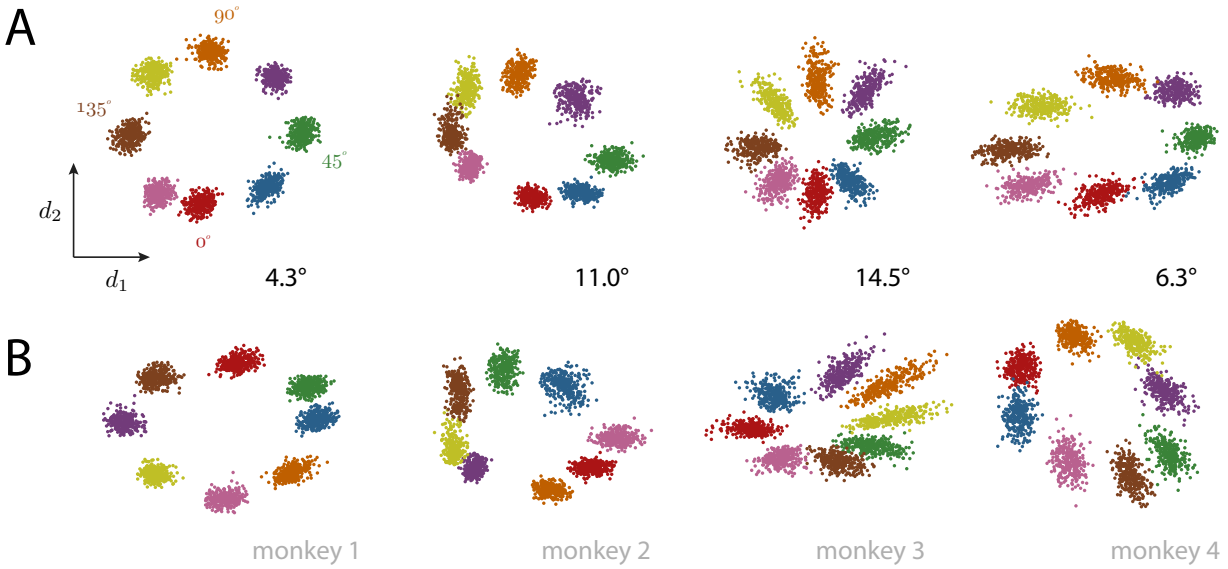


Figure 5.6: Top two DCA dimensions (A) of V1 population responses to gratings with 8 different orientations for 4 monkeys and (B) of the same V1 responses but with randomly permuted orientation labels. Mean principal angles were computed between dimensions in (A) and (B). Chance: ~83°.

(Zandvakili and Kohn, 2015). We applied DCA to the 4 datasets (taken in 1 sec bins). The ith trial for each monkey corresponded to the same orientation ($i = 1, \ldots, 2400$). Given no information about orientation, DCA was able to identify dimensions of each monkey's population activity that capture orientation (Fig. 5.6*A*), because these dimensions were the most strongly related across monkeys. These dimensions were approximately linearly related because the ordering of the clusters around the circle was the same across monkeys (i.e., the DCA dimensions could be rotated to align clusters based on color).

To make the task of aligning population activity more difficult, we introduced a nonlinear transformation by randomly permuting the orientation labels across trials for each monkey. Importantly, trials that previously had the same label still had the same label after permuting (i.e., we randomly permuted the colors across clusters). After permuting, the ith trial for one monkey might not have the same orientation as the ith trial for another monkey ($i = 1, \ldots, 2400$). As before, the color labels were not provided to DCA. DCA identified remarkably similar dimensions across monkeys (Fig. 5.6*B*) as those without permutation (Fig. 5.6*A*), quantified by the mean principal angle between the dimensions (chance angle: ~83°). Because the dimensions cannot simply be rotated to align the colors of the clusters, this shows that DCA is able to detect nonlinear relationships across datasets. These results illustrate how DCA can align neural activity.

5.8 Discussion

We proposed DCA, a dimensionality reduction method that combines the interpretability of linear projections with the ability to detect nonlinear interactions. The biggest advantage of DCA is its

applicability to a wide range of problems, including identifying dimensions in one or more sets of variables, with or without dependent variables. DCA is not regularized, unlike kernel-based methods (e.g., KDR, SPCA, and hsic-CCA), whose bandwidth parameters provide a form of regularization. However, fitting the bandwidth was computationally demanding, and when a heuristic was used to pre-select a kernel bandwidth, DCA was better able to capture nonlinear interactions. In addition, DCA may be directly regularized by the use of penalties, akin to sparse CCA (Witten and Tibshirani, 2009).

We optimized for $\{\mathbf{u}_1, \ldots, \mathbf{u}_K\}$ sequentially instead of directly optimizing for the orthonormal basis $U \in \mathbb{R}^{p \times K}$. The sequential approach, also used by other methods (Chang et al., 2013; Iaci et al., 2010; Mandal and Cichocki, 2013; Xia, 2008), has the advantages of many smaller optimization spaces rather that one large optimization space, and the removal of previously-identified dimensions that could otherwise mask other relevant dimensions (cf. Section 5.6.2). Directly optimizing for U no longer orders dimensions by relevance, making visualization and interpretation difficult. Still, directly optimizing for U can detect certain relationships between X and Y that would not be detected by the sequential approach (e.g., the continuous version of XOR: $Y = \text{sign}(x_1 x_2)$, where x_1 and x_2 are independent). Future work can extend DCA to directly optimize for the subspaces. The DCA source code for Matlab and Python can be found at `https://bit.ly/dca_code`.

Chapter 6

A slowly-varying arousal signal obstructs sensory information but is removed downstream

In Chapters 4 and 5, we proposed two new statistical methods to use for understanding population activity from brain areas far from the sensory periphery. Here, we apply these methods to understand population activity recorded from monkey V4. In particular, we study fluctuations in the V4 activity that can be considered noise, and how these fluctuations affect downstream decoding.

The search for neural correlates of behavior is at the heart of neuroscience. One start to this search was to compare a subject's performance on a task with the sensitivity of single neurons recorded during the same task (Newsome et al., 1989). The search expanded to looking at choice probabilities, which relate the fluctuations of a single neuron's activity to changes in the subject's decision across different repeats of the same stimulus (Britten et al., 1996; Nienborg and Cumming, 2006, 2009; Parker and Newsome, 1998). Recently, fluctuations shared across visual cortical neurons have been observed (Kohn et al., 2016; McGinley et al., 2015b), and many theoretical studies have asked how these internal signals in the visual cortex could help or hurt a subject's performance during a task (Averbeck et al., 2006; Moreno-Bote et al., 2014; Shadlen and Newsome, 1998; Zohary et al., 1994). For example, the internal signal of attention is thought to be "helpful", as a subject's ability to detect a change in a visual stimulus improves if the subject attends to the stimulus that is about to change (Desimone and Duncan, 1995). The neural correlates of attention include an increase in firing rate (Moore and Zirnsak, 2017; Snyder et al., 2016) and a decrease in mean noise correlation (Cohen and Maunsell, 2009; Mitchell et al., 2009), suggesting that attention increases the information content of visual cortical neurons (Shadlen and Newsome, 1998). On the other hand, the internal signal of adaptation can sometimes be "hurtful" to a subject's perception of a stimulus (Kohn et al., 2016), as a subject may report motion in a static stimulus shown immediately following a moving stimulus, known as the motion aftereffect (Anstis et al., 1998; Kanai and Verstraten, 2005). The neural correlate of adaptation includes a decrease in firing rate for consecutive repeats of the same stimulus and a shift in tuning curves (Kohn, 2007; Kohn and Movshon, 2004; Kohn et al., 2016). Thus, internal signals can be helpful or hurtful to perceptual readout of the stimulus.

Recent studies have employed statistical methods to identify fluctuations of activity of visual cortical neurons. These fluctuations account for a substantial amount of the covariability among neurons (Arandia-Romero et al., 2016; Ecker et al., 2014; Lin et al., 2015; Okun et al., 2015; Rabinowitz et al., 2015). Some have been attributed to anesthesia effects (Ecker et al., 2014; Goris et al., 2014), attention (Ecker et al., 2016; Rabinowitz et al., 2015), and arousal (McGinley et al., 2015a,b; Reimer et al., 2016), as well as to local connections of the neural circuit (Beaman et al., 2017; Goris et al., 2014; Gutnisky et al., 2017; Okun et al., 2015). Most of these fluctuations evolve on the time scale of hundreds of milliseconds to seconds (Ecker et al., 2014; McGinley et al., 2015b; Okun et al., 2015), although a small number have been reported to evolve on the time scale of minutes (Goris et al., 2014; Rabinowitz et al., 2015). It is currently unclear if the fluctuations arise from similar neural mechanisms and what role these fluctuations play in computation, if any.

The presence of these fluctuations in the visual cortex is puzzling. Consider a task in which the subject discriminates between two different stimuli at a certain location in visual space. Many tens of thousands of visual neurons are responsive to that region of visual space and are "read out" by a downstream area discriminating between the two stimuli. Fluctuations in the responses of these visual neurons may be read out as perceptual noise, impairing the subject's fidelity of discrimination. Many theoretical studies have explored the effects of noise on stimulus encoding (Averbeck et al., 2006; Ecker et al., 2011; Kanitscheider et al., 2015; Moreno-Bote et al., 2014; Shadlen and Newsome, 1998; Zohary et al., 1994). For example, a recent study found that only differential noise correlations (or noise aligned to the linear readout axis of a downstream area) can limit the amount of information of a population of neurons (Moreno-Bote et al., 2014). However, these studies typically make two assumptions. First, they assume that a downstream area only reads out the activity of the upstream population of visual neurons that process the stimulus input and has no access to activity from other brain areas. Second, they assume that this downstream area reads out the upstream population activity in a way that best discriminates between two arbitrary stimuli. The first assumption may be too conservative, as downstream areas provide rich feedback to sensory neurons and likely have access to this feedback (Bondy et al., 2018; Moore and Armstrong, 2003). The second assumption may not be conservative enough, as a downstream readout is likely informative about many different stimuli and not specialized to discriminate between two arbitrary stimuli (Ni et al., 2018). Thus, it is unclear how these assumptions hold in an empirical setting.

In this work, we identified a fluctuation in the activity of macaque V4 neurons on the order of tens of minutes and larger than changes in firing rate caused by attention. We found that this "slow drift" related to slow changes in criterion, suggesting that the slow drift is a cognitive signal of arousal (also called alertness, motivation, effort, or concentration). In addition, we found that two brain areas (V4 and prefrontal cortex) share the same slow drift. Thus, a likely source of the slow drift is the release of neuromodulators to large swaths of brain tissue to modulate arousal. We asked how this drift affected the ability of a readout of V4 population activity to discriminate between different stimuli. We found evidence that the slow drift resides along a linear axis in firing rate space (i.e., the slow drift axis) that is likely aligned to a downstream readout axis. For example, we found that the slow drift axis was aligned to axes along which the responses to natural images varied the most. In addition, we found that V4 activity along the slow drift axis predicted when false alarms would occur during a trial only when the slow drift

was removed. These results suggest that a possible mechanism to prevent the slow drift from interfering with the fidelity of downstream readout is for a downstream area to remove the slow drift from its readout. To reconcile the observations that the slow drift is removed but relates to slow changes in behavior, we propose a model in which the slow drift affects behavior via a pathway to the decision independent of perception. This work demonstrates the need for new theory about the information content of populations of visual neurons that accounts for the sources of neural fluctuations, and if these fluctuations are accessible by downstream readouts.

6.1 Results

We trained two adult, male rhesus macaque (macaca mulatta) monkeys to perform an orientation-change detection task (Fig. 6.1*A*). Briefly, on a given trial, the monkey was shown an initial 400 ms flash of two sample stimuli (drifting sinusoidal gratings), one in each visual field. The orientation angles of each pair of sample stimuli were randomly chosen to be either 45° (on the left visual hemifield) and 135° (on the right visual hemifield), or vice versa. After the initial flash, the subject remained fixated while observing consecutive flashes, which were interleaved with 300-500 ms blank screens. Each consecutive flash had a 30% (monkey 1) or 40% chance (monkey 2) of changing the orientation of one of the sample stimuli. For a 'correct' trial, the subject was required to make a saccade to the target whose orientation angle changed in order to receive a liquid reward. Other possible trials include 'miss' trials, in which the monkey does not saccade to the changed target, as well as 'false alarm' trials, in which the monkey incorrectly saccaded to an unchanged sample stimulus. The subject's ability to detect the changed target improved as the change in orientation angle increased (Fig. 6.1*B*, 'cued to changed target'). To probe the effects of spatial attention, we increased the chance of a changed target occuring in one of the visual hemifields to 80%. We alternated which visual hemifield was more likely to change its target in blocks of 100-200 trials, where the initial 5 trials of each block were used to cue the monkey to attend to the other visual hemifield. The subject employed spatial attention, as changes in orientation of uncued targets decreased performance (Fig. 6.1*B*, 'cued to unchanged target'). Results for the second monkey were similar (Supp. Fig. D.1). While the subject performed the task, we simultaneously recorded from a population of V4 neurons with a multi-electrode array (Blackrock Microsystems, Salt Lake City, UT).

We observed that the activity of many of the V4 neurons slowly drifted. For an example V4 neuron, its responses to the two sample stimuli (45° or 135°) drifted across the entire recording session (Fig. 6.1*C*, left plot). This slow drift likely impaired the neuron's ability to discriminate between the two stimuli (Fig. 6.1*C*, right plot, cross-validated decoding accuracy of 54%, where chance is 50%). To assess how much this slow drift impaired the fidelity of the neuron's stimulus encoding, we estimated the slow drift with kernel smoothing and subtracted it from the responses (Fig. 6.1*C*, right plot). The neuron's ability to discriminate between stimuli increased by ~10% (Fig. 6.1*C*, right histogram, $P(\text{correct}) = 68\%$). This suggests that one possible source of trial-to-trial variability of a neuron's activity is a slow drift that persists over an entire recording session, and that this drift can potentially be detrimental to stimulus encoding. Because we found that the slow drift was present in the activity of many neurons, we next assess the drift at the level of the population.

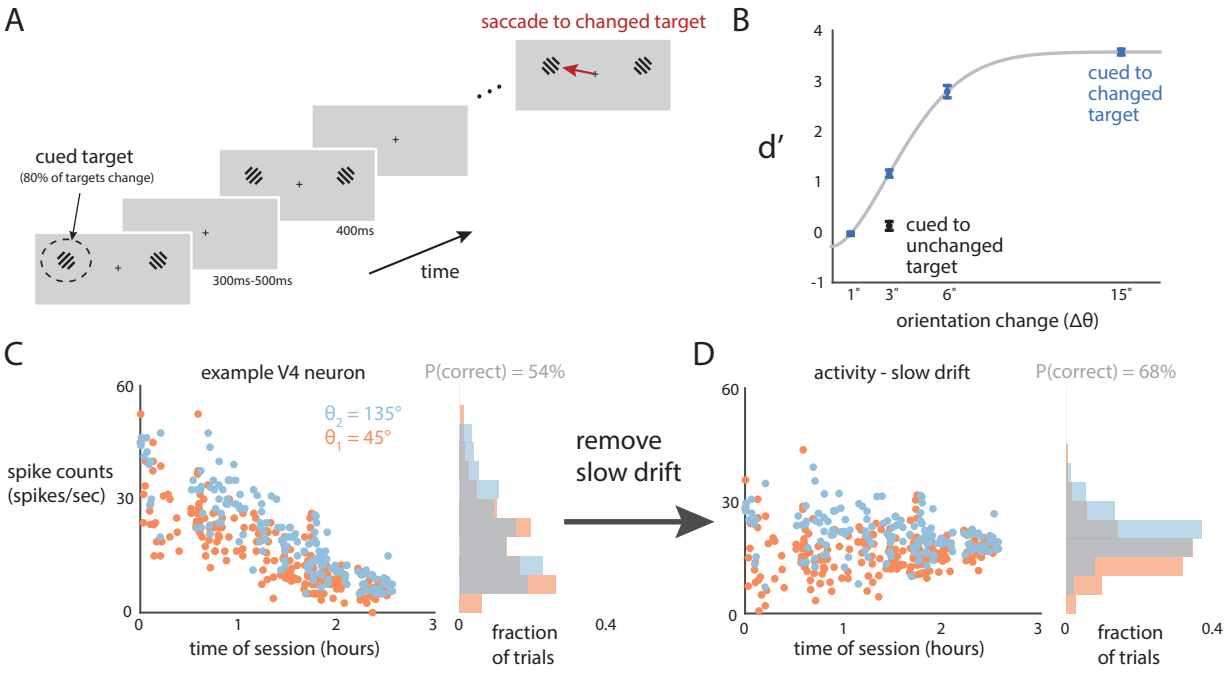


Figure 6.1: A source of noise in a neuron's responses. *A*. Illustration of attention discrimination task. The subject fixated during an initial 400 ms flash of two sample orientation gratings, where one target was cued to change 80% of the block's trials. After a 300-500 ms blank screen, another 400 ms flash of two targets appear. This continues until either the subject breaks fixation or one target changes in orientation. *B*. A subject's performance (measured with d') increased with larger orientation changes of the target stimulus (blue dots). The subject's performance decreased for trials in which the cued target did not change ($\Delta\theta = 3°$, black dot below blue dot). *C*. Left plot: Responses of example V4 neuron to two different orientations plotted against the time during the session. Right histograms: Cross-validated accuracy decoding the neuron's responses to the two different orientations. *D*. Left plot: Slow drift is removed from the raw responses, same data as in left plot. Right histograms: Cross-validated decoding accuracy increases for the neuron's responses without the slow drift.

6.1.1 V4 neurons slowly drift together.

We first asked if neurons in the population slowly drift their activity together. To assess this, we analyzed how the activity of different neurons drifts across a recording session. Because the slow drift was unrelated to the stimulus, we considered the residual activity (raw spike counts minus the trial-averaged stimulus responses). For an example session, we found one neuron whose residual activity slowly increased over the session (Fig. 6.2*A*, 'neuron 4'), one neuron whose residual activity slowly decreased (Fig. 6.2*A*, 'neuron 11'), and one neuron whose residual activity did not drift at all (Fig. 6.2*A*, 'neuron 5'). Thus, the slow drift affects neurons differently, and simply averaging the activity across these three neurons would not reveal the slow drift. Instead, we leveraged the fact that some neurons may drift their activity together, and took a linear combination of the neurons' activity (Fig. 6.2*B*, gray dots), which we then smoothed to characterize the slow drift (Fig. 6.2*B*, black line). The slow drift was not an attention signal, as the slow drift did not covary with the timing of the cued blocks (Fig. 6.2*B*, black line does not covary with blue line). The weights of the linear combination revealed that the activity of some neurons drifted together (Fig. 6.2*C*, positive weights, e.g., neurons 10 and 11), the activity of some neurons drifted opposite to one another (Fig. 6.2*C*, positive and negative weights, e.g., neurons 4 and 11), and the activity of some neurons did not slowly drift at all (Fig. 6.2*C*, weights close to zero, e.g., neuron 5).

To estimate the slow drift, we first applied principal component analysis to smoothed residual spike counts across the session, where the smoothing window was ~20 minutes. We define the *slow drift axis* as the normalized projection vector weights of the top principal component. For the example session, the weights of the slow drift axis are shown in Figure 6.2*C*. Across all sessions, the top principal component was dominant over the other principal components, and captured 70% of the smoothed residual spike count variance on average. (Supp. Fig. D.3). We then projected the residual activity along the slow drift axis (e.g., Fig. 6.2*B*, gray dots), and estimated the slow drift with kernel smoothing with a 9 minute bandwidth (e.g., Fig. 6.2*B*, black line). We further performed control analyses to rule out the potential alternative interpretation that the slow drift resulted from gradual changes in the distances between recording electrodes and neurons during the session (Supp. Fig. D.4).

We characterized three aspects of the slow drift. These included 1) the prominence of the slow drift relative to other co-fluctations among neurons, 2) the time scale of the slow drift, and 3) if the slow drift was a common gain to all neurons.

We first compared the prominence of the slow drift relative to the most prominent axis along which lied the most trial-to-trial shared variability. With this comparison, we can assess how prominent the slow drift is relative to other internal signals, such as attention and adaptation. To measure the prominence of the slow drift, we computed a ratio of the variance of the slow drift divided by the shared variance of the residual activity along the top axis that captures the most shared variance (Fig. 6.2*D*). We identified this top axis of the shared variance by applying factor analysis to the residual activity (see Methods). The shared variance represents co-fluctuations of the residual activity, including those arising from the slow drift, attention, and adaptation, but does not include Poisson-like variance that is private to each neuron. Because the denominator may include the slow drift variance, the ratio is between 0 and 1. A ratio close to 1 indicates that the slow drift is one of the most prominent signals, while a ratio close to 0 indicates that

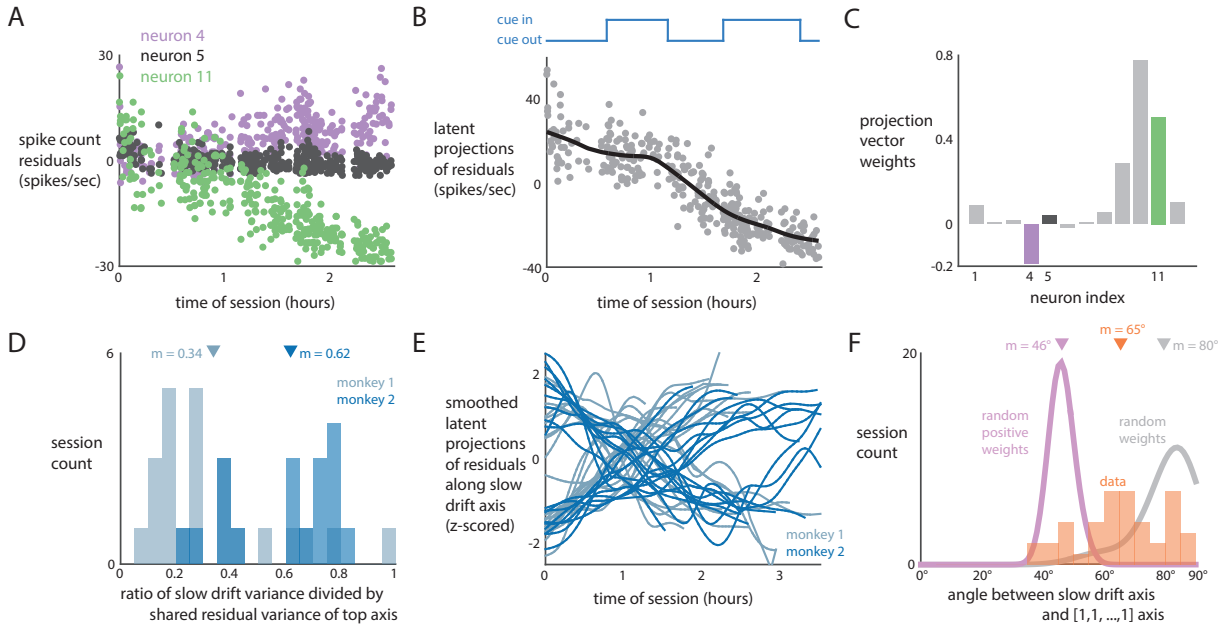


Figure 6.2: The activity of V4 neurons is modulated by a slow drift. *A*. The residual activity across one recording session of three example neurons. Each dot is the 400 ms-binned residual activity for one stimulus flash. Residuals are raw spike counts minus mean spike counts across repetitions of the same stimulus. For this plot, residual activity was shifted to visualize the drifts in activity. One neuron's activity slowly increased (neuron 4, purple), one neuron's activity remained constant (neuron 5, black), and one neuron's activity slowly decreased (neuron 11, green). *B*. A linear combination of the residual activity (i.e. latent projections) across the session of all recorded neurons. Same session as in *A*. Each gray dot is a projection of the residual spike count vector (with length equal to the number of neurons) for one stimulus flash. Projections were kernel smoothed to estimate the slow drift (black line). The slow drift did not match the cued blocks of trials in the task (blue line). *C*. The normalized weights of the linear combination used to take the projections in *B*. Neurons 4, 5, and 11 correspond to the neurons in *A*. The activity of a neuron whose weight is far from zero strongly drifts. *D*. Histogram of the ratio of the variance of the slow drift divided by the shared variance of the residual acitivity that lied along the top axis that captured the most shared variance. A ratio of 1 indicates that the slow drift is a prominent co-fluctuation of the population activity. *E*. Smoothed latent projections of residual activity for each session (one line corresponds to black line in *B*). Each smoothed trace is z-scored for each session. *F*. Histogram of angles between slow drift axis (e.g., the vector of weights in *C*) and an axis of weights of equal value (e.g., a normalized vector of $[1, 1, \ldots, 1]$. An angle close to 0°indicates that the slow drift axis has weights of similar magnitude and sign. An angle close to 90°indicates that the slow drift axis has weights of different magnitudes and signs. For reference, the angles between the $[1, 1, \ldots, 1]$ axis and axes whose weights have random magnitudes but the same sign (purple) as well as axes whose weights have random magnitudes and random signs (gray) are shown.

other fluctuations are larger. We found that the ratio of the slow drift differed across sessions and animals (Fig. 6.2*D*, mean ratios: 0.34 for monkey 1 and 0.62 for monkey 2), suggesting that the prominence of the slow drift may vary across sessions.

Next, we examined the time course of the slow drift along the slow drift axis (e.g., Fig. 6.2*B*, black line) for many sessions across both monkeys (Fig. 6.2*E*). Because the direction of the slow drift axis is identifiable up to a 180°rotation in firing rate space (i.e., a sign change), we chose a common reference frame across sessions based on the sample stimuli of the experiment, which were the same across all sessions (see Methods). All sessions showed slow drift on a time scale much longer than the 9 minute kernel bandwidth used for smoothing (Fig. 6.2*E*). To quantify the time scale of the slow drift, we performed cross-validated kernel smoothing and defined the time scale as the largest kernel bandwidth that retained 75% of the top performance. The time scale was on the order of 30 minutes for each subject (Supp. Fig. D.5*A*, monkey 1: 34 minutes, monkey 2: 41 minutes). A similar time scale was found with an auto-correlation analysis (Supp. Fig. D.5*B*).

Finally, we asked if the slow drift acted as a common gain factor, in which the activity of the neurons to co-fluctuate up and down together (e.g., all pairs of neurons have positive noise correlations). Such multiplicative fluctuations have been reported in mice (McGinley et al., 2015b) and macaque monkeys (Arandia-Romero et al., 2016; Ecker et al., 2014; Goris et al., 2014; Lin et al., 2015; Okun et al., 2015; Rabinowitz et al., 2015), and are thought to be caused by changing brain state (e.g., anesthesia or attention) (Kohn et al., 2016). One feature of multiplicative fluctuations is that the sign of the identified projection vector weight for each neuron is the same (e.g., all weights in Fig. 6.2*C* would have the same sign). We tested if this were the case for the slow drift by computing the angle in firing rate space between the slow drift axis and the axis in the $[1, 1, \ldots, 1]$ direction. An angle close to 0° indicates that the slow drift axis has projection weights with the same sign and value. An angle close to 90° indicates that the slow drift axis has both positive and negative projection weights (e.g., some pairs of neurons have negative noise correlations). For reference, we computed two chance distributions of the angles between the $[1, 1, \ldots, 1]$ axis and randomly-drawn vectors. The vectors of the first chance distribution had weights generated from a standard Gaussian but required to have the same sign (Fig. 6.2*F*, purple). The vectors of the second chance distribution had weights generated from a standard Gaussian with no restriction on their signs (Fig. 6.2*F*, gray). We found that the slow drift axis had an angle between the two distributions (Fig. 6.2F, orange), suggesting that the slow drift axis has both positive and negative weights. In further support of this, we found that when we required the weight with the largest magnitude to be positive, the minimum weight for the same session was almost always negative (Supp. Fig. D.6*A*). Roughly three-fourths of the neurons had slow drift present in their activity (Supp. Fig. D.6). These results suggest that the slow drift likely does not arise purely from a common gain factor. Instead, the slow drift could be additive or multiplicative with positive and negative gains.

The presence of the slow drift in the visual cortex raises two important questions. First, what is the source of the slow drift? Second, how may a downstream area overcome the slow drift to faithfully decode stimulus information, if at all? We begin to answer these questions in the following sections.

6.1.2 The slow drift relates to slow changes in behavior.

To identify the source of the slow drift, we first sought to link the slow drift to behavior. We can then identify possible neural mechanisms that govern the same behavior, and thus likely govern the slow drift. Specifically, we asked if any behavioral variables slowly changed across the session, and if so, if these changes covaried with the slow drift. We first focused on the behavioral variable of hit rate, which is computed by taking the number of correct saccades to a changed target divided by the total number of times the target stimulus changed. We computed a running estimate of the hit rate throughout a session with 30 minute windows shifted every 6 minutes (see Methods, results did not change with different time windows). We found that the hit rate varied over the session (Fig. 6.3*A*, teal lines). We then plotted the V4 slow drift (estimated with the same time bin as hit rate) alongside the hit rate, and found that both covaried over time (Fig. 6.3*A*, teal and black lines). To assess whether the hit rate and the slow drift were related, we made two comparisons. The first comparison asked if the magnitude of changes in the smoothed hit rate covaried with the size of the slow drift. In other words, if the fluctuations of the slow drift were larger for some sessions, then the hit rate should also have varied more for those same sessions, assuming the slow drift and smoothed hit rate were related. We measured the size of the slow drift by taking the fraction of shared variance explained by the slow drift, and we computed the variance of the hit rate over the entire session. We found these quantities to be correlated (Fig. 6.3*B*, $\rho = 0.39$).

We next asked to what extent the slow drift and hit rate covary over time. In other words, the slow drift and hit rate should fluctuate together, assuming the slow drift and hit rate are related. We found this to be the case (Fig. 6.3*C*, $\rho = 0.18$), and significantly above chance ($p < 0.02$). As mentioned previously, we alignedthe slow drift axes across sessions based on the mean responses to the two sample stimuli along the slow drift axis. After this alignment, the sign of the slow drift axis can be flipped only for all sessions collectively. We chose the direction of the slow drift axis to have a positive mean correlation with the hit rate. We could have equivalently flipped the signs of the slow drift for all sessions together such that the mean correlation had the same magnitude but was negative. In this way, we established an absolute reference frame across sessions for the slow drift that is independent of the strength of the relationship between the slow drift and behavior (i.e., the magnitude of the correlations).

In addition to hit rate, we performed the same comparisons between the slow drift and four other behavioral variables: false alarm rate (the number of incorrect saccades when the sample stimulus did not change divided by the total number of sample stimuli shown), trial length (i.e., the number of flashes observed by the monkey per trial), pupil diameter, and reaction time to make a saccade from stimulus onset. Running estimates of these behavioral variables used the same time windows as those for hit rate (see Methods). Similar to the results for hit rate, we found that the size of the slow drift covaried with the magnitude of slow changes in the behavioral variables across sessions (Fig. 6.3*D*). Within a session, we found that the slow drift and each behavioral variable covaried over time (Fig. 6.3*E*, magnitudes of correlations significantly above 0). Individual monkeys held the population trend (Supp. Fig. D.7). Importantly, the procedure in which we identified the slow drift had no access to the behavioral variables, indicating that the slow drift was a prominent brain signal. We also observed that slow fluctuations along the $[1, 1, \ldots, 1]$ axis were not coupled with behavior and could not explain these effects (Supp. Fig. D.8), suggesting the direction of the slow drift axis in firing rate space is important and not

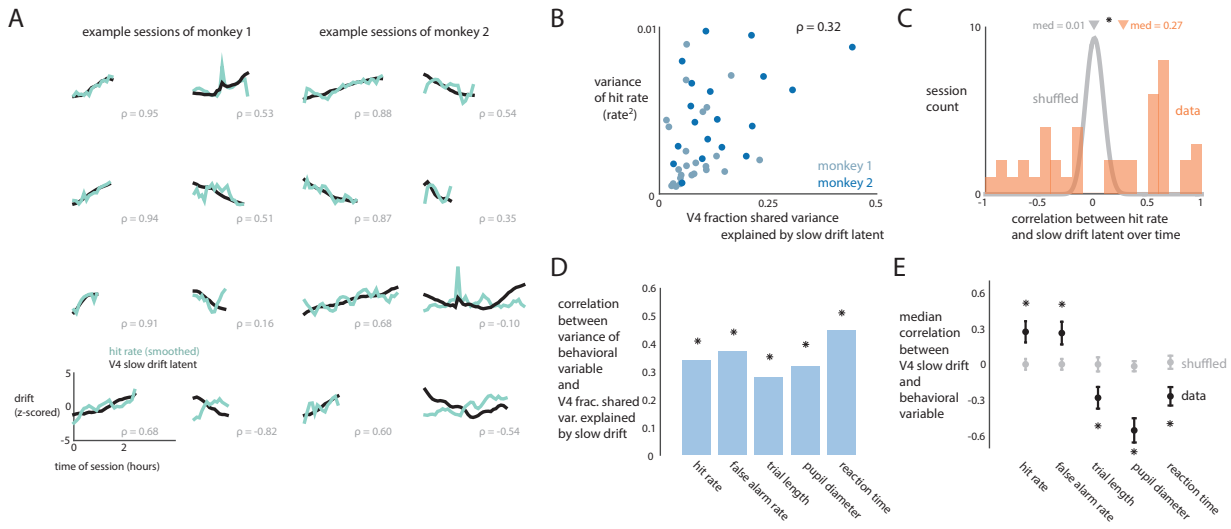


Figure 6.3: The slow drift covaried with slow changes in behavioral variables. *A*. For example sessions, the hit rate across time (teal) covaried with the V4 slow drift (black), z-scored for each session. *B*. To assess if the size of the changes in hit rate covary with the size of the slow drift across sessions, we computed the correlation between the variance of the smoothed hit rate and the fraction of shared variance of the V4 activity explained by the slow drift. Each dot represents one session. *C*. Correlations between the smoothed hit rate and the slow drift across sessions ('data', orange). As reference, a distribution of correlations between hit rates and slow drifts shuffled across sessions is plotted ('shuffled', gray). Median values ('med') are shown as triangles. *D*. The correlation between the variance of smoothed behavior variables and the fraction of shared variance explained by the slow drift. The first bar ('hit rate') corresponds to the same data as in *B*. *E*. Median correlations between the V4 slow drift and smoothed behavioral variables over time ('data', black). These correlations were significantly greater than the median correlations between behavioral variables and slow drifts shuffled across sessions ('shuffled', gray). Error bars denote 1 s.e.m. The first black dot ('hit rate') corresponds to the same data as in *C*.

simply aligned to the population firing rate averaged over neurons.

Because the slow drift was correlated with each behavioral variable, it suggests that the behavioral variables themselves covaried together The pattern of correlations matched with what we would expect if the slow drift was related to arousal. For example, the monkey may have been more or less hesitant to saccade (i.e., a change in decision criterion) during different time periods of the session. This change in criterion was reflected by the hit rate and false alarm rate going up and down together (Fig. 6.3*E*, positive mean correlations for hit rate and false alarm rate). Consistent with this interpretation, the slow drift covaried over time with criterion ($\rho = -0.21$, $p < 0.002$, difference of medians) and weakly correlated with sensitivity ($\rho = 0.11$, $p < 0.002$, difference of medians). As the hit and false alarm rate increased, the number of flashes within a trial (i.e., trial length) decreased (Fig. 6.3*D*, negative mean correlation for trial length). We also found that the pupil diameter was negatively correlated with hit rate and false alarm rate, suggesting that for a period of lower criterion, the pupil was constricted. This agrees with other studies that found a smaller pupil diameter correlates with disengagement of the task, albeit on a much smaller time scale (Aston-Jones and Cohen, 2005; Ebitz and Moore, 2017; Joshi et al., 2016; McGinley et al., 2015a,b; Reimer et al., 2014, 2016; Vinck et al., 2015). Finally, if the animal had a lower criterion, less time would be taken to integrate sensory evidence before executing saccades, resulting in faster response times (Fig. 6.3*D*, negative mean correlation for reaction time). We confirmed that this pattern of correlations was the most prominent pattern of co-fluctuations among the behavioral variables (Supp. Fig. D.9). This pattern indicates that gradual changes in criterion occur, consistent with slow changes in arousal. Because the slow drift is also correlated with these changes in criterion, it suggests that the slow drift is an arousal signal.

6.1.3 V4 and PFC neurons share the same slow drift.

In the previous section, we found that the slow drift related to gradual shifts in criterion throughout the session, potentially due to arousal. This suggests that one possible source for the slow drift is the release of neuromodulators, which have been linked to other neural fluctuations (McGinley et al., 2015b; Vinck et al., 2015). One candidate brain area that is a potential source of the neuromodulator is the locus coeruleus (LC). The activity in the locus coeruleus has been related to many behavior variables, including reaction time and pupil diameter, suggesting that the LC modulates arousal (Aston-Jones and Cohen, 2005; McGinley et al., 2015b). The LC releases the neuromodulator epinephrine on the time scale of the slow drift throughout large swaths of brain tissue (Aston-Jones and Cohen, 2005; Joshi et al., 2016). This suggests that if we recorded in another brain area, we should observe the same slow drift. In the same set of experiments, we simultaneously recorded in V4 and in dorsolateral prefrontal cortex (PFC) with a multi-electrode array in each brain area. The PFC is a central hub of the brain, with links to higher cognitive functions, such as working memory (Curtis and D'Esposito, 2003), multi-sensory integration (Raposo et al., 2014), and decision-making (Mante et al., 2013). Anatomically, PFC is interconnected with many different brain areas, including the LC (McGinley et al., 2015b; Ungerleider et al., 2007). Thus, a brain-wide release of neuromodulator would likely affect PFC activity. To test for this, we asked if V4 and PFC share the same slow drift. We applied the same spike waveform criteria to the PFC neurons as we applied to the V4 neurons (Supp. Fig. D.10).

We identified the slow drift of the PFC population activity in the same manner as for the

V4 slow drift, namely by applying PCA to smoothed spike counts (smoothed in ~20 min bins) and performing kernel smoothing on the PFC spike counts projected onto the slow drift axis. Because the PFC slow drift axes were also identified up to a 180°rotation in firing rate space, we flipped the sign of the PFC slow drift such that the correlation between the V4 and PFC slow drift was positive for each session. Visually, the slow drifts of V4 and PFC were similar across time (Fig. 6.4*A*). Such highly-correlated projections are expected when applying inter-area dimensionality reduction methods, such as canonical correlation analysis (Semedo et al., 2014) or distance covariance analysis (Cowley et al., 2017a), but here we apply PCA separately to V4 and PFC with no access to the activity of the other brain area.

We performed two comparisons to verify that the V4 and PFC neurons slowly drifted together. First, we asked if the size of the V4 slow drift was larger for some sessions, was the size of the PFC slow drift also larger for the same sessions. To address this question, we computed the fraction of shared variance explained by the slow drift for each brain area separately (Fig. 6.4*B*, same metric for the V4 slow drift as in Fig. 6.3*B*), and found a significant positive correlation between them ($\rho = 0.57$, $p < 0.002$). Results were consistent for both monkeys ($\rho = 0.61$, $\rho = 0.56$). Thus, when the V4 slow drift is large, the PFC slow drift is also large for the same session.

The second comparison assessed the extent to which the V4 and PFC slow drifts covaried. We found that the correlation over time between the V4 and PFC slow drift was high (Fig. 6.4*D*, orange histogram, median $\rho = 0.96$, $p < 0.002$). Results were consistent for both monkeys (median $\rho = 0.96$, median $\rho = 0.93$). The high correlations were not expected if we shuffled the PFC spike counts across time (the V4 activity remained unshuffled) and re-identified the PFC slow drift (Fig. 6.4*D*, gray histogram, median $\rho = 0.50$). A more conservative null hypothesis is that a slow drift exists in PFC activity but is uncorrelated with the V4 slow drift. To test for this, we computed the median correlation expected by this hypothesis (median $\rho = 0.87$) by shuffling the PFC slow drifts across sessions and re-computing the correlations. The correlations in the real data were significantly higher than for the shuffled data across sessions ($p < 0.002$). This suggests that V4 and PFC activity drift slowly together over time.

Taken together, these results indicate that PFC activity contains the same slow drift as that of the V4 activity. Because PFC receives input from V4 (Ungerleider et al., 2007), one possible explanation is that PFC may read out the slow drift from the V4 neurons. However, a more parsimonious explanation is that the slow drift is present in many brain areas, likely arising from neuromodulator release. The V4 slow drift was not related to slow fluctuations of PFC activity along the $[1, 1, \ldots, 1]$ axis (Supp. Fig. D.11), consistent with the finding that neuromodulators affect neurons differently (Totah et al., 2017) and that the slow drift is not a common gain to the neurons.

6.1.4 Three hypotheses about how the downstream readout deals with the slow drift.

The previous sections establish that the slow drift is likely a prominent, brain-wide arousal signal. However, the presence of the slow drift in V4 is puzzling, because the slow drift evolves on a time scale not useful to provide feedback about the stimulus. Even worse, the slow drift could

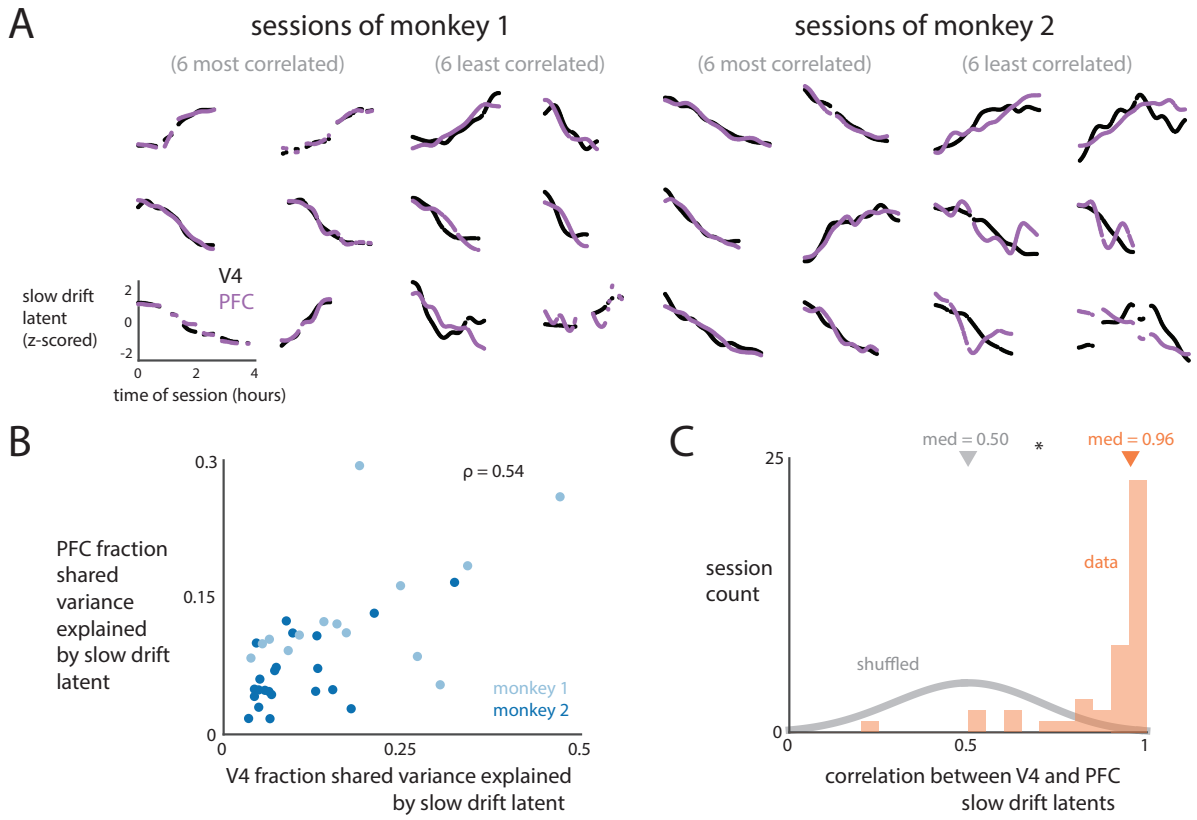


Figure 6.4: V4 and PFC neurons share the same slow drift. *A*. The slow drift of PFC neurons (purple) overlaid on top of the slow drift of V4 neurons (black) for the 6 most correlated and 6 least correlated sessions for each monkey. The slow drift is z-scored for each session. *B*. To assess if the size of the slow drift goes up and down both in V4 and PFC across sessions, the fraction of shared variances captured by the V4 and PFC slow drift are plotted against one another. Each dot represents one session. *C*. Absolute correlations between the V4 slow drift (black lines in *A*) and the PFC slow drift (purple lines in *A*) for all sessions ('data', orange). For reference, the distribution of absolute correlations between V4 and PFC slow drifts for which the PFC spike counts were shuffled across time is plotted ('shuffled', gray). Median values ('med') are shown as triangles.

potentially corrupt the stimulus information encoded by V4. For example, the responses of the V4 neurons to the same stimulus presented in the first and last trial can be different because of the slow drift. However, it is unlikely that the slow drift primarily influences behavior as perceptual noise. First, the brainstem nuclei known for modulating arousal can directly affect decision and motor circuits (Aston-Jones and Cohen, 2005; Joshi et al., 2016; McGinley et al., 2015b), making it unnecessary for arousal to affect decisions as perceptual noise. Second, the experimental design required a high level of performance from the animals, leading to a large number of false alarm trials observed during a session (42% of trials ended in a false alarm for monkey 1, 48% for monkey 2). These false alarms more likely arises from changes in decision criterion (i.e., arousal or motivation) than from perceptual noise. This is supported by our finding that hit rate and false alarm rate covaried throughout the session (Fig. 6.3*E*), a signature of criterion shift (Luo and Maunsell, 2015). Finally, it is unclear how the slow drift, if perceptual noise, could influence changes in pupil diameter (Fig. 6.3*E*) through the same circuit pathway in which the slow drift influences the perceptual readout.

This suggests that the slow drift, as a brain-wide arousal signal, primarily influences the decision to saccade through circuit pathways unrelated to perception. Nonetheless, this signal affects the activity of sensory neurons. How can the system prevent this signal from corrupting perception? Here, we outline three hypotheses in which the slow drift does not cause perceptual noise.

First, consider an illustrative example in which two neurons respond to two different stimuli (Fig. 6.5*A*). Repeated presentations of the same stimulus reveal a covariance structure of the responses (Fig. 6.5*A*, orange and blue ellipses). A downstream neuron may readout the two neurons' responses through a readout axis (Fig. 6.5*A*, purple dashed line and bottom histograms), which linearly combines the activity of both neurons. Because of the covariance structure, a response that lies within the overlap of the two response distributions may be incorrectly decoded (Fig. 6.5*A*, red X). Given this framework, we can ask how the slow drift changes the size and direction of the covariance ellipses. We begin by considering three hypotheses that follow from the possibility that a downstream area can disregard the slow drift from its readout. We then consider the hypothesis that follows from the possibility that the slow drift causes perceptual noise.

For all hypotheses, the V4 neurons receive information about a presented stimulus in the form of input from an upstream brain area, such as V1 or V2, and this input is likely corrupted (Fig. 6.5*B*, "stimulus input + noise"). The slow drift is also a common input of the V4 neurons (Fig. 6.5*B*, "slow drift"). We hypothesized three situations in which a downstream readout of V4 neurons can have high fidelity even in the presence of the slow drift. Hypothesis 1 predicts that the slow drift is small relative to the distances between responses from different stimuli (Fig. 6.5*B*). A downstream area would read out from the V4 neurons unhindered by the slow drift to determine if the stimulus changed with high fidelity, which can be relayed to further downstream areas that make the saccade decision. Hypothesis 1 predicts that the slow drift is small relative to the effects on V4 responses by other mechanisms, such as attention, as well as to the stimulus variance (i.e., the distance between the mean responses to each stimulus).

Hypothesis 2 predicts that the slow drift is large relative to the stimulus variance, but that the slow drift contributes to the covariance structure in a way that does not affect the fidelity of downstream readout (Fig. 6.5*C*). In other words, the slow drift axis is not aligned to the

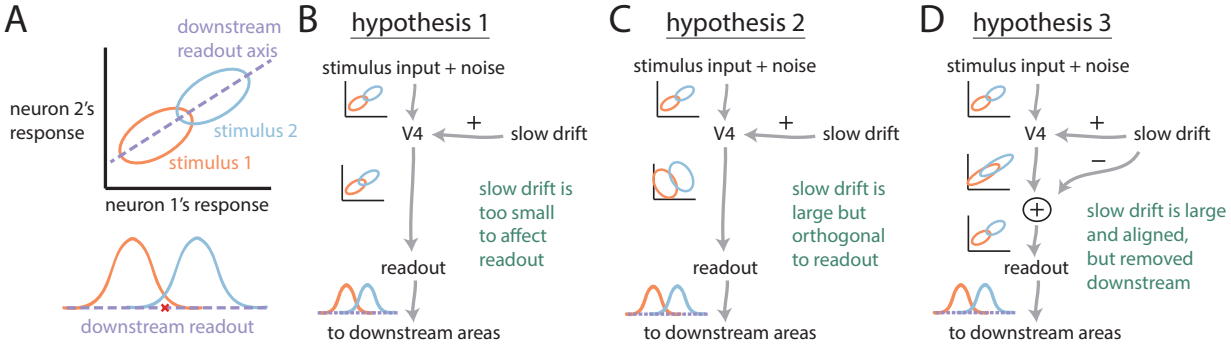


Figure 6.5: Illustrative hypotheses of how a downstream area can disregard the slow drift to achieve a high-fidelity readout. *A*. Top: Responses of two neurons to two different stimuli. The ellipses represent the covariances of responses. A downstream area may read out the responses of the neurons by taking a linear combination (purple dashed line). Bottom: Histograms representing the distributions of responses along the downstream readout axis. A response to stimulus 1 in the overlap of the two distributions (e.g., red X) may be incorrectly read out as a response to stimulus 2. The amount of overlap between the distributions is dependent on the mean responses to each stimulus (i.e., the center of the ellipses) as well as the covariance structure (e.g., the size and orientation of the ellipses). *B*. Hypothesis 1. Stimulus input and feedforward noise are input to the V4 neurons ('stimulus input + noise'). The slow drift is also an input, but the size of the slow drift is small and does not change the size or orientation of the covariance ellipses. Thus, the readout of V4 neurons can faithfully discriminate between stimuli. *C*. Hypothesis 2. The slow drift is large but orients the covariance ellipses in such a way as to not harm the fidelity of downstream readout. *D*. Hypothesis 3. The slow drift is large and orients the covariance ellipses to harm the fidelity of downstream readout (i.e., overlap between covariance ellipses, second response plot from the top). However, the downstream readout also has access to the slow drift, and can remove the slow drift from its readout (third response plot from the top) to faithfully discriminate between stimuli.

downstream readout axis, and thus the downstream readout is not aware of the slow drift. This hypothesis has been instrumental for theoretical studies about the helping or harmful presence of noise correlations (Averbeck et al., 2006; Moreno-Bote et al., 2014; Shadlen and Newsome, 1998). These studies find that correlated noise along the downstream readout axis (i.e., differential noise correlations) is the only noise that affects the fidelity of downstream readout. This hypothesis predicts that the slow drift does not contribute to differential correlations.

Finally, Hypothesis 3 predicts that the slow drift is large relative to the stimulus variance, and that the slow drift axis is aligned to a downstream readout axis. Importantly, the downstream readout also has direct access to the slow drift, which can then be removed from the readout of the V4 neurons. Under this hypothesis, the slow drift hinders the ability to decode the stimulus information of the V4 neurons. It also predicts that downstream brain areas have access to the slow drift.

In the following sections, we test each hypothesis, and find most evidence in support of Hypothesis 3. We focus our analyses on the size of the slow drift, whether or not the slow drift axis is aligned to a downstream readout axis, and how well the V4 activity can predict false alarms with or without the slow drift. We also provide evidence that rules out the possibility that the slow drift is read out from V4 as perceptual noise.

6.1.5 The slow drift is larger than spatial attention.

We first consider the prediction of Hypothesis 1 that the size of the slow drift is small relative to changes in V4 responses by other neural mechanisms, such as attention (Fig. 6.5*B*). The large effects of spatial attention on V4 responses have been extensively investigated (Cohen and Maunsell, 2009; Luo and Maunsell, 2015; Mitchell et al., 2009; Moore and Zirnsak, 2017). One of the most robust effects is that the evoked responses of V4 neurons increase when the subject attends within those neurons' receptive fields (Luo and Maunsell, 2015; Moore and Zirnsak, 2017; Reynolds et al., 2000), suggesting that the neurons are primed to better detect changes within their receptive fields (Reynolds and Chelazzi, 2004). We asked if the size of the slow drift was small relative to these changes in firing rate.

For one example session, we measured the variance of the slow drift over time (Fig. 6.6*A*, $\sigma^2_{\text{slow drift}}$). We compared this variance to the changes in firing rates between cue-in and cue-out blocks of trials (Fig. 6.6*B*, σ^2_{attn}). For a fair comparison, we identified the axis along which the responses differed the most between cue-in and cue-out blocks (defined as the attention axis), and computed σ^2_{attn} along the attention axis after removing the slow drift from the V4 activity (see Methods). For this example session, the size of the slow drift was 5 times larger than the effect size of attention ($\sigma^2_{\text{slow drift}}/\sigma^2_{\text{attn}} \approx 5$). Because the slow drift explains on average 30% of the total shared residual variance (Fig. 6.2*D*), this attention effect along the attention axis would explain ~6% of the total shared residual variance.

We computed the ratio $\sigma^2_{\text{slow drift}}/\sigma^2_{\text{attn}}$ for all sessions, and found that the slow drift was 2 to 6 times larger than the effect size of attention in our two animal subjects (Fig. 6.6*C*, medians: 1.8 and 6.6). This suggests that the slow drift is large relative to the effect of attention, at odds with the prediction of Hypothesis 1. It also suggests that the effect of the slow drift on V4 neurons is larger than that of attention, and that other effects of attention, such as noise correlations (Cohen and Kohn, 2011; Cohen and Maunsell, 2009; Mitchell et al., 2009), may be better estimated by

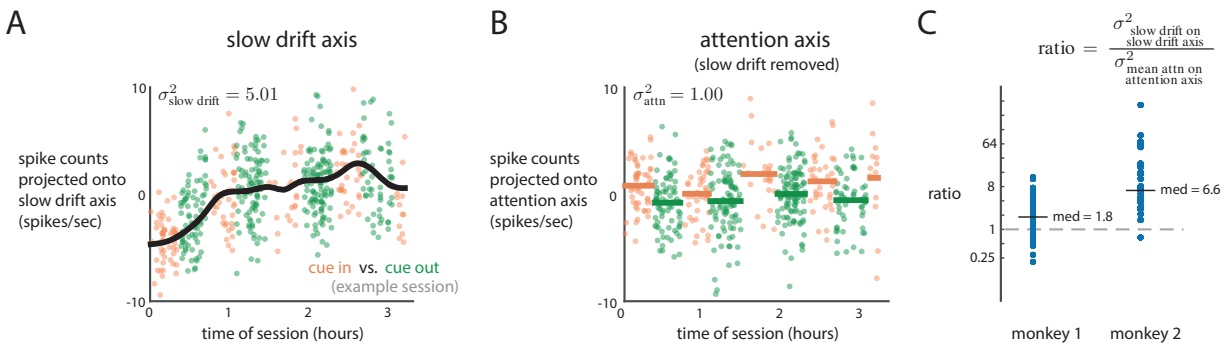


Figure 6.6: The size of the slow drift is larger than the effect of attention. *A*. For an example session, responses to each target stimulus (orange and green dots) along the slow drift axis had a slow drift (black line) over time. The slow drift was independent of how trials were cued to be within ('cue in', orange) or outside ('cue out', green) the recorded V4 neurons' receptive fields. The variance of the slow drift $\sigma^2_{\text{slow drift}}$ was computed over time. *B*. For the same example session, responses along an attention axis that captures the largest amount of difference between mean spike counts during cue in (orange) and cue out (green) blocks of trials are plotted over time of the session. Any fluctuations of the responses due to the slow drift were removed to avoid any effect of attention being masked by the slow drift. The effect of attention σ^2_{attn} was measured as the variance of the mean responses along the attention axis across the differently cued blocks (orange and green lines). *C*. Ratios $\sigma^2_{\text{slow drift}}/\sigma^2_{\text{attn}}$ across sessions for both monkeys.

accounting for the slow drift.

6.1.6 The slow drift axis is aligned with a downstream readout axis.

Although the size of the slow drift is large, a downstream area may orient its readout axis so as to be orthogonal to the slow drift axis (Fig. 6.5*C*, Hypothesis 2). In this way, the downstream neurons could disregard the slow drift and have a high-fidelity readout, regardless of how large the slow drift is. Thus, a key prediction by Hypothesis 2 is that the slow drift axis is not aligned with the downstream readout axis. This suggests that the responses along the slow drift axis should contain little information about the stimulus, as this information would be corrupted by the slow drift.

For one example session, we plotted the responses to the two target stimuli along the slow drift axis (Fig. 6.7*A*, left plot). While the differences between responses to the two stimuli were noticeable, these differences were small relative to the slow drift, resulting in an overlap between response distributions (Fig. 6.7*A*, left histograms). After subtracting the slow drift from the V4 population activity (Fig. 6.7*A*, right plot), we found a substantial increase in cross-validated decoding accuracy from 82% to 98% (Fig. 6.7*A*, right histograms). We confirmed this effect for both monkeys (Fig. 6.7*B*, red dots above black dots). This result is at odds with Hypothesis 1, which predicts that the slow drift is small relative to the differences of responses to different stimuli.

We also found that the decoding accuracy increased for randomly-chosen axes (Fig. 6.7*C*, red dots above black dots), suggesting that many possible readout axes were affected by the size of

the slow drift. These results are also at odds with the predictions of Hypothesis 1.

The finding that the responses along the slow drift axis encode stimulus information, and that this encoding improves when the slow drift is removed (Fig. 6.7*B*), also appears to be at odds with Hypothesis 2, which predicts that the slow drift axis is not aligned to a downstream readout axis (Fig. 6.5*C*). When we instead assumed that the downstream readout axis was randomly oriented to the slow drift axis, we still found that removing the slow drift increased decoding accuracy (Fig. 6.7*C*). However, it could be possible for a downstream area to read out an ideal axis that best decodes the stimulus information while disregarding the slow drift. In fact, it is expected that such an axis exists due to the large number of possible axes orthogonal to the slow drift axis (Kohn et al., 2016; Moreno-Bote et al., 2014). We found this to be the case. When we identified a decoding axis using the full population activity, decoding accuracies were much higher than those for the slow drift axis (~95% compared to ~65%), and removing the slow drift marginally increased decoding accuracy (Fig. 6.7*D*). Thus, the slow drift axis is not aligned with a decoding axis that decodes the two target stimuli.

Because the decoding axis is specific for the two arbitrarily-chosen stimuli in this experiment, it is unclear if a downstream area would specialize its readout for this particular axis (Ni et al., 2018). Instead, a downstream area likely reads out axes of the V4 activity that are informative about complex features present in a large number of stimuli (Roe et al., 2012; Yamins et al., 2014). Thus, to identify more realistic downstream readout axes, in a separate set of experiments we employed adaptive stimulus selection to choose a set of 2,000 natural images (out of a possible 10,000 candidate natural images) that elicited large and diverse responses of the V4 neurons (Cowley et al., 2017b). We presented these images randomly throughout a session, and computed the trial-averaged responses for each image (see Methods). We used PCA to identify the axes along which the trial-averaged responses vary the most, and defined the top axes as the stimulus-encoding axes. We plotted the responses along the first two stimulus-encoding axes, and overlaid the presented images over their corresponding responses (Fig. 6.7*E*). The responses along the top two stimulus-encoding axes appeared to encode complex features of the images, as we observed nearby responses encoding similar high-level features (Fig. 6.7*E*, images of blue backgrounds, eyes, cardinals, dogs, and arranged fruit). Because the stimulus-encoding axes represent the most prominent axes of V4 responses to a large number of natural stimuli, the stimulus-encoding axes are likely downstream readout axes.

We then tested the prediction of Hypothesis 2 that the slow drift axis is not aligned to a downstream readout axis. Theoretical studies predict that these axes are unlikely to be aligned, as differential correlations are presumed to be much smaller than other types of correlations (Kanitscheider et al., 2015; Kohn et al., 2016; Moreno-Bote et al., 2014). In addition, it is statistically unlikely that two vectors drawn randomly from a high-dimensional space are aligned (Cowley et al., 2013; Cunningham and Yu, 2014). Thus, we expected that the angle between the slow drift axis and each stimulus-encoding axis would be close to 90° (i.e., close to orthogonal). We first identified the slow drift axis, which revealed a slow drift similar to those we found previously (Fig. 6.7*F*). We then computed the fraction of the slow drift variance captured by each of the top 12 stimulus-encoding axes (Fig. 6.7*G*), and found that some of these fractions were above the distribution of fractions expected if the slow drift lied along a random axis (Fig. 6.7*G*, blue line below gray error bars). Interestingly, the slow drift axis was most aligned with the top stimulus-encoding axis. Unlike how global fluctuations may affect contrast encoding in V1

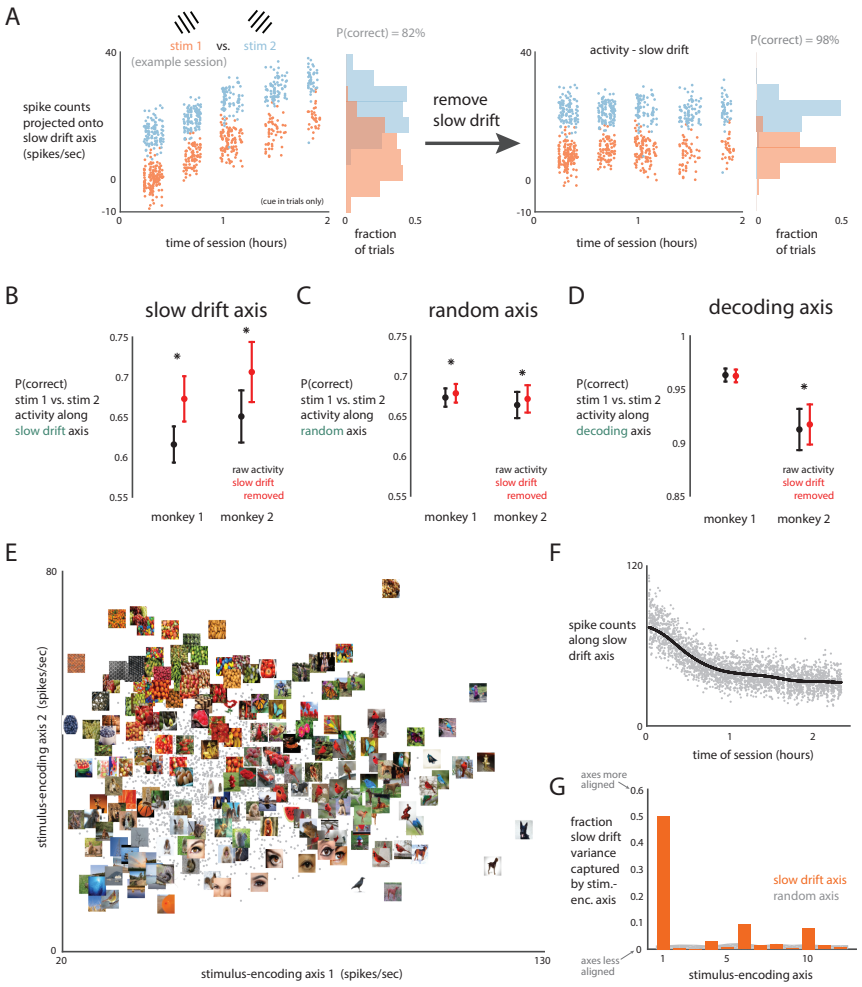


Figure 6.7: The slow drift axis is aligned with stimulus-encoding axes. *A*. For an example session, responses to two sinusoidal gratings with different orientations ('stim 1' and 'stim 2') along the slow drift axis are plotted over time (left plot). Each dot is the binned response (400 ms bins) to a stimulus flash. The gaps between responses are because only "cue in" trials in which the animal was cued to attend to the stimulus within the recorded V4 neurons' receptive fields are shown. If a downstream area read out these responses along the slow drift axis, the cross-validated decoding accuracy $P(\text{correct})$ would be 82% (left histograms). After removing the slow drift (right plot, same data as in left plot), the decoding accuracy increased to 98% (right histograms). *B*. Cross-validated decoding accuracy $P(\text{correct})$ for decoding the stimulus identity for responses along the slow drift axis across sessions and both monkeys. The mean decoding accuracies for the raw activity that contained the slow drift (black dots) and the same activity but with the slow drift removed (red dots) are shown. *C*. Same as in *B* except that the decoded responses were along a randomly-chosen axis in the high-dimensional firing rate space. *D*. Same as in *B* and *C* except that the decoded responses were along an axis identified by an SVM decoder given access to the full population activity. Note the scales in *D* are different from those of *B* and *C*. Error bars in *B*, *C*, and *D* are 1 sem. *E*. Trial-averaged responses to natural images (gray dots) along the first two stimulus-encoding axes (identified by PCA). Selected images were overlaid on top of their corresponding responses. (continued on next page...)

Figure 6.7: (...continued from previous page) Images were selected to highlight images with similar features eliciting similar responses (i.e., nearby one another along the stimulus-encoding axes). These images had features of arranged fruit, cardinals, squirrels, eyes, and blue backgrounds. *F*. Identified slow drift (black line) of the responses along the slow drift axis (gray dots) for the same session as in *E*. *G*. The fraction of the slow drift variance captured by each stimulus-encoding axis (orange) for the same session as in *E* and *F*. A higher fraction indicates that the stimulus-encoding and slow drift axis are more aligned. The top two stimulus-encoding axes are the same as in *E*, and the top 12 stimulus-encoding axes captured 62% of the neural variance due to changes in stimuli as well as 77% of the slow drift variance. For reference, the fraction of slow drift variance expected if the slow drift lied along a randomly-chosen axis is shown in gray.

(Lin et al., 2015), we found almost no relationship between the responses along the top stimulus-encoding axis and low-level image statistics, such as the luminance, contrast, and color energy of the images (Supp. Fig. D.12*A*). All of these results held for multiple sessions and another monkey for which adaptive stimulus selection was not used (Supp. Fig. D.12). We conclude from these results that the slow drift axis is likely aligned with a downstream readout axis, at odds with the predictions of Hypothesis 2.

6.1.7 V4 activity with the slow drift removed better predicts false alarms.

Another important aspect of identifying a downstream readout axis is that the readout should be predictive of behavior within a trial. Hypothesis 2 predicts that responses along the slow drift axis are not predictive of within-trial behavior, because the slow drift would be subtracted off at each moment by a readout axis that was insensitive to it (Fig. 6.5*C*). Meanwhile, Hypothesis 3 predicts that responses along the slow drift axis are predictive of within-trial behavior, especially when the slow drift is removed from the V4 activity (Fig. 6.5*D*).

To test these predictions, we considered how well the V4 activity can predict when the monkey incorrectly saccades to a target stimulus that did not change (i.e., a false alarm). We considered only false alarm trials (Fig. 6.8*A*), and decoded the responses to the final flash (in which a saccade was made) versus the responses to the preceeding flash (in which no saccade was made). To control for saccade artifacts, we binned spike counts for both flashes in a time window that began at a small lag after stimulus onset and ended 20 ms before saccade onset of the final flash.

To assess how the slow drift affects decoding behavior, we only considered false alarm trials in which the subject incorrectly saccaded to a target that did not change. Within these trials, we decoded responses projected along the slow drift axis between the final stimulus flash (i.e., when a saccade was made) and the stimulus flash previous to the final stimulus flash. *B*. Cross-validated percent correct for decoding responses as outlined in *A*. The same activity was decoded with the slow drift (black dots) and without the slow drift (red dots).

While a primary source of false alarms might be changes in a subject's criterion level (Luo and Maunsell, 2015), some fraction of false alarms may also be due to perceptual noise. For these trials, the subject incorrectly perceived that a change in the target stimulus occurred. This perceptual change, while small, should be evident in changes of neural activity in a visual area, such as V4 (Cumming and Nienborg, 2016). We first decoded the raw spike counts along the slow

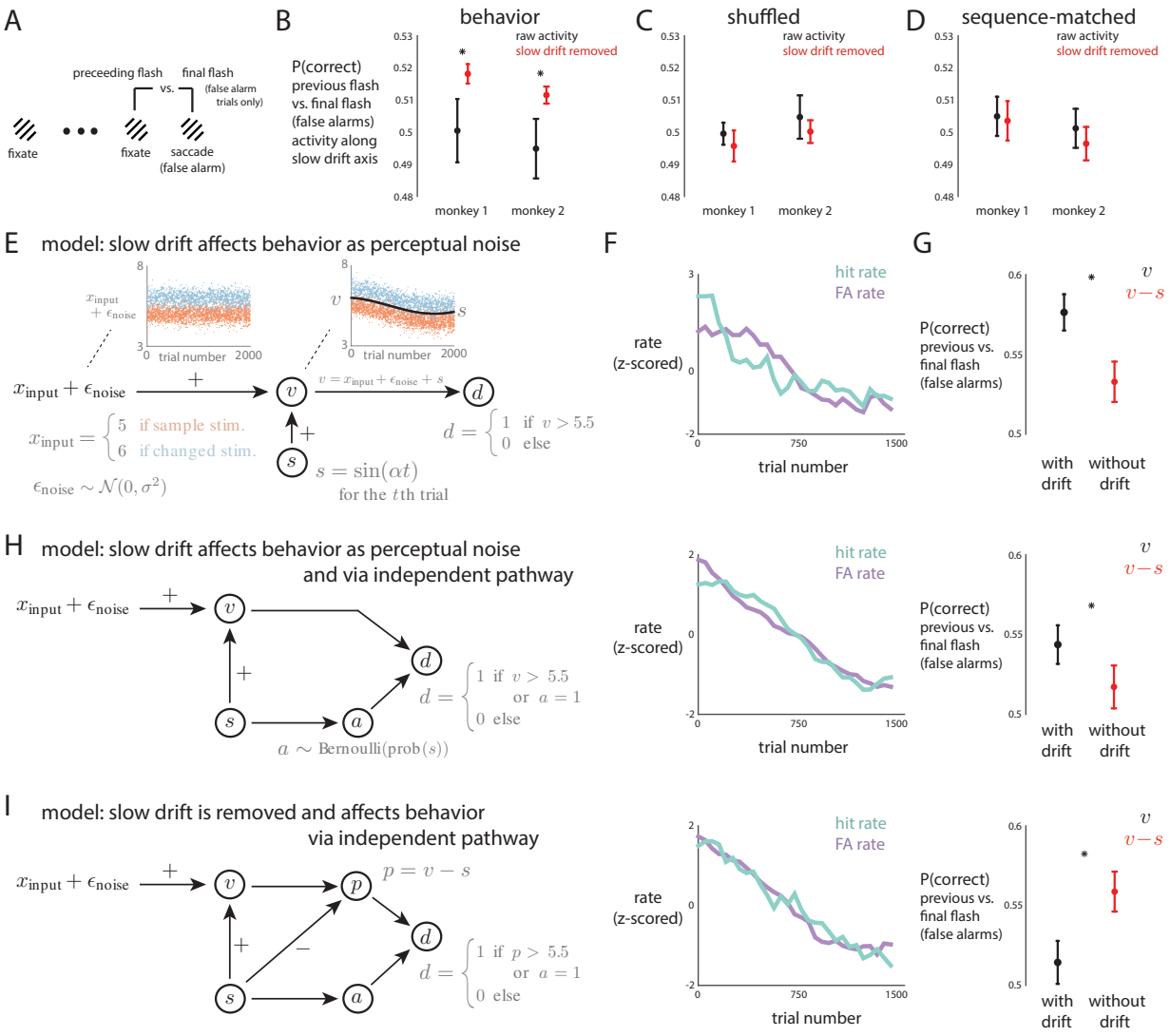


Figure 6.8: V4 activity predicts the occurrence of a false alarm within a trial only after the slow drift is removed. *A*. To assess how the slow drift affects the ability of V4 activity to predict behavior, only false alarm trials in which the monkey incorrectly saccaded to a target stimulus that did not change were considered. Within these trials, the responses along the slow drift axis were decoded to discriminate between the final stimulus flash when a saccade was made ('final flash') and the preceding stimulus flash ('preceding flash'). *B*. Cross-validated decoding accuracy $P(\text{correct})$ for decoding responses along the slow drift axis as outlined in *A*. The raw activity for which the slow drift was not removed (black dots) and activity for which the slow drift was removed (red dots) are shown. *C*. Same as in *B* except that the response to the final flash was randomly shuffled with the response to the preceding flash for each trial. *D*. To test if the effects observed in *B* are due to adaptation, responses to non-saccade flashes during other trials that matched the sequence position of the decoded flashes of the false alarm trials were decoded. *E*. First model in which stimulus input x_{input} is transformed into a decision output d. See text for details about variables. Left inset: $x_{\text{input}} + \epsilon_{\text{noise}}$ over flashes, colored by $x_{\text{input}} = 5$ (orange) and $x_{\text{input}} = 6$ (blue). Right inset: v over flashes, with s overlaid on top of it. Colors match those in the left inset. (contineud on next page...)

Figure 6.8: (...continued from previous page) *F*. Both hit and false alarm rates covaried with the slow drift over trials. Rates were computed with running estimates of 500 trials, with 50-trial strides. *G*. Decoding accuracies to predict d from v (black) or $v - s$ (red) following the same decoding paradigm in *A* in which only the final and previous-to-final flashes of false alarm trials were considered. Error bars are 1 s.d. over 50 runs. *H*. Second model in which the slow drift influences the model's decision through perception (i.e., stimulus readout v depends on slow drift s) and via an independent pathway (i.e., arousal signal a depends on slow drift s). Middle and right plots: Same analyses as in *F* and *G* except for the simulated data of the second model. *I*. Third model in which the slow drift is removed from the perceptual readout $p = v - s$ but influences the model's decision through an arousal signal a. Middle and right plots: Same analyses as in *F* and *G* except for the simulated data of the third model.

drift axis, and found that the cross-validated decoding accuracy was at the chance level (Fig. 6.8*B*, black dots at 0.5). However, when we removed the slow drift from the V4 activity, the decoding accuracy was significantly greater than that of the raw activity (Fig. 6.8*B*, red dots above black dots). We confirmed that this effect was not present when we shuffled the activity between flashes (Fig. 6.8*C*, black and red dots at 0.5). To test if the effects observed in *B* were due to adaptation effects, we matched each false alarm trialwith a corresponding trial in which an identical sequence of stimulus flashes was presented but for which the monkey did not make a saccade (i.e., the sequence of flashes for the the false alarm trial was smaller than that of the corresponding trial). We then decoded responses to the two flashes of the sequence-matched trials that had the same sequence position as that of the preceeding and final flashes of the false alarm trials. We found no effect due to adaptation (Fig. 6.8*D*). This suggests that responses along the slow drift axis are predictive of behavior, which is at odds with Hypothesis 2. In addition, it suggests that the V4 activity is more predictive of within-trial behavior with the slow drift removed, consistent with Hypothesis 3.

The data provides the most evidence for Hypothesis 3. However, this hypothesis seems to be at odds with our finding that the slow drift is an arousal signal. How can the slow drift affect behavior when it is also removed downstream? We proposed three different models to provide insight into this process. The models take as input the stimulus input x_{input} as well as some perceptual noise from upstream sources ϵ_{noise}, and outputs a decision variable d of whether the stimulus input changed or not. The models also contain internal noise in form of the slow drift s. Simulating the change detection task, each trial consisted of stimulus "flashes" in which $x_{\text{input}} = 5$ for a sample stimulus, with a 40% chance the next flash would change the stimulus to $x_{\text{input}} = 6$. The correct output of the model is to have $d = 1$ when $x_{\text{input}} = 6$, and $d = 0$ otherwise. Thus, we could measure the hit and false alarm rate of the model. We ran 2,000 trials for each model.

The first model regarded the slow drift as perceptual noise (Fig. 6.8*E*). The slow drift is added to $x_{\text{input}} + \epsilon_{\text{noise}}$ to form the representation of the V4 activity along the slow drift axis v. When v is greater than some threshold, the decision $d = 1$. We also incorporated noise in d to match decoding accuracies to those observed in the real data (see Methods). The slow drift acted as perceptual noise by biasing v to be closer or more distant from the threshold across trials. In this way, the hit and false alarm rates covaried together as well as with the slow drift (Fig. 6.8*F*). However, because the detection of a change in x_{input} (i.e., perception) was dependent on the slow

drift, removing the slow drift worsened the false alarm prediction of v (Fig. 6.8*G*, black dot above red dot). Thus, this model was not consistent with the real data (Fig. 6.8*B*).

The second model added a pathway in which the slow drift affected the model's decision independent of perception (Fig. 6.8*H*). The parameters of the second model were identical to the first model, except that decision d could be 1 if v surpasses a threshold *or* an arousal signal a was 1. The probability of a being 1 decreased as s decreased its value. In this way, the slow drift influenced the probability that $d = 1$, and thus covaried with the hit and false alarm rates (Fig. 6.8*H*, middle). However, because v was dependent on s, removing s from v impaired the extent to which v predicted the occurence of a false alarm (Fig. 6.8*H*, right, black dot above red dot). Thus, this model was also not consistent with the real data (Fig. 6.8*B*).

Finally, we proposed a model in which the slow drift was removed from the perceptual readout, but influenced the model's decision via an independent pathway (Fig. 6.8*I*). The parameters of this third model were same as the second model, except that d was dependent on p, a perceptual readout of v, which removed s from v (i.e., $p = v - s$). Because the slow drift influenced behavior through a, the hit and false alarm rates covaried with the slow drift (Fig. 6.8*I*, middle). However, because p no longer depended on s, $v - s$ better predicted the occurrence of a false alarm within a trial than v did (Fig. 6.8*I*, right, red dot above black dot). This model was consistent with the real data (Fig. 6.8*B*). We further varied how much the slow drift was removed (i.e., $p = v - \alpha s$, where $\alpha \in [0, 1]$), and found that over half of the slow drift needed to be removed for $v - s$ to better predict false alarm occurrence. Thus, through simple models, we found that a likely explanation for the real data is that the slow drift is removed from its V4 readout so as not to lead to perceptual noise, but that ultimately the slow drift influences the probability of making a saccade through separate decision- and motor-related pathways. The models proposed here are not exhaustive, but rather give us insight into the underlying computations of the decision-making process.

6.1.8 Discussion

We found a slow drift in V4 population activity that explained 15%-30% of trial-to-trial shared variability, evolved on the time scale of 30 minutes, and lead to positive and negative correlations between neurons. The existence of a slow drift in a visual area raises the following two questions. What is the source of the slow drift? And how does the brain deal with the slow drift (which could potentially be read out as perceptual noise)?

To address the first question, we hypothesize that the slow drift arises from the slow release of neuromodulators throughout the brain. One candidate neuromodulator is epinephrine, which is distributed by the locus coeruleus (LC) to many different brain areas on a similar time scale as that of the slow drift (Aston-Jones and Cohen, 2005; Joshi et al., 2016; McGinley et al., 2015b) and may target different neurons in different ways (Totah et al., 2017). It has also been proposed that the LC modulates arousal, as the activity of LC neurons has been linked to behavioral variables that reflect arousal, such as pupil diameter (Joshi et al., 2016; McGinley et al., 2015a) and task performance (Aston-Jones and Cohen, 2005). Thus, if the slow drift is an arousal signal, it should covary with slowly-varying behavioral variables measured during the recording session. We found this to be the case, as the V4 slow drift was correlated with slow changes in hit rate, false alarm rate, the number of flashes within a trial, pupil diameter, and reaction time. We also found that V4 and PFC neurons share the same slow drift. Thus, the slow drift is likely an arousal signal that

modulates the activity of many brain areas.

Does the brain need to deal with the slow drift, and if so, how? We first found that the slow drift was large relative to changes in firing rate caused by attention or by the presentation of different stimuli. This suggests that the slow drift is large enough to possibly corrupt a downstream readout. One possible mechanism to overcome the slow drift is for the downstream readout axis to be unaligned with the slow drift axis. However, we found evidence that the slow drift axis is aligned to stimulus-encoding axes, suggesting that the slow drift axis is likely to affect downstream readout. Interestingly, we found that only when we removed the slow drift, responses along the slow drift axis were able to predict the occurrence of a false alarm within a trial. This result has two implications. First, because activity along the slow drift axis was related to within-trial behavior, it suggests that the slow drift axis is likely read out by a downstream area. Second, it suggests that a downstream area removes the slow drift from its readout to and decodes the residual activity. The downstream area likely has access to the slow drift, which is a brain-wide signal (Fig. 6.4). We proposed different models to reconcile the findings that the slow drift was related to slow changes in false alarm rate but obscured the ability of V4 activity to predict the occurrence of a false alarm within a trial. The model that best described the data was the one in which the slow drift was removed from the perceptual readout but influenced behavior via a pathway independent to perception. This model is consistent with the slow drift being an arousal signal that influences decision- and motor-related pathways. For example, the arousal signal can bias a subject to have a higher criterion (e.g., waiting to integrate more sensory information) or a lower criterion (i.e., more willing to guess for reward) when making a decision. To remove the slow drift, many realistic neural mechanisms exist, such as subtraction via inhibition (Wilson et al., 2012) or division via normalization (Carandini and Heeger, 2012). Further experiments are needed to investigate how and at which point in the neural circuit the slow drift is removed.

What are the advantages for the brain to send a slowly-varying arousal signal to many brain areas, including sensory areas that process incoming stimulus information? Some studies suggest that arousal signals enhance stimulus encoding by increasing firing rates and reducing noise correlations (Beaman et al., 2017; McGinley et al., 2015b; Ni et al., 2018; Ruff and Cohen, 2014). However, these observed arousal signals typically act on faster time scales than the slow drift (e.g., on the order of 1 minute), and may utilize similar mechanisms as those of attention (Beaman et al., 2017; Harris and Thiele, 2011; Ruff and Cohen, 2014). Instead, the slow drift seems to operate on a time scale detrimental to process quickly changing stimuli. One possible explanation is that the brain is required to accomplish many tasks at once, and it may be unavoidable that multiple influences affect the same neurons. For example, the slow drift may be governed by the same mechanism that causes global fluctuations in sleep (McGinley et al., 2015b; Steriade et al., 2001). While these fluctuations are likely useful to replenish cellular supplies (Greene and Siegel, 2004) or strengthen synaptic plasticity (Huber et al., 2004), the cortex may have developed computational mechanisms for high fidelity readout, robust to a slowly-varying neuromodulator release during wakefulness. We propose that one such mechanism is that downstream areas, which have access to the slow drift, can remove it from their sensory readouts.

Many studies have analyzed the effect of trial-to-trial shared variability on the fidelity of stimulus encoding in a population of neurons (Averbeck et al., 2006; Shadlen and Newsome, 1998). Theoretical studies have found that the only noise correlations that limit the amount of information carried by a population of neurons are differential correlations, which align their

shared noise to the stimulus-encoding axes (Kanitscheider et al., 2015; Kohn et al., 2016; Moreno-Bote et al., 2014). Because of the large number of neurons in a population, these studies predict that most sources of noise, such as global fluctuations, likely fluctuate along axes orthogonal to the stimulus-encoding axes, and thus pose no problem for downstream readout (Kohn et al., 2016; Moreno-Bote et al., 2014). However, these studies assume an optimal downstream readout for two particular stimuli (e.g., sinusoidal gratings whose orientation angles differ by 1°), and not a "general" readout for many types of stimuli and tasks (Ni et al., 2018). These studies also make the conservative assumption that the noise present in the activity of a population of neurons cannot be accessed by downstream areas (Kanitscheider et al., 2015; Moreno-Bote et al., 2014), but recent experimental studies have found that feedback may play a role in shaping the structure of noise correlations (Bondy et al., 2018; Snyder et al., 2014). Our work provides evidence that downstream readout areas have access to at least some of the noise present in upstream population activity. Thus, measuring the amount of information encoded by a population of neurons must take into account the extent to which downstream areas have access to the same noise sources that affect the upstream sensory areas.

Finding signatures of the slow drift in previous studies

In this study, we have found that the slow drift appreciably contributes to the total amount of shared noise in a population of V4 neurons. Thus, we should find signatures of the slow drift in other studies that have analyzed population activity in the macaque visual cortex. Here, we remark on a few studies that we think show signatures of the slow drift. One such study is Rabinowitz et al. (2015), which modeled the quickly changing attention modulation of a population of macaque V4 neurons with gain factors. They incorporated a slow drift gain in their model which would have otherwise obscured the fast attention modulators. This drift evolved on a similar time scale as our identified slow drift. In Ruff and Cohen (2014), the authors manipulated task difficulty in blocks of trials and found that the mean noise correlation of pairs of macaque V4 neurons decrease when the task is more difficult. This result, taken with our results, suggest that the arousal signal remains constant during blocks of increased difficulty (i.e., high arousal) but fluctuates in blocks of decreased difficulty (i.e., low arousal). However, it could be that the slow drift and the results observed in Ruff and Cohen (2014) are caused by neural mechanisms on different time scales. Finally, the recent study of Ni et al. (2018) reports that the mean noise correlations of pairs of macaque V4 neurons decrease as the discrimination performance of the subject increases. They performed controlled experiments in which they found that both attention and perceptual learning can decrease noise correlations. Interestingly, when they removed the contributions of these factors, they still found that the mean noise correlation and behavioral performance were strongly inversely related across sessions. We hypothesize that another strategy, in addition to attention and perceptual learning, is for the subject to maintain constant arousal levels throughout the session to achieve high behavioral performance, resulting in almost no slow drift. For sessions with low behavioral performance, the subject's arousal levels likely fluctuated, resulting in a highly-varying slow drift.

We also reconcile the results of our work with other studies about V4 activity. For example, (Luo and Maunsell, 2015) found that V4 neurons are modulated by sensitivity (i.e., changes in d') but not criterion (i.e., changes in hit and false alarm rate that go up and down together). This seems

at odds with our observations that slow drift in V4 that covaries with both hit rate and false alarm rate. However, the experiment in (Luo and Maunsell, 2015) locally varied reward contigencies for criterion to avoid the effects of brain-wide arousal. Thus, their finding that criterion has no effect on V4 activity is independent of arousal, our hypothesized source of the slow drift. Another study reported to find no slow drift in V4 activity and no relationship between V4 activity and pupil diameter (Beaman et al., 2017). However, their metric, called the population synchrony index, normalizes the mean population firing rate over a 100 ms window, which would not capture slowly drifting firing rates. Studies in mice have found global fluctuations on the time scale of tens of seconds that correlate with changes in pupil diameter (McGinley et al., 2015b; Reimer et al., 2014, 2016; Vinck et al., 2015) and locomotion (McGinley et al., 2015a; Reimer et al., 2014), suggesting that these global fluctuations in mice are caused by arousal. However, it is unclear if these arousal signals in mouse arise from the same neural substrates as that of the slow drift signal in monkey. One possibility is that these signals are caused by the same mechanism of neuromodulator release, but that the neuromodulator release in mouse acts on a faster time scale than that in monkey. To investigate this possibility, one can record from multiple brain areas in mouse and test if the observed global fluctuations match across brain areas, as we observed in monkey.

Practical concerns of the slow drift for other experiments

The presence of the slow drift raises practical concerns when analyzing neural data. For example, experiments that vary task conditions across blocks of trials may be inadvertently affected by the slow drift. For example, effects of attention may be masked by the larger slow drift (Fig. 6.6). Because the slow drift is related to slow changes in behavior, decoding neural activity across trials (e.g., responses to correct versus incorrect trials for the same presented stimulus) is likely influenced by the slow drift. This is important because decoding accuracy is often used as a proxy for the fidelity of perceptual readout of V4 when instead the decoding accuracy may reflect changes in arousal, independent of the perceptual ability of V4 activity. Finally, the slow drift may be harmful to decoders used in closed-loop experiments, such as brain-computer interface experiments (Sadtler et al., 2014; Santhanam et al., 2006) and adaptive stimulus selection (Cowley et al., 2017b; Lewi et al., 2009; Park et al., 2014). These systems may need to account for the slow drift by recording from other brain areas or measuring slow changes in behavioral variables, such as reaction time or pupil diameter.

Chapter 7

Other related projects

Throughout this thesis, we have shown that dimensionality reduction yields sensible and interpretable output when applied to the activity of visual cortical neurons. However, questions remain. For example, can dimensionality reduction reveal more about the underlying signals of population activity than if we were to average across the correlations of pairs of neurons (a common practice) (Cohen and Kohn, 2011; Cohen and Maunsell, 2009; Kohn and Smith, 2005; Smith and Kohn, 2008)? Considering that we typically apply dimensionality reduction to a small number of recorded neurons (from a circuit of millions of neurons), how do the outputs of dimensionality change with increased number of neurons and number of trials? In this chapter, we consider our other work that begins to answer these questions. In addition, to make dimensionality reduction easily accessible to the neuroscience community, we conclude with a section about DataHigh, a graphical user interface that allows users to input raw spike trains, perform dimensionality reduction, and visualize its outputs.

7.1 Estimating shared firing rate fluctuations in neural populations

Because neurons in the brain are heavily interconnected, insight about the computations that a population of neurons performs will likely be gained by going beyond single-neuron statistics. A natural first step is to consider the interactions between two neurons. One common way to measure to what extent the activity of two neurons covary is to compute the spike count correlation (r_{SC}) of the pair of neurons across repeats of the same stimulus (Cohen and Kohn, 2011). Because the stimulus drive is the same for each repeat, r_{SC} is also known as the noise correlation. Theoretical studies have shown that the presence of non-zero noise correlations can be harmful or helpful for the information contained in the population activity about different stimuli, depending on the correlation structure (Averbeck et al., 2006; Kohn et al., 2016; Moreno-Bote et al., 2014). The measure of r_{SC} reflects both variability *shared* by underlying sources (e.g., common input), as well as variability that is *private* to each neuron (e.g., Poisson-like variability). However, when considering more than two recorded neurons, it is unclear how to partition the variability into shared and private components that reflect correlated activity across the entire population.

One approach is to leverage the simultaneous recordings of tens of neurons to identify

shared factors underlying their activity. This yields a partition of each neuron's variability into a shared and private components. This partition can be achieved by a dimensionality reduction method called factor analysis (FA) (Fig. 7.1*A*). FA partitions the raw covariance matrix Σ_{SC} into two components: a shared covariance matrix $\Lambda\Lambda^T$ and a private variance matrix Φ, where $\Sigma_{\text{SC}} \approx \Lambda\Lambda^T + \Phi$. The shared covariance matrix is a low-rank approximation of the raw covariance matrix, and the private variance matrix is defined such that its diagonal elements can take any nonnegative value but its off-diagonal elements are all zero by definition. This partitioning allows the variance for the ith neuron (Σ_{SC}^{ii}) to be partitioned into a shared variance component ($[\Lambda\Lambda^T]^{ii}$) and a private variance component (Φ^{ii}) (Fig. 7.1*B*). The neuron's percent shared variance is the percent of the neuron's total variance ($[\Lambda\Lambda^T]^{ii} + \Phi^{ii}$) explained by the shared variance ($[\Lambda\Lambda^T]^{ii}$). The percent shared variance is intuitively similar to Pearson's correlation, except that percent shared variance indicates to what extent a neuron co-fluctuates with the entire population of recorded neurons (instead of pairwise co-fluctuations). FA determines the number of shared factors (i.e., the rank of $\Lambda\Lambda^T$) through cross-validation based on the data (Fig. 7.1*C*).

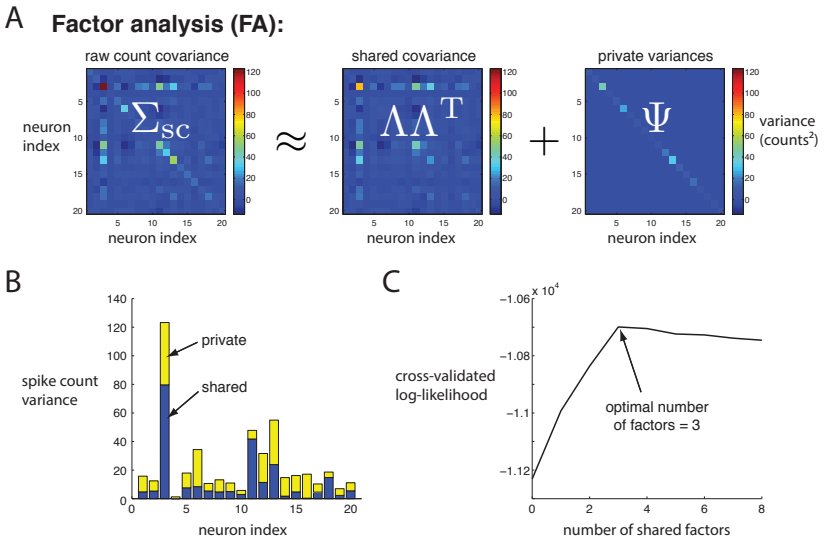


Figure 7.1: Illustration of partitioning raw spike count variance of N neurons into shared variance and private variance with factor analysis. *A*. Factor analysis partitions the raw spike count covariance matrix Σ_{SC} of size $(N \times N)$ into a shared covariance matrix $\Lambda\Lambda^T$ of size $(N \times N)$ and a private variance matrix Φ of size $(N \times N)$. Note that $\Lambda\Lambda^T$ is of lower rank than Σ_{SC} because $\Lambda \in \mathbb{R}^{N \times d_{\text{shared}}}$, where the number of factors $d_{\text{shared}} < N$. The private variance matrix by definition has non-negative diagonal elements and off-diagonal elements whose value are all zero. Factor analysis approximates the raw spike count covariance matrix as $\Sigma_{\text{SC}} \approx \Lambda\Lambda^T + \Phi$. *B*. For each neuron, we can compare its shared variance (i.e., its diagonal element of $\Lambda\Lambda^T$, blue bars) with its private variance (i.e., its corresponding diagonal element of Φ, yellow bars). We can compute the fraction of shared variance of each neuron by computing the ratio of the shared variance (blue bar) divided by the total variance (blue + yellow bar). *C*. Factor analysis determines the number of factors d_{shared} for Λ by choosing the d_{shared} with the highest cross-validated log-likelihood value.

This approach allows us to ask four questions about the structure of noise correlations on

the level of the population. First, what fraction of the raw spike count variance is shared among neurons (Fig. 7.2*A*)? A large fraction shared variance suggests that most of a neuron's variance can be explained by the activity of the other recorded neurons. Second, how dominant is the first factor of shared variability? In other words, can most of the noise correlations be captured by a single global fluctuation? We can measure this by assessing the fraction of shared variance explained by the top factor (i.e., the top eigenvalue of $\Lambda\Lambda^T$ divided by the trace of $\Lambda\Lambda^T$) (Fig. 7.2*B*). Third, does the entire population of neurons co-fluctuate their activity up and down together? We assess this by computing the angle between the mode of the top factor and the origin-to-mean-spike-count mode in firing rate space (Fig. 7.2*C*). Fourth, does the private variance swamp the shared variance? Because PCA can be affected by private variance, we can compute the angle between the axis of the top factor and the axis of the top principal component (Fig. 7.2*D*). An angle close to 90°suggests that the private variance masks the shared variance, while an angle close to 0°suggests that the shared variance dominates the private variance.

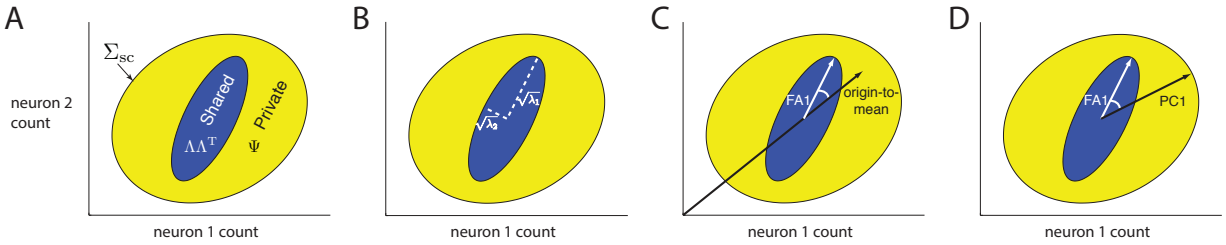


Figure 7.2: Illustrative plot of the activity of two neurons of that reflect the questions we can ask on the population level. *A*. Factor analysis partitions the raw spike count covariance matrix Σ_{SC} into a shared covariance matrix $\Lambda\Lambda^T$ and a private variance matrix Φ. We can ask to what extent is the shared variance (size of blue ellipse) larger than the private variance (size of yellow-shaded area of larger ellipse). *B*. We can ask how narrow is the shared covariance ellipse (blue ellipse). In other words, does most of the shared variance lie along one dominant axis? We can assess this by taking a ratio between the top factor's variance (λ_1) divided by the sum of all the factors' variances ($\lambda_1 + \lambda_2$). *C*. We can ask if the dominant axis (white vector, FA1) is aligned with the origin-to-mean axis (black vector, origin-to-mean). This is akin to asking if the activity of the neurons go up and down together (i.e., positively co-vary). *D*. We can ask how much does the private variance (yellow area) mask the shared variance (blue ellipse). Because PCA can succumb to large amount of private variability, we can compare the axis of the top principal component (PC1) with the axis of the top factor (FA1). A large angle between PC1 and FA1 indicates that the private variance masks the underlying shared variance.

We addressed these questions by analyzing multi-electrode array recordings in anesthetized macaque V1. We compared results between evoked responses (200 repeats of a 1 second drifting sinusoidal grating) and spontaneous activity (200 trials of 1 second blank screens) for the same 21 neurons. Previous studies have found that spontaneous activity typically has higher mean noise correlations that that of evoked responses (Kohn and Smith, 2005). Here, we ask how the outputs of factor analysis differ between spontaneous activity and evoked responses.

What fraction of the raw spike count variance is shared among neurons?

One may like to go beyond measuring how much one neuron's activity relates to another neurons activity to measuring how one neuron's activity relates to many other neurons' activity. Instead of measuring noise correlation, we can use factor analysis to measure fraction shared variance. An analogy that describes the difference between noise correlation and fraction shared variance is the following. Measuring the noise correlation is akin to predicting the activity of one neuron from the activity of another neuron using linear regression (with a 1×1 weight vector). On the other hand, measuring fraction shared variance is akin to predicting the activity of one neuron with the activity of all other recorded neurons using linear regression (with an $N \times 1$ weight vector, where N is the number of other recorded neurons). By measuring fraction shared variance, we can leverage the activity of the entire population to gain statistical power when relating neural activity between neurons.

We computed the mean fraction shared variance (averaged over neurons) by applying factor analysis separately to the spontaneous and evoked population activity. Similar to previous studies (Kohn and Smith, 2005), we found that the fraction shared variance was larger for spontaneous activity than for the evoked responses (mean fraction shared variance: ~0.40 $>$ ~0.35, $p < 0.05$ for matched levels of firing rates). We can also ask how many factors are needed to explain the population activity. Because the spontaneous activity has a larger fraction shared variance than that of the evoked responses, one might expect that the spontaneous activity is explained by a larger number of factors. However, we found that the number of factors needed to explain the shared variance was larger for evoked responses than for spontaneous activity (Fig. 7.3*A*). This suggests that the fluctuations of spontaneous activity are larger along a smaller number of modes than those of the evoked responses.

How dominant is the first factor of shared variability?

Beyond measuring the number of factors, the way in which the neurons co-fluctuate along their modes of shared variance may be different between spontaneous activity and the evoked responses. For example, fluctuations along the first mode may be larger relative to fluctuations along other modes, or the size of fluctuations may be equal along all the modes. To assess this, we computed the shared variance explained by the top factor divided by the total shared variance, which yields a fraction. This fraction was large for both spontaneous and evoked activity (Fig. 7.3*B*, > 0.5), suggesting that the shared variance is not spread equally across all modes but rather is most present along the first mode. We also found that the top factor of spontaneous activity was more dominant than that of the evoked responses (Fig. 7.3*B*, red dashed line $>$ black dashed line). This suggests that the way in which neurons co-fluctuate along their modes of shared variance differ between spontaneous activity and the evoked responses.

Does the entire population of neurons co-fluctuate their activity up and down together?

A pair of neurons can have either positive or negative noise correlations. In a similar way, neurons can co-fluctuate differently for one mode of the shared variance. For example, all the neurons could co-fluctuate up and down together, in which case each neuron contributes a small, positive amount to the mode of shared variance. On the other hand, the activity of some neurons could

increase while the activity of other neurons decrease, and vice versa. In this case, some neurons would contribute positively to the mode of the shared variance, while the other neurons would contribute negatively. To test which case is most likely for the top factor of spontaneous and evoked activity, we compute the angle between the top factor (i.e., mode of shared variance) and the origin-to-mean axis (i.e., a vector between the origin and the mean spike count in firing rate space). The origin-to-mean axis best captures the direction in firing rate space in which the activity of neurons go up and down together. We found that the top factor was more aligned with the origin-to-mean vector than expected by chance (Fig. 7.3*C*), suggesting that the fluctuations of neurons go up and down together. This co-fluctuation tended to be more pronounced for spontaneous activity than for the evoked responses (Fig. 7.3*C*, red dashed line $<$ black dashed line). This suggests that neurons are more likely to contribute positively to the top factor for spontaneous activity than for evoked responses.

Does the private variance swamp the shared variance?

An important question is whether the trial-to-trial variability of a population of neurons is mostly shared across neurons or private to each individual neuron. On one extreme, each neuron's activity is independent of the activity of all other recorded neurons, suggesting no common input or shared connections exist between neurons. On the other extreme, the activity of one neuron can be entirely explained by the activity of the other neurons, suggesting that these neurons directly reflect an external input or are tightly connected. On a practical concern, a large private variance may impact the ability of factor analysis to properly identify the shared variance. To assess this, we can compare the top factor identified by factor analysis (FA) with the top mode identified by principal component analysis (PC1). Because PCA has no notion of shared or private variance, the top mode of PCA can entirely reflect private variance, if the private variance is much larger than the shared variance. Thus, by comparing the top modes of FA and PCA in a similar way we compared the top factor and origin-to-mean mode (Fig. 7.3*C*), we can gain insight into how much the private variance "swamp" the shared variance. We found that the top modes were substantially more aligned than expected by chance (Fig. 7.3*D*), suggesting that identifying the shared variance is not largely affected by the private variance. Again, this trend was more pronounced for spontaneous activity than for the evoked responses (Fig. 7.3*D*, red dashed line $<$ black dashed line).

Overall, these results suggest that one dominant, shared factor underlies both spontaneous activity and evoked responses of anesthetized macaque V1, similar to other accounts (Arandia-Romero et al., 2016; Ecker et al., 2014; Lin et al., 2015; Okun et al., 2015). This dominant factor is more prominent during spontaneous activity than during evoked responses. In addition, we have shown that factor analysis extends the intuition of r_{SC} to partition the raw spike count variance into a shared and private component on the population level.

This is the work of Byron Yu, Adam Kohn, and Matthew Smith (Yu et al., 2011), with contributions from Benjamin Cowley.

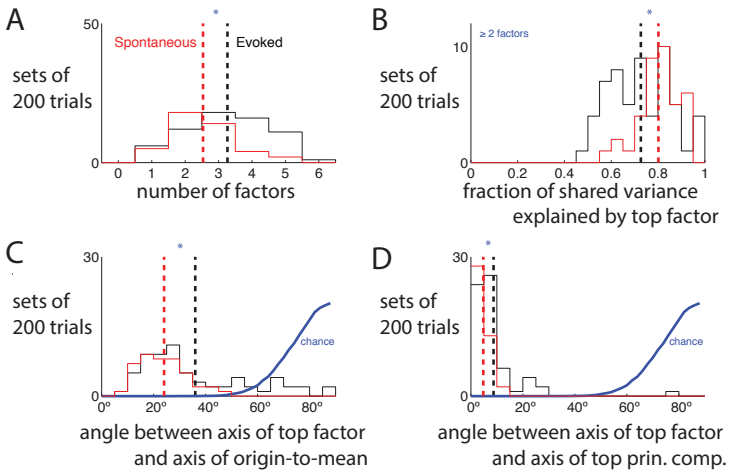


Figure 7.3: Factor analysis results for evoked responses and spontaneous activity for anesthetized macaque V1. We binned the spike counts of each 1 second trial, and grouped trials into sets of 200. *A*. The number of factors d_{shared} estimated separately for each set of 200 trials. *B*. The fraction of shared variance explained by the top factor for each set of 200 trials. The fraction is computed as the ratio of the variance of the top factor divided by the sum of the variances for all factors. *C*. Angle between the axis of the top factor and the axis of the origin-to-mean for each set of 200 trials. Chance is computed by taking the angle between two random vectors whose elements are drawn from a standard Gaussian and then normalized. *D*. Angle between the axis of the top factor and the axis of the top principal component. Chance is computed as in *C*.

7.2 Scaling properties of dimensionality reduction for neural populations and network models

In the previous section, we used factor analysis to characterize the population activity structure. However, our conclusions are based on a small number of neurons (21 neurons) from the presumably millions of neurons in the macaque primary visual cortex. This motivates us to see if these results change when we increase the number of neurons and number of trials. Because *in vivo* experiments are limited by the number of neurons and number of trials that we can record, we also rely on spiking neural network models (Litwin-Kumar and Doiron, 2012) for which we have access to thousands of neurons and unlimited trials. For this work, we consider the number of factors (d_{shared}) and the percent shared variance (averaged over neurons), as described in the previous section.

7.2.1 Varying neuron and trial count for *in vivo* neural recordings

We first studied how d_{shared} and percent shared variance scale with neuron count for *in vivo* recordings. To do this, we applied FA to spontaneous activity recorded in primary visual cortex (V1) of anesthetized macaque monkeys. We binned neural activity into 1 second epochs, where each bin is referred to as a 'trial'. Thus, the number of trials is equivalent to the recording time (in seconds). We sampled increasing numbers of neurons or trials from the recorded population

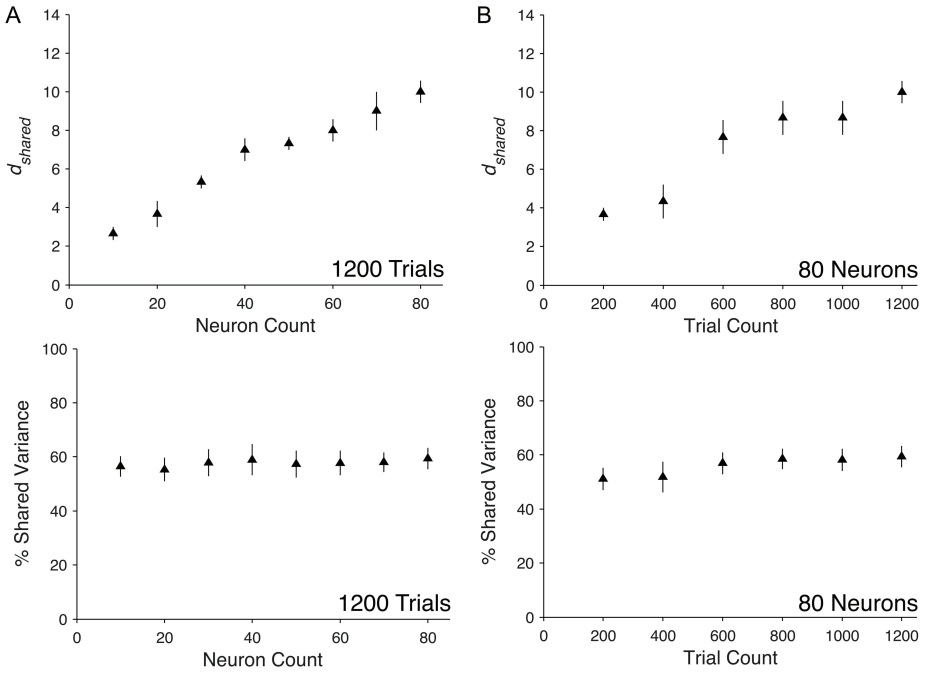


Figure 7.4: Scaling properties of shared dimensionality and percent shared variance with neuron and trial count in V1 recordings. The d_{shared} and percent shared variance over a range of (*A*) neuron counts and (*B*) trial counts from population activity recorded in V1. Each triangle represents the mean across single samples from each of three arrays. Error bars represent one standard error across the three arrays.

activity, and measured d_{shared} and percent shared variance for each neuron or trial count. We expected d_{shared} and percent shared variance to either saturate or to increase with increasing neuron or trial count. Saturating d_{shared} would suggest that we have identified all of the modes for the network (or networks) sampled by the recording electrodes and increasing d_{shared} would suggest that additional modes are being revealed by monitoring additional neurons or trials. We found that d_{shared} increased with neuron count (Fig. 7.4*A*, top), while percent shared variance remained stable with increasing neuron count (Fig. 7.4*A*, bottom). Similarly, additional trials resulted in increasing d_{shared} and stable percent shared variance (Fig. 7.4*B*). Together these results demonstrate that, within the range of neurons and trials availabe from our recordings, additional neurons and trials allow us to identify additional shared dimensions. This implies that we have not sampled enough neurons or trials to identify all of the modes of shared variability. However, given the stable percent shared variance observed in Figure 7.4*A* and 7.4*B* (bottom panels), the results suggest that the shared component is dominated by the first few modes and that additional modes do not explain substantial shared variance.

7.2.2 Varying neuron and trial count for network models within the experimental regime

In the previous section, we identified trends in d_{shared} and percent shared variance using *in vivo* recordings. Several experimental constraints limit the types of questions we can ask using in vivo recordings. First, we are limited in the number of neurons and the number of trials that are recorded. Second, in most experiments, we have no knowledge of the connectivity structure of the underlying network and cannot relate properties of the population activity to network structure. In this section we overcome these constraints by analyzing activity obtained from network models.

We consider recurrent spiking network models with distinct excitatory and inhibitory populations whose synaptic interactions are dynamically balanced (Renart et al., 2010; Vreeswijk and Sompolinsky, 1998). In particular, we focus on two subclasses of this model: one where excitatory neurons are grouped into clusters that have a high connection probability (clustered network) and one where the excitatory population has homogeneous connectivity (non-clustered network). Both the clustered and non-clustered networks have been shown to capture variability in spike timing (Litwin-Kumar and Doiron, 2012; Vreeswijk and Sompolinsky, 1998). Clustered networks have also been shown to demonstrate slow fluctuations in firing rate (Litwin-Kumar and Doiron, 2012) consistent with in vivo recordings (Churchland et al., 2011, 2010; Kohn and Smith, 2005).

In the particular clustered network studied here, each cluster resembles a bistable unit with low and high activity states that lead neurons in the same cluster to change their activity together. We expected to identify dimensions that reflected these co-fluctuations within clusters, resulting in d_{shared} bounded by the number of clusters (i.e., 50 dimensions) and high percent shared variance. In contrast, the non-clustered network lacks the highly correlated activity seen in the clustered network (Litwin-Kumar and Doiron, 2012; Renart et al., 2010; Vreeswijk and Sompolinsky, 1998), and so we expected to see little or no shared variance. Note that no shared variance would result in both percent shared variance and d_{shared} being zero. Small amounts of shared variance relative to total variance would result in low percent shared variance and either low or high d_{shared} depending on the multi-dimensional structure of the shared variance.

To test how clustered connectivity affects the population activity structure and to understand how the population-level metrics scale with the number of neurons and trials, we performed the following analysis. We applied FA to spike counts, from non-clustered and clustered network simulations. Each spike count was taken in a 1 second bin of simulation time, which we refer to as a 'trial' in analogy to physiological recordings. We then increased the neuron count, as we did in Figure 7.4 for the *in vivo* recordings, with the number of trials fixed at 1,200 to match the analyses shown in Figure 7.4*A*. We observed increasing d_{shared} with neuron count in the clustered network and a d_{shared} of zero for all neuron counts in the non-clustered network (Fig. 7.5*A*, top). The percent shared variance for the clustered network increased with neuron count and saturated at approximately 90% (Fig. 7.5*A*, bottom). In contrast, the non-clustered network showed zero percent shared variance at all neuron counts. In other words, in the range of trials and neurons studied, FA could not identify any shared population-level structure in the non-clustered network. These results agree with our predictions, namely non-zero d_{shared} and high percent shared variance in the clustered network and zero dshared and percent shared variance in the non-clustered network.

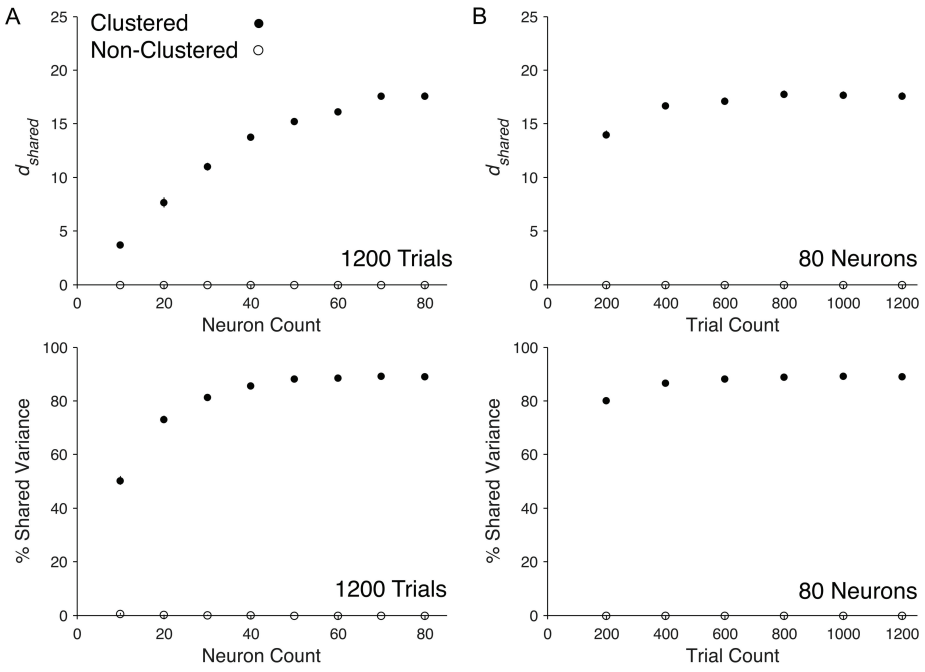


Figure 7.5: Scaling properties of shared dimensionality and percent shared variance with neuron and trial count in spiking network models. The d_{shared} and percent shared variance over a range of (*A*) neuron counts and (*B*) trial counts from clustered (filled circles) and non-clustered (open circles) networks. Circles represent the means across the five non-overlapping sets of neurons and five non-overlapping sets of trials (25 total sets) and error bars represent standard error across all sets. Standard error was generally very small and therefore error bars are not visible for most data points.

We next investigated how d_{shared} and percent shared variance change for an increasing number of trials, with the number of model neurons fixed at 80 to match the analyses shown in Fig 3B. We anticipated that d_{shared} and percent shared variance would increase to a saturation point after which enough trials would be available to reliably identify all of the modes of shared variability. In the clustered network, we observed that d_{shared} (Fig. 7.5*B*, top) and percent shared variance (Fig. 7.5*B*, bottom) initially increased and then saturated, indicating that fewer than 1,200 trials were needed to characterize d_{shared} and percent shared variance for 80 neurons sampled from the clustered network. In the non-clustered network, we observed zero d_{shared} and percent shared variance for all trial counts. Therefore, of the two networks studied, only the clustered network demonstrated population-level shared structure within the range of trials obtained in the *in vivo* recording.

Comparing the model network results (Fig. 7.5) with the experimental results (Fig. 7.4) obtained from equal numbers of neuron and trials, we observed similar trends in the clustered network and *in vivo* recordings. In both cases we observed increasing d_{shared} and saturating percent shared variance with neuron and trial count. Note that we did not tune network parameters (e.g., firing rates, number of clusters, etc.) in the clustered network to match the *in vivo* recordings and, therefore, we did not expect the magnitudes of d_{shared} or percent shared variance to match in the two cases. However, the trends of increasing d_{shared} with neuron and trial count accompanied by stable percent shared variance suggest that, in both cases, the population activity is largely governed by a few dominant modes that are well characterized within the range of neurons and trials obtainable with current recording technology.

7.2.3 Varying neuron and trial count for network models beyond the experimental regime

To better understand how the outputs of dimensionality reduction for limited sampling reflect larger portions of the network, we investigated how the trends from Figure 7.5 continued for larger numbers of neurons and trials. We first varied the number of neurons in the analysis up to 500 neurons. This required us to increase the number of trials from 1,200 to 10,000 trials in order to fit the larger number of parameters in the FA model. We found that d_{shared} in the clustered network saturated with roughly 100 neurons, whereas d_{shared} in the non-clustered network continued to increase with neuron count (Fig. 7.6*A*, top). In both networks, the percent shared variance remained stable with additional neurons, but the clustered network had higher shared variance than the non-clustered network (Fig. 7.6*A*, bottom). Within the experiment regime of neuron counts (10 to 80 neurons) we found non-zero d_{shared} and percent shared variance for both networks (Fig. 7.6*A*, inset). Overall, in the clustered network, we observed saturation in d_{shared} and percent shared variance with few neurons relative to the network size. This likely stemmed from the fact that neurons from the same cluster varied together. Therefore, we were able to identify the majority of shared variance once multiple neurons were sampled from most clusters. That result contrasts with our observation of increasing dimensionality and low shared variance in the non-clustered network, which lacks the defined structure of groups of co-varying neurons. It is therefore likely that we identified many modes that each explain small amounts of variability.

To study the effects of large trial count on population-level metrics, we computed d_{shared} and percent shared variance for 80 neurons while varying the trial count up to 20,000 (Fig. 7.6*B*). The

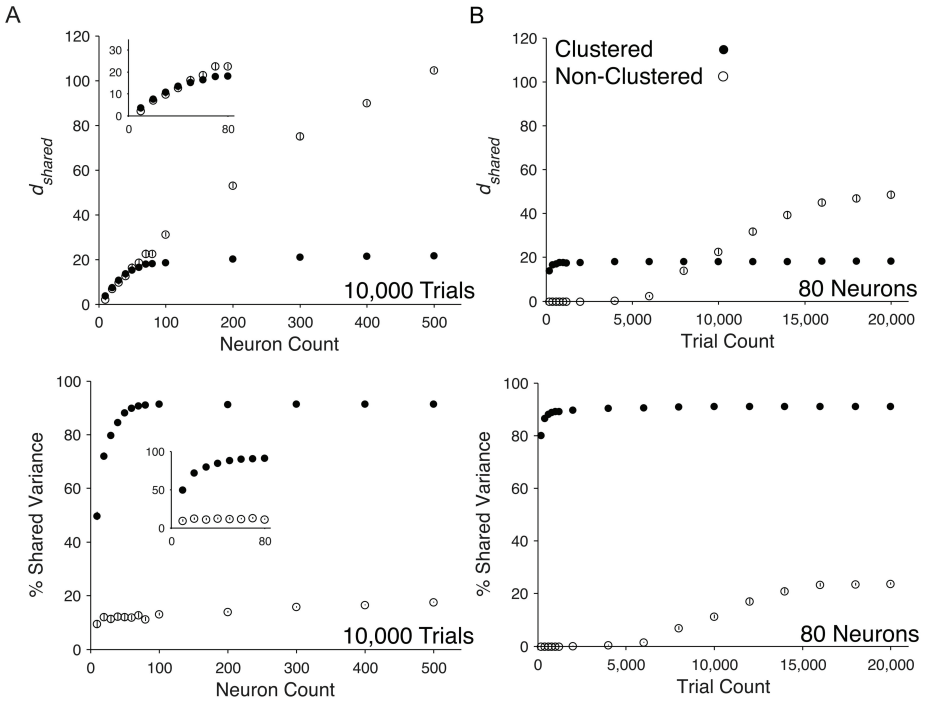


Figure 7.6: Scaling properties of shared dimensionality and percent shared variance with large neuron and trial counts in spiking network models. The d_{shared} and percent shared variance over a range of (*A*) neuron counts and (*B*) trial counts from clustered (filled circles) and non-clustered (open circles) networks. Insets zoom in on range of neurons used in in vivo recordings in Fig. 7.4. Circles represent the means across the five non-overlapping sets of neurons and five non-overlapping sets of trials (25 total sets) and error bars represent standard error across all sets. Standard error was generally very small and therefore error bars are not visible for most data points.

non-clustered network had no identifiable shared population activity structure when the trial count was less than 5,000, consistent with Figure 7.5; however, with 5,000 or more trials, we identified non-zero d_{shared}. It is clear from this result that trial counts within the experimental regime were insufficient to identify shared dimensions, but that additional trials revealed shared dimensions. Percent shared variance followed a similar trend, with zero percent shared variance below 5,000 trials, as expected given zero d_{shared}. These results show that many more trials were required to identify the small amounts of shared variability in the non-clustered network compared to the clustered network.

The above analyses showed substantial differences between the two model networks. In the clustered network, the shared population activity structure was salient (approximately 90% of the raw spike count variability was shared among neurons) and defined by a small number of modes (approximately 20 modes), all of which could be identified using a modest number of neurons and trials. In contrast, for the non-clustered network, the shared population activity was more subtle (approximately 20% of the raw spike count variability was shared among neurons), distributed across many modes, and required large numbers of trials to identify.

This is the work of Ryan Williamson (first author), Benjamin Cowley, Ashok Litwin-Kumar, Brent Doiron, Adam Kohn, Matthew Smith, and Byron Yu (Williamson et al., 2016).

7.3 DataHigh: A software tool to apply dimensionality reduction to neural data and visualize its outputs

Throughout this thesis, we have found that the outputs of dimensionality reduction were sensible when applied to neural population activity. This motivates us to develop a software tool that makes dimensionality reduction accessible to the neuroscience community. The tool has two complementary components. First, the user can input recorded spike trains and perform dimensionality reduction. Second, the user can then visualize the output of dimensionality reduction in an intuitive manner. This last step is important because dimensionality reduction typically outputs 4 or more latent variables, making it difficult to visualize how the latent variables interact together. The software tool overcomes this by allowing the user to quickly and smoothly navigate through a continuum of different 2-d projections of the latent space. The primary goal of this tool is to facilitate exploratory data analysis, which in turn can generate scientific hypotheses.

The basic setup is to first define an n-dimensional space, where each axis represents the firing rate of one of the n neurons in the population. A dimensionality reduction method is then applied to the n-dimensional population activity to determine the number of latent variables, k, needed to adequately describe the population activity ($k < n$), as well as the relationship between the latent variables and the population activity (Fig. 7.7). These latent variables define a reduced k-dimensional *latent space* in which we can study how the population activity varies over time, across trials, and across experimental conditions. Ideally, we would like to visualize the latent variables directly in the k-dimensional space. However, the number of latent variables, k, is typically greater than three (Machens et al., 2010; Santhanam et al., 2009; Yu et al., 2009) and

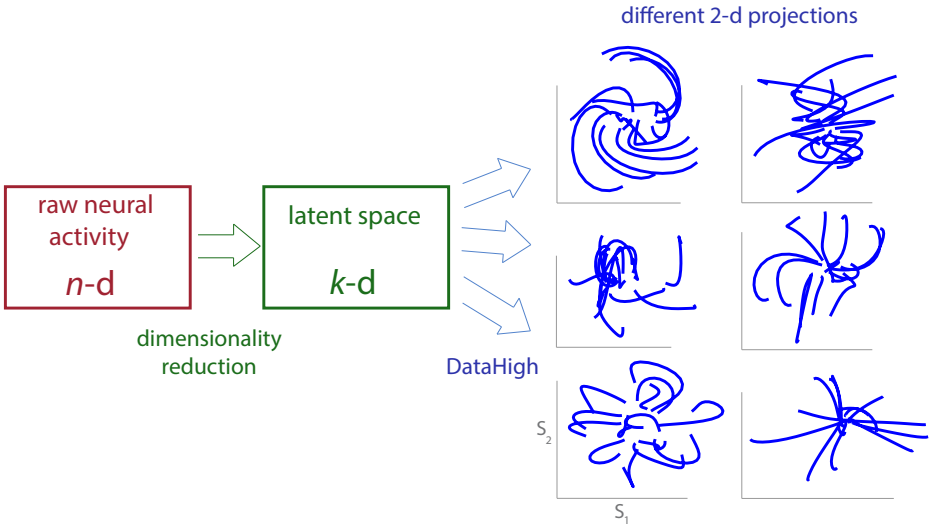


Figure 7.7: Flow diagram for visualization of population activity. Dimensionality reduction is performed on high-dimensional population activity (n-d, where n is the number of neurons) to extract a latent space (k-d, where k is the number of latent variables). Typically, k is less than n but greater than 3. We can then use DataHigh to visualize many 2-d projections of the same latent space. Shown here are six different 2-d projections of the same 6-d ($k = 6$) latent space described in Churchland et al. (2012).

direct plotting can only provide a 2-d or 3-d view. If specific features of interest of the population activity are known, one approach is to specify a cost function to find a 2-d projection of the k-d space that illustrates those features (Churchland et al., 2012). However, in exploratory data analysis, such features may be unknown in advance, and viewing a single 2-d projection can be misleading. As illustrated in Figure 7.7, the same 6-d latent space can yield rather different looking 2-d projections. This underscores the need to look at many 2-d projections to obtain a more complete picture of the high-dimensional structure of population activity.

To quickly and smoothly view many 2-d projections, we developed an interactive graphical user interface (GUI), called *DataHigh*, in Matlab for visualization of the k-d latent space. The user uploads raw spike trains to DataHigh, which then guides the user to perform dimensionality reduction and to quickly visualize a continuum of 2-d projections. We found DataHigh to be a valuable tool for building intuition about population activity, for hypothesis generation, and for development of models of population activity. Although high-dimensional visualization is a challenge across many scientific fields, DataHigh has tools tailored to neural data analysis that are currently not present in general-purpose high-dimensional visualization software (Swayne et al., 2003). DataHigh is versatile and can be used to study population activity recorded either simultaneously (using multi-electrode arrays) or sequentially (using conventional single electrodes). The population activity may be in the form of single-trial spike count vectors (taken in a single time bin on each experimental trial), single-trial timecourses (where spike counts are taken in small, non-overlapping time bins), or trial-averaged timecourses (PSTHs). For each class of population activity, DataHigh can extract the corresponding latent variables, termed *neural states*, single-trial

neural trajectories, and trial-averaged neural trajectories, respectively. We previously presented a preliminary version of this work in Cowley et al. (2012).

7.3.1 Data analysis procedure

We describe and motivate a neural data analysis procedure that incorporates DataHigh and involves four steps: preprocessing, dimensionality reduction, visualization, and testing (Fig. 7.8).

In the preprocessing step, the user first inputs single-trial raw spike trains into *DimReduce*, a plug-in tool of DataHigh (left-hand side of Fig. 7.8). DimReduce first takes spike counts in non-overlapping time bins, where the bin width is specified by the user in the GUI and determines if the neural activity will be represented as neural states or single-trial neural trajectories. For single-trial neural trajectories, DimReduce presents two optional actions: to average across trials (for trial-averaged neural trajectories) and to apply kernel smoothing.

In the dimensionality reduction step, the user selects a dimensionality reduction method to extract latent variables from the neural population activity (Section 7.3.4). After DimReduce has either performed cross-validation or extracted a large number of latent variables, the user can choose a latent dimensionality for visualization (Section 7.3.4). These latent variables are automatically uploaded to DataHigh for visualization. Alternatively, dimensionality reduction can be performed outside of the DataHigh environment (right-hand side of Fig. 7.8). The first steps are to take binned spike counts and to choose whether to average across trials and whether to apply kernel smoothing. Then, a dimensionality reduction method that has not yet been implemented in DataHigh can be applied to the data. Finally, the extracted latent variables need to be formatted correctly and input into DataHigh.

The next step in the data analysis procedure is to visualize the extracted neural states and trajectories, whose dimensionality is typically greater than three. By including visualization as a step in exploratory data analysis, the experimenter can potentially save a substantial amount of time in filtering out hypotheses about features that are not salient in the population activity and guiding the experimenter towards building hypotheses and intuition about features that are salient. This can help guide the development of algorithms and models that attempt to extract statistical structure from the population activity.

A standard way to incorporate visualization is to first hypothesize about a feature of the data and then define a cost function to search for a 2-d or 3-d projection that shows the existence of such a feature. If the hypothesized feature appears to be present, statistical tests can then be applied. However, there are two drawbacks to this approach. First, individual 2-d projections of high-dimensional data can be misleading. For instance, two points that are close together in 2-d visualization may not be close together in the high-dimensional space. Second, this “guess-and-check" approach may require the application of many cost functions before salient features are found, and can potentially miss features that were not hypothesized. Instead of limiting visualization to a small number of projections found by cost functions, the user can interactively view many 2-d projections of the latent space with DataHigh, and utilize a suite of built-in analysis tools that assist in the visualization process. The user can then use DataHigh to visually investigate existing hypotheses while building intuition and new hypotheses.

After hypothesis-building from visualization, the user can perform statistical tests on the hypotheses. If the user has an existing hypothesis about the population activity (either from

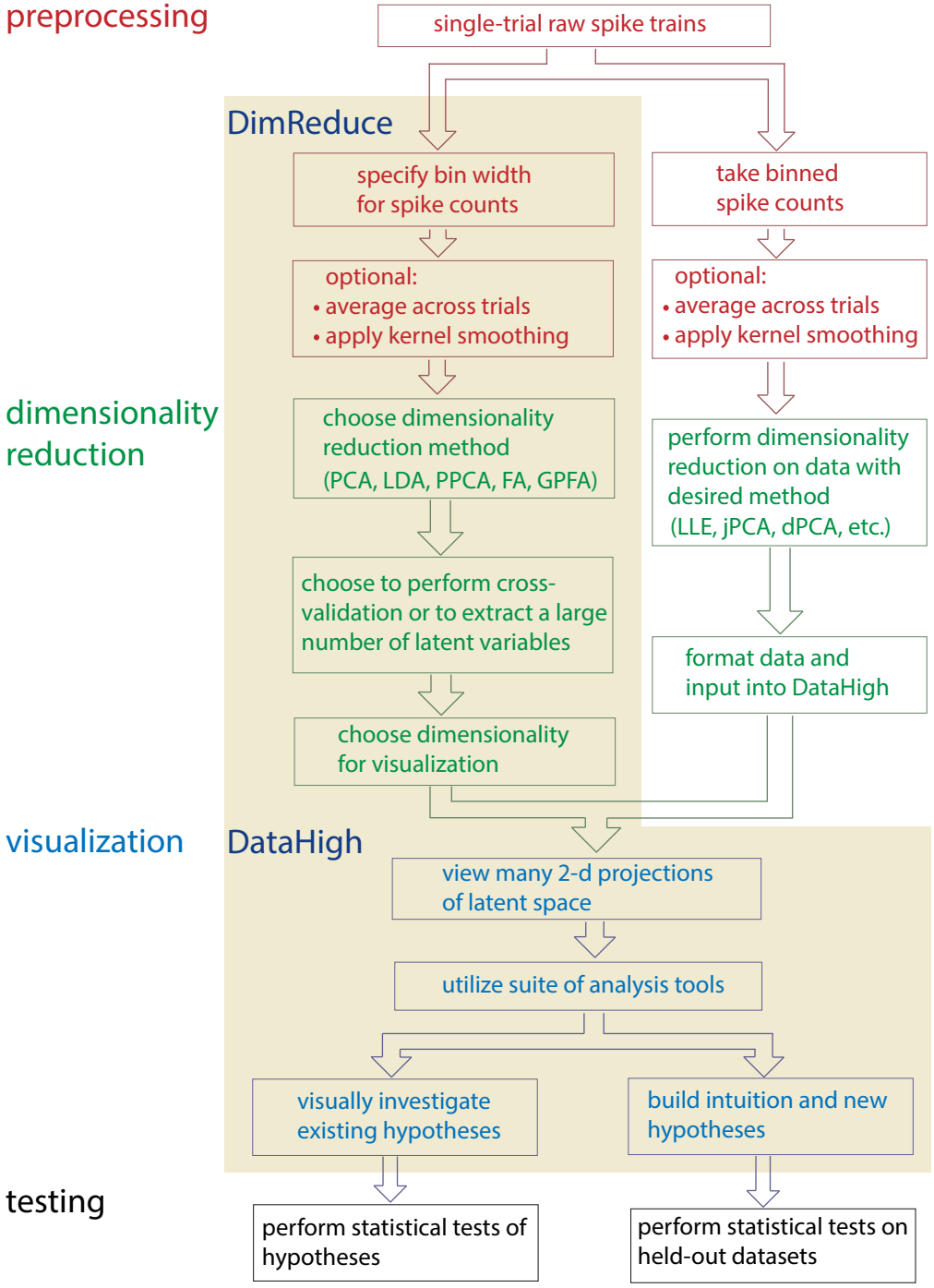


Figure 7.8: Flowchart for a data analysis procedure that utilizes visualization. The user may input raw spike trains into DataHigh, perform dimensionality reduction using the DimReduce tool (left-hand side of *dimensionality reduction*), and visualize many 2-d projections of the extracted latent space using DataHigh. The user may also perform dimensionality reduction outside the DataHigh environment (right-hand side of *dimensionality reduction*), and input the identified latent variables into DataHigh for visualization.

previous studies or from analyses of other datasets), the user can visually inspect whether the hypothesis holds and apply statistical tests to the data (left-hand side of *testing* in Fig. 7.8). These tests typically produce quantitative metrics (e.g., p-values, decoding accuracy), which can be complemented with visualization to add qualitative intuition about why a test succeeds or fails. However, if visualization suggests a *new* scientific hypothesis, statistical testing should be done on held-out datasets (either different datasets collected from the same subject or from different subjects), which avoids bias that comes from testing hypotheses suggested by the data (Berk et al., 2010) (right-hand side of *testing* in Fig. 7.8). Testing can either be done in the k-dimensional latent space or the original n-dimensional space. The advantage of testing in the latent space is that the data are "denoised" and effects may be more pronounced. However, the user needs to carefully consider whether the dimensionality reduction method can build the effect in question into the neural states or neural trajectories, either via simulations or analytical reasoning. It is often "safer" to use the latent space for hypothesis generation, and then test the hypothesis in the original high-dimensional space (Afshar et al., 2011).

Another benefit of DataHigh is that it can be used to quickly view and triage the large amounts of data produced by a neuroscience experiment. With single-electrode recordings, it is common to listen to each spike as it streams in and to plot a neuron's PSTHs and tuning curve during an experiment. With the advent of multi-electrode recordings, it has become less common to listen to and visualize neural activity as it streams in, simply due to the sheer quantity of the neural data (e.g., if there are 100 recording electrodes). Using dimensionality reduction in tandem with visualization allows experimenters to quickly inspect a large amount of data and perform error checking between each recording session of an experiment. This can help to guide changes in the design of an experiment, to build intuition about the population activity, and to detect any potential problems with the recording apparatus. For example, this approach led experimenters to identify individual, outlying trials (Churchland et al., 2010; Yu et al., 2009) and electrodes that contained cross-talk (Yu et al., 2009). As the number of sequentially- or simultaneously-recorded neurons increases, having methods for quickly building intuition about the population activity and assaying large datasets will become increasingly essential.

7.3.2 Rotating a 2-d projection plane

The main interface of DataHigh (Fig. 7.9) allows the user to quickly and smoothly rotate a 2-d projection plane in the k-dimensional space, where k is the number of identified latent variables. The goal is to provide the minimum set of "knobs" that allow the user to achieve all possible rotations within the k-dimensional space. We first describe the mathematical idea of our approach, then the implementation. The mathematical details presented in this section are not necessary to use DataHigh, and may be skipped without loss of intuition for the tool. We begin with two arbitrary orthonormal k-dimensional vectors, $\mathbf{v}_1$ and $\mathbf{v}_2$, which define the horizontal and vertical axes, respectively, of a 2-d projection plane. To rotate the projection plane, we keep one vector $\mathbf{v}_1$ fixed, while rotating the other vector $\mathbf{v}_2$. To maintain orthogonality, $\mathbf{v}_2$ must rotate in the $(k-1)$-dimensional orthogonal space of $\mathbf{v}_1$. In this space, any rotation of $\mathbf{v}_2$ can be fully specified by $(k-2)$ angles. Thus, we provide the user with $(k-2)$ "knobs" (right-hand panels in Fig. 7.9) to rotate $\mathbf{v}_2$ while keeping $\mathbf{v}_1$ fixed. Each panel shows a preview of the resulting 2-d projection if $\mathbf{v}_2$ were rotated by 180° in a particular rotation plane. The user can click and hold on a particular

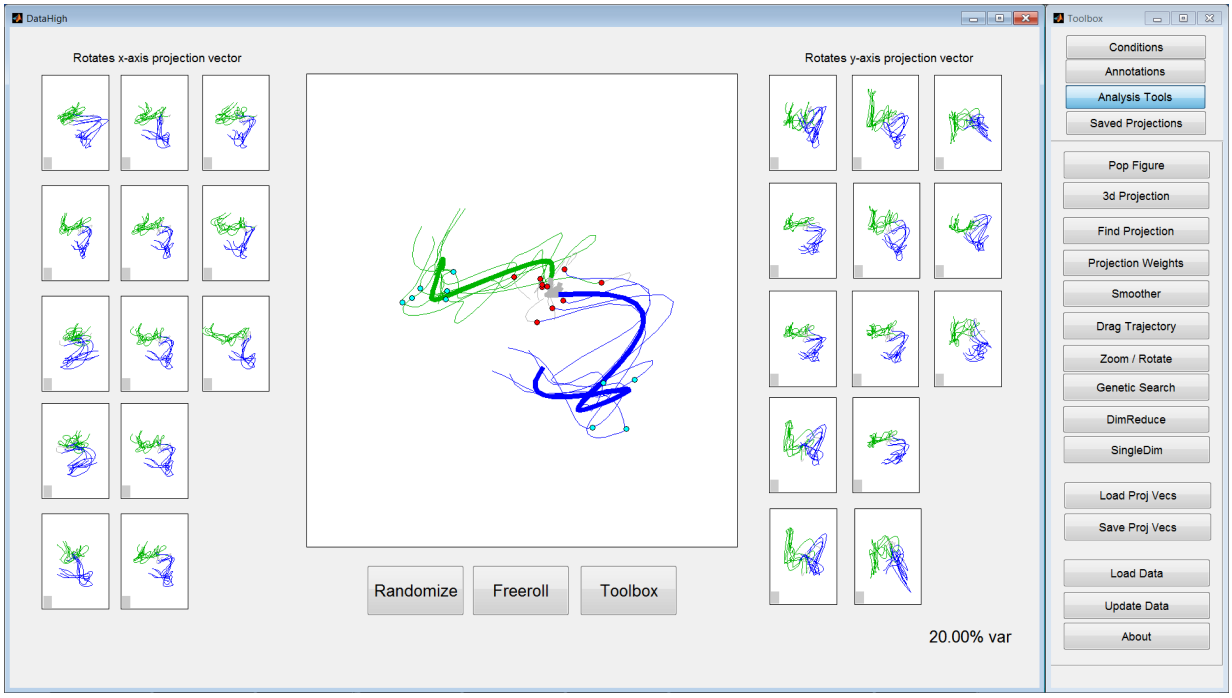


Figure 7.9: Main interface for DataHigh. Central panel: 2-d projection of 15-d single-trial neural trajectories extracted using GPFA from population activity recorded in premotor cortex during a standard delayed-reaching task for two different reach targets (green and blue). Dots indicate time of target onset (red) and the go cue (cyan). Gray indicates baseline activity before stimulus onset. Preview panels (left and right of central panel): clicking and holding on a preview panel instantly rotates one of the two projection vectors that make up the central 2-d projection. The bottom right corner shows the percent variance of the latent space that is captured by the central 2-d projection. The Toolbar (far right) allows the user to access analysis tools.

preview panel, which continuously updates the central panel as $\mathbf{v}_2$ is rotated smoothly in that plane. Similarly, we can fix $\mathbf{v}_2$ and rotate $\mathbf{v}_1$, which yields an additional $(k-2)$ preview panels (left-hand panels in Fig. 7.9). Thus, $2 \cdot (k-2)$ "knobs" are the least required to choose any possible 2-d projection of the k-d space.

Explicitly, we first use the Gram-Schmidt process to find a set of $(k-1)$ orthonormal vectors spanning the orthogonal space of $\mathbf{v}_1$; these vectors define the columns of $Q \in \mathbb{R}^{k \times (k-1)}$. We also define a rotation matrix $R_i(\theta) \in \mathbb{R}^{(k-1)\times(k-1)}$, which rotates a $(k-1)$ dimensional vector by an angle θ in the ith rotation plane.

$$R_i(\theta) = \begin{bmatrix} I_{i-1} & & & \\ & \cos(\theta) & -\sin(\theta) & \\ & \sin(\theta) & \cos(\theta) & \\ & & & I_{k-i-2} \end{bmatrix}, \tag{7.1}$$

where I_p is a $p \times p$ identity matrix and $i = 1, \ldots, k-2$. To rotate $\mathbf{v}_2$ by an angle θ in the ith

rotation plane, we compute

$$\mathbf{v}_2^{\text{new}} = Q\, R_i(\theta)\, Q^{\mathrm{T}}\, \mathbf{v}_2^{\text{old}}. \tag{7.2}$$

The neural trajectories shown in Fig. 7.9 are 15-dimensional ($k = 15$), leading to the use of 13 preview panels for $\mathbf{v}_1$ (the x-axis projection vector) and 13 preview panels for $\mathbf{v}_2$ (the y-axis projection vector). At present, DataHigh can support dimensionalities up to $k = 17$ (30 preview panels), which we found to be large enough for most current analyses, yet small enough to have all preview panels displayed simultaneously on a standard monitor. For $k > 17$, DataHigh applies PCA to the neural states or neural trajectories and retains the top 17 PCA dimensions for visualization. Alternatively, the user may implement a larger number of preview panels in DataHigh.

7.3.3 Visualization tools within DataHigh

In addition to the continuous rotation of a 2-d projection plane, DataHigh offers a suite of additional analysis tools that are useful for exploratory data analysis. For example, *Find Projection* automatically rotates the current projection vectors to static projections found by PCA, LDA, or other methods. *Genetic Search* allows the user to see many different views of the population activity and then to home in on features of interest. The initial projections are randomly chosen from all possible 2-d projections. Akin to a genetic algorithm, the user then selects the most 'interesting' projections, and a new iteration occurs in which 15 new 2-d projections are generated to be similar to the selected projections. This continues until the user uploads a projection to the main display. The *Evolve* tool highlights the timing differences between neural trajectories by displaying a movie of the neural trajectories playing out together over time. This movie can be saved for external viewing. Many other tools exist to help the user visualize and interact with neural population activity.

7.3.4 DimReduce

To facilitate the application of dimensionality reduction methods to neural data, we developed an optional dimensionality reduction tool to DataHigh, called DimReduce (Fig. 7.10). The user inputs raw spike trains, and DimReduce extracts the corresponding latent variables, which are automatically uploaded to DataHigh (left-hand side of *dimensionality reduction* in Fig. 7.8). DimReduce gives step-by-step instructions, embedded in the GUI with large red numbers and the "Next Step" button (Fig. 7.10), to perform dimensionality reduction, and guides the user to choose parameters that specify how to preprocess the data (e.g., time bin width, mean firing rate threshold, and smoothing kernel width), to select a dimensionality reduction method, and to specify a set of candidate dimensionalities for cross-validation. To identify the optimal number of latent dimensions, DimReduce computes a cross-validated data likelihood and a cross-validated leave-neuron-out prediction error (Yu et al., 2009) for each user-specified candidate dimensionality. DimReduce plots these cross-validated metrics versus latent dimensionality, and the user can inspect these plots to find the latent dimensionality which maximizes the cross-validated data likelihood or minimizes the leave-neuron-out prediction error. Using a slider in the GUI, the

user can then specify the number of latent variables DimReduce should extract and upload to DataHigh. To give the user a preview of the extracted latent variables, the tool displays a random 2-d projection of the latent space and a heat map of the loading matrix, which describes how each latent variable contributes to each neuron's activity. The tool also includes a viewer that plots the timecourse for each latent variable. The tool provides help buttons for how to choose a dimensionality reduction method and select preprocessing parameters, as well as how to interpret the cross-validated metrics.

DimReduce includes five linear dimensionality reduction methods that are computationally fast and have been fruitfully applied to analyze neural population activity. The five methods included are principal component analysis (PCA), Fisher's linear discriminant analysis (LDA), probabilistic principal component analysis (PPCA), factor analysis (FA), and Gaussian-process factor analysis (GPFA) (Bishop, 2006; Yu et al., 2009). If the user would like to use a dimensionality reduction method that is not currently available in DimReduce, the user may extract the latent variables outside the DataHigh environment and input them directly into DataHigh for visualization (right-hand side of *dimensionality reduction* in Fig. 7.8). For example, latent variables can be extracted by locally-linear embedding (Stopfer et al., 2003), jPCA (Churchland et al., 2012), demixed principal component analysis (dPCA) (Brendel et al., 2011), and linear dynamical systems (Macke et al., 2011; Yu et al., 2009). These methods may be added to future releases of DataHigh.

Selecting a dimensionality reduction method

The choice of dimensionality reduction method depends on the scientific questions of interest and the properties of the population activity to be visualized. Key considerations include whether the neurons were sequentially or simultaneously recorded, whether the user is interested in comparing single-trial or trial-averaged population activity, and whether the user is interested in the timecourse of the population activity. For each class of neural data, we shall recommend a corresponding dimensionality reduction method that we believe is appropriate for visualization purposes.

If the user is interested in trial-to-trial variability, where there is a single spike count vector (i.e., a point in multi-dimensional firing rate space) for each trial, we suggest FA. FA attempts to remove the independent Poisson-like spiking variability to identify a set of shared factors that describes the co-fluctuations of the activity across the neural population. FA has advantages over PCA, which assumes no observation noise and is thus less effective at removing Poisson-like spiking variability, and PPCA, which assumes that each neuron has the same observation noise variance. In contrast, FA allows neurons with different mean firing rates to have different levels of observation noise, which better agrees with Poisson-like spiking variability that depends on mean firing rate.

If the user is interested in single-trial neural trajectories, we recommend using GPFA (Yu et al., 2009). GPFA extends FA to include temporal smoothing of the latent variables, where the level of smoothness is determined by the data. Since the publication of Yu et al. (2009), we have accelerated the running time of the GPFA Matlab code by two orders of magnitude for the same hardware. For example, applying GPFA (i.e., running the EM algorithm once to convergence) to 100 neurons and 200 experimental trials now takes less than a minute on a standard single-processor laptop computer.

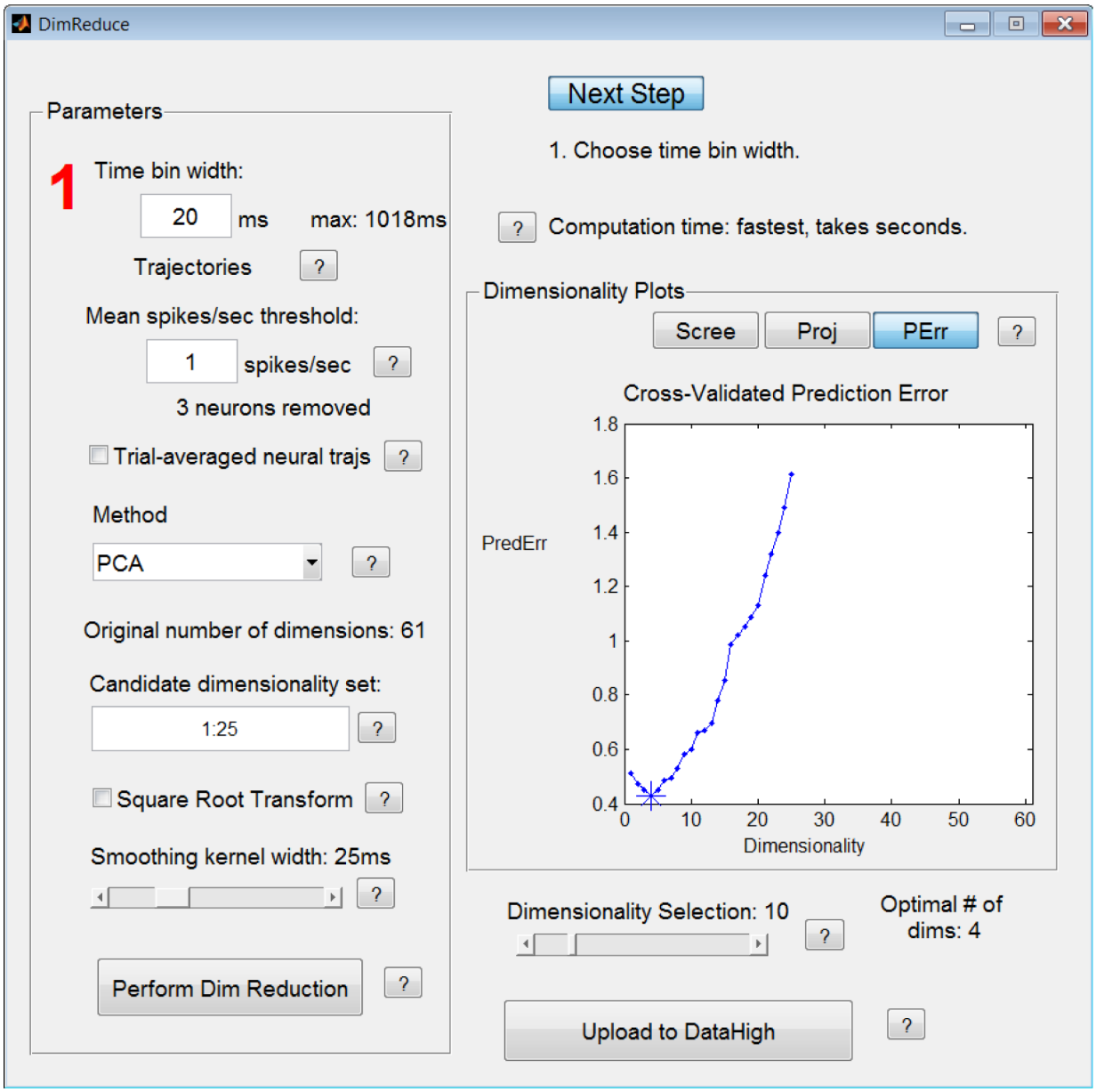


Figure 7.10: DimReduce allows the user to input raw spike trains, perform dimensionality reduction, choose the latent dimensionality, and upload the extracted latent variables to DataHigh. The large red “1" instructs the user where to complete the first step, which is to choose a bin width. Clicking the “Next Step" button increments the red step number and moves it to the next step. The example here shows a plot of leave-neuron-out prediction error versus candidate latent dimensionality. Using this metric, the optimal latent dimensionality is the dimensionality with the minimum cross-validated prediction error (starred on the plot).

If the user is interested in comparing trial-averaged population activity across different experimental conditions, we suggest using PCA, which identifies latent variables (i.e., principal components) that correspond to axes of greatest variance. In contrast to PPCA and FA, PCA does not have an explicit observation noise model to help remove the Poisson-like spiking variability. The use of PCA is well-justified here because trial-averaging (to produce the PSTHs that are input to PCA) likely removes much of the Poisson-like spiking variability, especially as the number of trials that are averaged together increases. If averaging over a small number of trials, the user can choose to apply kernel smoothing to the PSTHs.

Choosing the number of latent variables for visualization

One typically identifies the optimal latent dimensionality from the data by performing cross-validation on a range of candidate dimensionalities. Cross-validation can be time-consuming because each candidate dimensionality needs a separate set of model parameters to be fit for each cross-validation fold. For visualization purposes, finding the optimal dimensionality is not crucial, and choosing a number of latent variables that capture most of the richness in the data usually suffices. To do this, we suggest first performing dimensionality reduction with a large number of latent variables (e.g., a 50-d latent space for 100 neurons) without cross-validation. Then, choose a small subset of top latent variables that explain most of the variability. For PCA and PPCA, common methods for choosing this subset involve viewing the eigenspectrum of the sample covariance matrix and looking for a "bend at the elbow," or choosing a dimensionality that explains greater than 90% of the variance. Likewise for FA and GPFA, one may view the eigenspectrum of the shared covariance matrix and again look for an elbow. The shared covariance matrix is defined by CC^{T}, where C (number of neurons $\times$ number of latent variables) is the loading matrix. A complementary view is provided by plotting the timecourse separately for each orthonormalized latent variable (Yu et al., 2009). Orthonormalization has two key benefits: i) it provides an ordering of the latent variables from greatest to least covariance explained (similar to PCA), and ii) the units of the latent variables will be the same as the units of the observed variables (in this case, spike counts). After orthonormalization, the user can focus on visualizing the top dimensions. As one proceeds to the lower dimensions, one typically sees less variability in the latent variables over time, across trials, or across conditions. The user can estimate visually how many of the top dimensions are needed to capture the interesting structure in the population activity.

7.3.5 Discussion of DataHigh

We found DataHigh to be particularly useful for exploratory neural data analysis, where the relationship between the activity of any individual neuron and externally-imposed (e.g., sensory stimulus) or externally-measurable (e.g., subject's behavior) quantities may be unknown. By first applying dimensionality reduction to neural population activity, we attempt to relate the activity of each neuron to the activity of other neurons (either recorded simultaneously or sequentially). These relationships are captured by the identified latent variables, which describe how the activity of the neurons covary. In principle, we would like the population activity to first "speak for itself" via the identified latent variables. Then, we can attempt to relate the latent variables

to externally-imposed or externally-measurable quantities. DataHigh facilitates the process of building intuition about how the latent variables (and, by extension, the population activity) vary over time, across trials, and across conditions. More generally, DataHigh can be used to visualize the latent variables of any model of population activity, including generalized linear models (GLMs) (Vidne et al., 2012).

Visualization can be a valuable component of the data analysis procedure. For example, Bartho et al. (2009) presented sustained auditory tones and, as a first step of analysis, visualized the corresponding trial-averaged neural trajectories to gain insight into the population activity recorded from the auditory cortex. The visualization of the neural trajectories suggested that the sustained population response differed in magnitude and direction from the transient population response directly after stimulus onset. They then formed hypotheses about what they had seen in the visualizations and conducted statistical tests to either confirm or reject the hypotheses. Visualization with DataHigh, in tandem with statistical techniques (including statistical tests, decoding algorithms, models of population activity, and simulations), provides a powerful analysis framework to build intuition and to test scientific hypotheses.

DataHigh's code is released as open source, so that users may modify and add to the suite of available tools. This may include adding more dimensionality reduction methods to DimReduce, implementing projection-finding algorithms based on some cost function, such as jPCA (Churchland et al., 2012) or dPCA (Brendel et al., 2011), and extending DataHigh for real-time visualization of population activity during experiments. The DataHigh software package for Matlab can be downloaded from `http://www.cs.cmu.edu/bcowley/software/datahigh.html`.

This is the work of Benjamin Cowley, Matthew Kaufman, Zachary Butler, Mark Churchland, Stephen Ryu, Krishna Shenoy, and Byron Yu (Cowley et al., 2013).

Chapter 8

Discussion

To understand the computations of the brain, we must be able to understand the activity of its many neurons. One fruitful approach is dimensionality reduction, which reduces the number of variables one analyzes from the number of recorded neurons (typically tens to hundreds) to a smaller number of latent variables. Because dimensionality reduction relates the activity of a neuron to other neurons, it makes little to no assumptions about how neurons encode stimulus information or influence behavior. Thus, dimensionality reduction can uncover interactions among neurons, stimulus information, and behavior that single-neuron modeling may miss. Most neuroscientific studies have employed dimensionality reduction to neural activity primarily from the motor cortex (Churchland et al., 2010, 2012; Cunningham and Yu, 2014; Santhanam et al., 2009; Yu et al., 2009) and prefrontal cortex (Mante et al., 2013; Rigotti et al., 2013). This thesis investigated the sensibility and interpretability of dimensionality reduction for neural activity from the visual cortex. We performed many dimensionality reduction analyses to population activity from brain areas whose response properties are well-known (i.e., primary visual cortex, V1), as well as from brain areas whose response properties are less understood (i.e., visual area V4, prefrontal cortex). We found that the outputs of dimensionality reduction applied to neural activity were sensible and interpretable. This provides a strong foundation of when to apply dimensionality reduction, how to interpret its outputs, and which dimensionality reduction method to choose for different types of data. Because neural activity from the visual cortex posed new challenges for dimensionality reduction, we developed and employed new statistical methods to address these challenges. This thesis makes advances in both machine learning and neuroscience, which we outline below. We then provide caveats of dimensionality reduction, and describe its potential role in neural data analysis. Finally, we conclude with future extensions to the work in this thesis.

8.0.1 Advances in machine learning

This thesis makes two primary contributions to machine learning. The first contribution is an adaptive stimulus selection called Adept. Previous adaptive stimulus selection methods considered optimizing the response of a single neuron (Benda et al., 2007; DiMattina and Zhang, 2014), whereas Adept can optimize the responses of a *population* of simultaneously-recorded neurons. Experimenters can now choose from many different population objective functions to achieve

desired responses. The population objective functions can be intuitive, such as maximizing the responses for all neurons (e.g., the $\mathbb{L}_2$ norm of the response vector) or maximizing the scatter of responses (i.e., the average distance between observed response vectors). Adept is not dependent on a specific encoding model, but instead requires a notion of similarity between stimuli (e.g., a kernel function or psychometrically-defined similarity). For our experiments in which we optimized the responses of a population of V4 neurons, we used activity in the middle layer of a deep convolutional neural network (CNN) as feature embeddings for natural images. Previous studies have found that the representation of the CNN neurons resemble the representation of V4 neurons (Yamins and DiCarlo, 2016), but the representation is likely not perfect. To account for this representational mismatch, we incorporated kernel regression into the framework. Thus, our adpative stimulus selection paradigm combines active learning, deep learning, and kernel regression. This work is a successful example of integrating machine learning into closed-loop neuroscientific experiments.

The second contribution to machine learning is a novel dimensionality reduction method called distance covariance analysis (DCA). DCA is a general dimensionality reduction method (i.e., not specific to neuroscience) that combines the interpretability of linear dimensions with the ability to detect linear and nonlinear interactions. DCA is one of only a few dimensionality reduction methods that can identify dimensions for three or more sets of variables. It can also consider continuous or discrete variables, and take dependent variables into account. We tested DCA against other state-of-the-art methods on previously-used testbeds and on a novel testbed that systematically changed the interactions between sets of variables from linear to nonlinear. For these testbeds, DCA performed better than or equal to the state-of-the-art, especially excelling at identifying nonlinear interactions. DCA is computationally fast (performing faster than many of tested methods) and scales to many variables and samples. Thus, DCA is a powerful and flexible exploratory data analysis tool, and can also be used as a dimensionality reduction pre-processing step before performing regression or classification.

Overall, this thesis highlights new problems in neuroscience that can be solved with machine learning. New machine learning methods (and corresponding statistical theory) will be needed in the future to begin to understand the large quantity of data produced by current neuroscience experiments.

8.0.2 Advances in neuroscience

In addition to the advances in machine learning, this thesis also makes important contributions to neuroscience. We outline the primary contributions below.

First, this thesis builds a strong foundation for applying dimensionality reduction to neural activity. We first validated dimensionality reduction for neuroscience by applying it to neural activity recorded from a brain area for which the neuronal response properties are well-known (macaque primary visual cortex, V1). By performing systematic analyses that varied the number of neurons, trials, and stimulus classes, we found that the outputs of dimensionality reduction were sensible and interpretable. We further applied dimensionality reduction to different encoding models of V1 population activity. The outputs of dimensionality reduction, in comparison to the percent explainable variance used to characterize a model's output, revealed salient differences between the output of these models and the real neural activity. These differences can then be

examined in an effort to improve current models.

This thesis also proposes a novel dimensionality reduction method (DCA) that can be used to study interactions between population activity simultaneously recorded from multiple brain areas. These interactions are likely nonlinear, which cannot be captured by linear methods, such as canonical correlation analysis or partial least squares, but can be captured by DCA. DCA may also be used to identify latent variables of the population activity that are related to stimulus or behavioral variables of the experiment. Because DCA returns linear dimensions, the latent variables can be linearly mapped to the activity of the observed neurons. This is useful for asking scientific questions about which axes of firing rate space are employed by the population. To make DCA straightforward and easy to apply to neural activity, we ensured that DCA does not require hyperparameters and is computationally fast (e.g., < 10 minutes for 100s of neurons and 1000s of samples). While we have only applied DCA to spike count data, we intend to apply DCA to other types of neural activity, including calcium imaging traces and MEG data.

Next, to encourage a diversity of population responses, we propose in this thesis an adaptive stimulus selection algorithm (Adept) for optimizing the responses for a population of neurons. The intended use is to run Adept for a small amount of recording time (e.g., a few recording sessions for a chronic multi-electrode array or ~1 hour for acute experiments) to obtain a set of stimuli that drives a population of neurons, and then use this set of stimuli to probe other mechanisms of the population, such as attention, working memory, and other internal signals, whose effects likely depend on the response levels of the recorded neurons Cohen and Kohn (2011). Adept provides a principled approach for experimenters wanting to incorporate natural stimuli over artificial stimuli in their experiments (Felsen and Dan, 2005). Randomly-chosen natural stimuli may not elicit large responses from the recorded neurons because the chosen natural stimuli out of all possible natural stimuli are unlikely to be "preferred" by the neurons. Instead, one can use adaptive stimulus selection to efficiently navigate through the search space of natural stimuli to find stimuli that better resemble the "preferred" stimuli of the recorded neurons. As a technical note, the technology to run closed-loop experiments is no longer a hurdle for experimental labs desiring adaptive stimulus selection, as this technology is currently being utilized in many labs researching brain-computer interfaces. This is evidenced by our closed-loop experiments in a vision neuroscience lab.

The final contribution of this thesis to neuroscience is using dimensionality reduction to uncover a novel global fluctuation that slowly drifts the activity of neurons in at least two brain areas (V4 and PFC). Unlike previously-identified global fluctuations, this slow drift operates on the order of 30 minutes and co-modulates the activity of neurons in different ways, as opposed to a common gain on firing rates (Lin et al., 2015; Okun et al., 2015; Rabinowitz et al., 2015). We addressed two questions that arise from the existence of the slow drift. First, we asked how the slow drift affects the fidelity of downstream readout, and found evidence that the downstream area removes this slow drift from its readout. Second, we asked what neural mechanism could cause the slow drift. We found that the slow drift co-varies with slow changes in behavioral variables, such as hit rate, false alarm rate, pupil diameter, and reaction time. This suggests that the slow drift is an arousal signal. One candidate brain area from which the slow drift may originate is the locus coeruleus, a brainstem nucleus. The locus coeruleus releases the neuromodulator epinephrine on a similar time scale as that of the slow drift, and the activity of the locus coeruleus is closely linked with pupil diameter, a behavioral variable linked with arousal (Aston-Jones and Cohen,

2005; Joshi et al., 2016).

This work characterizes a global fluctuation likely present in population activity recorded in many brain areas and during many different experimental tasks. This suggests that the slow drift could influence effect size of other measured internal mechanisms, such as attention, as well as hinder closed-loop experiments involving adaptive stimulus selection by affecting estimates of mean responses to the presented stimuli. This work also speaks to measuring the information content of populations of sensory neurons (Kohn et al., 2016). Our work suggests that some correlated neural noise, which can be large and aligned to downstream readout axes, nonetheless can be removed by downstream areas for faithful readout of the stimulus information. Thus, even differential correlations, which are the only correlations to harm downstream readout (Moreno-Bote et al., 2014), may also be harmless if the downstream area also has access to the internal signals causing the differential correlations (Bondy et al., 2018; Kohn et al., 2016).

8.0.3 Caveats of dimensionality reduction

One over-arching conclusion of this thesis is that dimensionality reduction is a valuable tool to understand population activity. However, we do not advocate that computational neuroscientists should abandon other approaches and adopt dimensionality reduction as the only path forward in understanding population activity. On the contrary, we advocate that dimensionality reduction is complementary to many other approaches, such as other statistical models (Pillow et al., 2008; Yamins and DiCarlo, 2016) and estimating the information content of population activity (Kanitscheider et al., 2015; Kohn et al., 2016; Moreno-Bote et al., 2014). At the heart of performing dimensionality reduction is the idea that we desire to summarize the joint distribution of the activity of many neurons in a succint, interpretable way. By understanding the joint distribution, we can uncover sources of noise (either caused by artifacts in the recording technology or internal signals of the brain), begin to understand the geometry of responses to different stimuli in high-dimensional firing rate space, and analyze single-trial time courses of the neural activity. Dimensionality reduction is not only a visualization tool, but also can be used for basic science. For example, one pivotal study found monkeys were unable to learn how to express new population activity patterns (e.g., dimensions) in a short amount of time, even when those patterns were crucial for obtaining reward (Sadtler et al., 2014).

One natural goal for using dimensionality reduction for basic science is to estimate an absolute dimensionality of the neurons in a brain area. For example, one study recorded from higher-cortical visual neurons in the inferior temporal (IT) cortex, and asymptotically estimated that the total number of dimensions in IT was near 100 (Lehky et al., 2014). This suggests that experimenters may only need to record from ~100 IT neurons to capture all information present in the activity of all IT neurons. A theoretical study has found that the absolute dimensionality also likely depends on the task complexity (Gao et al., 2017). Under this theory, responses of sensory neurons to presented stimuli generated by varying a few parameters (e.g., sinusoidal gratings) are expected to be summarized by a smaller number of latent variables than responses of the same neurons to stimuli generated by varying many parameters (e.g., natural images), consistent with our results in Chapter 3. In this thesis, we make no claims about absolute dimensionality. Instead, we advocate for relative comparisons in dimensionality, in which we fix the number of neurons, number of trials, dimensionality reduction method, the metric used to assess the

number of dimensions, etc. We make conclusions about the dimensionality of the data only after controlling for these confounds, because small changes in these confounds can yield large changes in dimensionality.

8.0.4 Future work

We discuss at length three directions in which this thesis can be extended.

Dimensionality reduction for many, many neurons

Recording technologies have advanced to the point where experimenters can now record from thousands to tens of thousands of neurons simultaneously (Ahrens et al., 2013; Jun et al., 2017; Pachitariu et al., 2016). While these technologies will give us unprecedented access to the neural circuit, it raises new data analysis questions. For example, even though the number of recorded neurons has increased, the amount of recording time remains the same. This introduces difficulty in accurately estimating the covariance matrix of the neurons, which is used by many dimensionality reduction methods. This was exemplified by our work in Chapter 7, in which we had to increase the trial count to ~5,000 trials until factor analysis began to reveal the structure of the non-clustered spiking neural network (Williamson et al., 2016). Still, there are some theoretical guarantees that as we increase the number of neurons, we have more accurate estimates of the top identified dimensions (Bai et al., 2012; Williamson et al., 2016). New dimensionality reduction methods may need to be developed to identify latent variables that represent the activity of a sparse number of recorded neurons in an effort to make more interpretable conclusions about which neurons contribute to which latent variables. Some sparse dimensionality reduction methods already exist for this purpose, such as sparse PCA (Zou et al., 2006) and sparse CCA Witten and Tibshirani (2009). Likewise, scalable dimensionality reduction methods are needed to handle singular covariance matrices. Dimensions can be iteratively identified with stochastic gradient versions of PCA and DCA, and likely many other algorithms (Cunningham and Ghahramani, 2015).

Another important advance of recording technologies is the knowledge of the spatial and cell type information of the recorded neurons, typically with calcium imaging (Stosiek et al., 2003). New dimensionality reduction methods that can take this information into account are needed. For example, we can incorporate the assumption that neurons that are spatially close together are more likely to contribute to the same latent variables. Likewise, new dimensionality reduction methods that incorporate cell type information may better isolate latent variables reflecting activity purely from excitatory neurons, purely from inhibitory neurons, or both (Bittner et al., 2017; Williamson et al., 2016). These analyses may give us better understanding of the balance between excitation and inhibition (Litwin-Kumar and Doiron, 2012; Renart et al., 2010).

Because the dimensionality of neural activity likely depends on the task complexity (Gao et al., 2017), if too simple of a task is performed by the subject, recording from more neurons may not give us more insight into the computations of the circuit. Likewise, because the fundamental limit of data will be the amount of recording time, adaptive stimulus selection methods will be even more imperative to drive many neurons in diverse ways. For example, an experimenter may use adative stimulus selection to measure the number of total possible activity patterns that can be

expressed by the recorded population, which in turn can help elucidate computations carried out by the neural circuit and potentially the underlying functional connectivity (Okun et al., 2015; Sadtler et al., 2014).

Understanding interactions between brain areas

Another exciting direction in neuroscience is to simultaneously record from populations of neurons in multiple brain areas in order to understand the interactions among brain areas. This thesis provides initial work in this direction with a dimensionality reduction method that can identify linear and nonlinear interactions between brain areas (DCA). A key assumption of this work is that interactions between brain areas are captured by a small number of latent variables. While the validity of this assumption needs further investigation, our work on identifying interactions between macaque V4 and PFC appear to uphold this assumption for at least one kind of interaction. This also seems to be the case for interactions of trial-to-trial variability between V1 and V2 (Semedo et al., 2015, 2014).

One interesting aspect of multi-area interactions is identifying both linear and nonlinear interactions between brain areas. To separate the two types of interactions, one can first apply linear regression and remove any linear contribution of one brain area to another brain area. Then, one can apply DCA to identify any further dimensions that are related nonlinearly to one another. One can further make assumptions about the types of nonlinearites between brain areas and directly fit models based on those assumptions. For example, a commonly-used model is a generalized linear model (GLM) that takes a linear combination of upstream neurons and passes the combination through a pointwise nonlinearity to predict a downstream neuron's response (Pillow et al., 2008; Vidne et al., 2012). This compositional model of a linear-nonlinear interaction has been a critical component of deep neural networks (Yamins and DiCarlo, 2016). However, these types of models cannot capture all types of nonlinear interactions, including heteroskedastic relationships (see Chapter 5).

Another important aspect of multi-area interactions is to determine causality. For example, we found a slow drift between V4 and PFC. This slow drift may exist in PFC because PFC reads out the slow drift in the V4 activity. However, another possibility is that both V4 and PFC share a common source of the slow drift. Causal manipulations, such as electrical or optogenetic stimulation, are needed to determine causality (Deisseroth, 2011). Much work will be needed to manipulate neural circuits with realistic stimulation patterns that can be used to causally determine interactions between brain areas. These types of manipulations will help determine the different time scales on which the brain areas interact, as well as how the role of feedback plays in multi-area interactions.

Adaptive stimulus selection for temporal sequences of stimuli

To date, most adaptive stimulus selection methods have focused on the presentation of static stimuli. However, many of our sensory systems are active in how they collect and process stimulus information. For example, human eyes make multiple saccades per second to focus on visual objects of interest (Robinson, 1964). This motivates the development of methods that adaptively choose sequences of stimuli. Initial work on this topic involved choosing from a candidate pool of

stimulus sequences for each trial to optimize the firing rate of a single neuron to fit parameters of an encoding model (Lewi et al., 2011). Extending this work to optimize the responses of multiple neurons will require new population objective functions that account for responses across neurons and across time. For adaptive stimulus selection for a population of V4 neurons, one may pass natural movie sequences through a deep convolutional neural network to extract feature embeddings, and then use these feature embeddings to determine similarities between movie sequences. However, it is unclear if these feature embeddings will be predictive of neural responses, as the latter are dependent on adaptation effects (Kohn, 2007) and higher-cortical feedback (Moore and Armstrong, 2003). Also, because the number of possible movie sequences involves an exponentially larger search space than natural images (which itself is exponentially large), generating movie sequences rather than selecting from a finite pool of movie sequences may allow adaptive methods to better optimize their objective functions on the fly. These movies may be generated with generative adversarial networks (Goodfellow et al., 2014) or maximum-entropy models (Loaiza-Ganem et al., 2017).

These directions of future work represent opportunities to advance both machine learning and neuroscience. It is clear that dimensionality reduction will play a major role in understanding the activity of many neurons within one brain area and among multiple brain areas.

Appendix A

Materials and Methods for Chapter 3

A.1 Neural recordings

Details of the neural recordings have been described previously (Kelly et al., 2010; Smith and Kohn, 2008). Briefly, we recorded from primary visual cortex (V1) of anesthetized, paralyzed macaque male monkeys. Anesthesia was administered throughout the experiment with a continuous intravenous infusion of sufentanil citrate (6-18 μg/kg/hr). Eye movements were minimized with a continuous intravenous infusion of vecuronium bromide (100-150 μg/kg/hr). Experiments typically lasted 5-7 days. All experimental procedures followed guidelines approved by the Institutional Animal Care and Use Committee of the Albert Einstein College of Medicine at Yeshiva University, and were in full compliance with the guidelines set forth in the US Public Health Service Guide for the Care and Use of Laboratory Animals.

Neural activity was recorded using 96-channel multi-electrode arrays (Blackrock Microsystems, Salt Lake City, Utah), which covered 12.96 mm^2 and had an electrode length of 1 mm. The electrodes were inserted to a nominal depth of 0.6 mm to confine recordings mostly to layers 2-3. Recordings were performed in parafoveal V1, with RFs within 5 degrees of the fovea. Voltage waveform segments that passed a separately-chosen threshold for each channel were later spike-sorted offline. For comparisons of population responses to different orientation gratings, we spike sorted responses together across all orientations, thereby obtaining a common set of units. Similarly, we spike sorted responses together across all three movies. We included sorted units for which the voltage waveform had a signal-to-noise ratio (SNR) greater than 1.5, where SNR is defined as the ratio of the average waveform amplitude to the standard deviation of the waveform noise (Kelly et al., 2007). This SNR threshold yielded both single-unit and multi-unit activity (Wissig and Kohn, 2012, see) for comparison of these signals) with a median SNR near 2.5 for all datasets. We analyzed only neurons with mean firing rates greater than 1 spike per second.

A.2 Visual stimuli

We used two sets of visual stimuli. The first set (termed the individual gratings set) consisted of individual presentations of drifting sinusoidal gratings with different orientations. The second set (termed the movies set) consisted of different classes of visual stimuli, including a sequence of

drifting sinusoidal gratings with different orientations, a contiguous sequence of natural scenes, and white noise. All stimuli were presented on a CRT monitor with a frame rate of 100 or 120 Hz, and had mean luminance of approximately 40 cd/m^2. We used a look-up table for all stimuli to correct for the nonlinearity between input voltage and output luminance in the monitor.

Individual gratings Full-contrast (100%) drifting sinusoidal gratings with 12 equally-spaced orientations (30° between adjacent orientations, covering 360°) were presented. We use "orientation" to refer to the angle of drift and "direction" to refer to two orientations that are 180° apart (i.e., opposing drift) (Movshon and Newsome, 1996). Spatial frequency (1.3 cpd) and temporal frequency (6.25 Hz) were chosen to evoke robust responses from the population as a whole, and the position and size (8-10 degrees) were sufficient to cover the RFs of all recorded neurons. Gratings were block-randomized across the 12 orientations and presented for 1.28 seconds each, followed by a 1.5 second inter-trial interval consisting of an isoluminant gray screen. We conducted 200 trials for each of the 12 orientations.

Movies We presented three different 30-second grayscale movies: gratings, natural, and noise, as described previously (Kelly et al., 2010). Each movie comprised 750 unique images (each presented for four consecutive video refreshes of the CRT, for an effective framerate of 25 Hz). The movie frames were surrounded by a gray field of average luminance. The mean (normalized) RMS contrasts were 0.17, 0.17, and 0.09 for the gratings, natural, and noise movies, respectively. Thus, the global statistics were matched for the gratings and natural movies.

The gratings movie was a pseudo-randomly chosen sequence of full-contrast drifting sinusoidal gratings with 98 equally-spaced orientations (3.67° between adjacent orientations, covering 360°). The presentation of each drifting grating of a particular orientation lasted 300 ms. Spatial frequency (1.3 cpd), temporal frequency (6.25 Hz), position and size (8 degrees or 640 pixel diameter circular aperature) were chosen to evoke high firing rates and to sufficiently cover RFs of all recorded neurons. Two 300 ms periods of blank screen frames were included at a randomly-chosen position in the sequence, bringing the total to 100 stimuli in a block lasting 30 seconds. A 30-second movie with the same random sequence of gratings and blank screens was repeated 120 times.

The natural movie was a 30 second consumer film of a contiguous sequence of natural scenes converted to grayscale (a monkey wading through water). The movie was displayed in a square of 5 degrees (400×400 pixels) to cover all RFs, and repeated 120 times.

The noise movie was 30 seconds of white noise, where each 4 degree (320×320 pixel) frame comprised a 40×40 grid of 8-pixel squares. Each 8×8-pixel square displayed a random intensity drawn from a uniform distribution independently (in space and time) of other squares. The entire 40×40 grid was randomly shifted or "jittered" k pixels ($1 \leq k \leq 8$) horizontally and vertically between consecutive frames to avoid high-contrast grid effects. One noise movie was randomly generated, and the same movie was repeated 120 times.

A.3 Preprocessing of population activity and visual stimuli

Population activity for individual gratings We presented the individual gratings stimulus set to three monkeys (101r, 102l, 103r). After removing neurons that did not satisfy the SNR and firing rate criteria, the lowest number of neurons across monkeys was 61. Because measurements of dimensionality can be influenced by the size of the recorded population, we selected a random subset of 61 neurons from the data sets with larger populations, in order to combine results across monkeys (dimensionality trends were similar across monkeys for all analyses). We considered neural activity in a 1 second window starting at stimulus onset (discarding the remaining 0.28 seconds of response). Results were similar for a 1 second window starting 100 ms after stimulus onset to avoid onset transient effects. For each neuron and orientation, we took spike counts in 20 ms bins and averaged them across trials to create a peri-stimulus time histogram (PSTH), yielding 50 time points. Because a step in PCA subtracts from each PSTH the mean across the 50 time points, we can compare (across different orientations) fluctuations of trial-averaged activity around its mean.

Population activity for movies We presented the movies stimulus set to two monkeys (monkey 1: 103l, 61 neurons, monkey 2: 102l, 81 neurons). For each neuron, we took spike counts in 20 ms bins during the 30-second movie presentation and averaged them across trials to create a PSTH. For each movie, this yielded a PSTH with 1,500 time points for each neuron. PCA then centers each PSTH by subtracting its mean, enabling the study of fluctuations of the firing rate around the mean. We ensured the neural activity was centered just once for all analyses (i.e., for non-overlapping one-second time windows, the “local" mean was not subtracted again). In this way, all dimensionality measurements are made using a common reference frame, thereby allowing comparison of dimensionality across different time windows (analysis of Fig. 3.5*B*).

Visual stimuli for movies To relate neural complexity to stimulus complexity, we performed PCA on the movie stimuli. This required the following pre-processing steps in order to match the sizes of images across stimuli and remove incidental spatial correlations in the noise movie. The processed movies were only used for PCA analysis; the original, unprocessed movies were shown to subjects. We first cropped the grating and natural image frames to 320 x 320 pixels to match the size of the noise images. The noise movie had two incidental spatial correlations from the experimental design that prevented it from being “true" white noise: spatial correlations caused by noise images being comprised squares of 8×8 pixels with identical intensities and spatial correlations caused by jittering the squares to avoid grid effects. To weaken these correlations, we converted the images of all movies to 8×8 pixel squares. The average pixel intensity was computed for each pixel square, and each image was compressed from a matrix of 320×320 values to a matrix of 40×40 values. Because we averaged over fixed pixel squares, we did not eliminate all incidental spatial correlations in the noise movie, as the borders between the 8×8 pixels with identical intensities were jittered randomly from frame-to-frame. Thus, the red eigenspectrum curve in Fig. 3.3*B* is not perfectly diagonal, which is expected from true white noise. Each 40×40 pixel image was then reshaped into a vector (with size $1,600 \times 1$). Each movie stimulus was therefore represented as 750 vectors (one for each image) in a 1,600-dimensional

space, where each axis corresponds to the average intensity of one pixel block.

A.4 Assessing dimensionality and similarity of patterns

Assessing dimensionality We describe here how we assessed the dimensionality of the population activity and that of the visual stimuli. Conceptually, the dimensionality is the number of basis patterns needed to describe the population activity (Fig. 3.1) or the pixel intensities of the visual stimuli. There are many ways to assess dimensionality, including linear (Cunningham and Ghahramani, 2015) and nonlinear (Roweis and Saul, 2000; Tenenbaum et al., 2000) methods. Here, we focus on the most basic linear dimensionality reduction method, principal component analysis (PCA). PCA is appropriate for use with trial-averaged neural activity because trial averaging removes much of the Poisson-like spiking variability, consistent with PCA's implicit assumption of no observation noise (Cunningham and Yu, 2014). PCA has also been applied to pixel intensities in previous studies (Lehky et al., 2014; Simoncelli and Olshausen, 2001).

PCA dimensionality was assessed as the number of basis patterns needed to explain 90% of the variance in the population activity or visual stimuli. The threshold of 90% is arbitrary, and we verified that the results were qualitatively similar for other high-percentage thresholds (e.g., 70% and 80%). It is possible to use a data-driven approach to determine the variance threshold (Lehky et al., 2014). Under this approach, data which require a small or large number of dimensions to explain the majority of the variance can be deemed low-dimensional. To aid in interpreting dimensionality comparisons, we chose to use a pre-determined (90%) variance threshold in this work. It is important to note that the dimensionality depends on many factors, including the dimensionality reduction method (here, PCA), metric for assessing dimensionality (here, 90% variance explained), the number of neurons, and the number of data points used. Thus, we do not attempt to interpret the dimensionality in an absolute sense. Rather, we make relative comparisons of dimensionality across experimental conditions, while keeping these other factors fixed.

Assessing similarity of patterns In addition to computing the number of basis patterns needed for each condition, we sought to compare the patterns across conditions. The following is the intuition for performing this comparison. Consider two sets of patterns: condition A requires k_A patterns and condition B requires k_B patterns, where $k_A > k_B$. At one extreme, the space spanned by the condition A patterns includes the space spanned by the condition B patterns (i.e., the spaces overlap). In this case, the joint space is described by $k_{AB} = \max(k_A, k_B) = k_A$ patterns. At the other extreme, the space spanned by the condition A patterns is orthogonal to the space spanned by the condition B patterns. In this case, the joint space is described by $k_{AB} = k_A + k_B$ patterns. In general, k_{AB} will lie between $\max(k_A, k_B)$ and $k_A + k_B$. The closer k_{AB} is to $\max(k_A, k_B)$, the more similar the patterns are. Conversely, the closer k_{AB} is to $k_A + k_B$, the more dissimilar the patterns are.

To compute k_{AB}, we tried several different approaches. In the first approach, we aggregated the population activity (or visual stimuli) for different conditions, then applied the same PCA method to find the number of dimensions that explained 90% of the variance. A problem with this approach is that it does not account for possibly different variances of the population activity (or visual stimuli) across conditions. The consequence is that k_{AB} obtained by this method can be less

than $\max(k_A, k_B)$, thereby violating the intuition laid out above. To see this, consider a scenario where $k_A > k_B$ and condition B has much larger variance than condition A. The aggregated population activity (or visual stimuli) would be dominated by condition B, essentially ignoring the patterns of condition A. As a result, the aggregated dimensionality k_{AB} would be close to k_B, which is less than $\max(k_A, k_B) = k_A$. This motivated us to consider a second approach in which we normalized the population activity (or visual stimuli) such that the direction of greatest variance was 1 for each condition. However, there are still scenarios for which k_{AB} obtained by this method is less than $\max(k_A, k_B)$, due to how the variance is distributed across patterns.

Pattern aggregation method To overcome the issues described above, we developed an alternative approach which guaranteed that k_{AB} would be between $\max(k_A, k_B)$ and $k_A + k_B$. This method, termed the pattern aggregation method, is based on first identifying the k_A patterns for condition A (represented by $U_A : N \times k_A$, whose columns are orthonormal and where N is the number of pixels or the number of neurons) and k_B patterns for condition B (represented by $U_B : N \times k_B$, whose columns are orthonormal) separately using PCA. Then, we aggregate the patterns into a single matrix $V = [U_A U_B]$ (where $V : N \times (k_A + k_B)$) and designate k_{AB} to be the effective column rank (defined below) of V. In the case where $k_A > k_B$ and the space spanned by the condition *A* patterns includes the space spanned by the condition *B* patterns, then k_{AB} will be $\max(k_A, k_B) = k_A$. On the other hand, if the space spanned by the condition A patterns is orthogonal to the space spanned by the condition B patterns, then k_{AB} will be $k_A + k_B$. For more than two conditions, this method is easily extended by aggregating the patterns for all conditions into $V = [U_A U_B U_C \ldots]$.

The effective column rank of V is the number of "large" singular values of V. The definition of "large" requires setting a threshold (termed the rank threshold, between 0 and 1) for the singular values. The rank threshold determines how different two basis patterns (cf. Fig. 3.1, green) need to be before they define separate dimensions. To gain intuition for the rank threshold, we ran a simulation where we rotated a 2-d unit vector (v_B) from a position of overlap with another unit vector (v_A, 0°) to a position of orthogonality (90°) (Fig. B.1*A*). We varied the rank threshold for different angles between v_A and v_B, and assessed the effective column rank of V. A rank threshold $t = 0$ means that slight deviations of v_B away from v_A lead to $V = [v_A v_B]$ having an effective column rank of two (Fig. B.1*B*). In other words, if two patterns are slightly different, the dimensionality of the data will be two. On the other hand, a rank threshold $t = 1$ means that v_B needs to be orthogonal to v_A for V to have an effective column rank of two (Fig. B.1*B*). In other words, two patterns need to be orthogonal for the dimensionality of the data to be two; else, the dimensionality will be one. As a compromise between the two extremes, we used a rank threshold $t = 0.5$ throughout this study. This threshold means that the transition from one to two dimensions occurs when the angle between v_A and v_B is near 45 degrees (Fig. B.1*B*). We also verified that this intuition holds for higher-dimensional spaces by rotating a subspace towards an orthogonal subspace by taking a convex combination of the two subspaces and measuring the rank at intermediate rotations.

Computing dimensionality expected by chance To assess whether two sets of patterns are similar or dissimilar, it is necessary to compare k_{AB} to a chance level, rather than simply asking

whether k_{AB} is closer to the minimum dimensionality bound $\max(k_A, k_B)$ or maximum dimensionality bound $k_A + k_B$. The reason is that the chance level depends on the relative values of k_A, k_B, and N, and does not simply lie halfway in between $\max(k_A, k_B)$ and $k_A + k_B$. For example, if N is large relative to k_A and k_B, the chance level would lie near $k_A + k_B$ because randomly drawn patterns in a high-dimensional space tend to be orthogonal. We compute the chance level by drawing patterns randomly from the N-dimensional space, and compute a distribution of aggregated dimensionalities. Specifically, we repeatedly draw k_A orthonormal patterns and k_B orthonormal patterns in an N-dimensional space, and measure their aggregated dimensionality using the same method as applied to the data.

The above method makes the implicit assumption that all patterns within the N-dimensional space can be shown in the data. In some cases, there is reason to believe that not all patterns in the N-dimensional space can be achieved. For example, the underlying neural circuitry may constrain the space of activity patterns that a population of neurons is capable of producing (Luczak et al., 2009; Sadtler et al., 2014). In such settings, the chance level should be computed by drawing patterns randomly from an M-dimensional space, where $M < N$. Ideally, the range of M should be determined using a large number of stimuli and/or time points to obtain as accurate an estimate of the space of activity patterns that can be produced as possible. We can then assess whether the results hold for all values of M within this range.

Similarity index The similarity of two sets of patterns depends, relative to the two extremes ($\max(k_A, k_B)$ and $k_A + k_B$), on whether k_{AB} is larger or smaller than the chance level. We summarized this dependence with a single number, the similarity index. The similarity index is defined as $s = \frac{\hat{k}_{AB} - k_{AB}}{k_A + k_B - \max(k_A, k_B)}$. Intuitively, we take the difference between the mean chance dimensionality ($\hat{k}_{AB}$) and the actual dimensionality for the aggregated patterns of A and B (k_{AB}). Then, we normalize this difference by the difference between the two extreme dimensionalities, yielding an index between -1 and 1. A similarity index of $s > 0$ means that the patterns are more similar (i.e., more overlapping) than expected by chance, whereas $s < 0$ means that the patterns are less similar (i.e., closer to orthogonal) than expected by chance.

A.5 Statistical assessment of dimensionality

Error bars for the dimensionality of the population activity were computed by subsampling from all time points. We chose subsampling over bootstrapping, since bootstrapping led to biased estimates due to small sample size relative to the number of neurons. For population responses to the stimulus movies, we randomly subsampled 750 of the 1,500 time points, computed the dimensionality of the subsampled points, and repeated this 100 times (Fig. 3.4*C*). Similarly, for the dimensionality analysis in 1 second windows, we randomly sampled 25 of the 50 time points 100 times (Fig. 3.5*A*). We did not compute error bars for the visual stimuli because subsampling was not possible (only 750 available time points), and bootstrapping was not possible due to the small number of time points relative to the dimensionality (1,600) of the pixel space. All p-values were computed from 10^5 runs of a random permutation test.

A.6 Receptive field model

We considered a recently-proposed RF model of V1 neurons (Goris et al., 2015) that includes four components — oriented Gabor filtering, subtraction of untuned suppressive filtering, divisive normalization, and pointwise nonlinearity — common to many RF models of V1 neurons (Carandini et al., 2005). We input the individual gratings and movie stimuli into the model, and asked how the dimensionality ordering after each model component compares to that of the population activity. The parameter values for 100 model neurons were drawn from the distributions reported in Goris et al. (2015) rather than fit to data because we did not present the mixed gratings stimuli necessary for fitting the parameters. For this reason, we compare dimensionality trends between the model and population activity, rather than their absolute values.

Component 1: Oriented Gabor filtering The first component for each model neuron was an oriented Gabor filter applied to the input image. Because Goris et al. (2015) incorporates direction selectivity only when parameterizing the orientation tuning curve of a neuron, we needed a way to incorporate temporal filtering that would be applicable to any image sequence and allow for direction selectivity. We adopted a straightforward approach to extend the 2-d Gabor filter to a 3-d Gabor filter, which considers both space and time (Yun and Guan, 2013). The following equation describes the 3-d Gabor filter G for pixel location (x, y) and time index t:

$$G(x, y, t) = \text{real}\{H(x, y, t) \cdot S(x, y, t)\} \tag{A.1}$$

where

$$H(x, y, t) = \exp\big[-\frac{1}{2}(\tilde{x}^2 + \tilde{y}^2 + \tilde{t}^2)\big]$$
$$S(x, y, t) = \exp\big[-j2\pi(\gamma_s x' + \phi + \gamma_t t')\big]$$

and $[\tilde{x}, \tilde{y}, \tilde{t}]$ and $[x', y', t']$ are defined below. Thus, the Gabor filter is the product of a Gaussian envelope H and a sinusoid S. To orient the Gabor, the pixel locations (x, y) are first rotated by orientation angle θ. To incorporate direction selectivity, (x, y, t) are further rotated by angle β, which is either $0°$ or $180°$. These rotations are achieved with a rotation matrix, R:

$$R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\beta) & -\sin(\beta) \\ 0 & \sin(\beta) & \cos(\beta) \end{bmatrix} \times \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{A.2}$$

One can then compute the rotated $[x', y', t']^T = R \times [(x - x_{\text{loc}}), (y - y_{\text{loc}}), t]^T$ and the scaled rotated $[\tilde{x}, \tilde{y}, \tilde{t}]^T = R \times [(x - x_{\text{loc}})/\sigma_x, (y - y_{\text{loc}})/\sigma_y, t/\sigma_t]^T$, where (x, y, t) are scaled by $(\sigma_x, \sigma_y, \sigma_t)$ before rotation to align the Gaussian envelope with the orientation angle. The Gabor filter G is normalized by its Frobenius norm.

We chose the parameter values for each model neuron in the following manner to be biologically plausible. The center of the RFs $(x_{\text{loc}}, y_{\text{loc}})$ were drawn from a Gaussian with mean 160 and variance 20 to be partially overlapping at the center of the 320×320 pixel image. The spatial phase ϕ was drawn from a uniform distribution between 0 and 2π. The temporal frequency $\gamma_t = 6.25$ Hz

was fixed at the same temporal frequency as that for the drifting gratings. Orientation angle θ was drawn from a uniform distribution between 0 and π. The direction of drift β was randomly chosen from the set $\{0, \pi\}$. The sizes of the Gaussian envelope were $\sigma_x = 20 + |\epsilon|$ (where ϵ was drawn from a standard Gaussian) and $\sigma_y = c\sigma_x$ (where c is the aspect ratio drawn from the reported distribution in Goris et al. (2015)). The size of the temporal envelope was $\sigma_t = 8d$, where $d \in [0, 1]$ is the direction selectivity constant found in Goris et al. (2015). The spatial frequency γ_s was drawn from the reported distribution in Goris et al. (2015).

To compute the output of this model component at timestep t and considering the past $T = 15$ timesteps, the dot product of the 3-d Gabor filter (G) with the sequence of images ($I_{t,t-1,\ldots,t-T}$) was computed: $L_1 = \sum_{m=t-T}^{t} \sum_{x,y} G(x, y, m - t) I_m(x, y)$. This quantity was then passed through a linear rectifying function to yield the response $R_1 = \max(0, L_1)$.

Component 2: Subtraction of untuned Gabor filtering The next component of the model was to subtract from R_1 the response of an untuned suppressive filter to the current image. The untuned suppressive filter G_{untuned} was computed as a difference of 2-d Gaussians:

$$G_{\text{untuned}}(x, y) = G_{\sigma_1}(x, y) - G_{\sigma_2}(x, y) \tag{A.3}$$

$$\text{where } G_\sigma = \frac{1}{2\pi\sigma^2} \exp\left[-\frac{1}{2}\left(\frac{(x - x_{\text{loc}})^2}{\sigma^2} + \frac{(y - y_{\text{loc}})^2}{\sigma^2} \right) \right] \tag{A.4}$$

The RF centers $(x_{\text{loc}}, y_{\text{loc}})$ were kept the same as those for the first model component, and the spreads of the filters were fixed at $\sigma_1 = 20, \sigma_2 = 30$. The dot product of the untuned Gabor filter G_{untuned} and the current image I, computed as $L_2 = \sum_{x,y} G_{\text{untuned}}(x, y) I(x, y)$, was then passed through a weighted linear rectifying function $S_2 = \max\{0, \omega L_2\}$, where ω was drawn from the reported distribution in Goris et al. (2015). Finally, the output of this model component R_2 was the subtraction of the response of the untuned Gabor filter with the previous component's output: $R_2 = R_1 - S_2$.

Component 3: Divisive normalization The next component of the model is divisive normalization, which divides the output of the previous component with the mean output of all other model neurons from the first component and an additive offset. Let R_1^i and R_2^i refer to the output of the ith model neuron from the first and second model components, respectively. Then, the output of the ith model neuron after divisive normalization R_3^i was computed as follows:

$$R_3^i = \frac{R_2^i}{\sigma_{\text{offset}} + \frac{1}{N-1} \sum_{\substack{j=1, \\ j \neq i}}^{N} R_1^j} \tag{A.5}$$

where the additive offset σ_{offset} was drawn from the reported distribution in Goris et al. (2015), and the number of neurons $N = 100$.

Component 4: Pointwise nonlinearity The final output of the model was computed by passing the output of the previous model component through a pointwise nonlinearity:

$$R_4 = \left[\max\{0, R_3\}\right]^q \tag{A.6}$$

where the exponent q was drawn from the reported distribution in Goris et al. (2015).

Parametrically altering the visual stimuli To study how dimensionality of the model outputs changes as we parametrically alter the visual stimuli, we performed three analyses. First, we varied the contrast of the images of the natural movie in 5 different increments: 100%, 75%, 50%, 25%, 5%. For each image, we subtracted the mean luminance of that image, scaled the result by one of the percentages above, and added back the mean luminance. Second, we transformed the images of the natural movie to pink noise by randomizing the phase of the 2-d Fourier transform for each image. To avoid removing local contrast of the natural images (Wichmann et al., 2006), we added an offset to each phase, where the offset was randomly drawn from a uniform distribution over the range $[-\alpha, \alpha]$. We then considered 5 values for α: 0° (i.e., no change of the natural image), 45°, 90°, 135°, 180° (i.e., pink noise). Third, we transformed the pink noise images (which retained the power spectrum of the natural images) to white noise by raising the power spectrum of the pink noise to different fractional exponents. The intuition behind this is that the power spectrum of natural images falls off as $1/f^2$, where f is frequency (Simoncelli and Olshausen, 2001). Raising the power spectrum to a fractional exponent, for example $1/2$, transforms the $1/f^2$ fall-off to a $1/f$ fall-off. To ensure a transformed image did not exceed the pixel intensity range (i.e., $[0, 255]$), we normalized the power spectrum of each image by dividing by its sum of magnitudes λ. We then raised the normalized power spectrum to 5 different exponents: 1 (i.e., pink noise), $3/4$, $1/2$, $1/4$, 0 (i.e., white noise). Finally, we scaled the resulting power spectrum by λ.

A.7 Deep convolutional neural network

To assess how the ordering of dimensionality might change at different stages of visual processing, we studied a deep convolutional neural network (CNN). We used an instantiation of a CNN called GoogLeNet (Szegedy et al., 2015), trained by Xiao (2013), and available in Matlab with MatConvNet (Vedaldi and Lenc, 2015). The CNN had different processing units, including convolution, pooling, concatenation, and normalization/softmax. Each unit comprised many filters (from 10^3 to 10^6) that performed the same operation (e.g., convolution) but on different spatial regions of the input. Each layer comprised a group of units (shown in Fig. 3.8*A*). We assessed dimensionality of the outputs of eight consecutive layers of the deep network. For the first layer, we analyzed the outputs of 100 filters of the second normalization unit. For layers 2 to 8, we analyzed the outputs of 100 filters of the concatenation units. For each analyzed unit, we chose the 100 filters to have the closest RFs to the center of the image.

Appendix B

Supplemental figures for Chapter 3

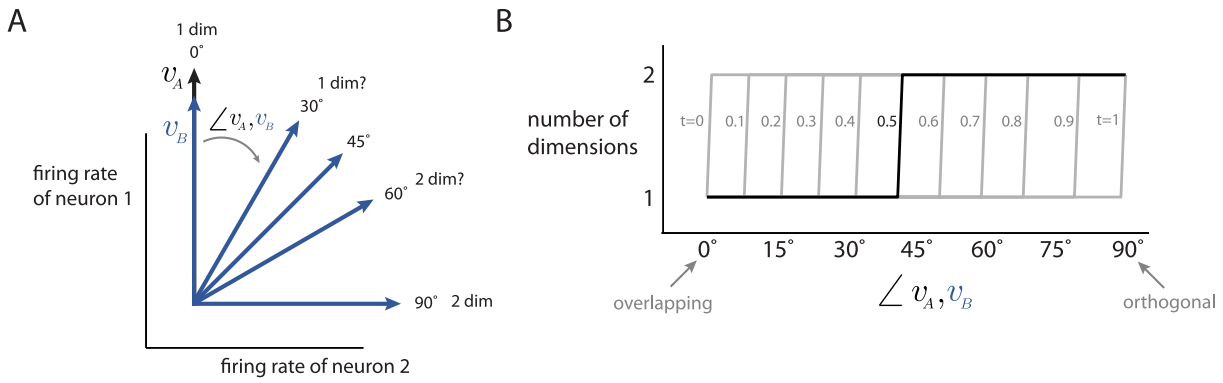


Figure B.1: Assessing similarity of basis patterns. *A*: Conceptual illustration for two neurons, where v_A denotes the basis pattern for one condition and v_B denotes the basis pattern for another condition. As v_B is rotated away from v_A, the question is at which point do we consider v_A and v_B to span two dimensions. *B*: The transition from one to two dimensions in *A* depends on the rank threshold t. For $t = 0.5$, the transition occurs when the angle between v_A and v_B is near 45 degrees.

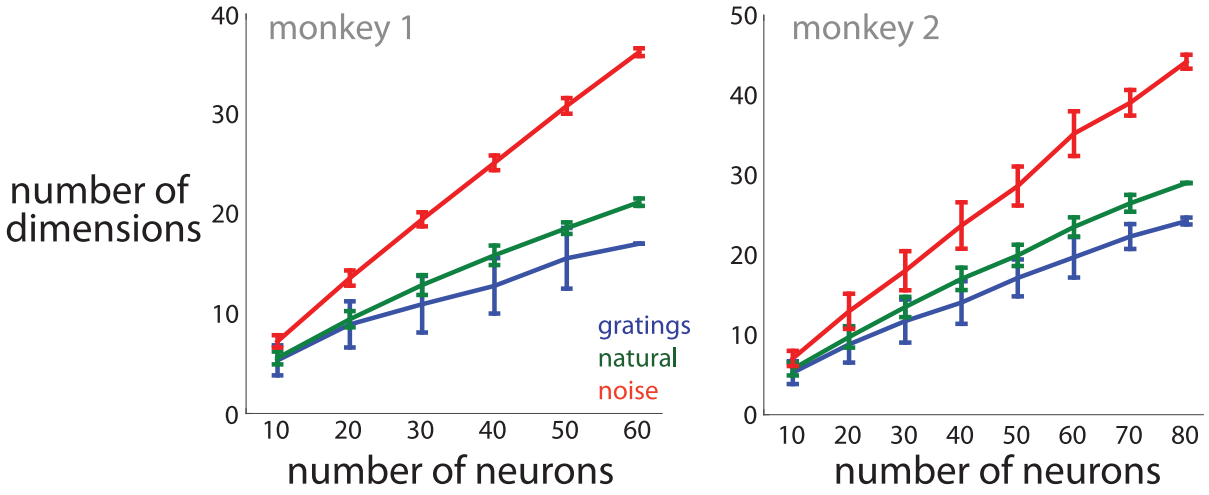


Figure B.2: The ordering of dimensionality of the population responses to the movie stimuli remained consistent for a wide range of neuron counts. We randomly subsampled a smaller number of neurons from the 61 neurons of monkey 1 (left panel) and from the 81 neurons of monkey 2 (right panel). We then computed the dimensionality for the subsampled population responses to each movie separately. We assessed dimensionality as the number of dimensions needed to explain 90% of the variance. Error bars represent the standard deviation for 50 different subsamples.

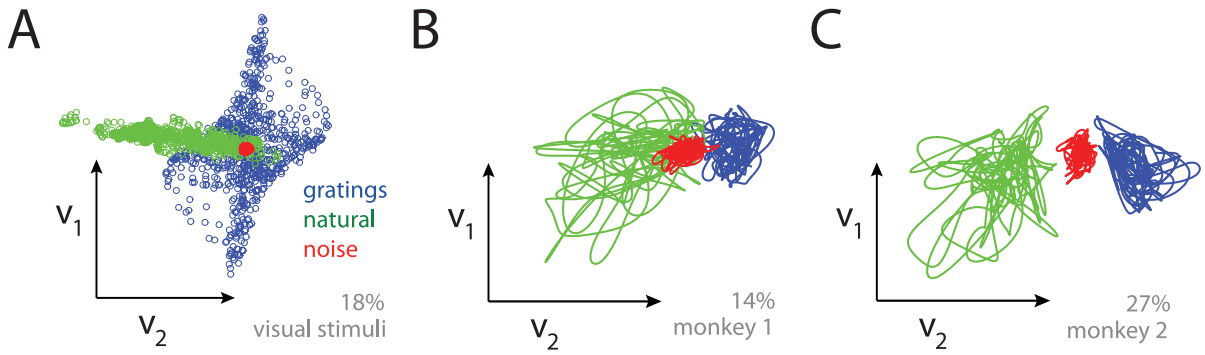


Figure B.3: Two-dimensional projections for the visual stimuli and population activity. Projections were found using the DataHigh software (Cowley et al., 2013), to show variance within each movie and separation between movies. *A*: Projection for the pixel intensities for the visual stimuli, capturing 18% of the total variance. Each dot is an image, and v_1 and v_2 are orthonormal projection vectors of the high-dimensional pixel space. *B*: Projections for the population responses for monkey 1 (left panel, 14% of the total variance) and monkey 2 (right panel, 27% of the total variance). Each trajectory traces out the population activity timecourse for one movie (gratings: blue, natural: green, noise: red). The orthonormal projection vectors v_1 and v_2 of the high-dimensional firing rate spaces are computed separately for each monkey.

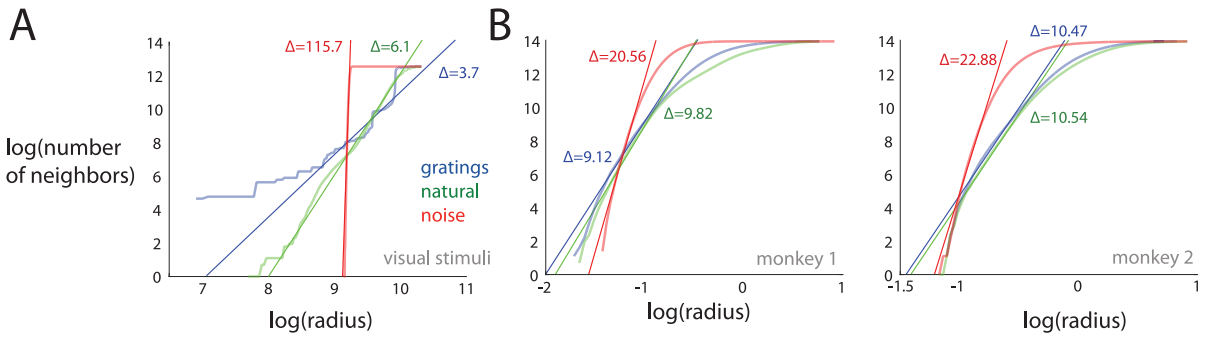


Figure B.4: Fractal dimensionality of the visual stimuli and population responses. Fractal dimensionality was computed by the same method as in Lehky et al. (2014). The intuition behind fractal dimensionality is to define a hypersphere with radius r and ask how the number of points N contained within that sphere grows as as r increases. The faster N grows with r, the higher the dimensionality. We plot $\log(N)$ versus $\log(r)$, where the slope is defined to be the fractal dimensionality. *A*: Fractal dimensionality of the pixel intensities of the gratings (blue), natural (green), and noise (red) movies. Slopes (Δ) were computed by linear regression between the bounds of 6 and 12 of the $\log$ number of neighbors. *B*: Fractal dimensionality of the population responses to the gratings, natural, and noise movies for monkey 1 (left panel) and monkey 2 (right panel). Slopes were computed by linear regression between the bounds of 4 and 10 of the $\log$ number of neighbors.

Appendix C

Supplemental figure for Chapter 5

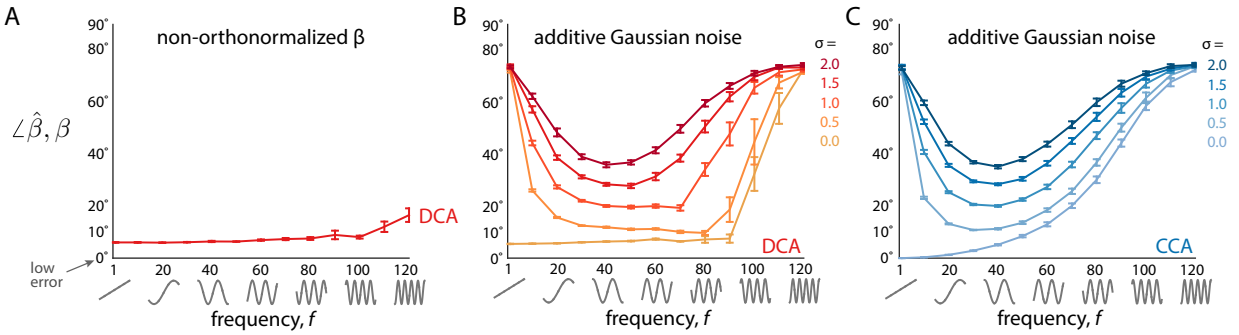


Figure C.1: Results of modified testbeds for identifying dimensions for two sets of variables (Section 6.2). (A) DCA shows low error (measured by angle of overlap) across many frequencies for non-orthonormalized β. The same testbed was used as that in Section 6.2, except that the columns of $\beta = [\beta_1, \ldots, \beta_5]$ were not orthnormalized. To ensure $\beta_i^T X \in [-\frac{\pi}{4}, \frac{\pi}{4}]$, each β_i was normalized to have a norm of 1. (B) DCA's performance decays gradually in the presence of additive Gaussian noise. The same testbed was used as that in Section 6.2, except that we corrupted Y with additive Gaussian noise: $Y = [\tilde{y}_1, \ldots, \tilde{y}_5]^T$, where $\tilde{y}_i = y_i + \sigma\epsilon$ and $\epsilon \sim \mathcal{N}(0, 1)$. We varied σ between 0 and 2.0. (C) CCA's performance decays in a similar manner as that of DCA. Same testbed as in (B).

Appendix D

Appendix for Chapter 6

D.1 Materials and Methods for Chapter 6

D.1.1 Subjects and electrophysiological recordings.

Two adult male rhesus monkeys (*Macaca mulatta*) were subjects in this study. All animal procedures were in accordance with the [Univ of Pitt Animal Care committee]. A monkey was trained on the task for several months, after which we implanted two 10×10 arrays of microelectrodes (Blackrock Microsystems), one in V4 (anatomical site) and one in PFC (anatomical site).

We recorded 47 sessions (one session per day) in total (24 from monkey 1, 23 from monkey 2). Monkeys were head-fixed and rewarded with water for correct trials. Each session comprised at least four blocks that alternated the cued target location. The cued target location switched after 150 successfully-completed trials. Spike sorting was performed semi-manually with in-house software (Kelly et al., 2007). Our data consisted of both well-isolated single units and multiunits, and we refer to each unit as a "neuron" in the paper. After applying rigorous spike waveform controls (see below), each session had 7-54 recorded neurons (24-54 for monkey 1 and 7-35 for monkey 2) with an average of 31 neurons (39 for monkey 1 and 20 for monkey 2).

D.1.2 Controls for electrode shift.

One possible cause of the slow drift is that the electrode arrays gradually shift their position throughout the recording session. Shifts in electrode position have been observed in mice, and linked to locomotion (Mante et al., 2018). However, in our experiments, the monkeys were head fixed, rarely moved their body during a session, and the electrode arrays were chronically implanted (i.e., fixed to the skull). To rigorously control for electrode drift, we also removed neurons whose differences in spike waveforms across a session may indicate electrode shift. For example, the distance between the electrode and neuron may change across a session, increasing or decreasing the neuron's spike waveform amplitude. We developed a metric (percent residual waveform variance) to capture this effect. To compute the percent residual waveform variance, we first divided the session into 10 non-overlapping time bins equal in size, and computed the mean spike waveform for each bin as well as the mean spike waveform across the entire session. We computed the variance of the residual mean waveform for each bin (i.e., the bin's mean waveform

minus the overall mean waveform). The percent residual waveform variance is the largest residual variance across bins divided by the variance of the overall mean waveform. Example mean spike waveforms and their corresponding percent residual variances are shown in Supp. Fig. D.4. We removed neurons whose percent residual variance was below 10%.

Another indication of electrode shift are sudden jumps in a neuron's activity. While our percent residual variance threshold removed most neurons with these jumps, we observed a small number of neurons with such jumps in their activity. We developed another metric to account for these jumps. First, we divided the session into 20 non-overlapping time bins equal in size, and computed the mean spike count of a neuron in each bin. We then normalized the spike counts by the largest spike count across bins. Finally, we computed the largest absolute difference between consecutive time bins, which could take values between 0 and 1. We removed any neurons whose difference was larger than 0.25 (i.e., a neuron whose activity jumped more than 25% between two consecutive time bins).

Finally, if an electrode shift gradually occurred throughout the session, the spike waveform shape would likely change throughout a session. For example, if the distance between an electrode and a neuron increases, the voltage amplitude of a spike waveform may decrease. We developed a metric (residual time correlation) that measures how gradually a spike waveform changes shape. To compute the residual time correlation, we computed the residual mean spike waveforms for the same 10 time bins as those used to compute the percent residual waveform variance. The residual time correlation is the absolute correlation between the starting time of each bin and the sums of the residual waveforms across the time of the spike. For example, if the amplitude of a neuron's spike waveforms is higher than the overall mean spike waveform at the start of the session but gradually decreases to be lower than the overall mean waveform at the end of the session, the residual time correlation will be close to 1 because early mean waveforms would integrate to a positive sum while later mean waveforms would integrate to a negative sum. We then correlated the slow drift's projection vector weight of each neuron with the percent residual variance and with the residual time correlation, and found weak correlations (Supp Fig. D.4). These results suggests that any slow drift present in the remaining neurons is likely not caused by electrode shift.

D.1.3 Orientation-change detection task.

The animal performed an orientation-change detection task. At the start of each trial, the animal fixated at a central cue (1.5 degree circular fixation window at the center of the display). The computer monitor of size 1024×768 pixels refreshed at a rate of 120 Hz, and was gamma corrected. After a period of fixation that lasted for 150 ms, two achromatic drifting sinusoidal gratings were presented with one stimulus inside the receptive field locations of the recorded V4 neurons and the other stimulus diametrically opposite. The size, location, as well as the spatial and temporal frequencies of the stimuli were optimized to elicit large firing rates across the population but fixed throughout all recording sessions. These sample stimuli were flashed on for 400 ms and off for a randomized period (300-500 ms picked from a uniform distribution between each stimulus flash). The sample stimuli could be one of two pairs of orientation angles (each equally likely to be shown): 45° on the left visual hemifield and 135° on the right hemifield, or vice versa. The next flash had a 40% chance of changing the orientation angle of one of the stimuli. The animal

was given a liquid reward for making a saccade to the changed stimulus within 400 ms of its appearance. If the monkey broke fixation during the inter-flash interval or did not saccade to one of the targets, the trial was aborted and no reward was given.

Trials were divided into blocks of "cue in" trials or "cue out" trials. For "cue in" trials, the stimulus inside the recorded V4 neurons' receptive fields had an 80% of changing its angle, whereas for "cue out" trials, the stimulus outside the receptive fields had an 80% chance of changing its orientation angle. To signal the start of a new block, the animal performed 5 instruction trials with a single stimulus presented for one flash before two-stimuli flashes began. The instruction trials were not analyzed in this work.

D.1.4 Estimating the slow drift.

To estimate the slow drift, we first computed the residual population activity by taking the mean spike counts across repeats of the same sample stimulus and subtracting these means from the spike counts of the corresponding flashes. The residual spike counts were then averaged in large time bins (21 minutes), where the start of each time bin was shifted by increments of 6 minutes. Principal component analysis was applied to the smoothed residual spike counts, and top principal component explained ~70% of the smoothed residual spike count variance (Supp. Fig. D.3). We defined the axis of the top principal component as the *slow drift axis*. The (non-smoothed) residual spike counts were then projected along the slow drift axis, and kernel smoothing (radial basis function with a kernel bandwidth of 9 minutes) was used to smooth the projections. We defined the kernel-smoothed projections as the *slow drift*. To measure the relative size of the slow drift, we applied factor analysis to the residual spike counts to identify the total variability shared across neurons. The size was computed as a fraction of the slow drift variance divided by the total shared residual spike count variance. Finally, the angle between the slow drift axis and the $[1, 1, \ldots, 1]$ axis was computed to assess if most of the projection vector weights of the slow drift axis had the same sign. For reference, the angles between the $[1, 1, \ldots, 1]$ axis and two axes that were randomly generated over many runs. The first axis had projection vector weights drawn from a standard Gaussian and then normalized. The second axis was generated in the same manner as the first axis, except the projection vector weights of the second axis were flipped to have the same sign.

D.1.5 Relating the slow drift to slow changes in behavioral variables.

We measured six behavioral variables throughout each session. 1. Hit rate was the number of correct saccades towards a changed target divided by the total number of times a sample stimulus changed. 2. False alarm rate was the number of saccades toward a sample flash that did not change divided by the total number of presented sample stimuli after the initial flash of the sample stimuli. 3. The trial length was the number of total flashes of stimuli (sample or changed) per trial. 4. Reaction time was the time between stimulus onset and saccade onset. 5. Pupil diameter was the mean diameter of the pupil during each flash of sample stimuli (not including initial flashes of sample stimuli) measured with the EyeLink eye-tracking software. 6. The d' was $\phi(\text{hit rate}) - \phi(\text{false alarm rate})$, where $\phi(\cdot)$ is the inverse cumulative distribution function of the Gaussian distribution and where hit rate and false alarm rate were computed as mentioned

above. A running mean of each behavioral variable (except pupil diameter, for which a running median was taken) was taken over long time bins (30 minutes). The length of the time bin was necessary to have accurate estimates of hit rate, false alarm rate, and d'. For fair comparison, we also estimated the slow drift as a running mean of residual spike counts in 30 minute time bins only for the analyses in Figure 6.3. The slow drift was the top principal component of the smoothed residual spike counts. To assess the relative size of the slow drift, we applied factor analysis to the spike counts taken in 400 ms bins during presentations of sample stimuli, and computed the fraction of the slow drift variance divided by the total shared variance of the spike counts.

Aligning the slow drift axes across sessions.

While the behavioral variables had an absolute reference (i.e., a higher hit rate indicates more correct saccades are being made), the slow drift axis was only identifiable up to a 180° rotation. In other words, the sign of the slow drift could be flipped. To establish an absolute reference frame for the slow drift across sessions that was independent of behavior, we flipped the sign of the slow drift such that when the mean spike counts of the two sample stimuli (45° and 135°) were projected along the slow drift axis, the mean spike count of the 45° sample stimulus had a higher projection value than that of the 135° sample stimulus. The reasoning behind this flipping procedure was that the slow drift is likely aligned to the stimulus representation of the population of neurons, and that this alignment is similar across sessions. The expected chance correlation between the slow drift and a smoothed behavioral variable over time was computed by randomly flipping the sign of the slow drifts for many runs. Thus, if the flipping procedure was completely random, we would expect correlations no different from chance.

D.1.6 PFC slow drift.

In the same experiment in which we recorded a population of V4 neurons, we simultaneously recorded from a population of neurons in the dorsolateral PFC (area 8ar). PFC neurons removed following the same spike waveform criteria as for the V4 neurons (Supp. Fig. D.10). The slow drift was computed in the same manner as the V4 slow drift. To compute the correlation between the relative sizes of the V4 and PFC slow drifts (Fig. 6.4*B*), we removed outlier sessions with a correlation between the V4 and PFC slow drift less than 0.8 (removed 5 sessions). Not removing these outliers resulted in a smaller but significant correlation (Pearson's $\rho = 0.38$, $p < 0.012$, random permutation test).

D.1.7 Comparing the size of the slow drift to that of attention.

To compare the relative sizes of the slow drift versus attention, we first defined the attention axis as the axis in firing rate space along which the mean spike count responses to the sample stimulus were the farthest apart between trials in which the sample stimulus was cued inside the V4 neurons' receptive fields and trials in which the sample stimulus was cued outside the receptive fields. This definition was used in another study (Cohen and Maunsell, 2010). We then computed the mean spike count response to the same sample stimulus along the attention axis for each block

of cued trials, and took the variance of the mean spike counts (i.e., the attention variance). For the slow drift variance, we computed the mean slow drift value for the same blocks of trials as that for attention, and took the variance of the mean values. To compare the relative sizes, we computed a ratio between the slow drift variance divided by the attention variance. A ratio greater than 1 indicates that the size of the slow drift is larger than the size of the attentional effect.

D.1.8 Decoding stimulus information of V4 population activity.

To assess the amount of stimulus information in V4 population activity with and without the slow drift, we computed decoding accuracies with a support vector machine (SVM) decoder and used leave-one-out cross-validation. For the slow drift axis and random axis, we first projected the V4 responses to the two sample stimuli (45° and 135°) onto the respective axis, and then decoded the projected activity. In this case, SVM acts as a 1-d threshold decoder. We only considered "cue in" trials. For the decoder axis, we allowed SVM access to the entire V4 population activity. In this case, SVM identified its own axis in firing rate space based on an objective function that minimized prediction error. Thus, the decoder axis need to be aligned to the slow drift axis.

D.1.9 Relating slow drift axis to stimulus-encoding axes

A downstream area likely optimizes its readout of V4 activity to extract information about many different stimuli rather than optimizing to discriminate between two arbitrary stimuli (e.g., sinusoidal gratings of 45° vs. 135° orientation angles). One assumption is that the downstream readout axis is aligned to an axis that best separates the mean spike count responses to many different images. This is akin to applying PCA to the mean spike count responses, and taking the top principal components as "stimulus-encoding axes." To have the V4 activity best express these stimulus-encoding axes, we employed adaptive stimulus selection to choose natural images that elicited large and diverse responses. On one recording session, we used an adaptive algorithm called Adept to choose a set of 2,000 natural images out of a candidate set of 20,000 (Cowley et al., 2017b). On following recording sessions, we showed repeats of these images randomly throughout a session. For each trial, the monkey fixated on a central cue, and remained fixating while we presented a 1 second clip of 10 natural images (each image lasted for 100 ms). After the clip, the fixation cue was replaced by a target cue, to which the monkey saccaded to receive a juice reward. Each image had 5 to 30 repeats throughout a session. For another monkey, we did not employ adaptive stimulus selection, but rather showed repeats of 550 natural images randomly throughout a session. The trial structure was the same as for the other monkey, except that the 1 second clip had three images shown for 200 ms interleaved with 200 ms blank screens.

During the sessions, we recorded from V4 using the Utah multi-electrode array (BlackRock Systems). We applied the same spike waveform criteria as for the other experiment. We then sought to compare the slow drift axis with the stimulus-encoding axes. Because it could be the case that some repeats were not uniformly distributed throughout a session, we computed the slow drift in the same manner as the other experiment except on the raw spike counts (not the residual activity). To compute the stimulus-encoding axes, we first subtracted from the mean spike count responses along each stimulus-encoding axis any slow drift, measured by kernel smoothing. This ensured that any similarity between the slow drift axis and a stimulus-encoding axis could

not be due to any residual drift in the mean spike count responses. We then applied PCA to the mean spike count responses, and took the loading vectors of the top PCs as the stimulus-encoding axes. To compare the similarity between the slow drift axis and each stimulus-encoding axis, we computed the fraction of slow drift variance captured by each stimulus-encoding axis. For reference, we rotated the slow drift axis to a random direction, and re-computed the fraction.

D.1.10 Predicting false alarms within a trial.

We asked to what extent did the V4 responses along the slow drift axis predict the occurrence of false alarms within a trial. In other words, we decoded spike count responses between the final flash and second-to-final flash. Spike counts were taken in time bins from 50 ms after stimulus onset to 20 ms before saccade onset or up to 200 ms, whichever came first. The same time bin was used for both final and previous-to-final flashes. Because the predictive ability of V4 responses was likely small, we employed two approaches. To gain statistical power for a small number of false alarm trials, we combined false alarm trials ("cue in" trials only) across the two sample stimuli (45° and 135°) by subtracting out the mean spike count responses to the previous-to-final flashes from the spike counts of both final and previous-to-final flashes. This procedure allowed us to roughly double the number of false alarm trials considered when decoding. In addition, the animal may have false alarmed because he perceived a positive or negative change in orientation angle. For the corresponding V4 responses, this could reflect an increase or decrease from the mean spike count response along the slow drift axis, for which a linear decoder would not be able to decode well. To account for this, we simply squared the re-centered responses, which caused outlying responses far from the mean response to both be heavily positive, which a linear decoder could decode well. We employed these approaches when decoding V4 activity with linear SVM using leave-one-out cross-validation. We performed two controls. First, we confirmed that these two approaches did not inject statistical bias into the decoding accuracies by first shuffling responses randomly between final and previous-to-final flashes. Second, we confirmed that the resulting decoding accuracies were not a by-product of adaptation-like effects. For each false alarm trial, we identified a matching trial whose sequence length M was at least one flash longer than the corresponding false alarm trial. We then decoded responses to the two flashes with identical sequence positions as the final and previous-to-final flash of the correpsonding false alarm trial. For example, if the final and previous-to-final flash of the false alarm trial occurred at positions M and $M-1$, respectively, then we found a corresponding trial of sequence length $\geq M+1$, and decoded responses to the M and $M-1$ flash. None of the matched trials were the same.

D.1.11 Models of perceptual decision-making

We proposed three models of perceptual decision-making. All models took as input stimulus input $\in \{5, 6\}$ and feedforward perceptual noise $\sim \mathcal{N}(0, \sigma^2)$ with $\sigma = 0.35$, and output a decision variable $d \in \{0, 1\}$. Similar to the experiment, each trial comprised a sequence of flashes where the next flash had a 40% chance of changing its stimulus input from $= 5$ to $= 6$. A correct trial is one in which the model outputs decision $d = 1$ when $= 6$. All models also had internal noise in the form of a slow drift variable $s = \frac{1}{2}\sin(2\pi\frac{1}{3500}(t - 500))$, where $t = 1, \ldots, 2,000$ is the

current trial number. To generate realistic decoding accuracies, all models also had internal noise affecting d by which there was a 35% chance each trial output either $d = 1$ or $d = 0$, independent of any other variables in the model. This internal noise reflects other internal processes in the brain from which we cannot record that may affect a subject's decision. We included a response variable v that represents V4 activity along the slow drift axis, which provides sensory information to the decision.

The first model treated the slow drift s as perceptual noise. Mathematically, $v = + + s$, and $d = 1$ if $v > 5.5$, $d = 0$ otherwise. An increase in s places v closer to the threshold 5.5, making it likelier for the model to false alarm. Likewise, a decrease in s places v farther from the threshold 5.5, requiring a high value of to register as a perceived change of stimulus, leading to a false alarm.

The second model builds upon the first model by adding a pathway independent on perception but dependent on the slow drift s. This pathway influences d via an arousal signal $a \sim$ Bernoulli(prob(s)). The probability of saccade prob(s) = $\frac{1}{2}(s + 0.5 + 0.01)$ is linearly dependent on s. Incorporating a, the updated decision $d = 1$ if $v > 5.5$ *or* $a = 1$, $d = 0$ otherwise. In this way, a can override the influence of perception $v > 5.5$.

The third model expands the second model to have a perception variable $p = v - s$ that removes the slow drift. In this model, s only affects the decision d through the arousal signal a. The decision output $d = 1$ if $p > 5.5$ or $a = 1$, $d = 0$ otherwise.

We computed the hit and false alarm rates with running estimates over the entire 2,000 trials (500-trial window length with 50-trial strides). We computed the extent to which v or $v - s$ predicted the occurrence of a false alarm within a trial by decoding the simulated activity for the final and previous-to-final flashes of false alarm trials. For all models, we used the optimal decoder, which predicted d with a threshold of 5.5 for v or $v - s$.

D.1.12 Statistical testing.

All hypothesis testing was conducted with random permutation tests. This included taking the difference between means or medians with paired and non-paired tests and assessing the significance of Pearson's correlations through shuffling.

D.2 Supplementary figures for Chapter 6

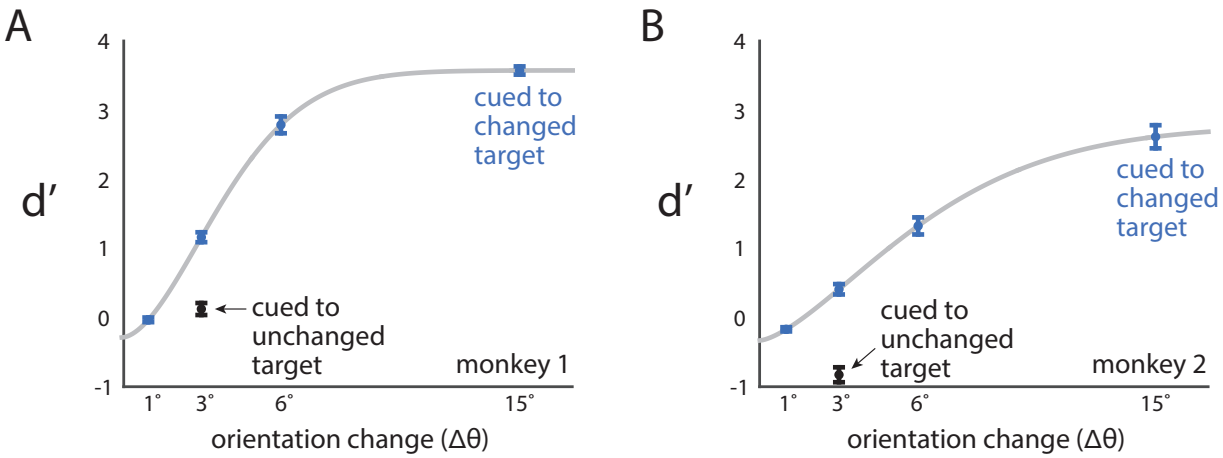


Figure D.1: Psychometric curves for both monkeys. *A*. Task performance (measured as d') versus the change in orientation angle of the cued stimulus (blue) and the uncued stimulus (black) for monkey 1. Data are fit with the Weibull function (gray). *B*. Same as in *A* for monkey 2.

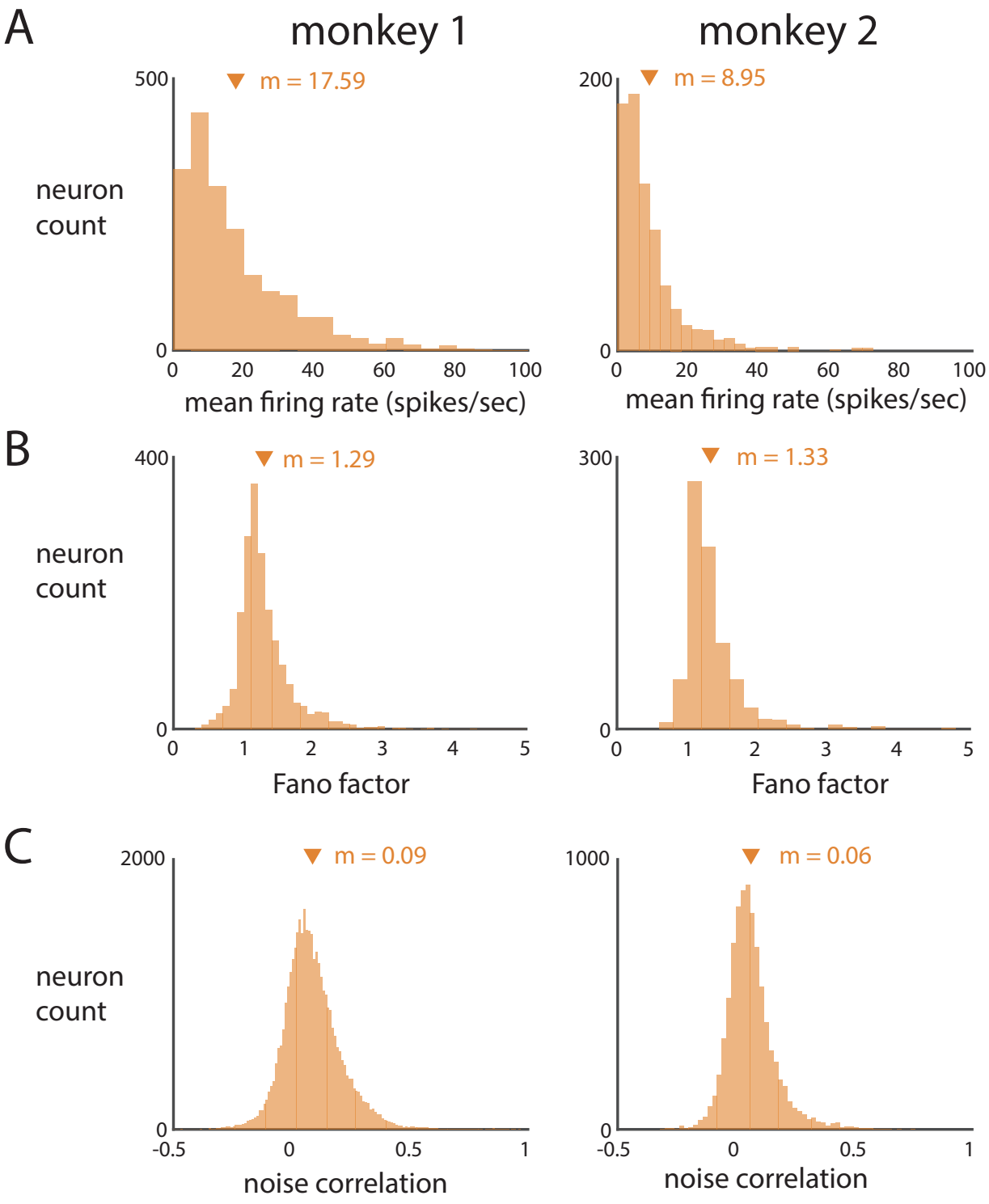


Figure D.2: Firing rate properties match those in previous studies. *A*. Mean firing rates across repeats of both sample stimuli for each neuron for monkey 1 (left) and monkey 2 (right). *B*. Fano factors (variance of spike counts divided by mean spike count) for each neuron. *C*. Noise correlations for each pair of simultaneously-recorded neurons. These results are consistent with those in previous studies (Cohen and Kohn, 2011; Cohen and Maunsell, 2009; Mitchell et al., 2009; Rabinowitz et al., 2015).

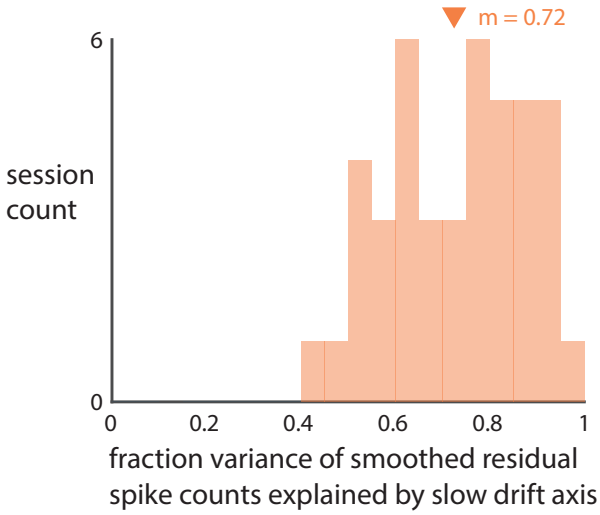


Figure D.3: The slow drift axis is defined as the axis along which the smoothed residual V4 spike counts (21 minute time bins with 6 minute strides) covary the most. To compare how well this axis captured the smoothed residual spike count variance versus other axes, we computed the fraction of smoothed spike count variance explained by the slow drift axis. This fraction was large (~0.72), suggesting the slow drift resides primarily along one dominant axis. However, other axes may have a non-negligible amount of slow drift as well.

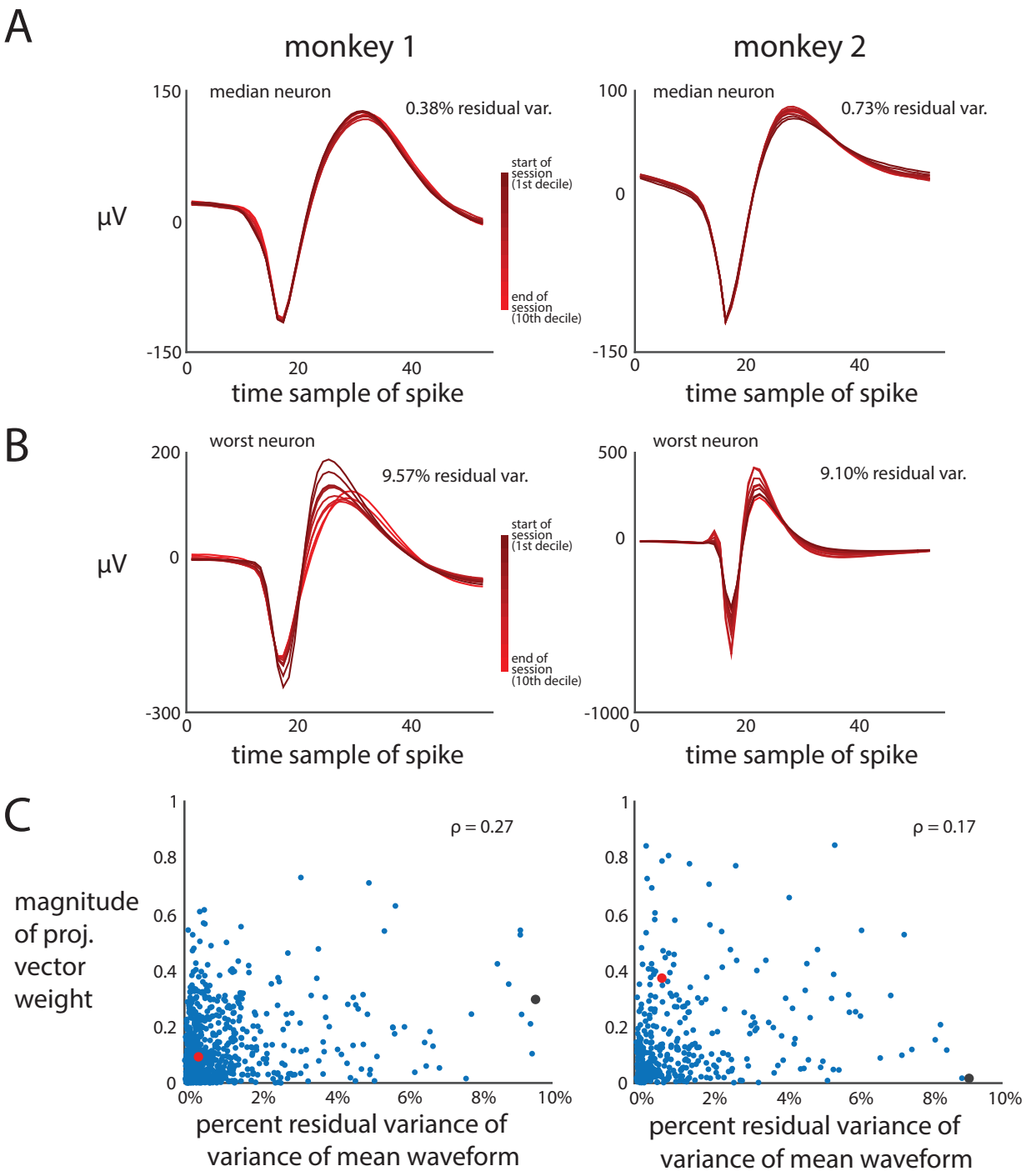


Figure D.4: Spike waveform controls for electrode shift of spike-sorted V4 units (referred to as "neurons"). Gradual changes in the distance between an electrode and a neuron throughout a session likely resulted in changes of the amplitudes of the neuron's spike waveform. For example, an far-away electrode would record smaller amplitudes of a neuron's membrane voltage, whereas an electrode nearer to the neuron would record larger amplitudes. To control for this, we assessed how much a neuron's amplitude changed throughout a session with a metric called the percent waveform variance. (continued on next page...)

Figure D.4: (...continued from previous page) To compute the percent waveform variance, we split the session into 10 non-overlapping, equal-sized time bins, and computed the mean spike waveform for each time bin. The percent waveform variance is computed as the ratio of maximum variance over deciles of the difference between that decile's mean spike waveform and the entire session's mean spike waveform divided by the variance of the mean spike waveform over the entire session. We removed any neurons whose percent waveform variance was greater than 10% from our analyses. *A*. Neurons with the median percent waveform variance for monkey 1 (left, 0.37%) and monkey 2 (right, 0.78%). *B*. Same as in *A* except for neurons with the largest percent waveform variance that met the 10% criteria for monkey 1 (left, 9.57%) and monkey 2 (right, 9.10%). *C*. Presumably, remaining neurons with a large percent waveform variance were the most likely to have electrode drift. If these neurons contributed the most to the slow drift, they should also have large projection vector weight magnitudes. We verified this was not true, as the percent waveform variance weakly correlated with the magnitudes of the projection vector weights ($p = 0.27$ for monkey 1, $p = 0.17$ for monkey 2). The red dots correspond to the neurons in *A* with median percent waveform variance, while the black dots correspond to the neurons in *B* with the lowest percent waveform variance. This suggests that even if we included some neurons with electrode drift, they did not largely contribute to our results. Because the shift of an electrode would likely change its distance to a neuron gradually over the session, we also measured to what extent a neuron's spike waveform linearly changes its shape over time. This measure (i.e., the correlation between time of session and the waveform's deviance from its mean waveform) was also weakly correlated with the neuron's projection weight magnitude ($\rho = 0.08$ and $\rho = -0.05$). These results suggest that any slow drift found in neural activity is unlikely caused by electrode shift.

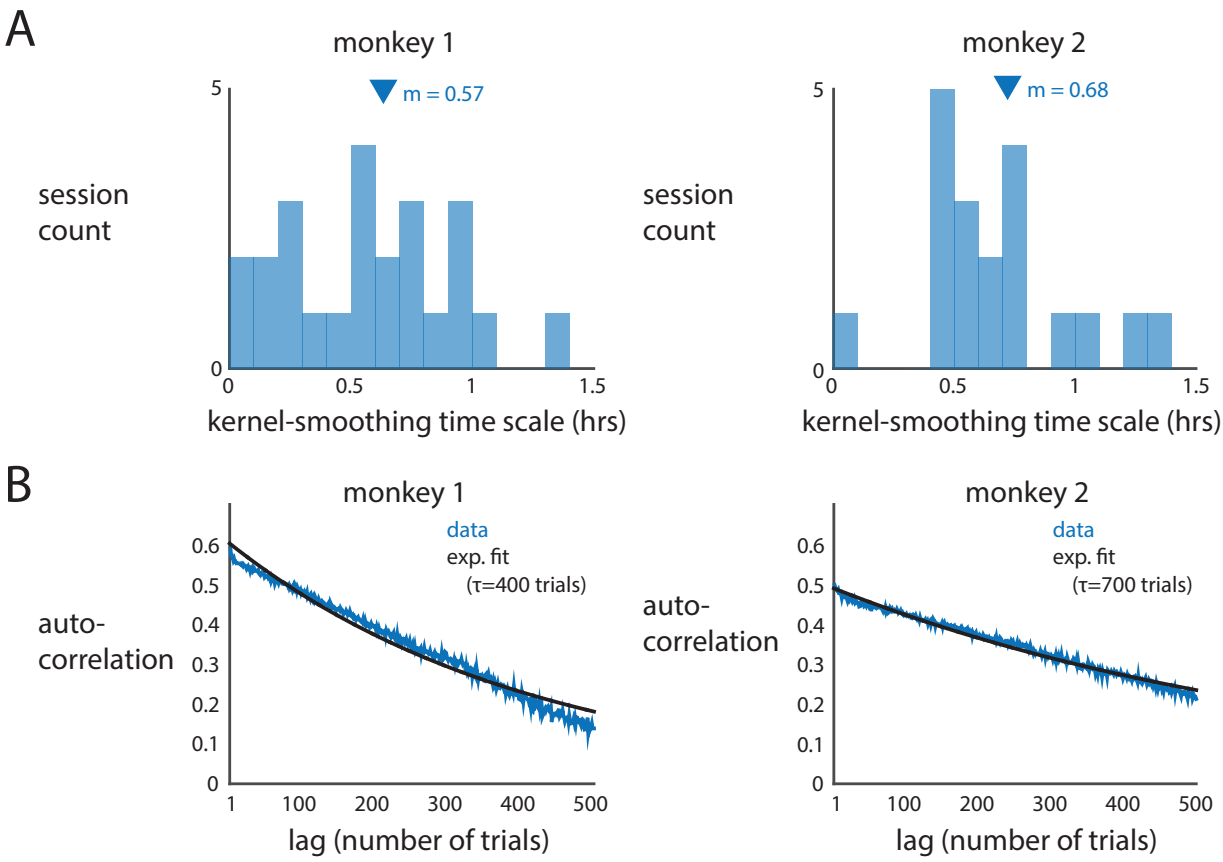


Figure D.5: The time scale of the slow drift is ~30 minutes. *A*. We used kernel smoothing with cross-validation to assess the time scale of the slow drift. After identifying the slow drift axis, we projected the residual spike count responses onto the slow drift axis. We performed 4-fold cross-validated kernel smoothing with different kernel bandwidths. Because a shorter kernel bandwidth can capture any fluctuations on longer time scales, a shorter time scale trivially yields a larger fraction of variance explained by the smoothing. To account for this, we first identified the optimal kernel bandwidth with cross-validation, and then selected the longest kernel bandwidth that performed no worse than 75% of the optimal kernel bandwidth. We found that this kernel bandwidth (i.e., the time scale of the slow drift) was typically above 30 minutes for both monkeys. *B*. As an alternate analysis, we performed an autocorrelation on the residual spike count responses along the slow drift axis (blue, averaged over sessions). Because an autocorrelation requires that responses are spaced at equal time points, we used trial number instead of the absolute time of the session. Thus, one caveat of assessing the time scale with autocorrelation was that different pairs of trials could be separated by different lengths in time. We fit an exponential function to each autocorrelation analysis, and found that decay rates $\tau = 400$ trials for monkey 1 and $\tau = 700$ trials for monkey 2 were good fits (black). On average, 100 trials took 10 minutes, indicating that the time scales of the slow drift, assessed by the autocorrelation analysis, was 40 minutes for monkey 1 and 70 minutes for monkey 2, consistent with the kernel smoothing results in *A*.

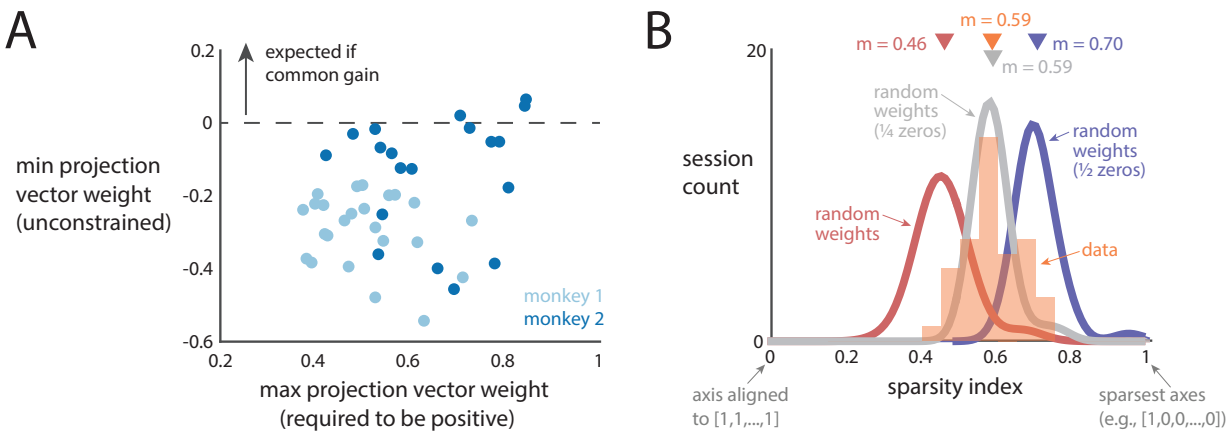


Figure D.6: The activity of some neurons drift oppositely other neurons, and the activity of three-fourths of neurons have some drift. *A*. If the slow drift affected the activity of neurons through a common gain factor, we would expect that all neurons have the same sign of their projection vector weight. To assess this, we identified the projection vector weight with the largest magnitude for each session. If this weight were negative, we flipped the sign of all projection vector weights to require that this weight be positive. We then compared this weight to the lowest projection vector weight. If the slow drift followed a common gain, we would expect that most of the lowest weights would be above 0 (above dashed line). Although true for some sessions, we found that most sessions had negative weights (below dashed line). This suggests that the slow drift is not a common gain, but can act as a positive or negative gain for different neurons or act as an additive signal. *B*. We assessed to what extent the projection vector weights of the slow drift were sparse. We defined a sparsity index SI for slow drift axis $\mathbf{s} \in \mathbb{R}^n$ for n neurons as the angle between $[|\mathbf{s}_1|, \ldots, |\mathbf{s}_n|]$ (where $\mathbf{s}_i$ is the ith element of $\mathbf{s}$ and $|\cdot|$ is the absolute value function) and the normalized $[1, 1, \ldots, 1]$ axis divided by the maximally-sparse angle. The maximally-sparse angle is the angle between the normalized $[1, 1, \ldots, 1]$ axis and the normalized $[1, 0, \ldots, 0]$ axis. Intuitively, for a non-sparse $\mathbf{s}$ that could include both positive and negative weights, we first flip all weights to be nonnegative, and assess how similar that axis is to the $[1, 1, \ldots]$ axis. With the contraint that all axes must be nonnegative, the axes that are farthest from the $[1, 1, \ldots, 1]$ axis are the unit vectors (e.g., $[1, 0, \ldots, 0]$), which gives us a maximum angle for which $\mathbf{s}$ can be from the $[1, 1, \ldots, 1]$ axis. A sparsity index of 0 indicates that $\mathbf{s}$ is dense with nonzero weights. A sparsity index of 1 indicates that $\mathbf{s}$ is highly sparse with many weights that are zero. For reference, we computed the sparsity index for randomly-generated vectors where each weight is drawn from a standard Gaussian, and then the weights are normalized (red). We computed two other reference distributions whose vectors were generated in the same way but a fraction of weights were forced to be zero ($\frac{1}{4}$ of weights are zero for gray, $\frac{1}{2}$ of weights are zero for blue). The sparsity indices of the slow drift axes (orange) overlapped the most with the $\frac{1}{4}$-weights distribution (gray), indicating that the activity of roughly $\frac{3}{4}$ of the neurons slowly drift.

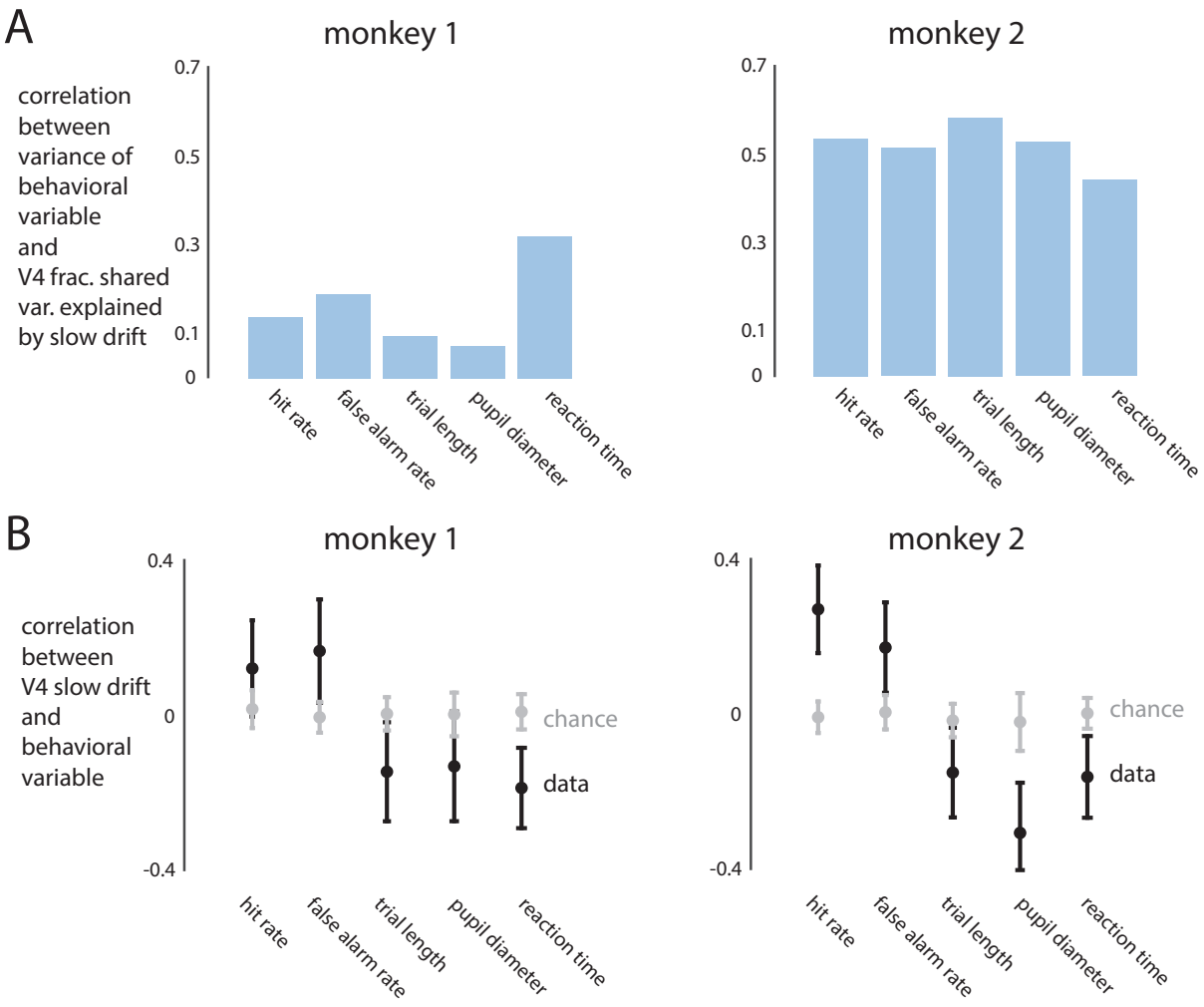


Figure D.7: Relationship between the slow drift and behavioral variables for individual monkeys. *A*. The fraction of shared variance explained by the slow drift correlated with the variance of five behavioral variables for monkey 1 (left) and monkey 2 (right). *B*. The slow drift covaried with the behavioral variables over the session for monkey 1 (left) and monkey 2 (right).

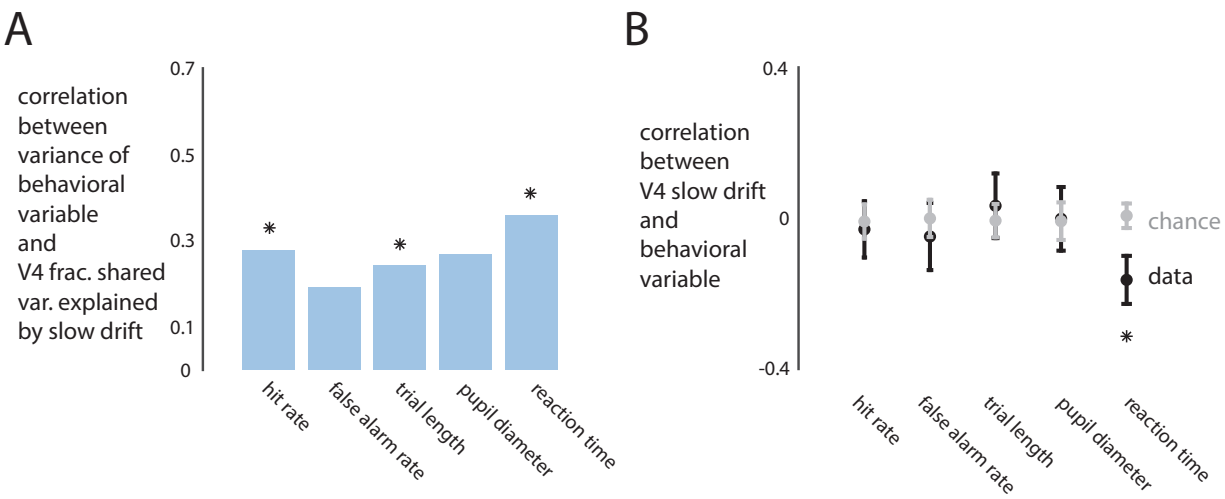


Figure D.8: We tested if the population firing rate also had a relationship between the slow drift and behavioral variables. The population firing rate (i.e., averaging the responses across neurons, akin to projecting the activity along the $[1, 1, \ldots, 1]$ axis) is a common measure to identify relationships between neural activity and behavior (Luo and Maunsell, 2015; Okun et al., 2015). Instead of the slow drift axis, we estimated the slow drift along the $[1, 1, \ldots, 1]$ axis. We found weak to no relationships between the slow drift and behavioral variables. *A*. The size of the slow drift across sessions weakly correlated with the size of changes in behavioral variables. *B*. Within a session, the slow drift weakly correlated with behavioral variables.

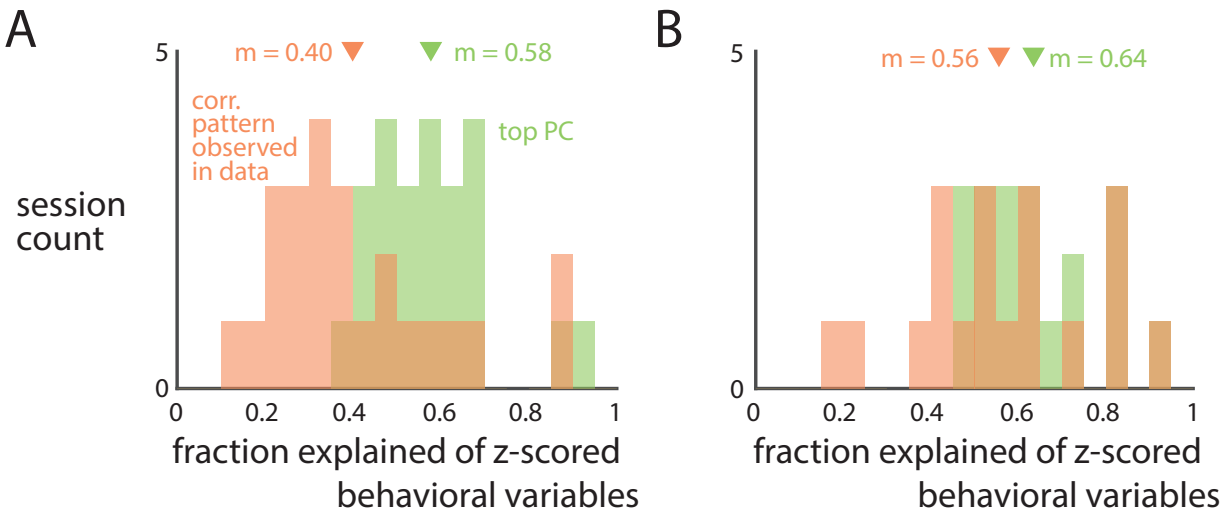


Figure D.9: Pattern of correlations observed among behavioral variables is a prominent pattern of behavioral correlations. We defined the top pattern of correlations among the behavioral variables as the top principal component of the five z-scored behavioral variables (hit rate, false alarm rate, trial length, pupil diameter, and reaction time) taken in 30 minute time bins. This top pattern explained ~60% of the behavioral variance (green, means: 0.58 for monkey 1, 0.64 for monkey 2). We defined the pattern of correlations between the slow drift and the behavioral variables (Fig. 6.3*E*) as the $[1, 1, -1, -1, -1]$ axis in behavioral space, where the elements correspond to the order of the behavioral variables (i.e., the first element corresponds to hit rate and the last element corresponds to reaction time). This pattern captured a substantial amount of variance (orange, means: 0.40 for monkey 1 and 0.56 for monkey 2). This suggests that the V4 slow drift corresponds to a large, slow drift in behavioral variables. *A*. Monkey 1. *B*. Monkey 2.

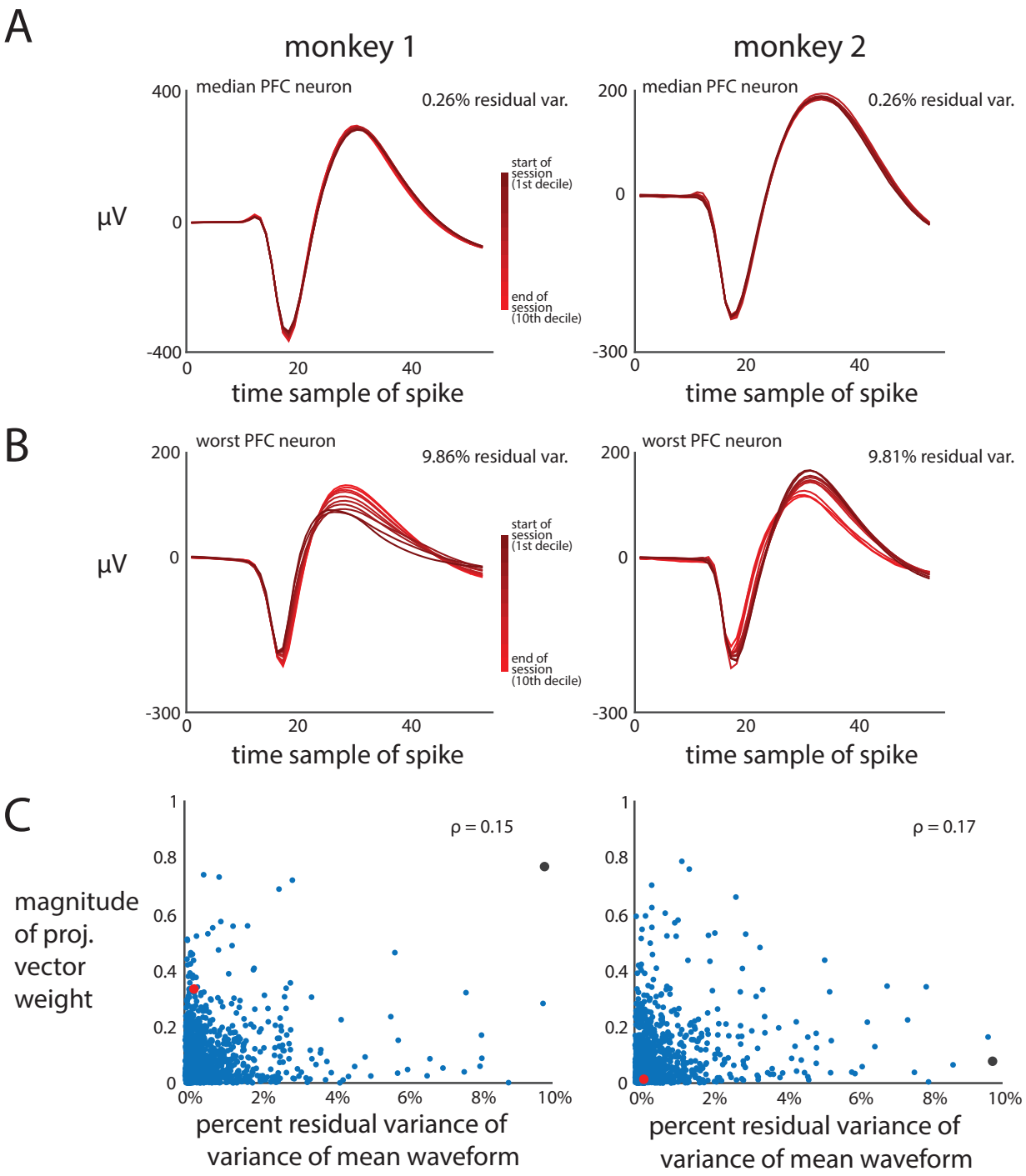


Figure D.10: Spike waveform analysis for PFC neurons. PFC neurons were subjected to the same removal criteria ($< 10\%$ waveform variance) as those of the V4 neurons. *A*,*B*, and *C* analyses for PFC neurons are the same as those for the V4 neurons (Supp. Fig. D.4). The correlation between the magnitude of projection vector weights and the percent waveform variance was weak ($\rho = 0.15$ for monkey 1, $\rho = 0.17$ for monkey 2). In addition, the magnitude of projection vector weights weakly covaried with the correlation between the time of session and the waveform's deviance from the mean waveform across the session ($\rho = 0.20$ for monkey 1, $\rho = 0.05$ for monkey 2). (continued on next page...)

Figure D.10: (...continued from previous page) These results suggest that any slow drift observed in the activity of the remaining neurons is not due to a gradual shift in the recording electrodes.

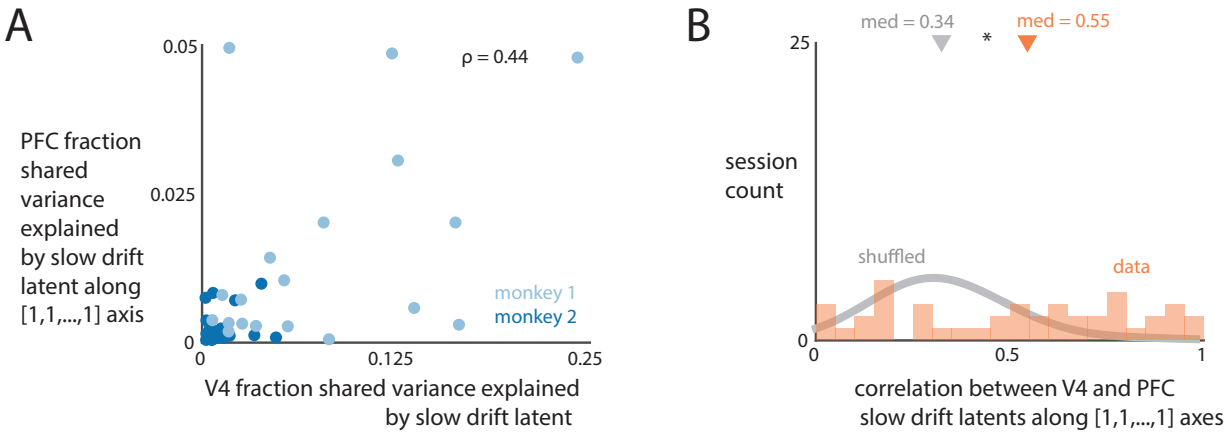


Figure D.11: We wanted to see if the mean population firing rates for the V4 and PFC activity shared the same slow drift. The mean population firing rate, a typical measure used in many studies (Luo and Maunsell, 2015; Okun et al., 2015), was computed by projecting spike count vectors onto the $[1, 1, \ldots, 1]$ axis. We found that slow drift along the $[1, 1, \ldots, 1]$ axis is not strongly correlated between V4 and PFC. *A*. The size of the V4 slow drift correlated with the size of the PFC slow drift ($\rho = 0.44$). *B*. Although the V4 and PFC slow drifts covaried over the course of the session (median $= 0.55$), this correlation was substantially smaller than that found for slow drift along the slow drif taxis (median $= 0.96$, Supp. Fig. D.11*C*). This suggests that analyses that compute the mean population firing rate may miss the effects of the slow drift. These results also help to rule out that the slow drift arises from gradual shifts in recording electrodes, as these shifts would likely affect the mean population firing rate.

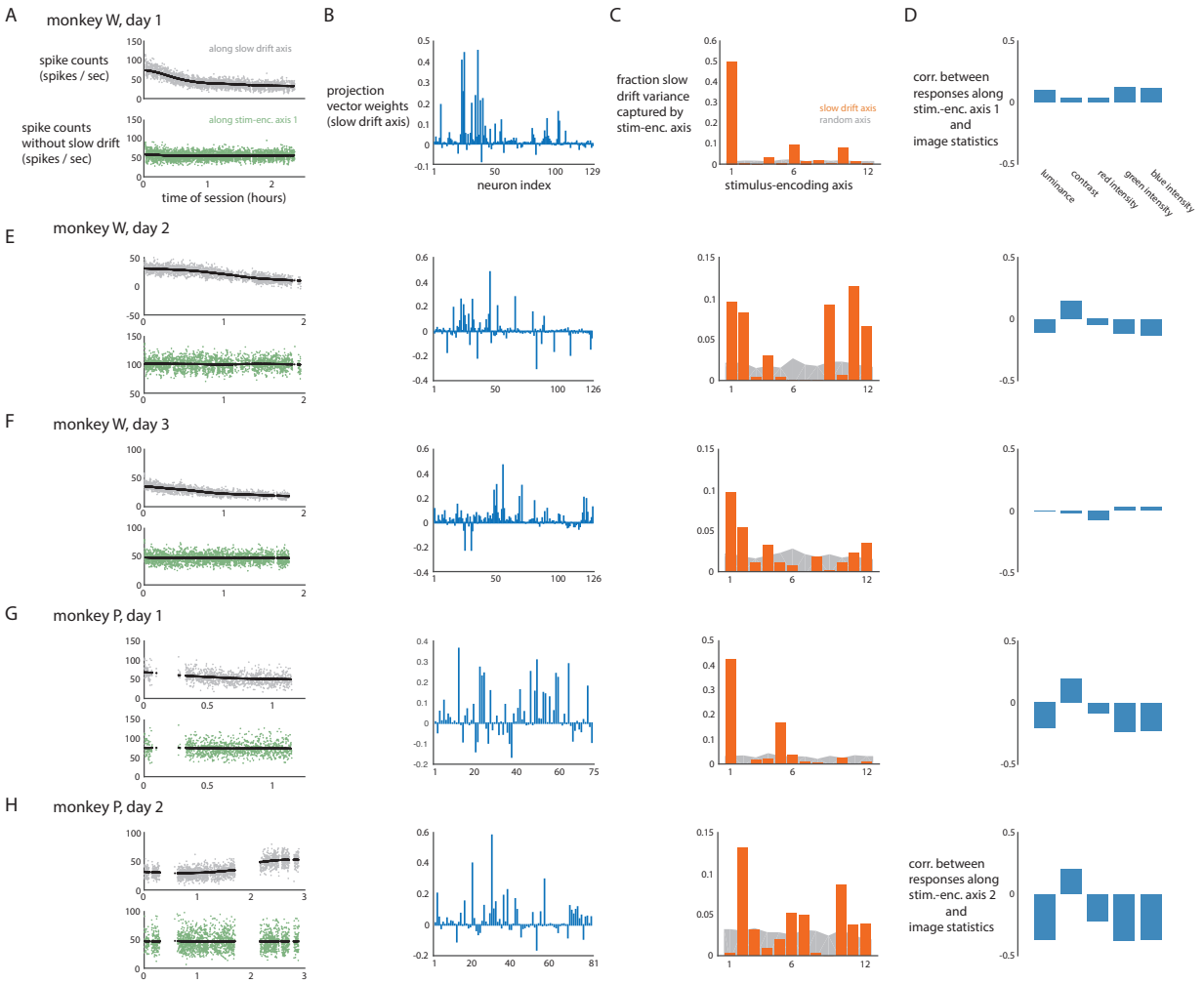


Figure D.12: The slow drift lied along stimulus-encoding axes (all sessions). For one monkey (monkey W), the adaptive stimulus selection algorithm Adept chose 2,000 natural images that elicited large and diverse responses (Cowley et al., 2017b). For following recording sessions, we showed these adaptively-selected natural images randomly throughout a session. For another monkey (monkey P), we did not use adaptive stimulus selection but rather randomly selected 550 images to show. Both monkeys were required to fixate while multiple natural images were shown for each trial. To compute the stimulus-encoding axes, we first removed the slow drift from each principal component axis, where PCA was applied to spike counts taken in long time bins (20 minutes). This was done to ensure that the stimulus-encoding axes were not trivially similar to the slow drift axis because the stimulus-encoding axes contained slow drift. We then computed the mean spike count vector to each natural image, and applied PCA to the mean spike count vectors to identify the stimulus-encoding axes (i.e., the top principal component axes). *A*. Top: Spike count vectors projected onto slow drift axis (gray), revealing a slow drift (black) across the session. Bottom: Spike count vectors (with slow drift removed) along the top stimulus-encoding axis (green) showed no slow drift (black), as expected. (continued on next page...)

Figure D.12: *B*. (...continued from previous page) Projection vector weights for the slow drift axis. *C*. The stimulus-encoding axes capture a larger fraction of slow drift variance (orange) than if the slow drift lied along an axis with a random direction in firing rate space (gray). *D*. The top stimulus-encoding axis did not covary with low-level image statistics, suggesting the top stimulus-encoding axis encoded more higher-level features of the natural images. The low-level image statistics included luminance (mean pixel intensity) and contrast (standard deviation of pixel intensity), as well as red, green, and blue intensities (mean pixel intensity for the corresponding colors).

Bibliography

Google google image search. http://images.google.com. Accessed: 2017-04-25. 4.4

Larry F Abbott and Peter Dayan. The effect of correlated variability on the accuracy of a population code. *Neural computation*, 11(1):91–101, 1999. 2.3

Afsheen Afshar, Gopal Santhanam, Byron M Yu, Stephen I Ryu, Maneesh Sahani, and Krishna V Shenoy. Single-trial neural correlates of arm movement preparation. *Neuron*, 71(3):555–564, 2011. 7.3.1

Misha B Ahrens, Michael B Orger, Drew N Robson, Jennifer M Li, and Philipp J Keller. Whole-brain functional imaging at cellular resolution using light-sheet microscopy. *Nature methods*, 10(5):413, 2013. 8.0.4

Julie A Alvarez and Eugene Emory. Executive function and the frontal lobes: a meta-analytic review. *Neuropsychology review*, 16(1):17–42, 2006. 2.1.3

Stuart Anstis, Frans AJ Verstraten, and George Mather. The motion aftereffect. *Trends in cognitive sciences*, 2(3):111–117, 1998. 6

Iñigo Arandia-Romero, Seiji Tanabe, Jan Drugowitsch, Adam Kohn, and Rubén Moreno-Bote. Multiplicative and additive modulation of neuronal tuning with population activity affects encoded information. *Neuron*, 89(6):1305–1316, 2016. 2.3, 6, 6.1.1, 7.1

Amos Arieli, Alexander Sterkin, Amiram Grinvald, and AD Aertsen. Dynamics of ongoing activity: explanation of the large variability in evoked cortical responses. *Science*, 273(5283): 1868–1871, 1996. 2.3, 3.2

Gary Aston-Jones and Jonathan D Cohen. An integrative theory of locus coeruleus-norepinephrine function: adaptive gain and optimal performance. *Annu. Rev. Neurosci.*, 28:403–450, 2005. 2.3, 6.1.2, 6.1.3, 6.1.4, 6.1.8, 8.0.2

Bruno B Averbeck, Peter E Latham, and Alexandre Pouget. Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, 7(5):358–366, 2006. 6, 6.1.4, 6.1.8, 7.1

Francis R Bach and Michael I Jordan. Kernel independent component analysis. *Journal of machine learning research*, 3(Jul):1–48, 2002. 5

Jushan Bai, Kunpeng Li, et al. Statistical analysis of factor models of high dimension. *The Annals of Statistics*, 40(1):436–465, 2012. 8.0.4

Rembrandt Bakker, Paul Tiesinga, and Rolf Kötter. The scalable brain atlas: instant web-based access to public brain atlases and related content. *Neuroinformatics*, 13(3):353–366, 2015. 2.1

Elnaz Barshan, Ali Ghodsi, Zohreh Azimifar, and Mansoor Zolghadri Jahromi. Supervised principal component analysis: Visualization, classification and regression on subspaces and submanifolds. *Pattern Recognition*, 44(7):1357–1371, 2011. 5.5

Peter Bartho, Carina Curto, Artur Luczak, Stephan L Marguet, and Kenneth D Harris. Population coding of tone stimuli in auditory cortex: dynamic rate vector analysis. *Eur J Neurosci*, 30(9): 1767–78, Nov 2009. doi: 10.1111/j.1460-9568.2009.06954.x. 7.3.5

Charles B Beaman, Sarah L Eagleman, and Valentin Dragoi. Sensory coding accuracy and perceptual performance are improved during the desynchronized cortical state. *Nature communications*, 8(1):1308, 2017. 2.3, 6, 6.1.8, 6.1.8

Jan Benda, Tim Gollisch, Christian K Machens, and Andreas VM Herz. From response to stimulus: adaptive sampling in sensory physiology. *Current Opinion in Neurobiology*, 17(4): 430–436, 2007. 4, 8.0.1

Richard Berk, Lawrence Brown, and Linda Zhao. Statistical inference after model selection. *Journal of Quantitative Criminology*, 26(2):217–236, 2010. 7.3.1

Pietro Berkes, Gerg*H*o Orbán, Máté Lengyel, and József Fiser. Spontaneous cortical activity reveals hallmarks of an optimal internal model of the environment. *Science*, 331(6013):83–87, 2011. 3.2

Christopher M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*, chapter 4, pages 191–2. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006. ISBN 0387310738. 2.2, 7.3.4

Sean R Bittner, Ryan C Williamson, Adam C Snyder, Ashok Litwin-Kumar, Brent Doiron, Steven M Chase, Matthew A Smith, and Byron M Yu. Population activity structure of excitatory and inhibitory neurons. *PloS one*, 12(8):e0181773, 2017. 8.0.4

Adrian G Bondy, Ralf M Haefner, and Bruce G Cumming. Feedback determines the structure of correlated variability in primary visual cortex. *Nature neuroscience*, page 1, 2018. 1, 6, 6.1.8, 8.0.2

Kristofer E Bouchard, Nima Mesgarani, Keith Johnson, and Edward F Chang. Functional organization of human sensorimotor cortex for speech articulation. *Nature*, 495(7441):327–32, Mar 2013. doi: 10.1038/nature11911. 2.2, 3, 3.2

Wieland Brendel, Ranulfo Romo, and Christian K. Machens. Demixed principal component analysis. In J. Shawe-Taylor, R.S. Zemel, P. Bartlett, F.C.N. Pereira, and K.Q. Weinberger, editors, *Advances in Neural Information Processing Systems 24*, pages 2654–2662. 2011. 2.2.2, 7.3.4, 7.3.5

Kenneth H Britten, William T Newsome, Michael N Shadlen, Simona Celebrini, and J Anthony Movshon. A relationship between behavioral choice and the visual responses of neurons in macaque mt. *Visual neuroscience*, 13(1):87–100, 1996. 6

Laura Busse, Alex R Wade, and Matteo Carandini. Representation of concurrent stimuli by population activity in visual cortex. *Neuron*, 64(6):931–942, 2009. 3.2

Santiago A Cadena, George H Denfield, Edgar Y Walker, Leon A Gatys, Andreas S Tolias, Matthias Bethge, and Alexander S Ecker. Deep convolutional models improve predictions of

macaque v1 responses to natural images. *bioRxiv*, page 201764, 2017. 2.1.1

Francesco Camastra and Alessandro Vinciarelli. Estimating the intrinsic dimension of data with a fractal-based method. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 24 (10):1404–1407, 2002. 3.2

Matteo Carandini and David J Heeger. Normalization as a canonical neural computation. *Nature Reviews Neuroscience*, 13(1):51–62, 2012. 6.1.8

Matteo Carandini, Jonathan B Demb, Valerio Mante, David J Tolhurst, Yang Dan, Bruno A Olshausen, Jack L Gallant, and Nicole C Rust. Do we know what the early visual system does? *Journal of Neuroscience*, 25(46):10577–10597, 2005. 1, 2.1.1, 3.1.6, 3.2, A.6

Eric T Carlson, Russell J Rasquinha, Kechen Zhang, and Charles E Connor. A sparse object coding scheme in area V4. *Current Biology*, 21(4):288–293, 2011. 4, 4.1, 4.4.1

Vivien A Casagrande and Jon H Kaas. The afferent, intrinsic, and efferent connections of primary visual cortex in primates. In *Primary visual cortex in primates*, pages 201–259. Springer, 1994. 2.1.1

Carmen Cavada, Teresa Compañy, Jaime Tejedor, Roelf J Cruz-Rizzolo, and Fernando Reinoso-Suárez. The anatomical connections of the macaque monkey orbitofrontal cortex. a review. *Cerebral Cortex*, 10(3):220–242, 2000. 2.1.3

Billy Chang, Uwe Kruger, Rafal Kustra, and Junping Zhang. Canonical correlation analysis based on hilbert-schmidt independence criterion and centered kernel target alignment. In *Proceedings of The 30th International Conference on Machine Learning*, pages 316–324, 2013. 5, 5.5, 5.6.2, 5.8

Anne K Churchland, Roozbeh Kiani, Rishidev Chaudhuri, Xiao-Jing Wang, Alexandre Pouget, and Michael N Shadlen. Variance as a signature of neural computations during decision making. *Neuron*, 69(4):818–831, 2011. 7.2.2

M. M. Churchland, B. M. Yu, M. Sahani, and K. V. Shenoy. Techniques for extracting single-trial activity patterns from large-scale neural recordings. *Curr. Opin. Neurobiol.*, 17(5):609–618, Oct 2007. 1

Mark M Churchland and Krishna V Shenoy. Temporal complexity and heterogeneity of single-neuron activity in premotor and motor cortex. *J Neurophysiol*, 97(6):4235–57, Jun 2007. doi: 10.1152/jn.00095.2007. 1

Mark M Churchland, Byron M Yu, John P Cunningham, Leo P Sugrue, Marlene R Cohen, Greg S Corrado, William T Newsome, Andrew M Clark, Paymon Hosseini, Benjamin B Scott, et al. Stimulus onset quenches neural variability: a widespread cortical phenomenon. *Nature neuroscience*, 13(3):369–378, 2010. 1, 7.2.2, 7.3.1, 8

Mark M Churchland, John P Cunningham, Matthew T Kaufman, Justin D Foster, Paul Nuyujukian, Stephen I Ryu, and Krishna V Shenoy. Neural population dynamics during reaching. *Nature*, 487(7405):51–6, Jul 2012. doi: 10.1038/nature11129. 1, 2.2, 2.2.2, 3, 3.2, 7.7, 7.3, 7.3.4, 7.3.5, 8

Ruben Coen-Cagli, Adam Kohn, and Odelia Schwartz. Flexible gating of contextual influences in natural vision. *Nature neuroscience*, 18(11):1648, 2015. 1, 3.2

Marlene R Cohen and Adam Kohn. Measuring and interpreting neuronal correlations. *Nature Neuroscience*, 14(7):811–819, 2011. 2.3, 4.5, 6.1.5, 7, 7.1, 8.0.2, D.2

Marlene R Cohen and John H R Maunsell. A neuronal population measure of attention predicts behavioral performance on individual trials. *J Neurosci*, 30(45):15241–53, Nov 2010. doi: 10.1523/JNEUROSCI.2171-10.2010. 1, D.1.7

Marlene R Cohen and John HR Maunsell. Attention improves performance primarily by reducing interneuronal correlations. *Nature Neuroscience*, 12(12):1594–1600, 2009. 1, 2.3, 4.5, 6, 6.1.5, 6.1.5, 7, D.2

Benjamin Cowley, Joao Semedo, Amin Zandvakili, Matthew Smith, Adam Kohn, and Byron Yu. Distance covariance analysis. In *Artificial Intelligence and Statistics*, pages 242–251, 2017a. 6.1.3

Benjamin Cowley, Ryan Williamson, Katerina Acar, Matthew Smith, and Byron M Yu. Adaptive stimulus selection for optimizing neural population responses. In *Advances in Neural Information Processing Systems*, pages 1395–1405, 2017b. 6.1.6, 6.1.8, D.1.9, D.12

Benjamin R Cowley, Matthew T Kaufman, Mark M Churchland, Stephen I Ryu, Krishna V Shenoy, and Byron M Yu. Datahigh: graphical user interface for visualizing and interacting with high-dimensional neural activity. *Conf Proc IEEE Eng Med Biol Soc*, 2012:4607–10, 2012. doi: 10.1109/EMBC.2012.6346993. 7.3

Benjamin R Cowley, Matthew T Kaufman, Zachary S Butler, Mark M Churchland, Stephen I Ryu, Krishna V Shenoy, and Byron M Yu. Datahigh: graphical user interface for visualizing and interacting with high-dimensional neural activity. *Journal of neural engineering*, 10(6):066012, 2013. 6.1.6, 7.3.5, B.3

Benjamin R Cowley, Matthew A Smith, Adam Kohn, and Byron M Yu. Stimulus-driven population activity patterns in macaque primary visual cortex. *PLOS Computational Biology*, 12(12): e1005185, 2016. 5

Bruce G Cumming and Hendrikje Nienborg. Feedforward and feedback sources of choice probability in neural population responses. *Current opinion in neurobiology*, 37:126–132, 2016. 6.1.7

John P Cunningham and Zoubin Ghahramani. Linear dimensionality reduction: Survey, insights, and generalizations. *Journal of Machine Learning Research*, 16:2859–2900, 2015. 5.2.1, 8.0.4, A.4

John P Cunningham and Byron M Yu. Dimensionality reduction for large-scale neural recordings. *Nature neuroscience*, 17(11):1500–1509, 2014. 1, 2.2, 2.2.2, 3, 3.2, 5, 5.7.1, 6.1.6, 8, A.4

Clayton E Curtis and Mark D'Esposito. Persistent activity in the prefrontal cortex during working memory. *Trends in cognitive sciences*, 7(9):415–423, 2003. 6.1.3

Kayvon Daie, Mark S Goldman, and Emre RF Aksay. Spatial patterns of persistent neural activity vary with the behavioral context of short-term memory. *Neuron*, 85(4):847–860, 2015. 2.2, 3

Stephen V David, William E Vinje, and Jack L Gallant. Natural stimulus statistics alter the receptive field structure of v1 neurons. *J Neurosci*, 24(31):6991–7006, 2004. 3.1.3, 3.1.6, 3.2

Karl Deisseroth. Optogenetics. *Nature methods*, 8(1):26, 2011. 8.0.4

Robert Desimone and John Duncan. Neural mechanisms of selective visual attention. *Annual review of neuroscience*, 18(1):193–222, 1995. 6

James J DiCarlo, Davide Zoccolan, and Nicole C Rust. How does the brain solve visual object recognition? *Neuron*, 73(3):415–434, 2012. 3

Christopher DiMattina and Kechen Zhang. Adaptive stimulus optimization for sensory systems neuroscience. *Closing the Loop Around Neural Systems*, page 258, 2014. 4, 8.0.1

Daniel Durstewitz, Nicole M Vittoz, Stan B Floresco, and Jeremy K Seamans. Abrupt transitions between prefrontal neural ensemble states accompany behavioral transitions during rule learning. *Neuron*, 66(3):438–48, May 2010. doi: 10.1016/j.neuron.2010.03.029. 2.2, 3, 3.2

R Becket Ebitz and Tirin Moore. Selective modulation of the pupil light reflex by microstimulation of prefrontal cortex. *Journal of Neuroscience*, 37(19):5008–5018, 2017. 6.1.2

Alexander S Ecker, Philipp Berens, Andreas S Tolias, and Matthias Bethge. The effect of noise correlations in populations of diversely tuned neurons. *Journal of Neuroscience*, 31(40): 14272–14283, 2011. 2.3, 6

Alexander S Ecker, Philipp Berens, R James Cotton, Manivannan Subramaniyan, George H Denfield, Cathryn R Cadwell, Stelios M Smirnakis, Matthias Bethge, and Andreas S Tolias. State dependence of noise correlations in macaque primary visual cortex. *Neuron*, 82(1): 235–248, 2014. 2.3, 6, 6.1.1, 7.1

Alexander S Ecker, George H Denfield, Matthias Bethge, and Andreas S Tolias. On the structure of neuronal population activity under fluctuations in attentional state. *Journal of Neuroscience*, 36(5):1775–1789, 2016. 2.3, 6

Gidon Felsen and Yang Dan. A natural approach to studying vision. *Nature Neuroscience*, 8(12): 1643–1646, 2005. 8.0.2

Gidon Felsen, Jon Touryan, Feng Han, and Yang Dan. Cortical sensitivity to visual features in natural scenes. *PLoS Bio*, 3(10):e342, 2005a. 3.2

Gidon Felsen, Jon Touryan, Feng Han, and Yang Dan. Cortical sensitivity to visual features in natural scenes. *PLoS biology*, 3(10):e342, 2005b. 4.5

József Fiser, Chiayu Chiu, and Michael Weliky. Small modulation of ongoing cortical dynamics by sensory input during natural vision. *Nature*, 431(7008):573–578, 2004. 3.2

Peter Földiák. Stimulus optimisation in primary visual cortex. *Neurocomputing*, 38:1217–1222, 2001. 4

Jeremy Freeman, Corey M Ziemba, David J Heeger, Eero P Simoncelli, and J Anthony Movshon. A functional and perceptual signature of the second visual area in primates. *Nature neuroscience*, 16(7):974–981, 2013. 3.1.8

Emmanouil Froudarakis, Philipp Berens, Alexander S Ecker, R James Cotton, Fabian H Sinz, Dimitri Yatsenko, Peter Saggau, Matthias Bethge, and Andreas S Tolias. Population code in mouse v1 facilitates readout of natural scenes through increased sparseness. *Nat Neurosci*, 2014. 3.2

Kenji Fukumizu and Chenlei Leng. Gradient-based kernel dimension reduction for regression.

Journal of the American Statistical Association, 109(505):359–370, 2014. 5.5

Kenji Fukumizu, Francis R Bach, and Michael I Jordan. Dimensionality reduction for supervised learning with reproducing kernel hilbert spaces. *The Journal of Machine Learning Research*, 5: 73–99, 2004a. 5.2.1

Kenji Fukumizu, Francis R Bach, and Michael I Jordan. Kernel dimensionality reduction for supervised learning. *Advances in Neural Information Processing Systems*, 16:81, 2004b. 2.2.2

Peiran Gao, Eric Trautmann, Byron M Yu, Gopal Santhanam, Stephen Ryu, Krishna Shenoy, and Surya Ganguli. A theory of multineuronal dimensionality, dynamics and measurement. *bioRxiv*, page 214262, 2017. 3, 3.2, 8.0.3, 8.0.4

Yuanjun Gao, Evan W Archer, Liam Paninski, and John P Cunningham. Linear dynamical neural population models through nonlinear embeddings. In *Advances in Neural Information Processing Systems*, pages 163–171, 2016. 2.2.1

Murray Gell-Mann and Seth Lloyd. Information measures, effective complexity, and total information. *Complexity*, 2(1):44–52, 1996. 3.2

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 8.0.4

Robbe LT Goris, J Anthony Movshon, and Eero P Simoncelli. Partitioning neuronal variability. *Nature Neuroscience*, 17(6):858–865, 2014. 2.3, 6, 6.1.1

Robbe LT Goris, Eero P Simoncelli, and J Anthony Movshon. Origin and function of tuning diversity in macaque visual cortex. *Neuron*, 88(4):819–831, 2015. 3.1.6, 3.2, A.6, A.6, A.6, A.6, A.6, A.6

Robert Greene and Jerome Siegel. Sleep. *Neuromolecular medicine*, 5(1):59–68, 2004. 6.1.8

Arthur Gretton, Kenji Fukumizu, Choon H Teo, Le Song, Bernhard Schölkopf, and Alex J Smola. A kernel statistical test of independence. In *Advances in neural information processing systems*, pages 585–592, 2007. 5

Diego A Gutnisky, Charles B Beaman, Sergio E Lew, and Valentin Dragoi. Spontaneous fluctuations in visual cortical responses influence population coding accuracy. *Cerebral cortex*, 27(2): 1409–1427, 2017. 6

David R Hardoon, Sandor Szedmak, and John Shawe-Taylor. Canonical correlation analysis: An overview with application to learning methods. *Neural computation*, 16(12):2639–2664, 2004. 5

Kenneth D Harris and Alexander Thiele. Cortical state and attention. *Nature reviews neuroscience*, 12(9):509, 2011. 6.1.8

Christopher D Harvey, Philip Coen, and David W Tank. Choice-specific sequences in parietal cortex during a virtual-navigation decision task. *Nature*, 484(7392):62, 2012. 2.2, 3, 3.2

MJ Hawken, AJ Parker, and JS Lund. Laminar organization and contrast sensitivity of direction-selective cells in the striate cortex of the old world monkey. *J Neurosci*, 8(10):3541–3548, 1988. 3.1.1

James V Haxby, J Swaroop Guntupalli, Andrew C Connolly, Yaroslav O Halchenko, Bryan R Conroy, M Ida Gobbini, Michael Hanke, and Peter J Ramadge. A common, high-dimensional model of the representational space in human ventral temporal cortex. *Neuron*, 72(2):404–416, 2011. 5.6.3, 5.7.3

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 770–778, 2016. 4.4.1

Inge S Helland. On the structure of partial least squares regression. *Communications in statistics-Simulation and Computation*, 17(2):581–607, 1988. 2.2.3, 5.5

Agnar Höskuldsson. Pls regression methods. *Journal of chemometrics*, 2(3):211–228, 1988. 2.2.3, 5.5

Harold Hotelling. Relations between two sets of variates. *Biometrika*, 28(3/4):321–377, 1936. 2.2.3, 5.5

Chonghai Hu, Weike Pan, and James T Kwok. Accelerated gradient methods for stochastic optimization and online learning. In *Advances in Neural Information Processing Systems*, pages 781–789, 2009. 5.2.1

Reto Huber, M Felice Ghilardi, Marcello Massimini, and Giulio Tononi. Local sleep and learning. *Nature*, 430(6995):78, 2004. 6.1.8

Chia-Chun Hung, Eric T Carlson, and Charles E Connor. Medial axis shape coding in macaque inferotemporal cortex. *Neuron*, 74(6):1099–1113, 2012. 4, 4.1, 4.4.1

Ross Iaci, TN Sriram, and Xiangrong Yin. Multivariate association and dimension reduction: A generalization of canonical correlation analysis. *Biometrics*, 66(4):1107–1118, 2010. 5.6.3, 5.8

Alan Julian Izenman. Reduced-rank regression for the multivariate linear model. *Journal of multivariate analysis*, 5(2):248–264, 1975. 2.2.3

Siddhartha Joshi, Yin Li, Rishi M Kalwani, and Joshua I Gold. Relationships between pupil diameter and neuronal activity in the locus coeruleus, colliculi, and cingulate cortex. *Neuron*, 89(1):221–234, 2016. 2.3, 6.1.2, 6.1.3, 6.1.4, 6.1.8, 8.0.2

James J Jun, Nicholas A Steinmetz, Joshua H Siegle, Daniel J Denman, Marius Bauza, Brian Barbarits, Albert K Lee, Costas A Anastassiou, Alexandru Andrei, cCaugatay Aydın, et al. Fully integrated silicon probes for high-density recording of neural activity. *Nature*, 551(7679): 232, 2017. 8.0.4

Ryota Kanai and Frans AJ Verstraten. Perceptual manifestations of fast neural plasticity: Motion priming, rapid motion aftereffect and perceptual sensitization. *Vision research*, 45(25-26): 3109–3116, 2005. 6

Ingmar Kanitscheider, Ruben Coen-Cagli, Adam Kohn, and Alexandre Pouget. Measuring fisher information accurately in correlated neural populations. *PLoS computational biology*, 11(6): e1004218, 2015. 6, 6.1.6, 6.1.8, 8.0.3

Masayuki Karasuyama and Masashi Sugiyama. Canonical dependency analysis based on squared-loss mutual information. *Neural Networks*, 34:46–55, 2012. 5.5

Matthew T Kaufman, Mark M Churchland, Stephen I Ryu, and Krishna V Shenoy. Cortical activity in the null space: permitting preparation without movement. *Nat Neurosci*, 2014. 3, 3.2

Ryan C Kelly, Matthew A Smith, Jason M Samonds, Adam Kohn, AB Bonds, J Anthony Movshon, and Tai Sing Lee. Comparison of recordings from microelectrode arrays and single electrodes in the visual cortex. *J Neurosci*, 27(2):261–264, 2007. 2.1.4, A.1, D.1.1

Ryan C Kelly, Matthew A Smith, Robert E Kass, and Tai Sing Lee. Local field potentials indicate network state and account for neuronal response variability. *Journal of computational neuroscience*, 29(3):567–579, 2010. 5.7.1, A.1, A.2

Jon R Kettenring. Canonical analysis of several sets of variables. *Biometrika*, 58(3):433–451, 1971. 5.4, 5.6.3

Seung-Jae Kim, Sandeep C Manyam, David J Warren, and Richard A Normann. Electrophysiological mapping of cat primary auditory cortex with multielectrode arrays. *Annals of biomedical engineering*, 34(2):300–309, 2006. 2.4

William F Kindel, Elijah D Christensen, and Joel Zylberberg. Using deep learning to reveal the neural code for images in primary visual cortex. *arXiv preprint arXiv:1706.06208*, 2017. 2.1.1

Dmitry Kobak, Wieland Brendel, Christos Constantinidis, Claudia E Feierstein, Adam Kepecs, Zachary F Mainen, Xue-Lian Qi, Ranulfo Romo, Naoshige Uchida, and Christian K Machens. Demixed principal component analysis of neural population data. *eLife*, 5:e10989, 2016. 2.2.2, 5, 5.7.1

Adam Kohn. Visual adaptation: physiology, mechanisms, and functional benefits. *Journal of neurophysiology*, 97(5):3155–3164, 2007. 1, 6, 8.0.4

Adam Kohn and J Anthony Movshon. Adaptation changes the direction tuning of macaque mt neurons. *Nature neuroscience*, 7(7):764, 2004. 6

Adam Kohn and Matthew A Smith. Stimulus dependence of neuronal correlation in primary visual cortex of the macaque. *Journal of Neuroscience*, 25(14):3661–3673, 2005. 7, 7.1, 7.1, 7.2.2

Adam Kohn, Ruben Coen-Cagli, Ingmar Kanitscheider, and Alexandre Pouget. Correlations and neuronal population information. *Annual Review of Neuroscience*, 39:237–256, 2016. 1, 2.3, 4.5, 6, 6.1.1, 6.1.6, 6.1.8, 7.1, 8.0.2, 8.0.3

Karthik C Lakshmanan, Patrick T Sadtler, Elizabeth C Tyler-Kabara, Aaron P Batista, and Byron M Yu. Extracting low-dimensional latent structure from time series in the presence of delays. *Neural computation*, 27(9):1825–1856, 2015. 2.2.1

Tai Sing Lee, David Mumford, Richard Romero, and Victor AF Lamme. The role of the primary visual cortex in higher level vision. *Vision research*, 38(15-16):2429–2454, 1998. 2.1.1

Sidney R Lehky, Roozbeh Kiani, Hossein Esteky, and Keiji Tanaka. Dimensionality of object representations in monkey inferotemporal cortex. *Neural computation*, 26(10):2135–2162, 2014. 3, 3.1.8, 3.2, 8.0.3, A.4, B.4

Jeremy Lewi, Robert Butera, and Liam Paninski. Sequential optimal design of neurophysiology experiments. *Neural Computation*, 21(3):619–687, 2009. 4, 4.4.3, 4.5, 6.1.8

Jeremy Lewi, David M Schneider, Sarah MN Woolley, and Liam Paninski. Automating the design of informative sequences of sensory stimuli. *Journal of computational neuroscience*, 30(1): 181–200, 2011. 8.0.4

Nuo Li, Kayvon Daie, Karel Svoboda, and Shaul Druckmann. Robust neuronal dynamics in premotor cortex during motor planning. *Nature*, 532(7600):459–464, 2016. 3.2

I-Chun Lin, Michael Okun, Matteo Carandini, and Kenneth D Harris. The nature of shared cortical variability. *Neuron*, 87(3):644–656, 2015. 2.3, 4.4.3, 6, 6.1.1, 6.1.6, 7.1, 8.0.2

Ashok Litwin-Kumar and Brent Doiron. Slow dynamics and high variability in balanced cortical networks with clustered connections. *Nature neuroscience*, 15(11):1498, 2012. 7.2, 7.2.2, 8.0.4

Gabriel Loaiza-Ganem, Yuanjun Gao, and John P Cunningham. Maximum entropy flow networks. *arXiv preprint arXiv:1701.03504*, 2017. 8.0.4

Artur Luczak, Peter Bartho, and Kenneth D Harris. Spontaneous events outline the realm of possible sensory responses in neocortical populations. *Neuron*, 62(3):413–25, May 2009. doi: 10.1016/j.neuron.2009.03.014. 2.2, 3, 3.1.3, 3.1.4, A.4

Thomas Zhihao Luo and John HR Maunsell. Neuronal modulations in visual cortex are associated with only one of multiple components of attention. *Neuron*, 86(5):1182–1188, 2015. 6.1.4, 6.1.5, 6.1.7, 6.1.8, D.8, D.11

Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of machine learning research*, 9(Nov):2579–2605, 2008. 2.2

Christian K Machens. Adaptive sampling by information maximization. *Physical Review Letters*, 88(22):228104, 2002. 4

Christian K Machens, Tim Gollisch, Olga Kolesnikova, and Andreas VM Herz. Testing the efficiency of sensory coding with optimal stimulus ensembles. *Neuron*, 47(3):447–456, 2005. 4

Christian K Machens, Ranulfo Romo, and Carlos D Brody. Functional, but not anatomical, separation of "what" and "when" in prefrontal cortex. *J Neurosci*, 30(1):350–60, Jan 2010. doi: 10.1523/JNEUROSCI.3276-09.2010. 1, 3.2, 7.3

Jakob H. Macke, Lars Buesing, John P. Cunningham, Byron M. Yu, Krishna V. Shenoy, and Maneesh Sahani. Empirical models of spiking in neural populations. In *NIPS*, pages 1350–1358, 2011. 7.3.4

Abhijit Mandal and Andrzej Cichocki. Non-linear canonical correlation analysis using alpha-beta divergence. *Entropy*, 15(7):2788–2804, 2013. 5.5, 5.8

V Mante, D Sussillo, KV Shenoy, and WT Newsome. Drift correction for electrophysiology and two-photon calcium imaging. Number III-82, Denver, CO, 2018. COSYNE. D.1.2

Valerio Mante, David Sussillo, Krishna V Shenoy, and William T Newsome. Context-dependent computation by recurrent dynamics in prefrontal cortex. *Nature*, 503(7474):78–84, 2013. 1, 1, 2.1.3, 2.2, 3, 3.2, 5.7.1, 6.1.3, 8

Kevan AC Martin and Sylvia Schröder. Functional heterogeneity in neighboring neurons of cat primary visual cortex in response to both artificial and natural stimuli. *The Journal of Neuroscience*, 33(17):7325–7344, 2013. 3.2

Ofer Mazor and Gilles Laurent. Transient dynamics versus fixed points in odor representations by locust antennal lobe projection neurons. *Neuron*, 48(4):661–73, Nov 2005. doi: 10.1016/j.neuron.2005.09.032. 1, 2.2, 3, 3.2

Matthew J McGinley, Stephen V David, and David A McCormick. Cortical membrane potential signature of optimal states for sensory signal detection. *Neuron*, 87(1):179–192, 2015a. 2.3, 6, 6.1.2, 6.1.8, 6.1.8

Matthew J McGinley, Martin Vinck, Jacob Reimer, Renata Batista-Brito, Edward Zagha, Cathryn R Cadwell, Andreas S Tolias, Jessica A Cardin, and David A McCormick. Waking state: rapid variations modulate neural and behavioral responses. *Neuron*, 87(6):1143–1161, 2015b. 1, 2.3, 6, 6.1.1, 6.1.2, 6.1.3, 6.1.4, 6.1.8, 6.1.8

Earl K Miller, Cynthia A Erickson, and Robert Desimone. Neural mechanisms of visual working memory in prefrontal cortex of the macaque. *Journal of Neuroscience*, 16(16):5154–5167, 1996. 2.1.3

Jude F Mitchell, Kristy A Sundberg, and John H Reynolds. Spatial attention decorrelates intrinsic activity fluctuations in macaque area v4. *Neuron*, 63(6):879–888, 2009. 6, 6.1.5, 6.1.5, D.2

Tirin Moore and Katherine M Armstrong. Selective gating of visual signals by microstimulation of frontal cortex. *Nature*, 421(6921):370, 2003. 6, 8.0.4

Tirin Moore and Marc Zirnsak. Neural mechanisms of selective visual attention. *Annual review of psychology*, 68:47–72, 2017. 6, 6.1.5

Rubén Moreno-Bote, Jeffrey Beck, Ingmar Kanitscheider, Xaq Pitkow, Peter Latham, and Alexandre Pouget. Information-limiting correlations. *Nature neuroscience*, 17(10):1410, 2014. 2.3, 6, 6.1.4, 6.1.6, 6.1.8, 7.1, 8.0.2, 8.0.3

J Anthony Movshon and William T Newsome. Visual response properties of striate cortical neurons projecting to area mt in macaque monkeys. *J Neurosci*, 16(23):7733–7741, 1996. 3.1.1, A.2

William T Newsome, Kenneth H Britten, and J Anthony Movshon. Neuronal correlates of a perceptual decision. *Nature*, 341(6237):52, 1989. 6

AM Ni, DA Ruff, JJ Alberts, J Symmonds, and MR Cohen. Learning and attention reveal a general relationship between population activity and behavior. *Science*, 359(6374):463–465, 2018. 2.3, 6, 6.1.6, 6.1.8, 6.1.8

Hendrikje Nienborg and Bruce G Cumming. Macaque v2 neurons, but not v1 neurons, show choice-related activity. *Journal of Neuroscience*, 26(37):9567–9578, 2006. 6

Hendrikje Nienborg and Bruce G Cumming. Decision-related activity in sensory neurons reflects more than a neuron's causal effect. *Nature*, 459(7243):89, 2009. 6

Kevin N O'Connor, Christopher I Petkov, and Mitchell L Sutter. Adaptive stimulus optimization for auditory cortical neurons. *Journal of Neurophysiology*, 94(6):4051–4067, 2005. 4

Michael Okun, Pierre Yger, Stephan L Marguet, Florian Gerard-Mercier, Andrea Benucci, Steffen Katzner, Laura Busse, Matteo Carandini, and Kenneth D Harris. Population rate dynamics and multineuron firing patterns in sensory cortex. *J Neurosci*, 32(48):17108–17119, 2012. 3.2

Michael Okun, Nicholas A Steinmetz, Lee Cossell, M Florencia Iacaruso, Ho Ko, Péter Barthó, Tirin Moore, Sonja B Hofer, Thomas D Mrsic-Flogel, Matteo Carandini, et al. Diverse coupling of neurons to populations in sensory cortex. *Nature*, 521(7553):511–515, 2015. 2.3, 4.5, 6, 6.1.1, 7.1, 8.0.2, 8.0.4, D.8, D.11

Adriana Olmos and Frederick AA Kingdom. A biologically inspired algorithm for the recovery of shading and reflectance images. *Perception*, 33(12):1463–1473, 2004. 4.4

Bruno A Olshausen and David J Field. How close are we to understanding v1? *Neural Computation*, 17(8):1665–1699, 2005. 1, 2.1.1, 2.2, 3.1.6

Bruno A Olshausen et al. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature*, 381(6583):607–609, 1996. 3.2

Marius Pachitariu, Dmitry R Lyamzin, Maneesh Sahani, and Nicholas A Lesica. State-dependent population coding in primary auditory cortex. *Journal of Neuroscience*, 35(5):2058–2073, 2015. 2.3

Marius Pachitariu, Carsen Stringer, Sylvia Schröder, Mario Dipoppa, L Federico Rossi, Matteo Carandini, and Kenneth D Harris. Suite2p: beyond 10,000 neurons with standard two-photon microscopy. *Biorxiv*, page 061507, 2016. 8.0.4

Liam Paninski. Maximum likelihood estimation of cascade point-process neural encoding models. *Network: Computation in Neural Systems*, 15(4):243–262, 2004. 2.2.2

Liam Paninski. Asymptotic theory of information-theoretic experimental design. *Neural Computation*, 17(7):1480–1507, 2005. 4

Mijung Park, J Patrick Weller, Gregory D Horwitz, and Jonathan W Pillow. Bayesian active learning of neural firing rate maps with transformed gaussian process priors. *Neural Computation*, 26(8):1519–1541, 2014. 4, 6.1.8

Andrew J Parker and William T Newsome. Sense and the single neuron: probing the physiology of perception. *Annual review of neuroscience*, 21(1):227–277, 1998. 6

Michael Pecka, Yunyun Han, Elie Sader, and Thomas D Mrsic-Flogel. Experience-dependent specialization of receptive field surround for selective coding of natural scenes. *Neuron*, 84(2): 457–469, 2014. 3.2

Jonathan W Pillow and Mijung Park. Adaptive bayesian methods for closed-loop neurophysiology. In A. El Hady, editor, *Closed Loop Neuroscience*. Elsevier, 2016. 4

Jonathan W Pillow, Jonathon Shlens, Liam Paninski, Alexander Sher, Alan M Litke, EJ Chichilnisky, and Eero P Simoncelli. Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995, 2008. 2.2.2, 8.0.3, 8.0.4

Vadim S Polikov, Patrick A Tresco, and William M Reichert. Response of brain tissue to chronically implanted neural electrodes. *Journal of neuroscience methods*, 148(1):1–18, 2005. 2.1.4

Neil C Rabinowitz, Robbe L Goris, Marlene Cohen, and Eero P Simoncelli. Attention stabilizes the shared gain of v4 populations. *Elife*, 4, 2015. 2.3, 6, 6.1.1, 6.1.8, 8.0.2, D.2

Alec Radford, Luke Metz, and Soumith Chintala. Unsupervised representation learning with deep

convolutional generative adversarial networks. *arXiv preprint arXiv:1511.06434*, 2015. 4.5

David Raposo, Matthew T Kaufman, and Anne K Churchland. A category-free neural population supports evolving demands during decision-making. *Nat Neurosci*, 17:1784–1792, 2014. 3.2, 6.1.3

Jacob Reimer, Emmanouil Froudarakis, Cathryn R Cadwell, Dimitri Yatsenko, George H Denfield, and Andreas S Tolias. Pupil fluctuations track fast switching of cortical states during quiet wakefulness. *Neuron*, 84(2):355–362, 2014. 2.3, 6.1.2, 6.1.8

Jacob Reimer, Matthew J McGinley, Yang Liu, Charles Rodenkirch, Qi Wang, David A McCormick, and Andreas S Tolias. Pupil fluctuations track rapid changes in adrenergic and cholinergic activity in cortex. *Nature communications*, 7:13289, 2016. 2.3, 6, 6.1.2, 6.1.8

Alfonso Renart, Jaime De La Rocha, Peter Bartho, Liad Hollender, Néstor Parga, Alex Reyes, and Kenneth D Harris. The asynchronous state in cortical circuits. *science*, 327(5965):587–590, 2010. 7.2.2, 8.0.4

John H Reynolds and Leonardo Chelazzi. Attentional modulation of visual processing. *Annu. Rev. Neurosci.*, 27:611–647, 2004. 6.1.5

John H Reynolds, Tatiana Pasternak, and Robert Desimone. Attention increases sensitivity of v4 neurons. *Neuron*, 26(3):703–714, 2000. 6.1.5

BARRY J Richmond and LANCE M Optican. Temporal encoding of two-dimensional patterns by single units in primate primary visual cortex. ii. information transmission. *Journal of Neurophysiology*, 64(2):370–380, 1990. 3, 3.2

Mattia Rigotti, Omri Barak, Melissa R Warden, Xiao-Jing Wang, Nathaniel D Daw, Earl K Miller, and Stefano Fusi. The importance of mixed selectivity in complex cognitive tasks. *Nature*, 497 (7451):585–90, May 2013. doi: 10.1038/nature12160. 1, 1, 2.2, 3, 8

Dario Ringach and Robert Shapley. Reverse correlation in neurophysiology. *Cognitive Science*, 28(2):147–166, 2004. 4

Dario L Ringach. Population coding under normalization. *Vision research*, 50(22):2223–2232, 2010. 3.2

DA Robinson. The mechanics of human saccadic eye movement. *The Journal of physiology*, 174 (2):245–264, 1964. 8.0.4

Kathleen S Rockland and Gary W Van Hoesen. Direct temporal-occipital feedback connections to striate cortex (v1) in the macaque monkey. *Cerebral Cortex*, 4(3):300–313, 1994. 2.1.1

Anna W Roe, Leonardo Chelazzi, Charles E Connor, Bevil R Conway, Ichiro Fujita, Jack L Gallant, Haidong Lu, and Wim Vanduffel. Toward a unified theory of visual area V4. *Neuron*, 74(1):12–29, 2012. 1, 2.1.2, 4.4.2, 5, 6.1.6

Sam T Roweis and Lawrence K Saul. Nonlinear dimensionality reduction by locally linear embedding. *science*, 290(5500):2323–2326, 2000. 2.2, 3.2, A.4

Douglas A Ruff and Marlene R Cohen. Global cognitive factors modulate correlated response variability between v4 neurons. *Journal of Neuroscience*, 34(49):16408–16416, 2014. 1, 2.3, 6.1.8, 6.1.8

Nicole C Rust and J Anthony Movshon. In praise of artifice. *Nature Neuroscience*, 8(12): 1647–1650, 2005. 4

Patrick T Sadtler, Kristin M Quick, Matthew D Golub, Steven M Chase, Stephen I Ryu, Elizabeth C Tyler-Kabara, Byron M Yu, and Aaron P Batista. Neural constraints on learning. *Nature*, 512(7515):423, 2014. 2.2, 3, 3.1.3, 3.1.4, 3.2, 6.1.8, 8.0.3, 8.0.4, A.4

Gopal Santhanam, Stephen I Ryu, Byron M Yu, Afsheen Afshar, and Krishna V Shenoy. A high-performance brain–computer interface. *Nature*, 442(7099):195–198, 2006. 6.1.8

Gopal Santhanam, Byron M Yu, Vikash Gilja, Stephen I Ryu, Afsheen Afshar, Maneesh Sahani, and Krishna V Shenoy. Factor-analysis methods for higher-performance neural prostheses. *J Neurophysiol*, 102(2):1315–30, Aug 2009. doi: 10.1152/jn.00097.2009. 1, 7.3, 8

Marieke L Schölvinck, Aman B Saleem, Andrea Benucci, Kenneth D Harris, and Matteo Carandini. Cortical state determines global variability and correlations in visual cortex. *Journal of Neuroscience*, 35(1):170–178, 2015. 2.3

Odelia Schwartz, Jonathan W Pillow, Nicole C Rust, and Eero P Simoncelli. Spike-triggered neural characterization. *Journal of Vision*, 6(4):13–13, 2006. 4

Dino Sejdinovic, Bharath Sriperumbudur, Arthur Gretton, and Kenji Fukumizu. Equivalence of distance-based and rkhs-based statistics in hypothesis testing. *The Annals of Statistics*, 41(5): 2263–2291, 2013. 5

J.D. Semedo, B.R. Cowley, Zandvakili A., C.K. Machens, B.M. Yu, and A. Kohn. Characterizing population-level interactions between v1 and v2. Number 331.07.2015, Washington, DC, 2015. Society for Neuroscience. 8.0.4

Joao Semedo, Amin Zandvakili, Adam Kohn, Christian K Machens, and Byron M Yu. Extracting latent structure from multiple interacting neural populations. In *Advances in neural information processing systems*, pages 2942–2950, 2014. 2.2.3, 3.2, 5.7.2, 6.1.3, 8.0.4

Michael N Shadlen and William T Newsome. The variable discharge of cortical neurons: implications for connectivity, computation, and information coding. *Journal of neuroscience*, 18(10): 3870–3896, 1998. 2.3, 6, 6.1.4, 6.1.8

Wenhui Sheng and Xiangrong Yin. Direction estimation in single-index models via distance covariance. *Journal of Multivariate Analysis*, 122:148–161, 2013. 5.2.1

Wenhui Sheng and Xiangrong Yin. Sufficient dimension reduction via distance covariance. *Journal of Computational and Graphical Statistics*, 25(1):91–104, 2016. 5.5

Oren Shriki, Adam Kohn, and Maoz Shamir. Fast coding of orientation in primary visual cortex. *PLoS Comput Biol*, 8(6):e1002536, 2012. 5.7.1

Eero P Simoncelli and William T Freeman. The steerable pyramid: A flexible architecture for multi-scale derivative computation. In *Image Processing, 1995. Proceedings., International Conference on*, volume 3, pages 444–447. IEEE, 1995. 4.3

Eero P Simoncelli and Bruno A Olshausen. Natural image statistics and neural representation. *Annual review of neuroscience*, 24(1):1193–1216, 2001. 3.1.7, A.4, A.6

Matthew A Smith and Adam Kohn. Spatial and temporal scales of neuronal correlation in primary

visual cortex. *The Journal of Neuroscience*, 28(48):12591–12603, 2008. 7, A.1

Darragh Smyth, Ben Willmore, Gary E Baker, Ian D Thompson, and David J Tolhurst. The receptive-field organization of simple cells in primary visual cortex of ferrets under natural scene stimulation. *J Neurosci*, 23(11):4746–4759, 2003. 3.1.3, 3.1.6, 3.2

Adam C Snyder, Michael J Morais, Adam Kohn, and Matthew A Smith. Correlations in v1 are reduced by stimulation outside the receptive field. *Journal of Neuroscience*, 34(34):11222–11227, 2014. 6.1.8

Adam C Snyder, Michael J Morais, and Matthew A Smith. Dynamics of excitatory and inhibitory networks are differentially altered by selective attention. *Journal of neurophysiology*, 116(4): 1807–1820, 2016. 6

Mircea Steriade, I Timofeev, and F Grenier. Natural waking and sleep states: a view from inside neocortical neurons. *Journal of neurophysiology*, 85(5):1969–1985, 2001. 6.1.8

Ian H Stevenson and Konrad P Kording. How advances in neural recording affect data analysis. *Nat Neurosci*, 14(2):139–42, Feb 2011. doi: 10.1038/nn.2731. 1, 4

Mark Stopfer, Vivek Jayaraman, and Gilles Laurent. Intensity versus identity coding in an olfactory system. *Neuron*, 39(6):991–1004, Sep 2003. 2.2, 7.3.4

Christoph Stosiek, Olga Garaschuk, Knut Holthoff, and Arthur Konnerth. In vivo two-photon calcium imaging of neuronal networks. *Proceedings of the National Academy of Sciences*, 100 (12):7319–7324, 2003. 8.0.4

David Sussillo, Rafal Jozefowicz, LF Abbott, and Chethan Pandarinath. Lfads-latent factor analysis via dynamical systems. *arXiv preprint arXiv:1608.06315*, 2016. 2.2.1

Deborah F. Swayne, Duncan Temple Lang, Andreas Buja, and Dianne Cook. GGobi: evolving from XGobi into an extensible framework for interactive data visualization. *Computational Statistics & Data Analysis*, 43:423–444, 2003. 7.3

Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015. 3.1.8, 4.2, 4.2, 4.4.1, A.7

Gábor J Székely and Maria L Rizzo. Brownian distance covariance. *The annals of applied statistics*, 3(4):1236–1265, 2009. 5, 5.1, 5.1

Vargha Talebi and Curtis L Baker. Natural versus synthetic stimuli for estimating receptive field models: a comparison of predictive robustness. *J Neurosci*, 32(5):1560–1576, 2012. 3.1.3, 3.1.6, 3.2

Joshua B Tenenbaum, Vin De Silva, and John C Langford. A global geometric framework for nonlinear dimensionality reduction. *science*, 290(5500):2319–2323, 2000. 2.2, 3.2, A.4

David J Tolhurst, J Anthony Movshon, and Andrew F Dean. The statistical reliability of signals in single neurons in cat and monkey visual cortex. *Vision research*, 23(8):775–785, 1983. 3.2

Nelson K Totah, Ricardo M Neves, Stefano Panzeri, Nikos K Logothetis, and Oxana Eschenko. Monitoring large populations of locus coeruleus neurons reveals the non-global nature of the

norepinephrine neuromodulatory system. *bioRxiv*, page 109710, 2017. 6.1.3, 6.1.8

Jon Touryan, Gidon Felsen, and Yang Dan. Spatial structure of complex cell receptive fields measured with natural images. *Neuron*, 45(5):781–791, 2005. 3.2

Wilson Truccolo, Uri T Eden, Matthew R Fellows, John P Donoghue, and Emery N Brown. A point process framework for relating neural spiking activity to spiking history, neural ensemble, and extrinsic covariate effects. *Journal of neurophysiology*, 93(2):1074–1089, 2005. 2.2.2

Leslie G Ungerleider, Thelma W Galkin, Robert Desimone, and Ricardo Gattass. Cortical connections of area v4 in the macaque. *Cerebral Cortex*, 18(3):477–499, 2007. 2.1.2, 6.1.3, 6.1.3

Laurens Van Der Maaten, Eric Postma, and Jaap Van den Herik. Dimensionality reduction: a comparative. *J Mach Learn Res*, 10:66–71, 2009. 5, 5.1

A. Vedaldi and K. Lenc. Matconvnet – convolutional neural networks for Matlab. In *Proceeding of the ACM Int. Conf. on Multimedia*, 2015. 4.4.1, A.7

Michael Vidne, Yashar Ahmadian, Jonathon Shlens, Jonathan W Pillow, Jayant Kulkarni, Alan M Litke, EJ Chichilnisky, Eero Simoncelli, and Liam Paninski. Modeling the impact of common noise inputs on the network activity of retinal ganglion cells. *Journal of computational neuroscience*, 33(1):97–121, 2012. 2.2.2, 7.3.5, 8.0.4

Martin Vinck, Renata Batista-Brito, Ulf Knoblich, and Jessica A Cardin. Arousal and locomotion make distinct contributions to cortical activity patterns and visual encoding. *Neuron*, 86(3): 740–754, 2015. 2.3, 6.1.2, 6.1.3, 6.1.8

William E Vinje and Jack L Gallant. Sparse coding and decorrelation in primary visual cortex during natural vision. *Science*, 287(5456):1273–1276, 2000. 2.1.1, 3.2

William E Vinje and Jack L Gallant. Natural stimulation of the nonclassical receptive field increases information transmission efficiency in v1. *J Neurosci*, 22(7):2904–2915, 2002. 3.2

C van Vreeswijk and Haim Sompolinsky. Chaotic balanced state in a model of cortical circuits. *Neural computation*, 10(6):1321–1371, 1998. 7.2.2

Geoffrey S Watson. Smooth regression analysis. *Sankhyā: The Indian Journal of Statistics, Series A*, pages 359–372, 1964. 4.2

Felix A Wichmann, Doris I Braun, and Karl R Gegenfurtner. Phase noise and the classification of natural images. *Vision research*, 46(8):1520–1529, 2006. 3.1.7, A.6

Ryan C Williamson, Benjamin R Cowley, Ashok Litwin-Kumar, Brent Doiron, Adam Kohn, Matthew A Smith, and Byron M Yu. Scaling properties of dimensionality reduction for neural populations and network models. *PLoS computational biology*, 12(12):e1005141, 2016. 7.2.3, 8.0.4

Nathan R Wilson, Caroline A Runyan, Forea L Wang, and Mriganka Sur. Division and subtraction by distinct cortical inhibitory networks in vivo. *Nature*, 488(7411):343, 2012. 6.1.8

Stephanie C Wissig and Adam Kohn. The influence of surround suppression on adaptation effects in primary visual cortex. *J Neurophysiol*, 107(12):3370–3384, 2012. A.1

Daniela M Witten and Robert J Tibshirani. Extensions of sparse canonical correlation analysis

with applications to genomic data. *Statistical applications in genetics and molecular biology*, 8 (1):1–27, 2009. 5, 5.8, 8.0.4

Anqi Wu, Nicholas G Roy, Stephen Keeley, and Jonathan W Pillow. Gaussian process based nonlinear latent structure discovery in multivariate spike train data. In *Advances in Neural Information Processing Systems*, pages 3499–3508, 2017. 2.2.1

Yingcun Xia. A semiparametric approach to canonical analysis. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, 70(3):519–543, 2008. 5.5, 5.8

Jianxiong Xiao. Princeton vision and robotics toolkit, 2013. Available from: `http://3dvision.princeton.edu/pvt/GoogLeNet/`. 4.4.1, A.7

Yukako Yamane, Eric T Carlson, Katherine C Bowman, Zhihong Wang, and Charles E Connor. A neural code for three-dimensional object shape in macaque inferotemporal cortex. *Nature Neuroscience*, 11(11):1352–1360, 2008. 1, 4, 4.1, 4.4.1

Daniel LK Yamins and James J DiCarlo. Using goal-driven deep learning models to understand sensory cortex. *Nature Neuroscience*, 19(3):356–365, 2016. 3.1.8, 4.4, 4.4.2, 8.0.1, 8.0.3, 8.0.4

Daniel LK Yamins, Ha Hong, Charles F Cadieu, Ethan A Solomon, Darren Seibert, and James J DiCarlo. Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, 111(23):8619–8624, 2014. 2.1.2, 6.1.6

B.M. Yu, A. Kohn, and M.A. Smith. Estimating shared firing rate fluctuations in neural populations. Number 483.18.2011/NN1, Washington, DC, 2011. Society for Neuroscience. 7.1

Byron M Yu, John P Cunningham, Gopal Santhanam, Stephen I Ryu, Krishna V Shenoy, and Maneesh Sahani. Gaussian-process factor analysis for low-dimensional single-trial analysis of neural population activity. *J Neurophysiol*, 102(1):614–35, Jul 2009. doi: 10.1152/jn.90941.2008. 2.2.1, 7.3, 7.3.1, 7.3.4, 7.3.4, 7.3.4, 7.3.4, 8

Tie Yun and Ling Guan. Human emotional state recognition using real 3d visual features from gabor library. *Pattern Recognition*, 46(2):529–538, 2013. A.6

Amin Zandvakili and Adam Kohn. Coordinated neuronal activity enhances corticocortical communication. *Neuron*, 87(4):827–839, 2015. 5.7.2, 5.7.3

Yuan Zhao and Il Memming Park. Variational latent gaussian process for recovering single-trial dynamics from population spike trains. *Neural computation*, 29(5):1293–1316, 2017. 2.2.1

Zhanxing Zhu, Timo Similä, and Francesco Corona. Supervised distance preserving projections. *Neural processing letters*, 38(3):445–463, 2013. 5.5

Corey M Ziemba, Jeremy Freeman, J Anthony Movshon, and Eero P Simoncelli. Selectivity and tolerance for visual texture in macaque v2. *Proceedings of the National Academy of Sciences*, 113(22):E3140–E3149, 2016. 2.2

Ehud Zohary, Michael N Shadlen, and William T Newsome. Correlated neuronal discharge rate and its implications for psychophysical performance. 1994. 6

Hui Zou, Trevor Hastie, and Robert Tibshirani. Sparse principal component analysis. *Journal of computational and graphical statistics*, 15(2):265–286, 2006. 8.0.4