The estimation and explanation of tuning curves in intermediate visual areas

Dean Pospisil

A dissertation
submitted in partial fulfillment of the
requirements for the degree of

Doctor of Philosophy
University of Washington 2021

Reading committee:

Wyeth Bair, Chair

Anitha Pasupathy

Gregory Horwitz

Eric Shea-Brown

Program Authorized to Offer Degree:
Neuroscience

University of Washington

Abstract

The estimation and explanation of tuning curves in intermediate visual areas

Dean Pospisil

Chair of the supervisory committee:
Wyeth Bair
Department of Biological Structure

In sensory neuroscience, the characterization of neuronal tuning curves is the de facto method for understanding the bulk of cortical sensory representation. In this thesis, I make two primary contributions to this approach. First, I provide validated statistical estimators that account for the corrupting effect of trial-to-trial variability on the correlation between neural tuning curves and between a noiseless model and a tuning curve. Second, I develop a single-neuron approach to fitting visual cortical responses to deep neural network (DNN) models and demonstrate a method to attribute tuning curve properties of single units in a DNN to the covariance structure of their inputs, which is key to understanding selectivity and invariance. Taken together these methods support the accurate quantification of fundamental summary statistics of noisy neural tuning curves and fine-grained characterization by DNN models of sensory tuning curves. I discuss how this can support the overarching goal of understanding biological intelligence.

Acknowledgements

I'll start from the beginning. Mom, Dad thanks for supporting me in being myself. You supported my interest in science from the moment I showed an interest and it has brought me so much joy and meaning.

Cameron, thanks for being the one in the family who gets 'it' (academia and science) and whose ear I could bend as I processed my journey through the PhD. Your work ethic, good humor, and perseverance were a constant inspiration to me.

Troy, thanks for helping get me on this path by providing guidance at key points in my life. The thoughtfulness and focus you bring to your endeavours have continuously aided and inspired me.

Emma, thank you for joining me on my journey through the last several years of my PhD. You kept me grounded and helped make the whole process probably the best time of my life. Even if you were the only thing that came out of my time spent in the health science building... worth it. Also, I'm sorry that I forgot to acknowledge your help with Figure 6.25 on that poster but I am setting the record straight here.

Wyeth, You have been an amazing mentor. You gave me a huge level of independence while at the same time generously give one-on-one time delving deeply into my research and through out you are consistently only interested in the truth of how any part of the visual system works. This latter quality permeates your style and mode of thinking, for example you're deeply critical of anything that isn't concretely new knowledge about the nervous system which (or that?) is a lot because we undoubtedly know very little about the nervous system. The interaction of these two qualities, generosity and unvarnished critical scientific thinking, is how I learned my most important lessons as a scientist under your tutelage. The thousands of hours we spent poring through research together I absorbed hopefully a small fraction of the way you think about science and it has and will serve me well. I hope this is only the beginning of many more conversations to come.

Anitha, thank you for informally advising me, giving feedback on grants and papers, generously sharing data, and most of all letting me into your lab. It was an invaluable experience that has shaped the way I think about the interplay between theory, experimentation, and the insight that lies somewhere closer to the data. Your ability to look at papers and grants and zoom out to see the forest for the trees saved Wyeth and I from getting sucked into our own navels many times and I aspire to match that vision. It was your guest lecture on V4 in my first year courses where the initial glimmering of my interest in the ventral stream began and it transformed into a fascinating journey.

Greg, I always look forward to your distinct brand of clarity, levity, and seriousness. I really appreciate the focus and thought you gave to every paper of mine you looked at and talk I gave. Also, thanks for encouraging me to take courses on statistics. It has clearly positively impacted my research (see Chapters 3-6). Your course on neuro-statistics was what initially got me excited about statistics and Your teaching style largely seemed to be you thinking through things out loud in a clearheaded manner and enjoying yourself while you did. The passion was contagious and I'm glad I caught it!

Eric, your enthusiasm and perspective were invaluable in committee meetings. Thanks for pushing me to explore other ways of thinking about my work, going out of your way to create inclusive meetings, and connecting me with other researchers who I would not have met otherwise. Also thanks for all the espressos!

John, before every thesis meeting you would send me an email telling me to keep the meeting under an hour and then after the meeting was over you would chat with

me for another hour. You dispensed several pieces of advice that I still keep in mind: there is no rushing science, my sense of what I can accomplish in a month is typically a wild overestimate, keep the thesis meeting under an hour, and take opportunities to look up from my research and think broadly about what direction I want to take it. It's criminal that you're retiring and I'll never forgive you for it.

My fellow lab mates in the Bair lab.

Pamela, I'm very happy I got to overlap with your tenure in the Bair lab. Your good humor made the lab fun to be in. Your informal mentorship on science and academia was invaluable as I tried to figure out my place in it. Most of all you've been a great friend throughout my time in the Bair lab and beyond.

Reza, I am very glad that on a you decided to lend your intellect to neuroscience and you chose Wyeth's lab. I'm sorry about that one time I made you go to SFN 3-days earlier because I thought my talk was then. I've always enjoyed our discussions on research, your a very clearheaded thinker but also open to wild ideas, which makes for great conversation.

Saba, thanks for the many impromptu and engaging discussion about research. Your levelheaded and thoughtful approach is much needed in our field and I look forward to your future contributions.

Zhaojie, thanks for fielding my random question with interest and your thoughtful questions during lab meetings.

The Pasupathy lab as a whole was a huge part of my PhD work and training.

Polina, worked closely with me to collect preliminary data, I always had a lot of fun when we were at the rig together, and she was a gifted scientist.

Dina, you were my introduction to electrophysiology, always refreshingly upbeat, and generously shared your V4 data on fill outline responses.

Yasmine, thank you for generously sharing your data on V4 invariance and always giving really good critical feedback whenever I presented research.

Taekjun, thank you for sharing your data on texture selectivity and also always listened carefully and asking insightful question.

Tony, you have a great sense of humor, are very easy going, and I've really enjoyed discussing deep neural networks and neuroscience with you.

Tomo thank you for always being ready to help out in the lab and for your incredible sense of humor.

My mentees, Teddy and Drew. Thanks for trusting me with your time and helping me learn how to support others in their research and goals.

The Neuroscience Graduate school, the environment at UW has been an amazing place to develop intellectually. From getting to learn brain anatomy through dissections of human brains to getting taught by world class experts at all levels of neuroscience made the nervous system a much more concrete and textured object. I know a lot goes on behind the scenes to support this environment, particularly by the directors, Lucia and Kyle, and I am deeply grateful.

My fellow batch mates, Abhishek, Chris, Jesse, Jane, Kali, and Rachel. Your camaraderie and excitement for science made my first year and on a pleasure.

During graduate school I have had inumerable conversations with the broader UW community which bit-by-bit have greatly shaped the way I think about neuroscience. These include memorable conversations with Dina Popovkina, Anthony Bigelow, Taekjun Kim, Tomo Namima, Iris Shi, Matthew Farrell, Ali Weber, Rich Pang, Timothy Oleskiw, Stephanie Seeman, Sierra Schleufer, Alison Duffy, Max Turner, Yoni Browning, Clare Gamlin, Kameron Decker-Harris, Gabriel Obregon-Henao, Lindsey Kishline, Mark Wronkiewicz, Patrick Weller, Andrew Bogard, Stavros Zanos, and Felix Darvas.

I'd like to thank the statistics PhD's and masters students who let me join their study group and provided the camaraderie needed to get through the first and second year courses: Sarah, Katina, Jonah, Jessica, Kristof, Jun, and Heather. The courses have really helped my research and I'm not sure I would have made it through on my own.

I had two main outside collaborators. The first was Adam Kohn who discussed initial drafts of the signal correlation paper and supplied the V1 data we analyzed. Yen-Chi Chen gave helpful comments on my initial work on neuron to model correlation.

My funding sources, I'm always amazed I get paid to do this job and I feel incredibly privileged and thankful for the efforts of the NSF and NIH to support basic science. Not to mention Anitha and Wyeth whose grants helped support me for the last few years of graduate school.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this introduction, I contextualize my thesis research into characterizing complex neural selectivity. I specify its contribution to understanding how the nervous system generates intelligent behavior. I first outline a conceptual framework for quantifying intelligence and then locate it within this framework. Keeping this framework in mind, I then outline the major topics of my thesis: (1) estimation of key parameters of neural tuning curves and (2) the use of linear-nonlinear cascades for gaining insights into complex tuning curves. To skip directly to a summary of the contents of this thesis see 'Summary of chapters'.

1.1 Making progress on intelligence

What is intelligence? Many definitions have been proposed and there is no clear consensus [1]. Most definitions include an agent in an environment that provides observations on the basis of which the agent acts on the environment, and finally some score of the agent on the basis of an environmental outcome. A measure of the intelligence of the agent can be the average value of the score (Figure 1). This measure cannot be evaluated on a given agent until the environment, available actions, observations, and score are defined. And what definition could be considered a scoring of true intelligence is a matter of debate.

The sensory neuroscientist has an advantage in the study of intelligence because primates and many other organisms exhibit intelligent behavior on the basis of their nervous system. While there may be some doubts exactly which environments or scoring methods reflect intelligence, there would be little doubt as to the observations the organism is making: the action potentials of all afferent sensory neurons. The brain only 'observes' incoming spike trains. Thus by studying the relation of the spiking activity to stimuli, the sensory neuroscientist can confidently make progress on one foundation of intelligence: observations of the environment.

Typically to characterize this relationship, the sensory neuroscientist will fix some environmental aspect of interest like the pattern of light impinging on the retina, termed a stimulus, and measure the subsequent responses of one or many neurons. A challenge comes from the fact that in general, the relationship is not a deterministic one. The same stimulus presented at different points in time to the same organism will often evoke a different number of spikes. Thus the spike count, randomizing across stimulus presentations, can be modeled as a random variable:

$$R|s$$

where R is the spike count conditioned on the presentation of stimulus s. Below I discuss a natural decomposition of this random variable into its expected value, termed the tuning curve, and variability.

1.2 The tuning curve and variability

The source of variation in the responses of sensory neurons is an active area of research. Some variation may be the result of low-level processes such as background synaptic noise, EPSC kinetics, and jitter of population excitatory inputs [2] but, in vitro, it has been found with deterministic variation in inputs neocortical neurons can be extremely precise in their spike timing [3]. Large amounts of variation in firing rate have been shown to be induced by changes in attentional state [4]. More generally sensitivity to long-term history preceding the presentation of a stimulus may cause the influence of the stimulus to change from trial to trial even in a deterministic system. This idea has been studied via the concept of 'chaotic dynamics' [5, 6]. Regardless, the source of variability is the result of factors the experimentalist cannot control, making it a difficult object of study. The sensory neuroscientist can side-step this issue by decomposing responses into their mean and variance:

$$R|s = \mathrm{E}[R|s] + (R|s - \mathrm{E}[R|s]).$$

Where $\mathrm{E}[R|s]$ is the expected number of spikes in response to a stimulus and $(R|s - \mathrm{E}[R|s])$ accounts for the 'residual' variability. $\mathrm{E}[R|s]$ is colloquially known as the tuning curve and by definition is not stochastic but a deterministic function of the stimulus.

A point along the tuning curve of a neuron can be estimated by presenting a stimulus to an organism and recording the number of spikes evoked from a neuron. The variance of this estimate will decrease if the stimulus is presented more than once and spike counts are averaged.

This thesis focuses on the tuning curve but it recognizes the influence that trial-to-trial variability can have on the inference of the tuning curve. Issues of inference will be discussed at length later, but here I consider issues involved in characterizing the tuning curve ignoring problematic influences of variability.

To experimentally characterize a tuning curve multiple, points can be estimated along it, but the choice of which points is a fraught one. The space of s is typically large. For example, a 10×10-pixel image with 10 discretized values (0 black, 9 white) includes 10^{100} unique possible images, more atoms than there are estimated to be in the universe. Yet it is possible to characterize the tuning curve with a small fraction of stimuli if we take advantage of statistical inference and image computable models.

Often stimuli are considered fixed and there is no consideration of the process by which, or pool from which, stimuli were drawn. If on the other hand a population of stimuli is defined and a subset is randomly drawn from it, inferences can be made about response properties with respect to the intractably larger population. A simple example would be taking the average firing rate across the responses to a sample of stimuli; this estimate will converge to the population mean. More to the point would be the correlation of a model with the responses of a neuron. As the correlation of model predictions to neural responses to a random sample of stimuli grows, we can be more confident the model matches the neuron for all stimuli from the population.

A population of interest is the set of 'natural stimuli', the stimuli corresponding to observations an organism makes as it goes through its life performing intelligent behaviors. What is the distribution of natural stimuli? Here I define an idealized scenario that gives an $R|s$ of interest as a basis of comparison. Imagine we are handed an artificial spiking neural network playing a video game in which a high score reflects intelligent behavior, the network on average achieves a high score. Now we want to determine what observations the ANN makes about the game, i.e. how it

depends on the input space. Though the ANN is a deterministic process we induce randomness by randomly sampling from different initialization of the game. Indeed we would expect intelligence to generalize across many different scenarios. We then choose a counting window of some duration d and a stimulus window of some duration w. We grab all identical stimuli in the stimulus window from within and across game initialization which can be considered stimulus 'repeats'. Thus each stimulus is identical within the stimulus window but the time preceding is uncontrolled. Randomly drawing from the repeats of each unique stimulus would be the distribution of natural stimuli in this case. Averaging across the repeats is analogous to that of the peri-stimulus time histogram (PSTH) triggered on identical stimuli occurring during behavior.

Many caveats result from the potential differences between the ideal case described above and an actual experiment. Chief-most is that responses to stimuli may be qualitatively different outside of the context of a naturalistic task. Whether responses in experimental conditions generalize to behaviourally relevant conditions is a question that can and should be asked of any experiment.

Setting this issues aside, if we perform 'artiphysiology' and record spiking responses from all units in this network then randomly sampling the responses in the counting window across repeats associated with a stimulus gives us the desired $R|s$. The $\mathrm{E}[R|s]$ gives us the tuning curve and $\mathrm{Var}[R|s]$ quantifies the remaining variation around the tuning curve resulting from history dependence longer than the stimulus window.

In some cases, there may be no variability and the neuron has a purely sensory response. The input space for example would be, by construction, purely sensory. In the case where there is some variability, $\mathrm{E}[R|s]$ is the purely sensory component of the response and the residual variability, the distribution about the mean, is 'sensory' to the degree that it depends on observations of the chosen time scale. Variability can hold information about stimulus parameters if it is a function of stimulus parameters [7, 8]. The key point here is that the tuning curve is defined by the spike counting window and the stimulus window, and it gives a systematic relationship between evoked responses and stimuli therein. For some intuition in to the relationship between the sensory component of a response and how it depends on the stimulus window consider a unit that is simply a linear filter on a 1-dimensional time varying input. If the linear filter is shorter then the stimulus window then the unit will be purely sensory, a deterministic function of the inputs but if it is longer then there can be some variability because of the unconstrained influence of stimuli outside the stimuli window.

A quantification of a floor on how 'sensory' a neuron is provided by the signal to noise ratio defined as:

$$\frac{\mathrm{Var}[\mathrm{E}[R|s]]}{\mathrm{E}[\mathrm{Var}[R|s]]}$$

which measures on average across stimuli what ratio of response variance is the result of the tuning curve vs trial-to-trial variability. If the ratio is greater than 1 it implies the neuron is predominantly sensory as changes in spike count are mostly a function of the stimulus at a short time scale.

Thus if we were to draw a subset of the stimuli from the population and found our model accounted for some fraction of variance of the tuning curve, which in turn accounted for some fraction of total variation in spiking, we could achieve an arbitrary level of confidence this was the case for all stimuli and that our model had captured some estimable fraction of how the organism represented its environment

on a short time scale. Thus achieving the goal of making quantifiable progress on the problem of intelligence.

Yet even if an image computable model was found which fit the mean responses of neurons to naturalistic stimuli, a real insight into sensory representation does not immediately follow. It is still a complex high dimensional tuning curve. It simply moved from the nervous system into a computer. There must be some means by which to organize and interpret the tuning curve. In tackling this problem I consider it from two complementary perspectives: invariance (absence of variation in the tuning curve with respect to stimuli) and selectivity (variation in the tuning curve with respect to stimuli). Below I discuss invariance and its relevance to understanding intelligent sensory representations and selectivity as its natural complement.

1.3 Invariance and selectivity

Invariance, the maintenance of a tuning curve with respect to irrelevant transformations of a stimulus, is thought to be important for a robust sensory representation and thus has been explicitly engineered into models of visual cortex [9–13] and more general theoretical work has focused on it [14]. A classical example is translation invariance, where shifting a visual stimulus does not change the tuning curve of a neuron with respect to the unshifted version. Invariance can be thought of as the null-space of the tuning curve: the dimensions implicit or explicit that the tuning curve does not vary with respect to.

Invariance being a very general concept overlaps with other topics under study in the neurosciences. For example contrast adaptation in the context of V1 [15] can be thought of as a means to becoming invariant to contrast and more generally normalization as a means to become invariant to overall variance in neural inputs. Hopfield networks generate an invariant representation by representing all inputs within a region of stimulus space defined by a basin of attraction as a canonical representation. Metamers are defined as stimuli that are different but produce the same percept and in color science helped specify the spectral selectivity of the three types of photoreceptors in the human retina.

The concept of invariance is not very useful until an irrelevant dimension of tuning is specified. One powerful and common conceptual tool for thinking about invariance is the 'object'. Examples would include a tree, or the sun, or my dog Teddy. The identity and properties of these objects are invariant to our position relative to them. A tree is still made of wood whether I look at it upside down or right side up. Thus to have an object-based encoding would require invariance to transformations of an image generated by changes in the position of the observer relative to those objects including scaling, translation, and rotation. Beyond these geometric transformations, there are many other 'identity preserving transformations' such as lighting conditions and physical deformation.

The complement to invariance is selectivity: variation in a tuning curve with respect to stimuli. Selectivity is necessary to discriminate between stimuli for example on the basis the properties of an object such as its category (e.g., tree versus dog). Whether it be single neurons that have tuning curves varying solely with respect to a category (e.g. the grandmother cell) or a population in which no neurons solely vary with respect to a category but a combination of them do, some variation with respect to a category is necessary to allow category selectivity. At the same time the very definition of a category implies no variation with respect to a host of transformations, some of which were discussed above. Thus the tuning curve can be

decomposed into its invariance and selectivity both of which inform how an effective sensory representation of objects is formed. Below I discuss the key properties of the tuning curve that I have expressed above, in the context of their estimation in the presence of trial-to-trial variability.

1.4 Parameterization of invariance and selectivity

I have described two properties that are fundamental to characterizing the tuning curve: invariance and selectivity. I now explicitly describe the ways in which I parameterize these properties. A unifying feature of these parameterizations is they are all a function of the covariance or variance of tuning curves across stimuli. I will discuss invariance primarily with respect to a single transformation where invariance is measured by the degree of change between responses to a reference set of stimuli and the transformed stimuli, that same set but transformed in some consistent manner (e.g. rotated).

In this work I quantify invariance in two ways. The first is within distribution invariance where reference and transformed stimuli come from the same distribution. In the context of an experiment this would be the case where the reference and transformed stimuli are presented randomly interleaved. The second is out of distribution invariance where the reference and transformed stimuli come from separate distributions. In the context of an experiment this would be the case where the reference and transformed stimuli are presented in different blocks and invariance is measured across the blocks. I discuss the interpretation of these two measurements with respect to gain control.

The most common measure of invariance I use is the correlation of the tuning curve in response to a set of stimuli and to that same set under some transformation (see Chapter 2, 3). This measure is not sensitive to differences in scale or location between tuning curves. There are several justifications of correlation as a measure of invariance despite allowing these changes in the tuning curve. If we assume there will be normalization of responses to a given stimulus distribution then a high correlation implies invariance. For example in the case of contrast gain control in V1, orientation tuning is the same regardless of contrast if enough time is given for responses to adapt [15]. Intuitively we can think of this parameterization of invariance as how invariant the representation will be when the distribution of stimuli changes between two states and gain control mechanisms can account for resulting shifts in mean and amplitude.

This measure of invariance also is specific to encodings that are 'separable'. For example, two motion tuned neurons may prefer the same motion direction but different motion speeds, thus they give the same motion tuning profile but a different mean rate and scale across speeds. Factorizing parameters of interest has a rich history of research in machine learning arguing parameters that factorize provide an efficient representation [16].

Alternatively, I use a measure that is sensitive to changes in mean and amplitude (see Ch. 6). This would be of relevance to tasks where the transformation of interest could be happening rapidly without a chance for gain control mechanisms to operate. This would be the case in an experiment where the reference and transformed stimuli were randomly interleaved. To quantify this more stringent form of invariance, I correlate the responses to the reference and transformed stimuli with the same stimuli except reindexed such that covariance is calculated across the transform (for an example see Figure 6.34). In addition to being more sensitive, this metric

is analytically simpler because variance does not change across the two correlated quantities. It explicitly models the reference and transformed stimuli as being drawn from the same distribution.

Finally, I measure invariance across multiple transformations by a parameter I term 'normalized total covariance'. I employ this measure in Chapter 2 to estimate translation invariance across multiple positions in V4.

I quantity selectivity in several ways. To give the 'scale' of selectivity in the case of noisy neural data, I measure the signal to noise ratio (SNR): the relative contribution of variation across the tuning curve vs. trial-to-trial variability. As discussed above this can be thought of as a measure of a floor on the fraction of variation in spiking that is 'sensory' and as I will discuss in Chapter 3, it is a practical measure of tuning reliability.

Selectivity across tuning curves is often redundant and thus constrains population tuning. I measure the correlation between neural tuning curves to quantify this redundancy; this parameter is also known as signal correlation [17]. Signal correlation has figured prominently in sensory neuroscience's move to understand population representations because signal correlation constrains decodability of parameters encoded in the responses of sensory neurons [18–21] but see [8]. Furthermore, signal correlation will be shown to have a link to invariance in Chapter 6.

While the prior two parameters of selectivity can provide a sense of scale and the relationships between sensory neurons, they do not strongly constrain selectivity. Magnitude and correlation of stimuli response vectors are invariant to rotation. A predictive model specifies the form of selectivity. To determine how well a model reproduces the selectivity of a neuron, I estimate the correlation between the tuning curve of the neuron and that of the fixed model.

1.5 Estimation of tuning curve parameters

To estimate any of the above parameters from data can be challenging because of trial-to-trial variability. In essence, the problem comes down to the fact that no matter how many trials are collected the estimated tuning curve found by averaging across trials is still a random variable that is a function of trial-to-trial variability. Thus statistics of the estimated tuning curve will also be a function of the nuisance parameter of trial-to-trial variability. The form of this dependence is a function of the chosen estimator, but I find that moment-based estimators of the above tuning curve properties, such as the sample covariance, will frequently be systematically biased by trial-to-trial variability. Chapter 2, 3, and 4 are all focused on reducing this bias in the estimation of one or more of the parameters described above.

Accounting for 'measurement error' in the estimation of statistical parameters overlaps method developments in classical statistics with a history of at least 100 years [22]. For example, Spearman (1904) [23] considered the very problem of estimating the underlying correlation between two noisy quantities (the same problem as estimating the signal correlation between neurons) and his solution has been adapted by the fields of quantitative ecology, genomics, and psychology for their purposes [24–27]. I improve on this and other estimators based on calculations of bias for small samples as opposed to asymptotic limits.

I will show that accounting for these biases both reveal interesting relationships between these fundamental quantities in neural data and avoids confounds while giving a detailed insight into their source.

I now discuss the class of models I focus on in this thesis.

1.6 Deep neural networks

A popular class of models is the deep neural network. It is a straightforward extension of the linear-nonlinear model which has been used to some success in early visual regions [28]. It takes a linear combination of its inputs and then applies a non-linearity, such as rectification, to the scalar output::

$$g(\vec{s} \cdot \vec{w})$$

where g is the non-linear function, $\vec{w}$ is a vector of weights, and $\vec{s}$ the vector of inputs. A feed-forward neural network is simply a cascade of linear-nonlinear models where the inputs to a unit in the second layer are the outputs in the first layer. While initially these models were fit directly to neural data later researchers would find neural-like selectivity emerging for networks trained on tasks thought to be a function of a brain region [29–34]. The favored architectures in this recent spate of work are called deep convolutional neural networks (DCNNs), essentially a linear non-linear cascade except linear filters are convolved across the input from previous layers. Historically, this work is a subset of earlier research into how neural response properties could be predicted on the basis of hypothesized optimization goals of populations of neurons, beginning with mutual information between neural responses and stimuli [35].

In Chapter 2, I discuss how this class of models can be seen as complementary to the more established approach of parametric models and stimuli.

I now discuss the regions of cortex studied in this thesis and how they lend themselves to the approach and models I have laid out.

1.7 Intermediate visual sensory regions

In this work, I study for the most part an intermediate regions of the visual processing stream, area V4. A fundamental roadblock to understanding V4 is the heterogeneity of its tuning curves. In nearly every electrophysiological study, some neurons are strongly modulated by the chosen stimulus set and others are not. For example in V4 tuning for blur, color, equiluminance, shape, spatial frequency, and texture, to name a few, have all been found to show great diversity of tuning strength across neurons [36–40]. While this heterogeneity is perhaps not surprising, given the richness with which the nervous system represents the visual world, it is theoretically challenging because the number of potential cell types grows exponentially with each new property, and the experimentalist's task becomes one of unending enumeration.

The approach of employing image computable models to understand V4 provides an opportunity to, temporarily, sidestep these issues. Instead of a purely electrophysiological approach to characterize these tuning curves, image computable models fit to the responses to naturalistic stimuli both focus questions of tuning on ethologically relevant stimuli and give guarantees about wholly capturing the tuning curve without having to exhaustively sample it. Surprisingly we even find that single units in DCNNs trained to perform object recognition on naturalistic images but fit to V4 responses on the basis of simple parametric stimuli appear to show a generalization of V4 response properties to natural images. The high dimensional tuning of V4 requires many stimuli at the expense of repeats; I discuss in Chapter 3 how this is the circumstance under which the corrected estimators I develop

are most helpful. Thus heterogeneous intermediate sensory regions are particularly amenable to the modeling and statistical approaches I have described above.

In terms of my focus on the class of DCNN models, there are obvious analogies to be made between their structure and that of these heterogeneous visual areas. At its core a DCNN assumes, through convolution, relevant features are translation invariant i.e. may appear at any position in the visual field, and that features increase in complexity with relative scale. At small scales features are simple, whereas more complex features composed of combinations of these simple features arise in later layers. Similarly, in the visual cortex across the retinotopic map similarly tuned neurons will be found, and neurons' receptive fields for a given eccentricity tend to grow along with the complexity of their selectivity.

On the other hand, there are many ways in which the DCNNs I study here are not phsyiologically realistic. To name a few: they do not spike, there is no feedback, units can switch between being inhibitory and excitatory, and no cortical regions or cell types are specifically modeled. Thus a conservative position would be that I should solely consider these models on their functional merits: do they fit tuning curves well and are they amenable to interpretation. A less conservative position would be that, all things considered, these models are a relatively simple model of sensory cortex and thus their use should be exhausted until an additional mechanism is shown to be necessary to explain a specific feature of a tuning curve.

1.8 Summary of chapters

In Chapter 2, I study the similarity between V4 and a DCNN on the basis of the representation of object-centered curvature. I follow a seminal study by Pasupathy and Connor (2001) [41] that showed that V4 neurons have tuning consistent with an object-centered encoding of boundary curvature. I show that by the criteria of these experiments a DCNN trained only to recognize images shows the same form of selectivity. I provide example units which qualitatively display the spirit of this selectivity across units. We suggest shape selectivity is jointly encoded with other surface properties, and hypothesize this may be the case in V4. I discuss how the non-image computable parametric models, with the example of the angular-position and curvature (APC) model developed by Pasupathy, can be complementary to the image computable DCNN model.

In Chapter 3, motivated by the problem of attenuation of correlation between a noiseless model and neural responses by trial-to-trial variability, I develop a novel estimator $\hat{r}^2_{\text{ER}}$ which I find in simulation outperforms previous estimators, some of which grossly overestimate the fit of models. In addition, I provide validated confidence intervals for this estimator. I also develop an estimator of SNR as a means to judge the quality of data for the purpose of estimating correlation to a model.

I extend the estimator $\hat{r}^2_{\text{ER}}$ in Chapter 4 to the correlation between two neural tuning curves. I compare this estimator to the popular estimator developed by Spearman (1904) [23] and find $\hat{r}^2_{\text{ER}}$ has less bias. I demonstrate how it avoids confounds in the estimation of translation invariance and fill-outline invariance in V4.

In Chapter 5, I study the estimation of signal correlation and the distinct confounds that arise from simultaneous recordings where trial-to-trial responses are often correlated. Using estimators developed in previous chapters, I demonstrate a novel relationship between SNR and signal correlation in area MT. I explain this as the result of 'tuning curve noise': imperfections in the tuning curve which attenuate

model fit and signal correlation, even accounting for trial-to-trial variability, and in addition to systematic differences in tuning.

In Chapter 6, I develop a deep neural network response variance attribution method on the basis of covariance between inputs. I demonstrate how it allows insight into the selectivity of a DCNN model of V4 responses to natural images and the invariance of a DCNN to changes in the luminance of shape surfaces. I discuss how this provides a path to characterizing tuning curve fits by DCNNs.

Finally, in Chapter 7 I conclude with a discussion of my major findings and indicate promising avenues for future work.

1.9 Figures

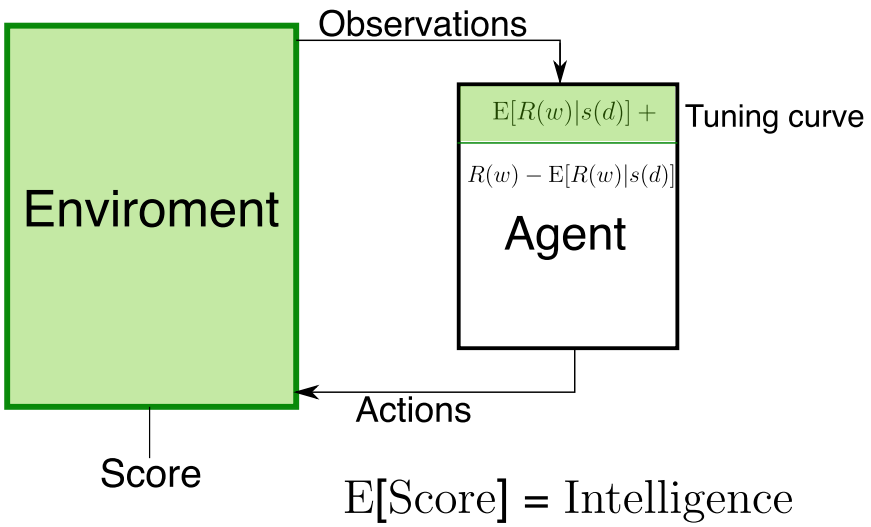


FIGURE 1.1: A conceptual decomposition of intelligence. The 'environment' is some process that generates observations on the basis of the agent's actions and its own internal state and rules. Observations are a minimal sufficient statistic of the state of the environment for the internal state of the deterministic agent. For example in a video game the value of the pixels on the screen. The agent takes observations and transforms them into actions. Some means of scoring the state of the environment, perhaps resources made available to the agent, is taken as a trial-to-trial measure of intelligence. Taking an average across many initialization of the environment gives the expected intelligence of the agent. Thus this scheme to evaluate an agent's intelligence requires the environment, score, available actions and observations be defined. If the agent is a spiking neural network then the tuning curve is defined as the expected number of spikes counted in a window of duration w for a stimulus pattern, $s(d)$, where d is duration. This component of the response is a systematic function of the environment within the counting window, thus it is highlighted green indicating it as an extension of the environment into the internal state of the agent. The deviation of the spike count resulting from a stimulus history longer than d is captured by $R(w) - \mathrm{E}[R(w)|s(d)]$. This thesis is focused on constraining and gaining insights into the tuning curve.

Chapter 2

'Artiphysiology': V4 like shape selectivity in a DNN

2.1 Summary

Deep networks provide a potentially rich interconnection between neuroscientific and artificial approaches to understanding visual intelligence, but the relationship between artificial and neural representations of complex visual form has not been elucidated at the level of single-unit selectivity. Taking the approach of an electrophysiologist to characterizing single CNN units, we found many units exhibit translation-invariant boundary curvature selectivity approaching that of exemplar neurons in the primate mid-level visual area V4. For some V4-like units, particularly in middle layers, the natural images that drove them best were qualitatively consistent with selectivity for object boundaries. Our results identify a novel image-computable model for V4 boundary curvature selectivity and suggest that such a representation may begin to emerge within an artificial network trained for image categorization, even though boundary information was not provided during training. This raises the possibility that single-unit selectivity in CNNs will become a guide for understanding sensory cortex.

This work was previously published in *eLife* [32].

2.2 Introduction

Deep convolutional neural networks (CNNs) are currently the highest performing image recognition computer algorithms. While their overall design reflects the hierarchical structure of the ventral ("form-processing") visual stream [42, 43], the visual selectivity (i.e., tuning) of single units within the network are not constrained to match neurobiology. Rather, single-unit properties are determined by a performance-based learning algorithm that operates iteratively across many pre-classified training images, tuning the parameters of the network to decrease the error between the network output and the target classification. Nevertheless, first-layer units in these CNNs, following training, often show selectivity for orientation and spatial frequency (Figure 2.1; see also Krizhevsky et al., 2012) like neurons in primary visual cortex (V1). Attempts to visualize features encoded by single units deeper in such networks [44, 45] show that selectivity becomes increasingly complex and categorical, similar to the progression along the ventral stream. Solidifying this idea, Güçlü and van Gerven [46] found a corresponding hierarchy of visual features between BOLD signals in the human ventral stream and layers within a CNN. This raises the tentative but exciting possibility that units deeper in the network may approximate tuning observed at mid-level stages of the ventral stream, e.g., area V4.

This is not unreasonable given that artificial networks that perform better at image classification also have population-level representations closer to those in area IT [29–31]. V4 is a primary input to IT [47] , yet there has been no systematic examination of whether specific form-selective properties found in V4 emerge within a CNN.

To address this, we tested whether two properties of shape selectivity in V4, tuning for boundary curvature [13, 41, 48] and translation invariance [41, 49–53] arise within a CNN. In particular, many V4 neurons are selective for boundary curvature, ranging from concave to sharply convex, at particular angular positions around the center of an object. This angular position and curvature (APC) tuning may be important for supporting entire object representations deeper in the ventral stream [54, 55], but it remains uncertain how it arises or is used. Finding APC-like tuning in the middle of an artificial network could help to relate mid-level visual physiology to pressures on visual representation applied by image statistics at the front end and by categorization performance downstream. It could also relate to the recent observation that human perception of shape similarity correlates with response similarity in CNNs [56].

We take an approach to characterizing single units in an artificial deep network that we refer to as "artiphysiology" because it closely mirrors how an electrophysiologist approaches the characterization of single neurons in the brain. In particular, we presented the original 362 shape stimuli used by Pasupathy and Connor ([41]) to AlexNet, a CNN that was the first of its class to make large gains on general object recognition [57] and that continues to be well-studied, [44, 46, 58–64], we found that many units in AlexNet would be indistinguishable from good examples of boundary-curvature-tuned V4 neurons. We applied a CNN visualization technique [44] to examine whether natural image features that best drive such APC-like units are consistent with the notion of selectivity for curvature of object boundaries. We identify specific V4-like units so that other researchers may utilize them for future studies.

2.3 Results

AlexNet contains over 1.5 million units organized in eight major layers (Figure 2.2), but its convolutional architecture means that the vast majority of those units are spatially offset copies of each other. For example, in the first convolutional layer, Conv1, there are only 96 distinct kernels (Figure 2.1), but they are repeated everywhere on a 55 x 55 grid (Figure 2.2E). Thus, for the convolutional layers, Conv1 to Conv5, it suffices to study the selectivity of only those units at the spatial center of each layer. These units, plus all units in the subsequent fully-connected layers comprise the 22,096 unique units (Figure 2.2D) that we analyzed.

2.3.1 Responses of CNN units to simple shapes

We first establish that the simple visual stimuli used in V4 electrophysiology experiments (Figure 2.3A) do in fact drive units within the CNN, which was trained on a substantially different set of inputs: natural photographic images from the ImageNet database (Deng et al., 2009). Across the convolutional layers and their sublayers, we found that our shape stimuli typically evoked a range of responses that was on average similar to, or larger than, the range driven by ImageNet images (e.g., Figure 2.4, Conv1, compare red to dark blue). The ranges for shapes and images

became more similar following normalization layers (e.g., Figure 2.4, Norm1). In contrast, in the subsequent fully-connected layers, the natural images drove a larger range of responses (Figure 2.4, FC6, dark blue) than did shapes (red line), and from FC6 onwards the range of responses to shapes was about 1/2 to 1/3 of that for images. The wider dynamic range for images in later layers may reflect the sensitivity of deeper units to category-relevant conjunctions of image statistics that are absent in our simple shape stimuli. These results were robust to changes in stimulus intensity and size (see Figure 2.4, legend); therefore, we settled on a standard size of 32 pixels so that stimuli fit within all RFs from Conv2 onwards (Figure 2.2B) with room to spare for translation invariance tests (see Methods).

Although our shapes drove responses in all CNN layers, many units responded sparsely to both the shapes and natural images. Across all layers, 13% of units had zero responses to all shape stimuli and 7% had non-zero response to only one stimulus, i.e., one shape at one rotation. Because we aim to identify CNN units with V4-like responses to shapes, we excluded from further analysis units with response sparseness outside the range observed in V4 (see Methods and Figure 2.4 supplement 1).

2.3.2 Tuning for boundary curvature at RF center

To assess whether CNN units have V4-like boundary curvature selectivity, we measured responses of each unique CNN unit to our shape stimuli (up to 8 rotations for each shape in Figure 2.3A), centered in the RF. We then fit responses with the angular position and curvature (APC) model ([41]), which captures neuronal selectivity as the product of a Gaussian tuning curve for curvature and a Gaussian tuning curve for angular position with respect to the center of the shape (Figure 2.3B, C and Methods). We found that the responses of many units in the CNN were fit well by the APC model. For example, the responses of Conv2 unit 113 (i.e., Conv2-113) were highly correlated ($r = 0.78$, $n = 362$) to those of its best-fit APC model (Figure 2.6A). The fit parameters indicate selectivity for a sharp convexity ($\mu_c = 1.0$, $\sigma_c = 0.39$) pointing to the upper left ($\mu_a = 135$, $\sigma_a = 23°$), and indeed the 8 most preferred shapes all include such a feature (Figure 2.6B, pink), whereas the least preferred shapes (cyan) do not. A second example unit, FC7-3591 (Figure 2.6C) with a high APC r-value (0.77) had fit parameters (see legend) reflecting selectivity for concavities roughly toward the top of the shape, consistent with most of its preferred shapes (Figure 2.6D). These results were similar to those for well-fit V4 neurons. For example, the V4 unit *a1301* (Figure 2.6E, F) had an APC fit ($r = 0.76$, $p < 0.001$, $n = 362$) reflecting a preference for a sharp convexity, like the first CNN example unit, except with a different preferred angular position ($\mu_a = 180°$).

For each layer of the CNN, we computed the distributions of the APC fit r-values across units (Figure 2.6G). There is a clear but modest trend for the cumulative distribution functions to shift rightward for higher layers (orange lines, Figure 2.6G), indicating that deeper layer units fit better on average to the APC model. The first CNN layer, Conv1 (black line) stands apart as having a far leftward-shifted r-value distribution, but this occurs simply because most of the stimuli overfill the small Conv1 RFs. Compared to V4 neurons studied with the same shape set (red line, Figure 2.6G), the median r-values (corresponding to 0.5 on the vertical axis) for layers Conv2 to FC8 were somewhat higher than that for V4, but the V4 and CNN curves matched closely at the upper range, with the best V4 unit having a higher APC r-value than any CNN unit.

One factor that could influence our CNN to V4 comparison is that CNN responses are noise-free, whereas V4 responses have substantial trial-to-trial variability. We extended the method of Haefner and Cumming [65] to remove the bias that variability introduces into the correlation coefficient (see Methods). The distribution of the corrected estimates of the r-values across the V4 population (pink line, Figure 2.6G) has a higher median than that for any of the CNN layers. This suggests that, had it been possible to record many more stimulus repeats to eliminate most of the noise in the V4 data, then one would find that the V4 population somewhat out-performs even the deep layers in AlexNet in fitting the APC model. Overall, regardless of whether we consider the raw or corrected V4 r-values, we would still conclude that the CNN contains units that cover the vast majority of the range of APC r-values found in V4 when tested with the same stimuli.

To determine whether the goodness of fit to the APC model was a result of the network architecture alone or if training on the object categorization task played a role, we fit the model to units in an untrained network in which weights were assigned random initial values (see Methods) and found that only ∼14% had APC r-values above 0.5 (Figure 2.6H, blue trace) and none reached the upper range of r-values observed in the trained CNN (Figure 2.6H, black line, aggregate of all layers) or in V4 (red line). This suggests that training is important for achieving an APC r-value distribution consistent with V4.

To control for over fitting, we re-fit the APC model to all CNN units after shuffling the responses of each unit across the 362 shapes. After shuffling, 99% of units had $r < 0.07$ (Figure 2.6H, green), whereas in the original data (Figure 2.6H, black) 99% of units had $r > 0.07$. Thus, the APC model largely reflects specific patterns of responses of the units to the shapes, and not an ability of the model to fit any random or noisy set of responses (see also [41]).

2.3.3 Translation Invariance

To have V4-like boundary curvature tuning, a CNN unit must not only fit the APC model well for stimuli centered in the RF, but must maintain that selectivity when stimuli are placed elsewhere in the RF, i.e., it must show translation invariance like that found in V4 for our stimulus set [41, 66]. For example, responses of a V4 neuron to 56 shapes centered in the RF are highly correlated ($r = 0.97, p < 0.0001$, n=56) with responses to the same shapes presented at a location offset by 1/6 of the RF diameter (Figure 2.7A), indicating that shapes that drive relatively high (or low) responses at one location also tend to do so at the other location. This can be visualized across the RF using the position-correlation function (Figure 2.7B, red), which plots response correlation as a function of distance from a reference position (e.g., RF center). For this V4 neuron, the RF profile, measured by the mean response across all stimuli at each position (Figure 2.7B, green; see Methods), falls off faster than the position-correlation function, consistent with a high degree of translation invariance.

A similar analysis for the example CNN unit, Conv2-113, reveals a steep drop-off in its position-correlation function (Figure 2.7E, red) compared to its RF profile (green). In particular, when stimuli were shown 13 pixels to the left of center (black arrow) the aggregate firing rate (see Methods) was 87% of maximum, but the correlation was near zero. The largely uncorrelated selectivity at two points within the RF indicates low translation invariance. Thus, despite its high APC r-value (Figure 2.6A), its low translation invariance diminishes it as a good model for V4 boundary contour tuning. This behavior was typical in layer Conv2, as demonstrated by

the position-correlation function averaged across all units in the layer (Figure 2.7F). Specifically, the correlation (red) falls off rapidly compared to the RF profile (green) even for small displacements of the stimulus set.

For deeper layers, RFs tend to widen and translation invariance increases. This is exemplified by unit 369 in the fourth convolutional layer (Figure 2.7G) and the Conv4 layer average (Figure 2.7H): on average the correlation (red) more closely follows the RF profile (green) and does not drop to zero near the middle of the RF. In the deepest layers, exemplified by the FC7 unit from Figure 2.6C, the RFs become very broad (Figure 2.7I, green) and there is very little fall-off in correlation (red) even for shifts larger than the stimulus size. This is true for the layer average as well (Figure 2.7J). These plots show that shape selectivity becomes more translation invariant relative to RF size, and not just in terms of absolute distance, as signals progress to deeper layers.

To quantify translation invariance for each unit with a single number, we defined a metric, TI, based on the normalized average covariance of the response matrix across positions (see Methods). The values of this metric, which would be one for perfect (and zero for no) correlation across positions, are shown for the example CNN units in Figure 2.7E, G and I. The trend for increasing TI with layer depth seen in Figure 2.7 (panels F, H and J) is borne out in the cumulative distributions of TI broken down by CNN layer (Figure 2.8A). For comparison, the cumulative distribution of our TI metric for 39 V4 neurons from the study of El-Shamayleh and Pasupathy (2016) is plotted (red). Only the deepest four layers (Conv5 to FC8) had median TI values that approximated or exceeded that of our V4 population. Conv1 is excluded because its RFs are far too small to fully contain our stimuli at multiple positions (see Methods). The substantial increase in TI for deeper layers is striking relative to the modest progression in APC r-values observed in Figure 2.6G.

An intuitive motivation for CNN architecture, chiefly convolution (repetition of linear filtering at translated positions) and max pooling, is the desire to achieve a translation invariant representation [9–13]. This might lead to the idea that responses of units within these nets are translation invariant by design, but the observation that strong translation invariance only arises in later layers begins to deflate this notion. Furthermore, we computed TI for the same units and stimuli but in the untrained network. We found that the degradation of TI in an untrained network (Figure 2.8B) was even more dramatic than the degradation of APC tuning (Figure 2.6H). Specifically, it was very rare for any FC-layer unit in the untrained network to exceed the median TI values for those layers in the trained network.

To assess the influence of neuronal noise on our comparison of TI between V4 and AlexNet, we estimated an upper bound on how much TI could have been reduced by V4 response variability (see Methods). TI tended to be less influenced by noise for neurons having higher TI, in particular the upward correction of the r-value was negatively correlated with the raw TI value ($r = -0.6$, $p < 0.001$, $n = 39$). Thus, for cells at the upper range of TI, we do not expect sampling variability to strongly influence our measurements. The distribution of V4 TI values corrected for noise is superimposed in Figure 2.8A and B (pink line). The modest rightward shift in the corrected distribution relative to the original raw distribution (red line) does not change our conclusion that only the deepest several layers in AlexNet have average TI values that match or exceed that of V4.

Our TI metric above was measured for horizontal stimulus shifts; however, we also measured TI for vertical shifts and verified that there was a high correlation between these two (r=0.79) (Figure 2.8 supplement 1), particularly for high TI values.

2.3.4 Identifying and visualizing preferences of candidate APC-like units

We now plot the joint distribution of our metrics for boundary contour tuning and translation invariance described above to identify candidate APC-like CNN units. Figure 2.10 shows a unit square with APC r-value on the vertical axis and translation invariance, TI, on the horizontal axis. An ideal unit would be represented by the upper right corner, (1,1). The hypothetical best V4 neurons lie within this space at the red X (TI= 0.97, $r = 0.80$). This best V4 point is a hybrid of the observed highest APC r-value from the Pasupathy and Connor (2001) [41] study, and the highest TI value from our re-analysis of the El-Shamayleh and Pasupathy (2016) [66] data. In comparison, the most promising CNN unit lies at the orange star (TI= 0.91, $r = 0.77$), very close to the hypothetical best V4 point. To demonstrate how the CNN population falls on this map, we plotted 100 randomly chosen units from an early layer, Conv2 (dark brown), and a deep layer, FC7 (orange). Although only a few FC7 units approach the hypothetical best V4 point, many units are better than the average V4 neuron (red lines, Figure 2.10). In contrast, most units from Conv2 are much further from ideal V4 behavior, but they span a large range, indicating that even in the second convolutional layer, some units have ended up, after training, having high TI and high APC r-values.

To determine whether units identified as being the most APC-like, i.e., those closest to (1,1) in Figure 2.10, respond to natural images in a manner qualitatively consistent with boundary curvature selectivity in an objected-centered coordinate frame, we identified image patches that were most facilitatory (drove the greatest positive responses) and most suppressive (drove the greatest negative responses) for the 50,000 image test-set from the 2012 ImageNet competition. We then used a visualization technique [44] to project back ("deconvolve") from the unit onto each input image through the connections that most strongly contributed to the response, thereby revealing the regions and features supporting the response. We examined the ten most APC-like units in each of seven layers from Conv2 to FC8. Below we describe major qualitative observations as a function of layer depth.

Visualizing the ten most APC-like units in Conv2 revealed selectivity for orientation, conjunctions thereof, or other textures. For example, unit Conv2-113 (from Figure 2.6A and 8E), was best driven by lines at a particular orientation (Figure 2.11A) and most suppressed by oriented texture running roughly orthogonal to the preferred. This explains why this unit responded well only to shapes that have long contours extending to a point at the upper left, and poorly to shapes having a broad convexity or concavity to the upper left (Figure 2.6B). Another Conv2 example (Figure 2.11B) was driven best by the conjunction of a vertical that bends to the upper left and a horizontal near the top of the RF that meet at a point in the upper left. Examining the input images reveals that textures and lines (e.g., the bedspread and rocking chair cushion) are as good at driving the unit as are boundaries of objects. A third unit (Figure 2.11C) preferred conjunctions of orientations and was suppressed by lines running orthogonal to the preferred vertical orientation. The preferred pattern was usually not an object boundary, but could surround negative space or be surface texture. These observations, taken together with the poor translation invariance of Conv2 relative to deeper layers, suggest that units at this early stage are not coding boundary conformation in an object-centered way, but that any pattern matching the preferred features of the unit, regardless of its position with respect to an object, will drive these units well.

From Conv3 to Conv5, the visualizations of the most APC-like units were more often consistent with an encoding of portions of object boundaries. Unit Conv3-156

Layer	Unit	APC r	μ_c	σ_c	μ_a	σ_a	TI
Conv2	108	0.67	0.7	0.72	134	34	0.76
Conv2	113	0.76	0.9	0.39	134	22	0.70
Conv2	126	0.67	0.1	0.72	337	51	0.81
Conv3	20	0.68	0.5	0.01	224	171	0.90
Conv3	156	0.67	0.5	0.01	337	171	0.79
Conv3	334	0.73	0.2	0.12	157	171	0.74
Conv4	203	0.71	0.2	0.16	292	171	0.77
Conv5	144	0.65	0.9	0.29	89	30	0.89
Conv5	161	0.72	0.2	0.16	112	87	0.85
FC6	3030	0.73	-0.1	0.16	89	26	0.89
FC7	3192	0.75	0.2	0.16	112	171	0.91
FC7	3591	0.78	-0.1	0.16	112	44	0.89
FC7	3639	0.76	-0.1	0.16	112	114	0.92
FC8	271	0.73	-0.1	0.16	112	114	0.91
FC8	433	0.70	0.3	0.21	112	130	0.91
FC8	722	0.72	0.2	0.08	112	130	0.93

TABLE 2.1: Fit parameters and TI metric for example CNN units. Unit numbers are given starting at zero in each sublayer. The APC model parameters, μ_c, σ_c, μ_a and σ_a, correspond to those in Equation 2. The TI metric is given by Equation 3. For visualization of preferred stimuli for example units, see Figures 9-12.

was driven best by the broad downward border of light objects (Figure 2.12A), particularly dog paws. The most suppressive features for this "downward-dog-paw" unit were dark regions, often negative space, with relatively straight edges. The deconvolved features tended to emphasize the lower portion of the object border. A similar example, Conv3-020, had a preference for the upper border of bright forms (e.g., flames; Figure 2.12B) and was suppressed by the upper border of dark forms (often dark hair on heads). This unit was representative of a tendency for selectivity for bright regions with broad convexities (e.g., Conv4-171, not shown). We assume that more dark-preferring units would have been found had our stimuli been presented as black-on-white. These trends continued with greater category specificity in Conv5. For example, Conv5-161 was driven best by the rounded, convex tops of white dog heads (Figure 2.12C), including some contribution from the eyes, and was most suppressed by human faces below the eyebrows. Unit Conv5-144 was best driven by the upward facing points of the tops of objects, particularly wolf ears and steeples (Figure 2.12D). This "wolf-ear-steeple" unit was most suppressed by rounded forms, and may be important for distinguishing between the many dog categories with and without pointed ears. In addition to units like these, which appeared to be selective for portions of boundaries, there were several units that appeared to detect entire circles (Figure 2.13), and thus fit well to an APC model with specificity for curvature but broadly accepting of any angular position.

In the FC layers, the most excitatory images were revealing about unit preferences, but the deconvolved features provided less insight because power in the back projection was typically widely distributed across the input image. For example, unit FC6-3030 (Figure 2.13A) responded best to hourglasses, but deconvolution did not highlight a particular critical feature. The shape stimuli driving the highest and

lowest five responses (Figure 2.14A, bottom row) suggest that a cusp (convexity) facing upward is a critical feature, consistent with the APC model fit (Table 1). The most suppressive natural images (not shown) were more diverse than those for the Conv layers, and thus provided little direct insight. Broadly, many of the top ten APC-like units in the FC layers fell into two categories: those preferring images with rounded borders facing approximately upwards (we refer to these as the "balls" group) and those associated with a concavity between sharp convexities, also facing approximately upwards (the "wolf-ears" group). For example, FC7-3192 (Figure 2.14B) responded best to images of round objects (e.g., golf balls) and to shapes having rounded tops. FC7-3591 (Figure 2.14C), which was the most APC-like unit by our joint TI-APC index (orange star in Figure 2.10), responded best to starfish and rabbit-like ears pointing up. Shapes with a convexity at 112° drove the unit most strongly, whereas shapes with rounded tops and overall vertical orientation yielded the most negative responses. FC7-3639 (Figure 2.14D) is an example of a wolf-ears unit, and its preferred shapes include those with a convexity pointing upwards flanked by one or two sharp points. In FC8, where there is a one-to-one mapping from units onto the 1000 trained categories, the top ten APC units were evenly split between the wolf-ears group (categories: kit fox, gray fox, impala, red wolf and red fox) and the balls group (categories: ping-pong balls, golf balls, bathing caps, car mirrors and rugby balls). For example, unit FC8-271 (Figure 2.14E) corresponds to the red wolf category and units FC8-433 and FC8-722 correspond to the bathing cap and ping-pong ball categories, respectively.

What is most striking about the deep-layer (FC) units is that, in spite of their tendency to be more categorical, i.e., to respond to a wolf in many poses or a ping-pong ball in many contexts, they still showed systematic selectivity to our simple shapes. We hypothesized that these FC units were driven by a range of image properties that correlated well with the target category, and that shape was simply one among others such as texture and color. We examined how much better the units were driven by the best natural images compared to our best standard shapes. Figure 2.15 shows for the top-10 APC-like units in each layer, that the best image drove responses on average about 2 times higher than did the best shape for Conv2-4, about 4-5 times higher for FC6-7 and more than 8 times higher for FC8. This is consistent with the hypothesis that shape tuned mechanisms contribute to the selectivity of these units, but are not sufficient in the absence of other image properties to drive the FC layers strongly. Nevertheless, the selectivity for simple shapes at the final layer appears to be qualitatively consistent with the category label. Notably, only two APC-like units responded better to a shape than to any natural image, but both were Conv4 units selective for bright circular regions (not shown), and the best stimulus was our large circle (Figure 2.3A, second from upper left).

2.3.5 CNN fit to V4 responses

Above, we examined the ability of CNN units to approximate the boundary curvature selectivity of V4 neurons as described by the APC model, but while an APC model provides a good description of the responses of many V4 neurons, there are also neurons for which it explains little response variance across our shape set. We therefore examined whether the CNN units might directly provide a better fit (than the APC model) to the responses of the V4 neurons. We used cross-validation (see Methods) to put these very different models on equal footing. Figure 2.16 shows the cross-validated, best fit r-values for the APC model plotted against those for the CNN units. Neither model is clearly better on average: just over half (56/109) of

neurons were better fit by the APC model, while just under half (53/109) were better fit by a CNN unit. Only 21 of 109 neurons had significant deviations from the line of equality (Figure 2.16, red) and these were evenly split: 11 better fit by the APC model and 10 by the CNN. The similar performance of the APC model and CNN could be a result of the CNN and APC model explaining the same component of variance in the data, or explaining largely separate components of the variance. To assess this, for each V4 neuron, we removed from its response the component of variance explained by its best-fit APC model. For this APC-orthogonalized V4 response, the CNN model had a median correlation to V4 of $r = 0.29$ (SD=0.11), much lower than the APC model's $r = 0.47$ (SD=0.12) median . For 94/109 neurons, the APC model explained more variance than the variance uniquely explained by the CNN. Overall, we conclude that the APC model and the CNN explain similar features of V4 responses for most neurons.

2.4 Discussion

We examined whether the CNN known as AlexNet, designed to perform well on image classification, contains units that appear to have boundary curvature selectivity like that of V4 neurons in the macaque brain. Although our simple shape stimuli were never presented to the network during training, we found that many units in the CNN were V4-like in terms of quantitative criteria for translation invariance and goodness of fit to a boundary curvature model. While units throughout AlexNet had good fits to the APC model, relatively poor translation invariance in the early layers meant that only the middle to deeper layers had substantial numbers of units that came close to matching exemplary APC-tuned V4 neurons. Based on our quantitative criteria and on the qualitative visualization of preferred features identified in natural images, we believe that APC-like units within middle layers of trained CNNs currently provide the best image-computable models for V4 boundary curvature selectivity.

Finding such matches at the single unit level is striking because the deep net and our macaques differ dramatically in their inputs, training and architecture. The animals never saw ImageNet images and probably never saw even a single instance of the overwhelming majority of the 1000 output categories of AlexNet. They did not see the forest, ocean, sky nor other important contexts for AlexNet categories, nor had AlexNet been trained on the artificial shapes used to characterize V4. While the macaque visual system may be shaped by real-time physical contact with a 3D dynamic world, AlexNet cannot and was not even given information about the locations nor boundaries of the targets to be classified within its images during categorization training. AlexNet lacks a retina with a fovea, an LGN, feedback from higher areas, dedicated excitatory and inhibitory neurons, etc., and it does not have to compute with action potentials. Our results suggest that image statistics related to object boundaries may generalize across a wide variety of inputs and may support a broad variety of tasks, thereby explaining the emergence of similar selectivity in such disparate systems.

2.4.1 Visualization of V4-like CNN units.

By applying a CNN visualization technique to APC-like units identified by our quantitative criteria, we found that some of these CNN units appeared, qualitatively, to respond to shape boundaries in natural images whereas many others did

not. In early layers, particularly Conv2, the strongest responses were not driven specifically by object boundaries but instead by other image features including texture, accidental contours and negative space. In contrast, candidate APC units in intermediate layers often responded specifically to natural images patches containing object boundaries, suggesting that these units are APC-like. In the deeper (FC) layers, units were poorly driven by our shape stimuli relative to natural images, and the preferred natural images for a given unit appeared similar along many feature dimensions (e.g., texture, background context) beyond simply the curvature of object boundaries. We speculate that these units are jointly tuned to many features and that object boundaries alone account for only part of their tuning. More work is needed to understand the FC-layer units with high APC r-values; however, we believe units in the middle layers, Conv3-5, provide good working models for understanding how APC-tuning might arise from earlier representations, how it may depend on image statistics and how it could support downstream representation.

2.4.2 Training and translation invariance

Training dramatically increased the number of units with V4-like translation invariance, particularly in the FC layers (Figure 2.8A vs. B). Since the trained and untrained nets have the same architecture, the increase in TI is not simply a result of architectural features meant to facilitate translation invariance, e.g., max-pooling over identical, shifted filters. Thus, while CNN architecture is often associated with translation invariance [9–13, 59, 67], we find that high TI for actual single unit responses is only achieved in tandem with the correct weights. We are currently undertaking an in-depth study comparing the trained and untrained networks to elucidate statistical properties of weight patterns that support translation invariance. Our preliminary analyses show that spatial homogeneity of a unit's kernel weights across features correlates with its TI score, but this correlation is weaker in higher layers. Alternative models of translation-invariant tuning in V4 include the spectral receptive field (SRF) model [68] and HMax model [13]. The former made use of the Fourier spectral power, which is invariant to translation of the input image, but this phase insensitivity prevents the SRF model from explaining APC-like shape tuning [69]. The HMax model of Cadieu et al. (2007) is a shallower network with the equivalent of two convolutional layers and does not achieve the strong translation invariance found in deeper layers here [70]. Overall, translation invariance at the single-unit level is not a trivial result of gross CNN architecture, yet it is crucial for modeling V4 form selectivity.

2.4.3 Other studies of TI in CNNs

Although other studies have examined translation invariance and related properties (rotation and reflection invariance) in artificial networks [44, 46, 59, 67, 71–75], we are unaware of any study that has quantitatively documented a steady layer-to-layer increase of translation invariant form selectivity, measured for single units, across layers throughout a network like AlexNet. For example, using the invariance metric of Goodfellow et al. (2009) [67], Shang et al. (2016, their Fig. 4c)[73] averaged over multiple types of invariance (e.g., translation, rotation) and over all units within a layer and found a weak, non-monotonic increase in invariance across layers in a CNN similar to AlexNet. Using the same metric but different stimuli, Shen et al. (2016) [74] found no increase and no systematic trend in invariance across layers

of their implementation of AlexNet (their Fig. 5). Although Güçlü and van Gerven (2015) [46] plot an invariance metric against CNN layer, their metric is the half-width of a response profile, and thus it is unlike our TI selectivity metric. In spite of the importance of translation invariance in visual processing and deep learning [43], there currently is no standard practice for quantifying it. An important direction for future work will be to establish standard and robust methods for assessing translation invariance and other transformation invariances to facilitate comparisons across artificial networks and the brain.

2.4.4 Comparison to previous work

One way our approach to comparing the representation in a CNN to that in the brain differs from previous work is that we examined the representation of specific visual features at the single-unit level, whereas previous studies took a population level approach. For example, Yamins et al. (2014) [29] modeled IT and V4 recordings using weighted sums over populations of CNN units, and Khaligh-Razavi & Kriegeskorte (2014) [30] examined whether populations of CNN units represented categorical distinctions similar to those represented in IT (e.g., animate vs. inanimate). Also, Kubilius et al. (2016) [56] examined whether forms perceived as similar by humans had similar CNN population representations. Our work is the first to quantitatively compare the single-unit representation in a CNN to that in a mid-level visual cortical area. We tested whether an artificial network matched the neural representation at a fundamental level—the output of single neurons, which are conveyed onward to hundreds or thousands of targets in multiple cortical areas. Unlike previous studies, we focused on specific physiological properties (boundary curvature tuning and translation invariance) with a goal of finding better models where a robust image-computable model is lacking. Furthermore, we use visualization of unit responses to natural images to qualitatively validate whether the representation that these response properties are intended to capture (an object-centered representation of boundary) does in fact hold across natural images. We believe this level of model validation, which includes quantitative and conceptual registration to documented neuronal selectivity, pushes the field beyond what has been done before. Our results allow modelers to focus on specific neural selectivities and work with concrete, identified circuits that have biologically plausible components.

Another major difference with prior work is that we fit the CNN to the APC model as opposed to directly to neural responses. This might seem like an unnecessary layer of abstraction, but the purpose of a model is not just predictive power but also interpretability, and the CNN's complexity runs counter to interpretability. The CNN is necessarily complex in order to encode complex features from raw pixel values, whereas the APC model has five interpretable parameters. The APC model describes responses to complex features while ignoring the details of how those features were computed from an image. By identifying APC-tuned units in the CNN, we gain an image-computable model of neural responses to interpretable features; these units can be studied to understand how and why such response patterns arise. When we separately tested whether the CNN units were able to directly fit the responses of V4 neurons, we found they were no better on average than the APC model, thus for a gain in interpretability, we did not suffer an overall loss of predictive power. Nevertheless, some V4 neurons were better fit directly to a CNN unit than to any APC model, suggesting there may be V4 representations beyond APC tuning that can be synergistically studied with CNNs.

2.4.5 Value of artiphysiology

Comparing artificial networks to the brain can serve both computer and biological vision science ([31]). What can an electrophysiologist learn from this study? First, our results demonstrate that there may already exist image-computable models for complex selectivity that match single-neuron data better than hand-designed models from neuroscientists. Second, finding matches between neuronal selectivity in the brain and artificial networks trained on vast amounts of natural data provides one method for electrophysiologists to validate their findings. For example, our findings support the hypothesis that an encoding of boundary curvature in single units may be generally important for the representation of objects. Third, once a match is found based on limited sets of experimentally practical stimuli, units within deep nets can then be tested with vast and diverse stimuli to attempt to gain deeper understanding. For example, finding the downward-dog-paw and wolf-ear-steeple units raises the question of whether boundary curvature is encoded independent of other visual traits in V4 or in the CNN. Specifically, is it possible that V4 neurons that appear to encode curvature at a particular angular position are in fact also selective for texture or color features associated with a limited set of objects that have relevance to the monkey? Longer experimental sessions with richer stimulus sets will be required to test this in V4. Fourth, concrete, image-computable models can be used to address outstanding debates that may otherwise remain imprecise. For example, by visualizing the preferences of single units for natural stimuli after identifying and characterizing those units with artificial stimuli, our results speak to the debate on artificial vs. natural stimuli [76] by showing that artificial stimuli are often able to reveal critical characteristics of the selectivity of units involved in complex mid-level (parts-based) to high-level (categorical) visual encoding, even when the visual dimensions of the artificial set explore only a minority of the feature space represented by the units. As another example, our results can help to address the debate of whether the visual system explicitly represents object boundaries [77–79], which Movshon and Simoncelli describe as follows: "In brief, the concept is that the visual system is more concerned with the representation of the "stuff" that lies between the edges, and less concerned with the edges themselves (Adelson and Bergen 1991)." The models we have identified can now be used to pilot experimental tests of this rather complex, abstract idea.

Our approach also provides potentially valuable insight for machine learning. The connection between deep nets and actual neural circuits is often downplayed, but we found a close match at the level of specific single-unit selectivity. This opens the possibility that future studies could reveal more fine-scale similarities, i.e., matches of sub-types of single-unit selectivity, between artificial networks and the brain, and that such homology could become a basis for improving network performance. Second, translation invariance, seen as critical for robust visual representation, has not been systematically quantified for units within artificial networks. Determining why deeper layers in the network maintain a wide diversity of TI across units could be important for understanding how categorical representations are built. More generally, the art of characterizing units within complex systems using simple metrics and systematic stimulus sets, as practiced by electrophysiologists, can provide a useful way to interpret the representations learned in deep nets, thereby opening the black box to understand how learned representation contributes to performance.

2.4.6 Further work

Our findings are consistent with the hypothesis that some CNN units share a representation of shape in common with V4 that is captured by the APC model. Examining whether these CNN units demonstrate additional V4 properties, beyond those examined here, would further test this hypothesis. For example, curvature-tuned V4 cells have been shown to (1) have some degree of scale invariance [66], (2) suppress the representation of accidental contours, e.g., those resulting from occlusion that are unrelated to object shape [80], (3) be robust against partial occlusions of certain portions of shape [81], and (4) maintain selectivity across a spectrum of color [37]. Further studies like these are needed to more deeply probe whether the intermediate representation of shape and objects in the brain is similar to that in artificial networks. In addition to further study of functional response properties, it is important to understand how the network achieves these representations. For example, translation invariance was a key response property that allowed the trained network to achieve a V4-like representation, yet we are just beginning to understand what aspects of kernel weights, receptive field overlap, and convergence are critical to matching the physiological data. For CNNs to be valuable models of the nervous system, it will be important to understand what network properties support their ability to match representations observed in vivo.

2.5 Methods and Materials

2.5.1 The convolutional neural network

We used an implementation of the well-known CNN referred to as "AlexNet," which is available from the Caffe deep learning framework (http://caffe.berkeleyvision.org) [82]. Its architecture (Figure 2.2) is purely feed forward: the input to each layer consists solely of the output from the previous layer. The network can be broken down into 8 major layers (Figure 2.2A, left column), the first five of which are called convolutional layers (Conv1 through Conv5) because they contain linear spatial filters with local support that are repeatedly applied across the image. The last three layers are called fully connected (FC6 through FC8) because they receive input from all units in the previous layer. We next describe in detail the computations of the first major layer, which serves as a model to understand the later layers.

The first major convolutional layer consists of four sublayers (Figure 2.2A, orange, and Figure 2.2C-F, top 4 rows). The first sublayer, Conv1, consists of 96 distinct linear filters (shown in Figure 2.1) that are spatially localized to 11 x 11 pixel regions and that have a depth of three, corresponding to the red, green and blue (RGB) components of the input color images. The input images used for training and testing are 227 x 227 (spatial) x 3 (RGB) pixels. The output of a Conv1 unit is its linear filter output minus a bias value (a constant, not shown). Conv1 has a stride of 4, meaning that neighboring units have filters that are offset in space by 4 pixels. The output of each Conv1 unit is processed by a rectified linear unit in the second sublayer, Relu1, the output of which is simply the half-wave rectified value of Conv1. These values are then pooled by units in the third sublayer, Pool1, which compute the maximum over a 3 x 3 pixel region (Figure 2.2A, gray triangles) with a stride of 2. The outputs of the Pool1 units are then normalized (see below) to become the outputs of units in the fourth sublayer, Norm1. These normalized outputs are the inputs to the Conv2 units in the second major layer, and so on. Figure 2.2A shows a scale diagram of the spatial convergence in the convolutional layers (major layers are color coded)

along one spatial dimension. Starting at the top, the 11 x 11 pixel kernels (orange triangles) sample the image every 4 pixels, reducing the spatial representation to a 55 x 55 element grid (Figure 2.2A, column 4, lists spatial dimensions). The Pool1 layer reduces the representation to 27 x 27 because of its stride of 2. The Conv2 unit linear filters are 5 x 5 in space (red triangles) and are 48 deep (not depicted), where the depth refers to the number of unique kernels in the previous layer that provide inputs to the unit (see [57], for details and their Figure 2.2 for a depiction of the 3D kernel structure).

These operations continue to process and subsample the representation until, after Pool5, there is a 6 x 6 spatial grid that is 256 kernels deep. Given the convergence between layers, the maximum possible receptive field (RF) size (i.e., extent along either the horizontal or vertical dimension) for units in each convolutional layer ranges from 11 to 163 pixels (Figure 2.2B) for Conv1 to Conv5, respectively. For example, the pyramid of support is shown for the central Conv5 unit (Figure 2.2A, dark blue triangle shows tip of upside-down pyramid), which has access to the region of width 163 pixels covered by Conv1 kernels (orange triangles). The receptive field sizes of units in the FC layers are unrestricted (not shown in Figure 2.2B). The last major layer, FC8, has a Prob8 sublayer that represents the final output in terms of the probability that the visual input contains each of 1000 different categories of object (e.g., Dalmation, Lampshade, etc.; see [57], for details).

Units in the Norm1 and Norm2 sublayers carry out local response normalization by dividing their input value by a function (see [57], their section 3.3) of the sum of squared responses to 5 consecutive kernels (indices from +2 to -2) along the axis of unique kernel indices (e.g., in Conv1, the indices go from 0 to 95 for the filters shown in Figure 2.1, from the upper left towards the right and down), thereby creating inhibition among kernels. Figure 2.2D (bottom row) lists the total number of units with unique kernels in each layer, and this defines the number of units that we examine here. In the Conv layers, we only test the units that lie at the central spatial position because they perform the same computation as their spatially offset counterparts. We analyzed a total of 22,096 units. To identify units for reproducibility in future studies, we refer to units by their layer name (e.g., Conv1) and a unit number, where the unit number is the index, starting at zero, within each sublayer and proceeding in the order defined in Caffe.

We tested the network in two states: untrained and fully trained. The untrained network has all weights (i.e., values within the convolutional kernels and input weights for FC layers) initialized to Gaussian random values with mean 0 and SD 0.01, except for FC6 and FC7 where SD=0.005, and all bias values initialized to a constant of 0 (Conv1, Conv3, FC8) or 1 (Conv2, Conv4, Conv5, FC6, FC7). These initial bias values are relatively low to minimize the number of unresponsive units, which in turn guarantees a back propagation gradient for each unit during training. The fully trained network (available from Caffe) has been trained with stochastic gradient descent on large database of labeled images, ImageNet [83], with the target that the final sublayer, Prob8, has value 0 for all units except for a value of 1 for the unit corresponding to the category of the currently presented training image. To speed up training and mitigate overfitting, an elaborate training procedure was used that included a number of heuristics described in detail in Krizhevsky et al. (2012) [57].

2.5.2 Visual stimuli

Our stimulus set (Figure 2.3A) is that used by Pasupathy and Connor (2001) [41] to assess tuning for boundary curvature in V4 neurons. The set consists of 51 different simple closed shapes that are presented at up to 8 rotations (fewer rotations for shapes with rotational symmetry), giving a total of 362 unique stimulus images. We rendered the shapes within a 227 by 227 pixel field with RGB values set to the same amplitude, thus creating an achromatic stimulus. The background value was 0, and the foreground amplitude was varied up to 255, the maximum luminance. This format matched the size and amplitude of the JPEG images on which the CNN was originally trained. The center of each shape was taken to be the centroid of all points on the finely sampled shape boundary. We fixed the foreground amplitude to 255 after varying it to lower values and finding that it made little difference to the response levels through the network because of the normalization layers (see Results).

We set the size of our stimuli to be 32 pixels, meaning that the largest shape, the large circle (Figure 2.3A second shape from upper left), had a diameter of 32 pixels and all stimuli maintained the relative scaling shown in Figure 2.3A. This ensured the stimuli fit within the calculated RF of all layers except Conv1 with additional room for translations (see Maximum RF size, Figure 2.2B) and allowed all layers to be compared with respect to the same stimuli. We excluded Conv1 from our analysis because fitting the stimuli within the 11 by 11 pixel RFs would corrupt their boundary shape, would not allow room for testing translation invariance, and Conv1 is of less interest because of its simple function. In the V4 electrophysiological experiments of Pasupathy and Connor, stimuli were sized proportionally to each neuronal RF, as it can be difficult to drive a cell with stimuli that are much smaller than the RF. We tested sizes larger than 32 pixels (see Results) and found it did not substantially change our results.

2.5.3 Electrophysiological data

For comparison to the deep network model, we re-analyzed data from two previous single-unit, extracellular studies of parafoveal V4 neurons in the awake, fixating rhesus monkey (Macaca mulattta). Data from the first study, Pasupathy and Connor (2001) [41], consists of the responses of 109 V4 neurons to the set of 362 shapes described above. There were typically 3-5 repeats of each stimulus, and we used the mean firing rate averaged across repeats and during the 500 ms stimulus presentation to constrain a model of tuning for boundary curvature in V4 (Figure 2.3C). To constrain translation invariance, we used data from a second study, El-Shamayleh and Pasupathy (2016) [66], because the first study used only two stimuli (one preferred and one antipreferred) to coarsely assess translation invariance. The data from the second study consists of responses of 39 neurons tested for translation invariance. The stimuli were the same types of shapes as the first study, but where the position of the stimuli within the RF was also varied. Each neuron was tested with up to 56 shapes (some of which are rotations of others) presented at 3-5 positions within the receptive field. Each unique combination of stimulus and RF position was presented for 5-16 repeats, and spike counts were averaged over the 300 ms stimulus presentation. Experimental protocols for both studies are described in detail in the original publications.

2.5.4 Response sparsity

While many units in the CNN responded well to our shape set, there were also many units, particularly in the rectified (Relu) sublayers, that responded to very few or none of our shape stimuli. It was important to identify the very sparse responding units because they could bias our comparison between the CNN units and V4 neurons. We quantified response sparsity using the fourth moment, kurtosis (Field, 1994),

$$K = \frac{1}{n}\sum_{i}^{n}\frac{(x_i - \bar{x})^4}{\sigma^4}, \tag{2.1}$$

where x_i is the response to the i^{th} stimulus, n is the number of stimuli, and $\bar{x}$ and σ are the mean and SD of the response across stimuli. This metric works for both non-negative and signed random variables, thus covering the outputs of all layers of the CNN. We excluded CNN units where response sparsity was outside the range observed in V4: 2.9 to 42 (Figure 2.4, supplement 1; see Results). We also found that such units gave degenerate fits to the APC model.

2.5.5 Placing stimuli in the classical receptive field

In keeping with neurophysiology, we defined the classical receptive field (CRF) of a CNN unit as the region of the input from which our stimuli can elicit a response different from baseline, where baseline is defined as the response to the background input (all zeros). For example, to determine the horizontal extent of the CRF of a unit, we started with our stimulus set centered (in x and y) on the spatial location of the unit and determined whether there was a driven response (deviation from baseline) to any stimulus. We then moved the stimulus set left and right to cover a 100 pixel span in 2 pixel increments to find the longest set of contiguous points from which any response was elicited at each point. In other words, stimuli were centered on pixels ranging from 64 to 164 in the 227 pixel wide image. To account for the finite width of the stimuli, we subtracted the maximum stimulus width from the length of the contiguous response region and added one to arrive at the estimated extent of the CRF in pixels along the horizontal axis. Any unit with a CRF wide enough to contain three 2-pixel translations of our stimulus set was included in our analyses. Generally, this provided a conservative estimate of the receptive field, because most stimuli were narrower than the maximal-width stimulus, as observed in Figure 2.3A.

All analyses and plots of responses to translated shapes were made with respect to horizontal shifts of our vertically centered shape set. To verify that our conclusions did not depend on testing only horizontal shifts, we recalculated our metrics for vertical shifts and found them to be strongly correlated with those for horizontal shifts (Figure 2.8, supplement 1).

2.5.6 The APC model

Our study focuses on the ability of CNN units to display a particular physiological property of V4 neurons—tuning for boundary conformation—which has been modeled using the angular position and curvature (APC) model introduced by Pasupathy and Connor (2001) [41]. Conceptually, APC tuning refers to the ability of a neuron to respond selectively to simple shape stimuli that have a boundary curvature feature (a convexity or concavity) at a particular angular position with respect to the center of the shape. Unlike the CNN, the APC model does not operate on raw

image pixel values, but instead on the carefully parameterized curvature and angular position of diagnostic elements of the boundaries of simple closed shapes (see example shape, Figure 2.3B). Each boundary element along the border of a shape can be mapped to a point in a plane heretofore referred to as the APC plane (Figure 2.3C). The responses, R_i, of a unit to the i^{th} shape is given by:

$$R_i = k \max_j \left[\exp\left(\frac{-(c_{i,j} - \mu_c)^2}{2\sigma_c^2} \right) \exp\left(\frac{-(a_{i,j} - \mu_a)^2}{2\sigma_a^2} \right) \right], \tag{2.2}$$

where the expression inside the square brackets is the product of two Gaussian tuning curves, one for curvature with mean μ_c and SD σ_c, and one for angular position with mean μ_a and SD σ_a. The curvature axis extends from -1 (sharp concavity) to +1 (sharp convexity), and the angular position is defined with respect to the center of the shape. The j^{th} curvature value of the i^{th} shape is encoded as $c_{i,j}$ and the angular position of that curvature element is $a_{i,j}$. The factor k is a scaling constant. The max over these boundary elements is taken, thus the response depends only on the most preferred feature. In the original study [41], these parameters were fit using a gradient descent method, the Gauss-Newton algorithm, from a grid of starting points across the APC plane. We instead discretely sampled the parameter space taking the Cartesian product of 16 values of μ_c, σ_c, μ_a and σ_a, where the means were linearly spaced, the SDs were logarithmically spaced, and the end-points were set to match the range of values observed for the V4 cells when fit by the original Gauss-Newton method ($\mu_c \in [-0.5, 1]$, $\sigma_c \in [0.01, 0.98]$, $\mu_a \in [0°, 338°]$ and $\sigma_a \in [23°, 171°]$). We defined the best-fit model to be that which maximized Pearson's correlation coefficient between observed and predicted responses. We then found k using a least squares fit. We found this to be more rapid, and the median correlation of the original V4 neurons to be the same to two decimal places as the Gauss-Newton fits (0.48), and had the assurance that the same models were tested on all units. We used Pearson's correlation coefficient two-tailed p-value to test for significance.

2.5.7 Measuring translation invariance

To visualize translation invariance we created position-correlation functions by plotting the r-value of responses between a reference and an offset location as a function of distance (e.g., Figure 2.7B and E-J, red). To compare the fall-off in correlation to the fall-off in the RF profile (e.g., Figure 2.7E-J, green) of the CNN unit, we computed an aggregate firing rate metric—the square root of the sum of the squared responses across the stimulus set at each spatial position. For CNN units, this was used rather than the mean firing rate because responses could be positive or negative.

To quantify translation invariance in neuronal and CNN unit responses, we defined a metric, TI, that can be thought of as a generalization of the correlation coefficient. The correlation coefficient,

$$r = \frac{\text{Cov}(\vec{p}_1, \vec{p}_2)}{\text{SD}(\vec{p}_1)\ \text{SD}(\vec{p}_2)}, \tag{2.3}$$

which is bounded between -1 and 1, measures how similar the response pattern is across two locations, where $\vec{p}_1$ and $\vec{p}_2$ are vectors containing the responses to all stimuli at positions 1 and 2. Our TI metric is,

$$\text{TI} = \frac{\sum_{i \neq j} \text{Cov}(\vec{p}_i, \vec{p}_j)}{\sum_{i \neq j} \text{SD}(\vec{p}_i)\ \text{SD}(\vec{p}_j)}, \tag{2.4}$$

where the sums are taken over all unique pairs of locations, and $\vec{p}_i$ is the mean-subtracted column of responses at the i^{th} RF position. The numerator is the sum of the non-diagonal entries in the covariance matrix of the responses, and the denominator is the sum of the products of each corresponding pair of SDs. Thus, this metric is also bounded to lie between -1 and 1, but it has an advantage over the average r-value across all unique pairs of locations because the latter would weight the r-value from RF locations with very weak responses just the same as those with very strong responses. For a simple model of neuronal translation invariance in which the variations of responses are described as the product of a receptive field profile and a shape selectivity function, our TI metric would take its maximum possible value, 1. If responses at all positions were uncorrelated, it would be 0.

We also evaluated an alternative metric, the separability index [84, 85] based on the singular value decomposition of the response matrix, but we found that it was biased to report higher translation invariance values for response matrices that reflected tuning that was more confined in space (i.e., smaller RF sizes) or more limited to a small range of shapes (i.e., higher shape selectivity). According to our simulations, our TI metric has the benefit of being unbiased with respect to receptive field size or selectivity of our response matrices, thereby facilitating comparisons across layers and diverse response distributions.

In testing the CNN, we finely sampled horizontal shifts of the stimulus set, as described above in 'Placing stimuli in the CRF'. The TI metric for any neuron was computed only for the set of contiguous locations for which the entire shape set was within the RF of the unit.

2.5.8 Comparing CNN and APC model fits to V4 data

We examined whether the CNN units might directly provide a better fit to the V4 neural responses than does the APC model. This required us to compare, for each of the 109 V4 units, the best-fit unit in the pool of CNN units to the best fit provided by the APC model. In the case of the CNN, there are 22,096 units to consider (Figure 2.2D). In the case of the APC model, there are 5 parameters (see "The APC model" above). We employed cross-validation to ensure that any differences in fit quality were not the result of one fitting procedure being more flexible than the other. In particular, we performed 50 fits on a random subset of 4/5 of the neural data, then measured the correlation of the fit model on the remaining 1/5. We took the mean of these 50 fits for each unit to be the estimate of test correlation, and the 95^{th} percentiles of the distribution of fits for identifying cells that deviate in their fit quality between two models (e.g., APC model and the CNN). To judge whether the variance explained by the CNN was largely distinct from that explained by the APC model we fit a V4 neurons best-fit CNN model to the residual of the fit of the APC model to a V4 neuron. If the correlation of the CNN unit to the V4 neuron remains high then the APC model and CNN explain different features of the response of the V4 neuron.

2.5.9 Estimating the effect of the stochastic nature of neuronal responses

AlexNet produces deterministic, noise-free responses, whereas the responses of V4 neurons are stochastic. This raises the possibility that our conclusions might have been different if more trials of V4 data had been collected to reduce the noise in the estimates of the mean neuronal responses. In particular, trial-to-trial variability will tend to lower the correlation coefficient (r-value) between model and data.

To address this, we used the methods of Haefner and Cumming (2009) [65] to remove the downward bias that trial-to-trial variability imparts on the r-value for our fits of the APC model to neuronal data. The method of Haefner and Cumming assumes that the neural responses have been appropriately transformed to have equal variance across stimuli and that the averaged responses for each stimulus are normally distributed. For the case where the variance-to-mean relationship is, $\sigma^2(\lambda) = a\lambda$, where λ is the mean response and a is a constant (i.e., Fano factor is constant across firing rates), an often used transformation is the square root of the responses. Empirically, we have found that this transformation works well even when neural responses have a quadratic variance-to-mean relationship. After taking the square root of the responses, we estimated sample variance for each stimulus across trials and then averaged across stimuli to get $\bar{s}^2$. We made a least-squares fit of the model to the centered mean responses (grand mean subtracted from the mean for each stimulus). We then calculated the corrected estimate of explained variance:

$$\hat{R}_c^2 = \frac{\hat{\beta}^2 - \frac{\bar{s}^2}{n}}{\hat{\alpha}^2 + \hat{\beta}^2 - \frac{\bar{s}^2}{n}(m-1)}, \tag{2.5}$$

where $\hat{\beta}^2$ is the sum of squares of the model predictions (explained variance), $\hat{\alpha}^2$ is the sum of squares of the residuals from the model (unexplained variance), $\bar{s}^2$ is sample variance across trials, averaged for all stimuli, m is the number of stimuli, and n is the number of trials.

We used a different approach to estimate how much our TI metric for V4 neurons might be degraded by noise because TI is not a correlation coefficient and does not lend itself to the methods described above. In particular, for each V4 neuron tested with stimuli at multiple positions, we built an ideal model with perfect TI by taking the responses at the position that produced the greatest response and replicating them at the other positions, but scaling them to match the original mean at each RF position. We then used this set of sample means, which has TI = 1, to generate Poisson responses, simulating the original experiment 100 times and computing the TI value for each case. We took the average drop in TI (compared to 1) to be an estimate of the upper bound of how much the V4 neuron TI values could have been degraded by noise.

2.5.10 Visualization

To visualize the features that drove a particular unit in the CNN to its highest and lowest response levels, we first ranked all images (or image patches) based on the response of the unit to the standard test set of 50,000 images for AlexNet. For units in the convolutional layers, we considered the responses at all x-y locations for a particular unique kernel. Thus, we found not just the optimal image, but also the optimal patch within the image that drove the kernel being examined. We then performed a visualization technique on the 10 most excitatory images and on the 10 most suppressive images. We followed the methods of Zeiler and Fergus (2013) [44], and used a deconvnet to project the response of the unit onto successive layers until we reached the input image. The deconvolved features can then be examined, as an RGB image, to provide a qualitative sense of what features within the image drove the unit to such a large positive or negative value.

2.6 Figures

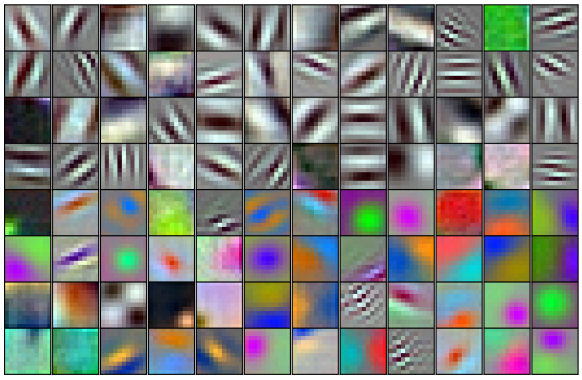
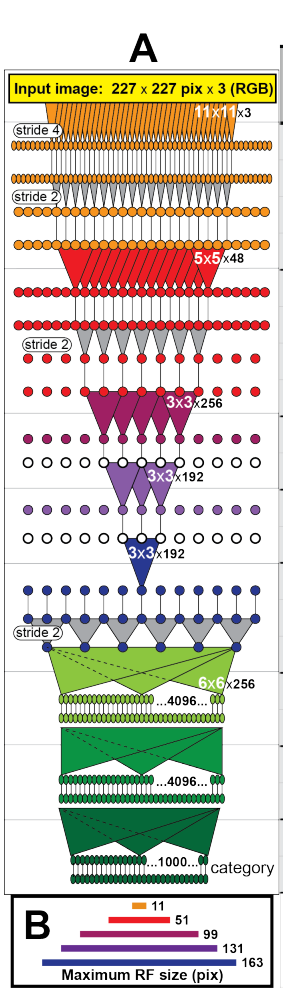


FIGURE 2.1: The 96 kernels (11 x 11 pixels, by 3 color channels) of the 1st layer, Conv1, of the AlexNet model tested here. Like many V1 receptive fields, many of these kernels are band-limited in spatial frequency and orientation. Each kernel was independently scaled to maximize its RGB dynamic range to highlight spatial structure.



C	D	E	F
Sub-layer name	**Number of unique units**	**Spatial locations (x x y)**	**Kernel size (x x y x depth)**
Conv1	96	55 x 55	11 x 11 x 3
Relu1	"	55 x 55	1
Pool1	"	27 x 27	3 x 3
Norm1	"	27 x 27	1
Conv2	256	27 x 27	5 x 5 x 48
Relu2	"	27 x 27	1
Pool2	"	13 x 13	3 x 3
Norm2	"	13 x 13	1
Conv3	384	13 x 13	3 x 3 x 256
Relu3	"	13 x 13	1
Conv4	384	13 x 13	3 x 3 x 192
Relu4	"	13 x 13	1
Conv5	256	13 x 13	3 x 3 x 192
Relu5	"	13 x 13	1
Pool5	"	6 x 6	3 x 3
FC6	4096	1	6 x 6 x 256
Relu6	"	1	1
FC7	4096	1	4096
Relu7	"	1	1
FC8	1000	1	4096
Prob8	"	1	1
21 sub-layers	**22,096** unique units	**1,553,986** total units	**60,954,656** unique weights; **724,406,816** total weights

FIGURE 2.2: See below for legend.

Figure 2. Architecture of the Caffe AlexNet CNN. **(A)** A one-dimensional scale view of the fan-in and spatial resolution of units for all 21 sublayers, aligned to their names listed in column (C). The color-filled triangles in convolutional (Conv) layers indicate the fan-in to convolutional units, gray triangles indicate the fan-in to max pooling units, and circles (or ovals) indicate the spatial positions of units along the horizontal dimension. For the Conv layers and their sublayers, each circle in the diagram represents the number of unique units listed in column (D). For example, for each orange circle/oval in the four sublayers associated with Conv1, there are 96 different units in the model (the Conv1 kernels are depicted in Figure 2.1). The 227 pixel wide input image (top, yellow), is subsampled at the Conv1 sublayer (orange; "stride 4" indicates that units occur only every 4 pixels) and again at each pooling sublayer ("stride 2"), until the spatial resolution is reduced to a 6 x 6 grid at the transition from Pool5 to FC6. The pyramid of support converging to the central unit in Conv5 (dark blue triangle) is indicated by triangles and line segments starting from Conv1. Each unit in layers FC6, FC7 and FC8 (shades of green; not all units are shown) receives inputs from all units in the previous layer (there is no spatial dimension in the FC layers, units are depicted in a line only for convenience). Green triangles indicate the full fan-in to three example units in each FC layer. **(B)** The maximum width (in pixels) of the RFs for units in the five convolutional layers (colors match those in (A)) based on fan-in starting from the input image. For the FC layers, the entire image is available to each unit. **(C)** Names of the sublayers, aligned to the circuit in (A). Names in bold correspond to the eight major layers, each of which begins with a linear kernel (colorful triangles in (A)). **(D)** The number of unique units, i.e., feature dimensions, in each sublayer (double quotes repeat values from previous row). **(E)** The width and height of the spatial (convolutional) grid at each sublayer, or "1" for the FC layers. The total number of units in each sublayer can be computed by multiplying the number of unique kernels (D) by the number of spatial positions (E). **(F)** The kernel size corresponds to the number of weights learned for each unique linear kernel. Pooling layers have 3 x 3 spatial kernels but have no weights—the maximum is taken over the raw inputs. The Conv2 kernels are only 48 deep because half of the Conv2 units take inputs from the first 48 feature dimensions in Conv1, whereas the other half take inputs from the last 48 Conv1 features; inputs are similarly grouped in Conv4 and Conv5 (see Krizhevsky et al.'s Fig. 2). The bottom row provides totals. In addition to the weights associated with each kernel, there is also one bias value per kernel (not shown), which adds 10,568 free parameters to the $\sim$60.9 million unique weights.

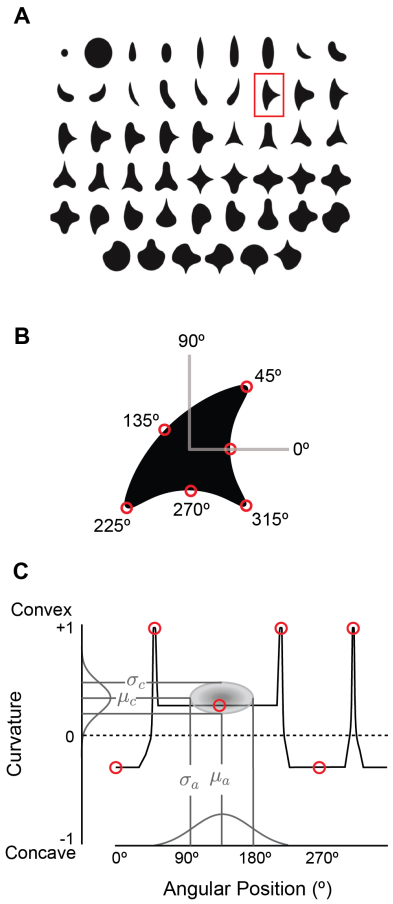


FIGURE 2.3: The angular position and curvature (APC) model and associated stimuli. **(A)** The set of 51 simple closed shapes from Pasupathy and Connor (2001). Shapes are shown to relative scale. Shape size, given in pixels in the text, refers to the diameter of the big circle (top row, 2nd shape from the left). Each shape was shown at up to eight rotations as dictated by rotational symmetry, e.g., the small and large circles (upper left) were only shown at one rotation. This yielded a set of 362 unique shape stimuli. Stimuli were presented as white-on-black to the network (not as shown here). **(B)** Example shape with points along the boundary (red circles) indicating where angular position and curvature values were included in the APC model. **(C)** Points from the example shape in (B) are plotted in the APC plane where x-axis is angular position and y-axis is normalized curvature. Note the red circle furthest to the left at 0° angular position and negative curvature corresponds to the concavity at 0° on the example shape in (B). A schematic APC model is shown (ellipse near center of diagram) that is a product of Gaussians along the two axes. This APC model would describe a neuron with a preference for mild concavities at 135°.

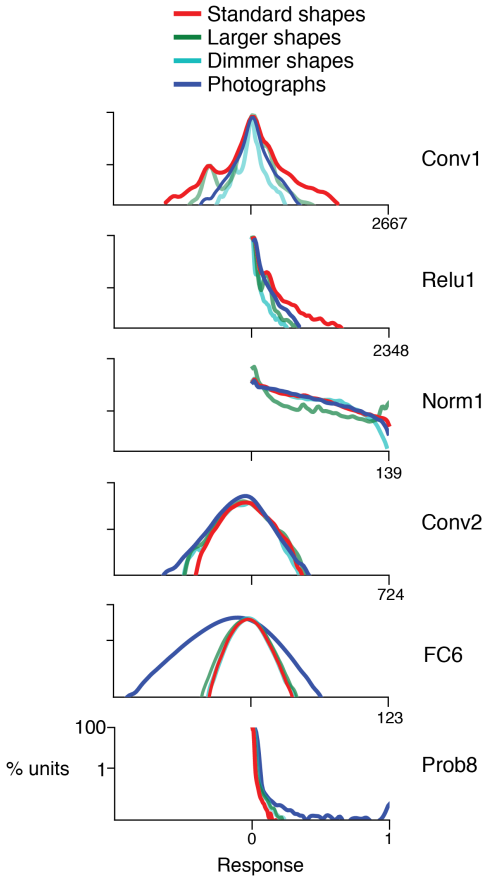


FIGURE 2.4: Response distributions for shapes and natural images in representative CNN layers. In each panel, the frequency distribution of the response values across all unique units in a designated CNN sublayer is plotted for four stimulus sets: our standard shape set (red; size 32 pixels, stimulus intensity 255, see Methods), larger shapes (cyan; size 64 pixels, intensity 255), dimmer shapes (green; intensity 100, size 32 pixels) and natural images (dark blue). Natural images (n = 362, to match the number of shape stimuli) were pulled randomly from the ImageNet 2012 competition validation set. From top to bottom, panels show results for selected sublayers: Conv1, Relu1, Norm1, Conv2, FC6 and Prob8 (Figure 2.2C lists sublayer names). The number of points in each distribution is given by the number of stimuli (362) times the number of unique units in the layer (Figure 2.2D). The vertical axis is log scaled as most distributions have a very high peak at 0. For Conv1, standard shapes drove a wider overall dynamic range than did images because of the high intensity edges that aligned with parts of the linear kernels (Figure 2.1). This was not the case for larger shapes because they often over-filled the small Conv1 kernels. For Relu1, negative responses are removed by rectification after a bias is added. At Conv2, there is little difference between the four stimulus sets on the positive side of the distribution. This changes from FC6 forward, where natural images drive a wider range of responses. For Prob8, natural images (dark blue line) sometimes result in high probabilities among the 1000 categorical units, whereas shapes do not.

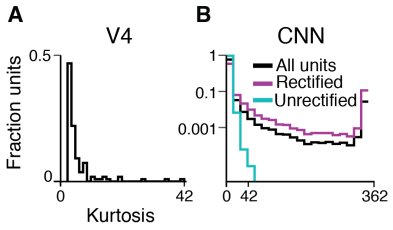


FIGURE 2.5: Sparsity of CNN and V4 unit responses to shape stimuli. **(A)** The distribution of K (kurtosis, Eqn. 1) for all 109 V4 neurons from Pasupathy and Connor (2001) was skewed strongly to the left. Most V4 neurons have values clustered around the mean, 5.9 (SD 6.1), whereas a few outliers have high sparsity. **(B)** Distribution of K for all CNN units (black), and separately for units in rectified layers (purple) vs. non-rectified layers (cyan). Note change in x-axis and log y-axis compared to (A). Rectified layers include all Relu, Pool and Norm sublayers (they have no negative responses); non-rectified layers include all Conv sublayers (Figure 2.2C). The substantial peak at the maximal kurtosis value (K=362) corresponds to units with one non-zero response among 362 stimuli. There were no such extremely sparse-responding units in the non-rectified layers (cyan; mean 4.1, SD 3.7), which had a K distribution that covered a range closer to that observed in (A) for V4.

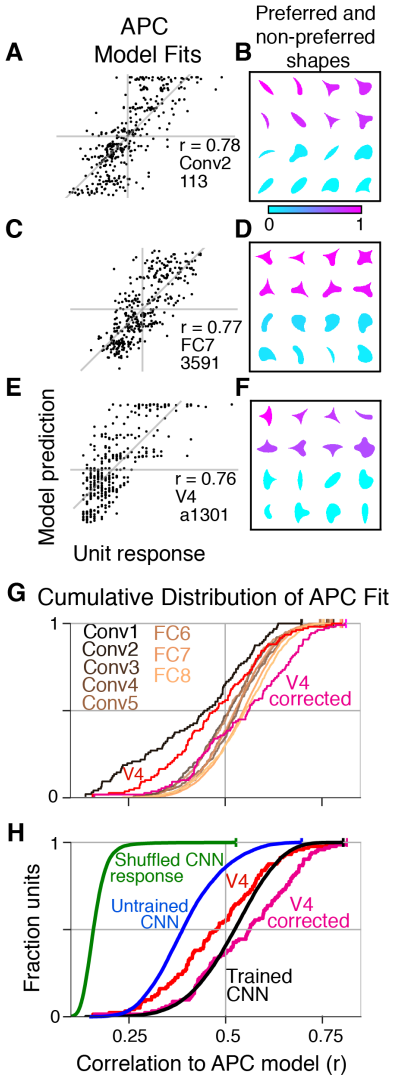


FIGURE 2.6: See below for figure legend.

Figure 5. Boundary curvature selectivity for CNN units compared to V4 neurons. **(A)** APC model prediction vs. CNN unit response for an example CNN unit from an early layer (Conv2-113). **(B)** The top and bottom eight shapes sorted by response amplitude (most preferred shape is at upper left, least at lower right) reveal a preference for convexity to the upper left (such a feature is absent in the non-preferred shapes). This is consistent with the APC fit parameters, $\mu_c = 1.0$, $\sigma_c = 0.53$, $\mu_a = 135°$, $\sigma_a = 23°$. **(C)** Predicted vs. measured responses for another well-fit example CNN unit (FC7-3591) but in a later layer. **(D)** Top and bottom eight shapes for example unit in (C). The APC model fit was $\mu_c = -0.1$, $\sigma_c = 0.15$, $\mu_a = 112°$, $\sigma_a = 44°$. **(E)** Model prediction vs. neuronal mean firing rate (normalized) for the V4 neuron (a1301) that had the highest APC fit r-value. **(F)** The top eight shapes (purple) all have a strong convexity to the left, whereas the bottom eight (cyan) do not. The APC model fit was $\mu_c = 1.0$, $\sigma_c = 0.39$, $\mu_a = 180°$, $\sigma_a = 23°$. **(G)** The cumulative distributions (across units) of APC r-values are plotted for the first sublayer of each major CNN layer (boldface names in Figure 2.2C) from Conv1 (black) to FC8 (lightest orange). The other sublayers (distributions not shown for clarity) tended to have lower APC r-values but the trend for increasing APC r-value with layer was similar. For comparison, red line shows cumulative distribution for 109 V4 neurons (Pasupathy and Connor, 2001), and pink line shows V4 distribution corrected for noise (see Methods). **(H)** The cumulative distribution of r-values for the APC fits for all CNN units (black), CNN units with shuffled responses (green), units in an untrained CNN (blue) and V4 (red and pink). The far leftward shift of the green line shows that fit quality deteriorates substantially when the responses are shuffled across the 362 stimuli within each unit.

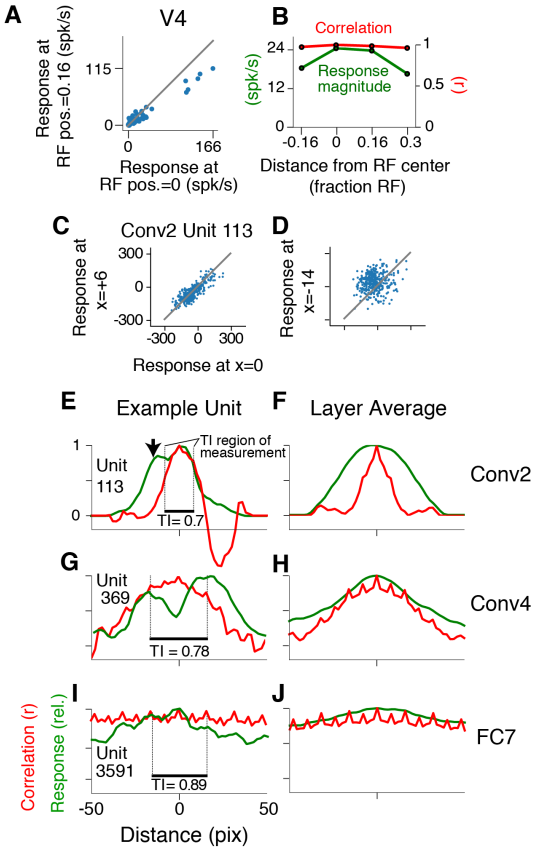


FIGURE 2.7: see below for figure legend

Figure 7. Translation invariance as a function of position across the RF. **(A)** For an example neuron from the V4 study of El-Shamayleh and Pasupathy (2016), the responses to stimuli shifted away from the RF center by 1/6 of the estimated RF size are plotted against those placed in the RF center. The overall response magnitude decreases with shift, but a strong linear relationship is maintained between responses at the two positions. **(B)** In green, the RF profile of the same neuron from (A) is plotted (average response at each position). In red, the correlation of the responses at each position with the responses at RF center. **(C)** For unit Conv2-113, responses to stimuli shifted 6 pixels to the right are plotted against responses for centered stimuli. **(D)** For the same unit in (C), responses for stimuli shifted 14 pixels to the left vs. responses for centered stimuli. **(E)** For unit Conv2-113, the position-correlation function is plotted in red. The RF profile, i.e., the normalized response magnitude (square root of sum of squared responses) across all shapes is plotted in green. The region over which TI is measured, where all stimuli are wholly within the CRF (see Methods), is within dotted lines bookending horizontal black bar. The unit is less translation invariant because it continues to have a large response even when correlation drops quickly from center. This is reflected in the lower TI score of 0.7. **(F)** The averages of the correlation and RF profiles across all units in the Conv2 layer show that correlation drops off much more rapidly than the RF profile. **(G)** Same as in (E) but for a unit in the 4th convolutional layer (Conv4-369). There is a broadened correlation profile compared to the Conv2 unit. **(H)** For Conv4, the average position-correlation function (red) has a wider peak than that for Conv2, more closely matching the shape of the average RF profile (green). It also has serrations that occur 8 pixels apart, which corresponds to the pixel stride (discretization) of Conv2 (Figure 2.2A; see Methods). **(I)** The shape-tuned example unit FC7 3591 (Figure 2.6C) in the final layer is highly translation invariant (TI=0.89). **(J)** The response profile and correlation stay high across the center of the input field on average across units in FC7.

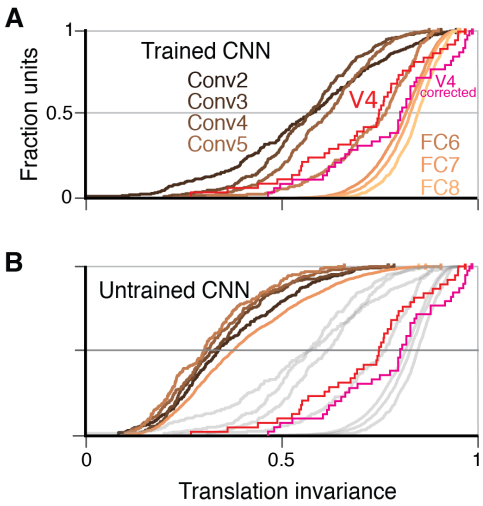


FIGURE 2.8: Cumulative distributions of the TI metric for the CNN and V4. **(A)** The cumulative distributions (across units) of TI are plotted for the first sublayer of each major CNN layer (boldface names in Figure 2.2C) from Conv2 (black) to FC8 (lightest orange). There is a clear increase in TI moving up the hierarchy. The TI distribution for V4 is plotted in red, and an upper bound for noise correction is plotted in pink (see Methods). The other sublayers (distributions not shown for clarity) tended to have lower TI values but the trend for increasing TI with layer was similar. **(B)** The cumulative distribution of TI across layers in the untrained CNN. There is a large shift toward lower TI values in comparison to the trained CNN (faint grey and red and pink lines reproduce traces from panel A).

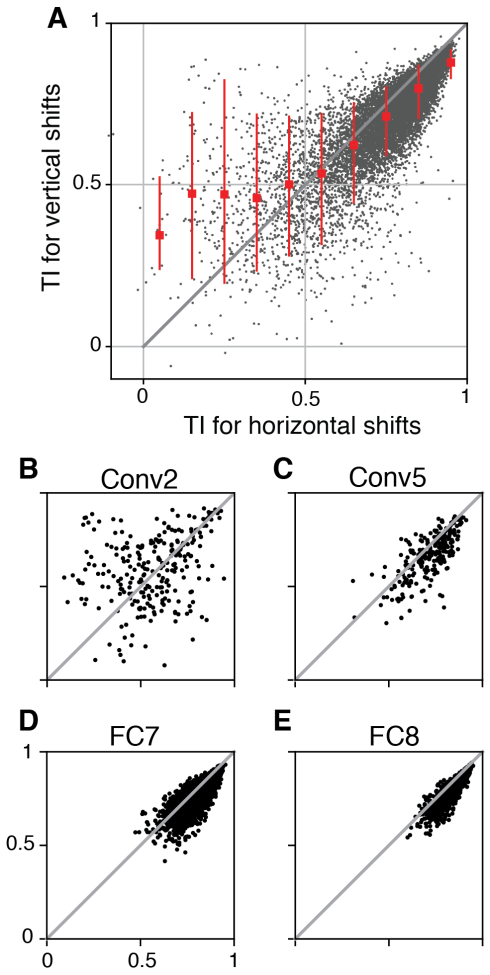


FIGURE 2.9: See below for figure legend.

Figure 7s. Translation invariance as a function of position across the RF. **(A)** For an example neuron from the V4 study of El-Shamayleh and Pasupathy (2016), the responses to stimuli shifted away from the RF center by 1/6 of the estimated RF size are plotted against those placed in the RF center. The overall response magnitude decreases with shift, but a strong linear relationship is maintained between responses at the two positions. **(B)** In green, the RF profile of the same neuron from (A) is plotted (average response at each position). In red, the correlation of the responses at each position with the responses at RF center. **(C)** For unit Conv2-113, responses to stimuli shifted 6 pixels to the right are plotted against responses for centered stimuli. **(D)** For the same unit in (C), responses for stimuli shifted 14 pixels to the left vs. responses for centered stimuli. **(E)** For unit Conv2-113, the position-correlation function is plotted in red. The RF profile, i.e., the normalized response magnitude (square root of sum of squared responses) across all shapes is plotted in green. The region over which TI is measured, where all stimuli are wholly within the CRF (see Methods), is within dotted lines bookending horizontal black bar. The unit is less translation invariant because it continues to have a large response even when correlation drops quickly from center. This is reflected in the lower TI score of 0.7. **(F)** The averages of the correlation and RF profiles across all units in the Conv2 layer show that correlation drops off much more rapidly than the RF profile. **(G)** Same as in (E) but for a unit in the 4th convolutional layer (Conv4-369). There is a broadened correlation profile compared to the Conv2 unit. **(H)** For Conv4, the average position-correlation function (red) has a wider peak than that for Conv2, more closely matching the shape of the average RF profile (green). It also has serrations that occur 8 pixels apart, which corresponds to the pixel stride (discretization) of Conv2 (Figure 2.2A; see Methods). **(I)** The shape-tuned example unit FC7 3591 (Figure 2.6C) in the final layer is highly translation invariant (TI=0.89). **(J)** The response profile and correlation stay high across the center of the input field on average across units in FC7.

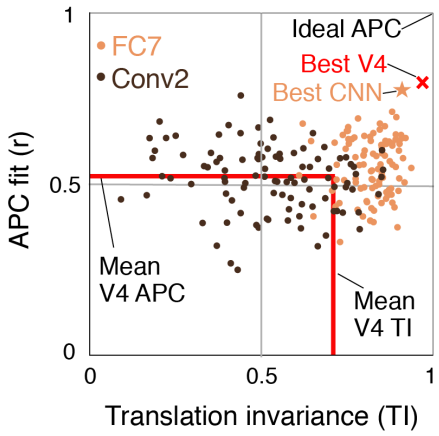


FIGURE 2.10: Summary of the similarity of CNN units to V4 neurons in terms of translation invariance (TI) and fit to the APC model. For 100 randomly selected CNN units from Conv2 (brown) and FC7 (orange), APC r-value is plotted against TI. The hypothetical highest scoring V4 unit (red $\times$) is the combination of the highest TI score and the highest APC fit from separate V4 data sets (0.97, 0.80). The highest scoring unit in the CNN (FC7-3591, from Figure 2.6C, Figure 2.7I and Figure 2.13C) is indicated by the orange star (0.91, 0.77) and is close to the hypothetical best V4 unit. The red lines indicate the mean V4 values along each axis, not including any correction for noise (see Figures 5 and 7 for estimated noise correction, pink lines).

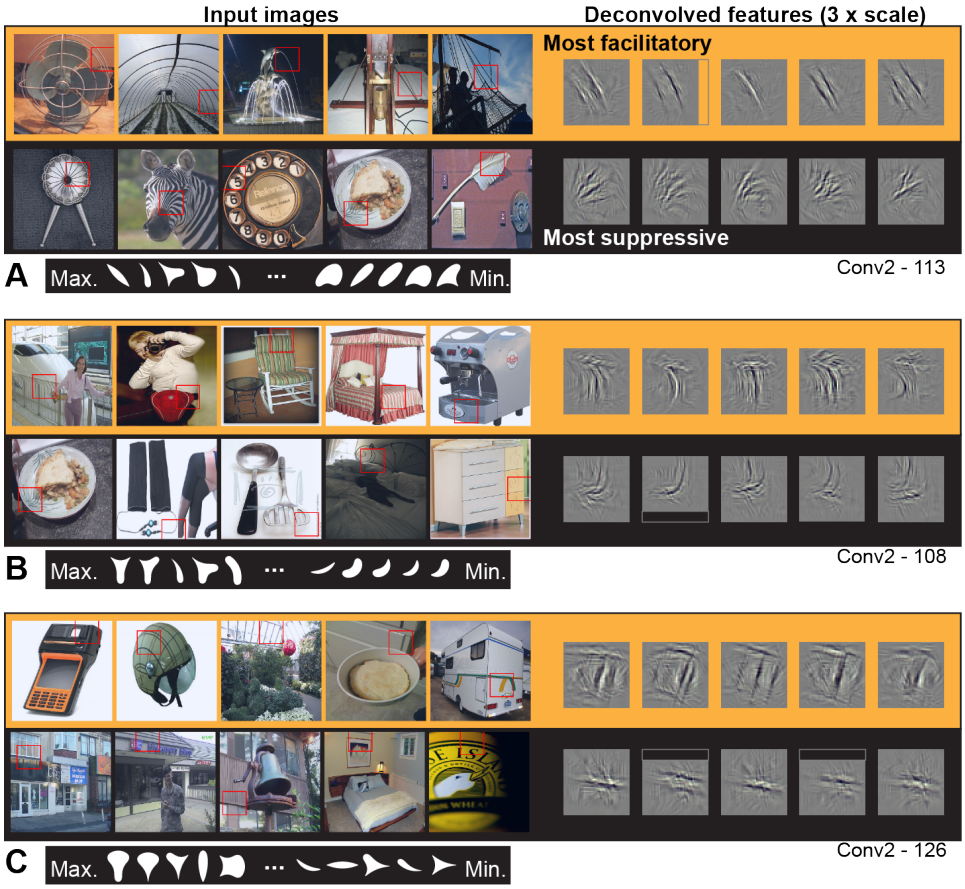


FIGURE 2.11: Visualization of APC-like units in layer Conv2. **(A)** For unit Conv2-113, the 5 most excitatory image patches are indicated by red squares superimposed in the raw images (top row, left side, from left to right). The size of the red square corresponds to the maximal extent of the image available to Conv2 units (see Figure 2.2B). In corresponding order, the five deconvolved features are shown at the upper right, with a 3x scale increase for clarity. The blank rectangular region at the right side of the second feature indicates that this part of the unit RF extended beyond the input image (such regions are padded with zero during response computation). For the same unit, the lower row shows the 5 most suppressive image patches and their corresponding deconvolved features. We examined the top 10 most excitatory and suppressive images, and for all examples in this and subsequent figures, they were consistent with the top 5. Below the natural images are the top 5 and bottom 5 shapes (white on black background) in order of response from highest (at left) to lowest (at right). Shapes are shown at 2x scale relative to images, for visibility. **(B)** Same format as (A), but for unit Conv2-108. **(C)** Same format as (A), but for unit Conv2-126. In all examples, the most suppressive features (bottom row in each panel) tend to run orthogonal to, and at the same RF position, as the preferred features (top row in each panel) For APC fit parameters, see Table 1 in Results text. The un-redacted input image thumbnails were accessed via the ImageNet database and the original image URLs can be found through this site (http://image-net.org/about-overview). These thumbnails may be subject to copyright. They are not available under CC-BY and are exempt from the CC-BY 4.0 license.

FIGURE 2.12: Visualization of APC-like units in layers Conv3 to Conv5. **(A)** Visualization for unit Conv3-156, using the same format as Figure 2.11. Deconvolved features are scaled by 1.8 for visibility. **(B)** Same as (A), for unit Conv3-020. **(C)** Same for unit Conv5-161, but deconvolved features are scaled by 1.15. **(D)** Same as (C), but for unit Conv5-144. For APC fit parameters, see Table 1 in main text. The un-redacted input image thumbnails were accessed via the ImageNet database and the original image URLs can be found through this site (http://image-net.org/about-overview). These thumbnails may be subject to copyright. They are not available under CC-BY and are exempt from the CC-BY 4.0 license.

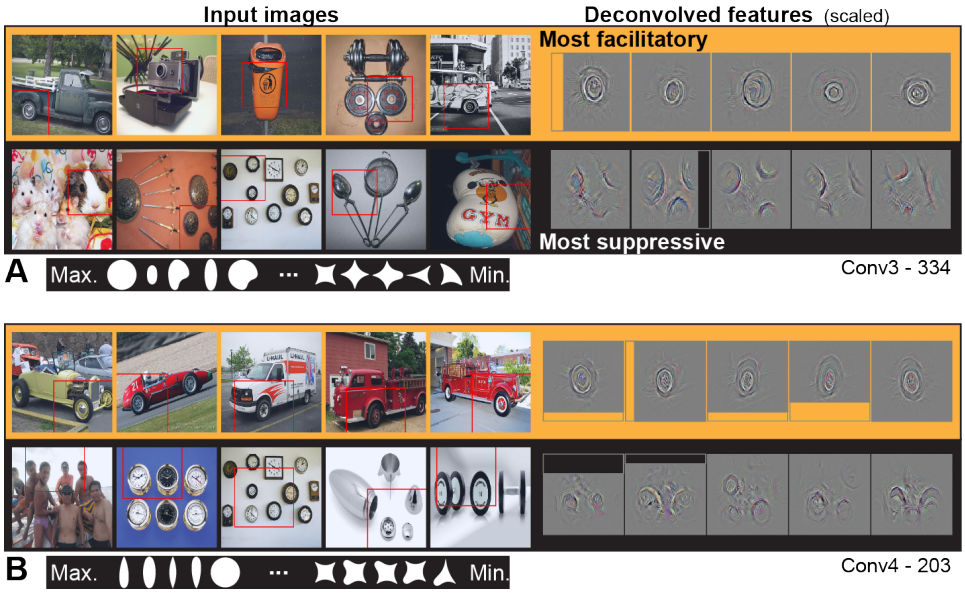


FIGURE 2.13: Visualization of APC-like units: circle detectors. These examples are representative of many units that were selective for circular forms. **(A)** Unit Conv3-334 was selective for a wide variety of circular objects near its RF center and was suppressed by circular boundaries entering its RF from the surround. Deconvolved feature patches are scaled up by 1.8 relative to raw images. **(B)** Unit Conv4-203 was also selective for circular shapes near the RF center, but showed category specificity for vehicle wheels. Suppression was not category specific but was, like that in (A), related to circular forms offset from the RF center. The higher degree of specificity in (B) is consistent with this unit being deeper than the example in (A). Deconvolved features are scaled by 1.4 relative to raw images. APC fit parameters are given in Table 1. The un-redacted input image thumbnails were accessed via the ImageNet database and the original image URLs can be found through this site (http://image-net.org/about-overview). These thumbnails may be subject to copyright. They are not available under CC-BY and are exempt from the CC-BY 4.0 license.

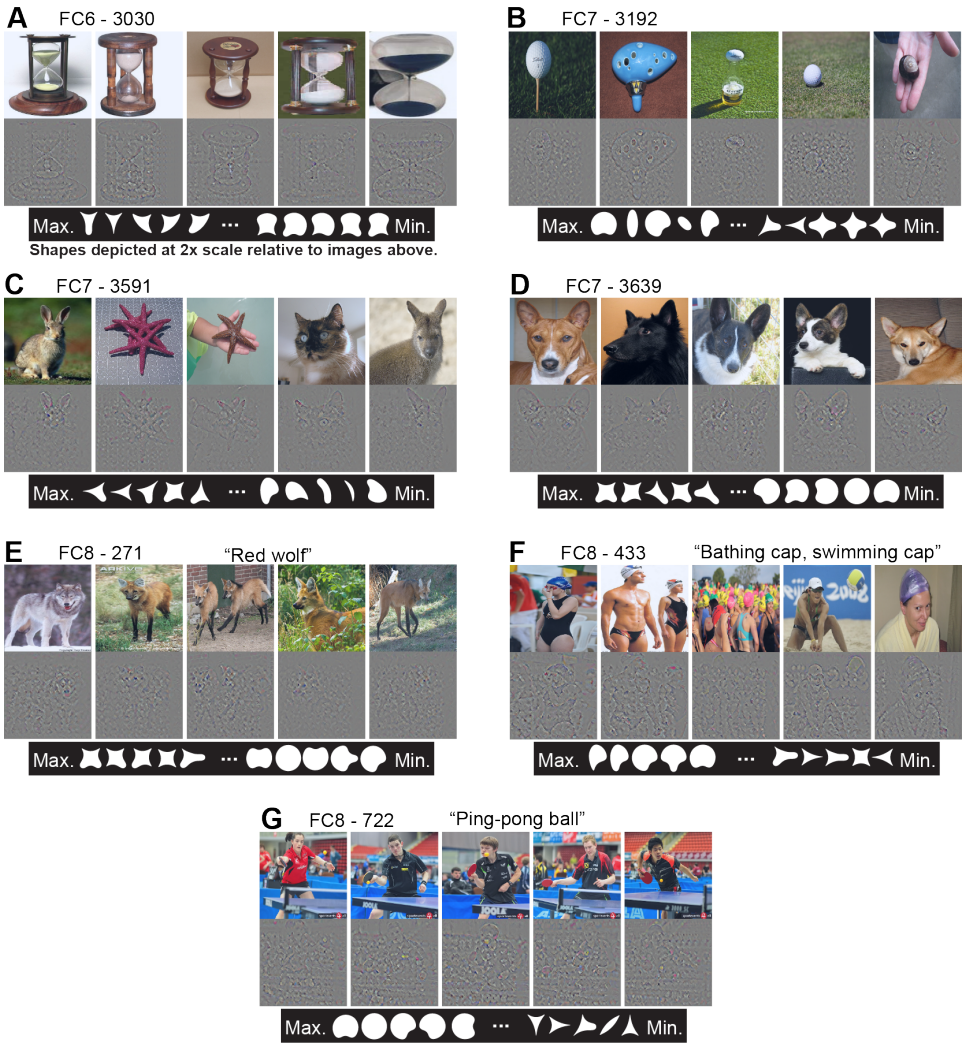


FIGURE 2.14: Visualization of APC-like units in the FC layers. **(A)** For unit FC6-3030, the top five images from the test set are shown above their deconvolved feature maps. The maximal RF for all FC units includes the entire image. At bottom, the top five shapes are shown in order from left to right, followed by the bottom 5 shapes such that the shape associated with the minimum response is the rightmost. For visibility, shapes are shown here at twice the scale relative to the images. **(B)** For unit FC7-3192, same format as (A). **(C)** For unit FC7-3591, same format as (A). **(D)** For unit FC7-3639, same format as (A). **(E)** For unit FC8-271, same format as (A), except the category of this output-layer unit is indicated as "Red wolf." **(F)** For unit FC8-433, same format as (E). **(G)** For unit FC8-722, same format as (E). See Table 1 for APC fit values for all units. The un-redacted input image thumbnails were accessed via the ImageNet database and the original image URLs can be found through this site (http://image-net.org/about-overview). These thumbnails may be subject to copyright. They are not available under CC-BY and are exempt from the CC-BY 4.0 license.

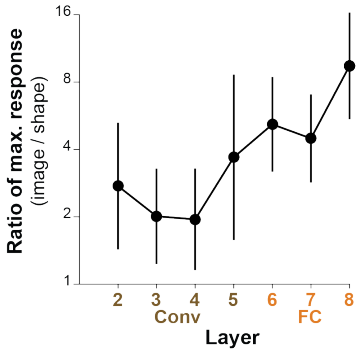


FIGURE 2.15: Comparing the maximum responses driven by images to those driven by shapes for APC-like units. For a given CNN unit, we computed the ratio of the maximum response across natural images (50,000 image test set) to the maximum response across our set of 362 shapes. The average of this ratio across the top ten APC-like units in each of seven layers (Conv2 to FC8) is plotted. Error bars show SD. In a few cases, the maximum response to shapes was a negative value and these cases were excluded: one unit for Conv3 and two for FC6 and FC7.

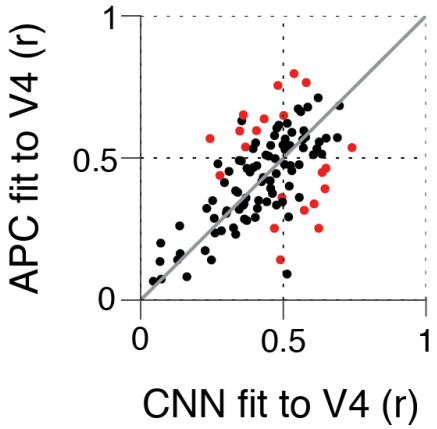


FIGURE 2.16: Comparing the ability of the APC model vs. single CNN units to fit V4 neuronal data. Showing r-values for cross-validated fits from both classes of model, black points correspond to V4 neurons for which neither model performed significantly better at predicting responses to the shape set. The APC model provided a better fit for red points above the line of equality, whereas points below the line correspond to neurons for which at least one unit within the trained CNN provided a better fit than any APC model.

Chapter 3

The unbiased estimation of the fraction of variance explained by a model

3.1 Summary

The correlation coefficient squared, r^2, is commonly used to validate quantitative models on neural data, yet it is biased by trial-to-trial variability: as trial-to-trial variability increases, measured correlation to a model's predictions decreases. As a result, models that perfectly explain neural tuning can appear to perform poorly. Many solutions to this problem have been proposed, but no consensus has been reached on which is the least biased estimator. Some currently used methods substantially overestimate model fit, and the utility of even the best performing methods is limited by the lack of confidence intervals and asymptotic analysis. We provide a new estimator, $\hat{r}^2_{\text{ER}}$, that outperforms all prior estimators in our testing, and we provide confidence intervals and asymptotic guarantees. We apply our estimator to a variety of neural data to validate its utility. We find that neural noise is often so great that confidence intervals of the estimator cover the entire possible range of values ([0,1]), preventing meaningful evaluation of the quality of a model's predictions. This leads us to propose the use of the signal-to-noise ratio (SNR) as a quality metric for making quantitative comparisons across neural recordings. Analyzing a variety of neural data sets, we find that up to $\sim 40\%$ of some state-of-the-art neural recordings do not pass even a liberal SNR criterion. Moving toward more reliable estimates of correlation, and quantitatively comparing quality across recording modalities and data sets, will be critical to accelerating progress in modeling biological phenomena.

3.2 Introduction

Building an understanding of the nervous system requires the quantification of model performance on neural data, and this often involves computing Pearson's correlation coefficient between model predictions and neural responses. Yet this typical estimator, $\hat{r}^2$, is fundamentally confounded by the trial-to-trial variability of neural responses: a low $\hat{r}^2$ could be the result of a poor model or high neuronal variability.

One approach to this problem is to average over many repeated trials of the same stimulus in order to reduce the influence of trial-to-trial variability. With a finite number of trials, this approach will never wholly remove the influence of noise and its confounding effect, moreover, the collection of additional trials is expensive. A more principled approach has been to account for trial-to-trial variability in the estimation of the fraction of explainable variance or r^2. Most often, this takes the form

of attempting to estimate what the r^2 would have been in the absence of trial-to-trial variability. Here we call this quantity r^2_{ER}, the r^2 between the model prediction and the expected response (ER) of the neuron (i.e., the 'true' mean, or expected value, of the estimated tuning curve). While a variety of solutions have been proposed to estimate this quantity [29, 33, 34, 41, 65, 86–90], they have not been quantitatively compared, thus there is no basis to reach a consensus on which methods are appropriate, or more importantly inappropriate. We find that several estimators still in recent use have large biases. Estimators that did have relatively small biases lacked associated confidence intervals, thus the degree of uncertainty in these sometimes highly variable estimates remains ambiguous. Finally, none of these methods have been analyzed asymptotically to give a theoretical guarantee that they will converge to r^2_{ER}, i.e., it has not been shown that they are consistent estimators.

To address these substantial problems, we introduce $\hat{r}^2_{\text{ER}}$, which is a simple analytic estimator of r^2_{ER}, along with a method for generating α-level confidence intervals. We validate our estimator in simulation, prove that it is consistent, and provide head-to-head comparisons to prior methods. We then demonstrate the use of $\hat{r}^2_{\text{ER}}$ and its confidence interval on two sets of neural data. We find many cases where neuronal data is so noisy that estimates of r^2_{ER} provide little inferential power about the quality of a model fit. This naturally leads to a useful metric of the quality of a neuronal recording that we will refer to as the signal-to-noise ratio (SNR), and which can be interpreted in terms of the number of trials needed to reliably detect tuning. Across a diverse set of neural recordings, we find that many neurons do not pass even a liberal criterion for providing meaningful insight into the quality of a model fit.

3.3 Results

Our results are organized as follows. First, we give the essential intuition into the source of the bias in $\hat{r}^2$ and we explain how $\hat{r}^2_{\text{ER}}$ removes this bias. Next, we evaluate $\hat{r}^2_{\text{ER}}$ through simulation and compare it to prior methods. We then demonstrate the method on two neural data sets: one from a study of motion direction tuning in area MT and one from a study of responses to natural images in area V4. Finally, we develop an estimator, $\widehat{\text{SNR}}$, based on the signal-to-noise ratio (SNR), as a metric to determine the inferential power of a given neuronal recording.

3.3.1 Bias of $\hat{r}^2$ and its correction

Consider a typical scenario in sensory neuroscience where the responses of a neuron to m stimuli across n repeated trials of each stimulus have been collected and the average of these responses, the estimated tuning curve (Figure 3.1, dashed green line), is compared to responses predicted by a model (red line). Even if the m expected values of the neuronal response, μ_i (solid green trace), perfectly correlate with the model predictions, ν_i (red trace is scaled and shifted relative to green), the m sample averages, Y_i (dashed green trace), will deviate from their expected value owing to the sample mean's variability. Here, we quantify this variability using the variance, σ^2, of the distribution of responses from trial-to-trial (see Methods, "Assumptions and terminology for derivation of unbiased estimators"). We assume σ^2 is constant across responses to different stimuli, which can be achieved by applying a variance stabilizing transform to the data. The variance of the sample mean for all stimuli will thus be $\frac{\sigma^2}{n}$. Owing to the variance of the sample mean, the reported $\hat{r}^2$ can be

appreciably less than 1 even though the r^2 between the underlying expected values of the neuronal response and the model is 1.

The quantity we attempt to estimate in this paper is r^2 between the model predictions (ν_i) and the expected neuronal responses (μ_i). We will call this quantity r^2_{ER}, the fraction of variance of the 'expected response' explained by the model:

$$r^2_{\mathrm{ER}} = \frac{(\sum_{i=1}^{m}(\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2}{\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 \sum_{i=1}^{m}(\mu_i - \bar{\mu})^2}. \tag{3.1}$$

We will show that the naive sample estimator, which uses Y_i in place of μ_i,

$$\hat{r}^2 = \frac{(\sum_{i=1}^{m}(\nu_i - \bar{\nu})(Y_i - \bar{Y}))^2}{\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2}, \tag{3.2}$$

has an expected value that can be well approximated as the ratio of the expected values of its numerator and denominator as follows (for asymptotic justification see Methods, "Inconsistency of $\hat{r}^2$ in m"):

$$\begin{aligned} \mathrm{E}[\hat{r}^2] &\approx \frac{\mathrm{E}[(\sum_{i=1}^{m}(\nu_i - \bar{\nu})(Y_i - \bar{Y}))^2]}{\mathrm{E}[\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2]} \\ &= \frac{(\sum_{i=1}^{m}(\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2 + \frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2}{\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 \sum_{i=1}^{m}(\mu_i - \bar{\mu})^2 + \frac{\sigma^2}{n}(m-1)\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2}. \end{aligned} \tag{3.3}$$

While the terms on the left in the numerator and denominator are the same as r^2_{ER}, the terms on the right are proportional to the trial-to-trial variability (σ^2) and cause $\hat{r}^2$ to deviate from r^2_{ER}. This is the essential problem: $\hat{r}^2$ is biased away from r^2_{ER} by terms proportional to the amount of variability, $\frac{\sigma^2}{n}$, in the estimated responses.

The strategy we take to solve this problem is straightforward: find unbiased estimators of these noise terms and subtract them from the numerator and denominator of Eqn. 4.2 for $\hat{r}^2$, thus:

$$\hat{r}^2_{\mathrm{ER}} = \frac{(\sum_{i=1}^{m}(\nu_i - \bar{\nu})(Y_i - \bar{Y}))^2 - \frac{\hat{\sigma}^2}{n}\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2}{\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - \frac{\hat{\sigma}^2}{n}(m-1)\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2}, \tag{3.4}$$

where $\hat{\sigma}^2$ is an unbiased estimator for trial-to-trial variability, after a variance stabilizing transform if necessary. Typically $\hat{\sigma}^2 = \hat{s}^2$, the sample variance, but not necessarily. For example if stimuli are shown only once ($n = 1$), then an assumed value of trial-to-trial variability could be substituted into $\hat{\sigma}^2$. The numerator and denominator of the fraction $\hat{r}^2_{\mathrm{ER}}$ are unbiased estimators of the numerator and denominator of r^2_{ER}; therefore, this solution is approximate since the expected value of a ratio is not necessarily the ratio of the expected values of the numerator and denominator (see Methods, "Bias of $\hat{r}^2_{\mathrm{ER}}$"). Yet we show in simulation that the approximation is very good for typical neural statistics, and we show analytically that, unlike $\hat{r}^2$, our estimator $\hat{r}^2_{\mathrm{ER}}$ converges to the true r^2_{ER} as the number of stimuli $m \to \infty$ (see Methods, "Consistency of $\hat{r}^2_{\mathrm{ER}}$ in m"). We next evaluate this estimator in simulation.

3.3.2 Validation of $\hat{r}^2_{\mathrm{ER}}$ by simulation

To demonstrate the effectiveness and key properties of $\hat{r}^2_{\mathrm{ER}}$, we ran a simulation with $m = 362$ stimuli, $n = 4$ repeats, and $\sigma^2 = 0.25$ (the trial-to-trial variance of Poisson

neuronal response after a variance-stabilizing transform, see Methods: "Assumptions and terminology for derivation of unbiased estimator"). This corresponds, for example, to a minimal experiment to characterize shape tuning in V4 neurons, which requires hundreds of unique shapes and takes on the order of 1 hour [41]. In the case where the model prediction (v_i) and expected response (μ_i) were perfectly correlated (as in Figure 3.1) and SNR was moderate at 0.5, the distribution of the naive estimator, $\hat{r}^2$, is centered well below 1 (Figure 3.2A, blue). Thus, the model appears to be a poor fit to data that it in fact generated, indicating that $\hat{r}^2$ is a poor estimator of r^2_{ER}. On the other hand, the distribution of our corrected estimator, $\hat{r}^2_{\text{ER}}$, is appropriately centered at 1 (Figure 3.2A, orange). Approximately 50% of the time our estimator exceeds 1, taking on impossible values of $r^2_{\text{ER}} \in [0, 1]$, but this is necessary to achieve unbiased estimates for high r^2_{ER} because truncating the values would shift the mean below 1.

We evaluated the estimators $\hat{r}^2$ and $\hat{r}^2_{\text{ER}}$ at five values of r^2_{ER} (0, 0.25, 0.5, 0.75, 1) and plotted the mean and 90% quantiles. Figure 3.2B shows that $\hat{r}^2$ (blue line) grossly underestimates r^2_{ER} (black line) at all levels except for $r^2_{\text{ER}} = 0$, whereas $\hat{r}^2_{\text{ER}}$ (orange line) correctly estimates the true correlation r^2_{ER} (orange and black lines overlap). Thus the estimator $\hat{r}^2_{\text{ER}}$ performs favorably in this simulation. Next, we characterize the performance of $\hat{r}^2_{\text{ER}}$ relative to $\hat{r}^2$ in simulations that cover a wide range of the parameters, m, n and SNR.

3.3.3 Asymptotic properties of $\hat{r}^2_{\text{ER}}$ and $\hat{r}^2$

We ran simulations to determine the bias and variance of $\hat{r}^2_{\text{ER}}$ relative to $\hat{r}^2$ as a function of the parameters SNR, n, and m. Figure 3.3A shows that as SNR increases, $\hat{r}^2$ (blue) and $\hat{r}^2_{\text{ER}}$ (orange) converge ($r^2_{\text{ER}} = 0.75$, $n = 4$, $m = 362$). Thus, for neuronal recordings where variation in response strength across stimuli is large relative to trial-to-trial variability, these two estimators should have similar values. At low values of SNR, e.g., 0.1, $\hat{r}^2$ has a large downward bias (mean $\hat{r}^2$=0.23), whereas $\hat{r}^2_{\text{ER}}$ has a small upward bias relative to its own variability and to the bias of $\hat{r}^2$ (for the source of this bias see Methods, "Bias of $\hat{r}^2_{\text{ER}}$"). This small upward bias of $\hat{r}^2_{\text{ER}}$ quickly diminishes as SNR increases, whereas the large negative bias of $\hat{r}^2$ remains across a much wider range of SNR. The essential problem this simulation reveals is that if SNR varies widely from neuron to neuron, the bias in the naive estimate will cause apparent variation in r^2 across neurons that depends on SNR and not on the underlying tuning curve. Neuronal SNR is not typically under experimental control, making this problem difficult to avoid.

The number of repeats, n, is under the experimenter's control but is expensive to increase. Figure 3.3B shows how $\hat{r}^2$ and $\hat{r}^2_{\text{ER}}$ converge as n increases. Thus the bias in $\hat{r}^2$ can be reduced by increasing the number of repeats, but to achieve this requires a very high number of repeats for low SNR. An advantage of $\hat{r}^2_{\text{ER}}$ is that even for low n, it on average estimates the true correlation to the model (orange trace overlaps black trace, Figure 3.3B), providing a large gain in total trial efficiency for estimating the quality of model fit.

When increasing the number of stimuli, m, unlike the previous two cases, $\hat{r}^2$ and $\hat{r}^2_{\text{ER}}$ do *not* converge to the same value (Figure 3.3C). While variability of both estimators decreases (90% quantiles narrow), it is clear in simulation that $\hat{r}^2$ is not a consistent estimator of r^2_{ER} in m since it does not converge to $r^2_{\text{ER}} = 0.75$. While there is a small upward bias of $\hat{r}^2_{\text{ER}}$ for low m, as m increases this bias is reduced (see Methods, "Consistency of $\hat{r}^2_{\text{ER}}$ in m").

3.3.4 Comparison to prior methods

Accounting for noise when interpreting the fit of models to neural data has been examined and applied in the neuroscientific literature for some time [29, 33, 34, 41, 65, 86–90]. Several studies have followed the approach of attempting to estimate the upper bound on the quality of fit of a model given noise and then referencing the measure of fit to this quantity. Roddey et al. [86] estimate this upper bound by computing their estimate of model fit, 'coherence', across split trials then plotting coherence of the data to the model predictions relative to the split trial coherence. Yamins et al. [29] normalize r^2 with split-trial correlation transformed by the Spearman-Brown prediction formula (averaged across randomly resampled subsets of trials); we will call this $\hat{r}^2_{\text{norm-split-SB}}$. Hsu et al. [88] also use split-half correlation (averaged across randomly resampled subsets of trials), to estimate an upper bound (CC_{max}) by a transformation they derive attempting to estimate the correlation of the true mean with the firing rate of the neuron. For purpose of comparison, we square this estimator and call it $\text{CC}^2_{\text{norm}}$. Schoppe et al. [90] improve upon this method by giving an analytic form, thus removing the need for resampling. They do this by using the 'signal-power' (SP) estimate developed by Sahani and Linden [87], thus we call their estimator $\text{CC}^2_{\text{norm-SP}}$. Kindel et al. [34] take inspiration from Schoppe et al., except to estimate CC_{max} they measure the correlation of responses from a Gaussian simulation (based on the sample mean and variance of the neural data) with the sample mean. We square their estimator and call it $\text{CC}^2_{\text{norm-PB}}$ (PB for parametric bootstrap). Pasupathy and Connor [41] estimate the fraction of total variance accounted for by trial-to-trial variability, intuitively the fraction of unexplainable variance, then use it to normalize $\hat{r}^2$. We call this estimator $\hat{r}^2(1 - \frac{SE^2}{SS_{\text{total}}})$. With a similar motivation, Cadena et al. [33] provide a metric they call "fraction explainable variance explained" (FEVE). They form the ratio of mean squared prediction error over total variance of the response (except subtracting off an estimate of trial-to-trial variability from both) and subtract this ratio from one. While all of these methods might be intuitively appealing, the quantities to which they converge, and their relationship to r^2_{ER} is unclear.

Unlike the above approaches, we follow a line of research [65, 87] that explicitly attempts to construct an unbiased estimator of r^2 in the absence of noise (see Methods, "Prior analytic methods of estimating r^2_{ER}"). Heretofore many of the methods reviewed above have not been quantitatively validated and none have been directly compared. We now compare all these methods with respect to estimating r^2_{ER}. We exclude from this comparison David and Gallant [89] because their method depends on a large number of repeated trials, at which point the estimators' utility decreases.

We quantified the ability of all methods to estimate r^2_{ER} in a simulation with $n = 4$ trials and $m = 362$ stimuli (see Methods, "Simulation procedure"). We sort the estimators (Figure 3.4 y-axis) by their MSE in a test case where $r^2_{\text{ER}} = 1$. We generally find, $\hat{r}^2_{\text{ER}}$, Y, SPE_{norm}, $\text{CC}^2_{\text{norm-SP}}$, FEVE, and $\text{CC}^2_{\text{norm-split}}$ are all comparable in their performance (red trace, top 6 points) with $\hat{r}^2_{\text{ER}}$ performing slightly, but significantly, better. SPE_{norm} and $\text{CC}^2_{\text{norm-SP}}$ are numerically identical in their performance and their trial-to-trial results. On the other hand, $\hat{r}^2(1 - \frac{SE^2}{SS_{total}})$ and $\hat{r}^2_{norm-split-SB}$ both over estimate r^2_{ER}, and the naive estimator $\hat{r}^2$, as expected, yields an under estimate (mean=0.50). In addition $\text{CC}^2_{\text{norm-PB}}$ underestimates r^2_{ER}. When the true r^2_{ER} is 0.5, we find similar results, where $\hat{r}^2(1 - \frac{SE^2}{SS_{\text{total}}})$ and $\hat{r}^2_{\text{norm-split-SB}}$ produce overestimates (0.63 and 1.04 on average, respectively) and the mean $\hat{r}^2$ is 0.25. Thus, serious caution should be taken when interpreting these last two estimators. We conclude $\hat{r}^2_{\text{ER}}$ is as

good as any estimator of r^2_{ER} available, has a simple analytic form, and in contrast to Y, can still be calculated without calculating the sample variance, for example, if no repeats are collected and variance must be assumed (see Discussion, "Relationship to prior methods"). None of the top five prior estimators we reviewed have associated confidence intervals, and thus we now provide confidence intervals for $\hat{r}^2_{ER}$.

3.3.5 Confidence intervals for $\hat{r}^2_{ER}$

In order to interpret point estimates such as $\hat{r}^2_{ER}$, it is important to be able to meaningfully quantify uncertainty about the estimate relative to the true parameter r^2_{ER}. An α-level confidence interval (CI) provides an interval that will contain the true parameter $\alpha \times 100\%$ of the time for IID estimates. We considered three typical generic approaches to forming CIs for $\hat{r}^2_{ER}$: the non-parametric bootstrap, the parametric bootstrap, and BC_a [91]. We found all methods to be lacking because they did not achieve the desired α in simulations with ground truth. Motivated by these problems, we developed a novel Bayesian method. We first recount the issues we found with the bootstrap methods and then provide a basic account of the Bayesian method we use throughout the paper. For more detailed exposition, see Methods: "Quantifying uncertainty in the estimator".

The non-parametric bootstrap is a commonly used method to approximate CIs. In our case, it involves randomly re-sampling with replacement from the n trials in response to each of the m stimuli then calculating $\hat{r}^{2(b)}_{ER}$ for the bootstrap sample. Repeating this many times allows the quantiles of the bootstrap distribution of $\hat{r}^{2(b)}_{ER}$ to be used as CIs. We applied this method across a simulated population of 3000 neurons with $m = 40$ and $n = 4$ and found it suffered from two problems. First, the CIs were not centered around r^2_{ER}, specifically the interval was too low (Figure 3.5A), with the upper and lower bounds of the interval (orange and blue traces, respectively) almost always falling below the true value (green). Secondly, as the true r^2_{ER} increased from 0 to 1, CIs contained r^2_{ER} at rates far lower than the desired $\alpha = 0.8$ (Figure 3.6 cyan trace, open-circles under the trace indicate a significant difference, $p < 0.01$ Bonferoni corrected z-test). Thus at practically all levels of correlation, the non-parametric bootstrap performs poorly. The problem is a result of the expected value of the empirical distribution (the sample mean) being typically much lower than r^2_{ER}. To overcome this, we turned to the parametric bootstrap where we could explicitly estimate r^2_{ER} with our estimator $\hat{r}^2_{ER}$. This method approximates CIs by estimating the parameters of an assumed distribution from which samples are generated. In our case it involves estimating σ^2, r^2_{ER}, and the variance of the neuronal tuning curve d^2 (see Results, "Signal-to-noise ratio as recording quality metric") and then simulating observations from the distribution with these parameters to calculate $\hat{r}^{2(PB)}_{ER}$. Drawing many $\hat{r}^{2(PB)}_{ER}$ we again use the sample quantiles as CI estimates. Figure 3.5B shows that this overcomes the main failure of the non-parametric bootstrap, but this method tended to be too conservative for low r^2_{ER} values (Figure 3.6 red trace below 0.8 at left side) and too liberal for high values (red trace above 0.8 at right side). Deviations such as these are well known for bootstrap percentile methods when the variance is a non-constant function of the mean and/or the distribution of the estimator is skewed [91]. The correction to the bootstrap, the bias-corrected and accelerated bootstrap (BC_a) , can help ameliorate these issues by implicitly approximating the skewness and the mean-variance relationship from bootstrap samples. We employed BC_a with our parametric bootstrap and found that

performance improved but still deviated from the desired α for low and high r^2_{ER} (Figure 3.6, green trace).

We aimed to create a CI with better α-level performance. To do this, we assumed uninformative priors on the parameters σ^2 and d^2 so that, conditioned on estimates of these parameters, we can draw from the distribution of $\hat{r}^2_{ER}|r^2_{ER}$ for an arbitrary r^2_{ER} (see Methods, "Confidence Intervals for $\hat{r}^2_{ER}$"). This allows us to compute the highest true r^2_{ER} that would have given an observed $\hat{r}^2_{ER}$ or a lower value in $\alpha/2$ proportion of IID samples. We take this as the high end, $r^2_{ER(h)}$, of our CI. Similarly we determine the low end, $r^2_{ER(l)}$, of the CI as the lowest r^2_{ER} that produces a value greater than or equal to $\hat{r}^2_{ER}$ in $\alpha/2$ of the samples. In Methods we give conditions under which this procedure will provide α-level CIs (see Methods, "Confidence intervals for $\hat{r}^2_{ER}$"). In our simulations, this method consistently achieves the desired α at all levels of r^2_{ER} (Figure 3.6, orange trace). We use this CI method, which we call the estimate-centered credible interval (ECCI), throughout the rest of the paper.

3.3.6 Application of estimator to MT data

We have shown in simulation that the use of $\hat{r}^2$ introduces ambiguity as to whether a low correlation value was the result of trial-to-trial variability or a poor model, whereas $\hat{r}^2_{ER}$ removes this ambiguity. Here we demonstrate in neural data how this, in tandem with confidence intervals, allows investigators to distinguish between units that systematically deviate from model predictions versus those that simply have noisy responses. We re-analyzed data from single neurons in the visual cortical motion area MT responding to dot motion in eight equally spaced directions [92, 93]. A classic model of these responses is a single cycle sinusoid as a function of the direction of dot motion with the free parameters phase, amplitude, and average firing rate. We chose this MT data set as the first example application because it has a high number of repeats ($n = 10$) and a low dimensional model, thus it is simple to visually inspect whether the neuronal tuning curves agree with the model predictions.

An example of a typical MT neuron direction tuning curve (Figure 3.7A, orange trace) has an excellent sinusoidal fit (blue trace), as reflected in its estimated $\hat{r}^2_{ER} = 1.0$. Furthermore, the short confidence interval ([0.99, 1.0]) quantifies the lack of ambiguity about the quality of the fit. On the other hand, the tuning curve of a neuron with $\hat{r}^2_{ER} = 0.05$ (Figure 3.7B) has a clear systematic deviation from the least-squares fit. Here the tuning curve is double-peaked and thus largely orthogonal to any single cycle sinusoid. It is important to notice that this neuron has far lower SNR (2.8 here vs. 20 for the example in A), as quantified by our estimator, $\widehat{SNR}$ (Eqn. 3.16), which corrects for trial-to-trial variability (described below and defined in Methods, "Estimators of correction terms"). Thus without r^2_{ER}, there would be plausible doubts about whether the correlation was lower because of noise or systematic deviation. Furthermore, with low SNR it would be plausible that the estimate itself is noisy (Figure 3.3A), but the short confidence interval ([0.01, 0.11]) unambiguously characterizes the fit as being systematically poor.

In some cases, neurons show little tuning for direction and thus have very low SNR over a set of directional stimuli. This in turn can cause $\hat{r}^2_{ER}$ to give wild estimates (Figure 3.7C, $\widehat{SNR}$=0.05, $\hat{r}^2_{ER} = 1.81$). If we truncate the value to the nearest possible $r^2_{ER} = 1$, we might be tempted to interpret this as a well-fit direction selective neuron. But, the CI covers most of the interval of possible values ([0.3, 1]),

making it clear that little information can be gleaned about r^2_{ER} from this data. Extreme $\hat{r}^2_{ER}$ values themselves can indicate when the estimator is unreliable, but even a reasonable seeming $\hat{r}^2_{ER}$ value, for example $\hat{r}^2_{ER} = 0.59$ (Figure 3.7D), can be unreliable when there is a low $\widehat{SNR}$ (0.12). In this case, the confidence interval covers the maximal range ([0,1]), indicating that the point estimate is unreliable. Thus, $\hat{r}^2_{ER}$ and its associated confidence interval quickly and unambiguously show how well the model fits the MT data, avoiding the tiresome and unreliable process of judging each fit by eye for the 162 neurons.

While we have shown to a good approximation that $\hat{r}^2_{ER}$ is unbiased and its expected value is largely invariant to SNR, this is definitely not the case for the variance of the estimator. Figure 3.3A shows clearly that the variability of the estimator is larger for lower SNR. This fact should be kept in mind when interpreting the spread of $\hat{r}^2_{ER}$ values. For example, we calculated $\hat{r}^2_{ER}$ and confidence intervals across our entire population of MT neurons. Of the estimates with high SNR (Figure 3.8, right side, $\widehat{SNR} > 3.5$), most neurons are well fit to the model and only a few have less than 3/4 of their variance accounted for (8/81). For the estimates with low SNR ($\widehat{SNR}$<3.5), left side of Figure 3.8), this fraction is substantially higher (39/81), but the increased variability of these estimates will spread out the distribution, thus this difference in quantiles must be interpreted carefully. When estimating population dispersion, conclusions may be confounded by the SNR-dependence of the variability of $\hat{r}^2_{ER}$.

Comparing the naive $\hat{r}^2$ to our unbiased $\hat{r}^2_{ER}$ (Figure 3.9), the high SNR units (red points), lie close to the diagonal. Thus for these units, one could exchange the two estimates and come to similar general conclusions about model fits. The utility of $\hat{r}^2_{ER}$ is that it removes ambiguity about whether trial-to-trial variability may be spuriously pushing fits down (black points). The interpretation of the naive estimator $\hat{r}^2$ remains ambiguous for any given unit until it can be confirmed it does not suffer from this issue.

The MT data considered here has relatively few stimuli and many repeats, but other experimental paradigms involve a larger number of stimuli and, consequently, fewer repeats. Below we apply $\hat{r}^2_{ER}$ in these more challenging conditions.

3.3.7 Application of estimator to V4 data

The primate mid-level visual cortical area V4 is known to have complex, high-dimensional selectivity for visual inputs. To rigorously assess models of neuronal responses in areas like V4, validation needs to be performed on responses to a large corpus of natural images to ensure that models capture ecologically valid selectivity [76, 94]. Thus, the number of unique stimuli, m, will be large at the expense of having relatively few repeats, n, and SNR can be low because stimuli are not customized to the preferences of a given neuron. These are the challenging conditions under which $\hat{r}^2_{ER}$ avoids the major confounds of $\hat{r}^2$. Here we estimate $\hat{r}^2_{ER}$ and associated 90% confidence intervals for a model that won the University of Washington V4 Neural Data Challenge by most accurately predicting single-unit (SUA) and multi-unit activity (MUA) for held-out stimuli (see Methods, "Electrophysiological data"). Plotting $\hat{r}^2_{ER}$ against $\hat{r}^2$ (Figure 3.10A) shows that the corrected estimates are higher than the naive estimates (points lie above diagonal line). Using $\hat{r}^2_{ER}$ here is important because it provides confidence that the poor fit quality is not a result of noise and that the best performing model often did not explain more than 50% of the variance in the tuning curve.

While we have examined $\hat{r}^2_{ER}$ for individual recordings, it can also be useful to estimate the average quality of model fit across a population of neurons. Since the

individual estimates are unbiased, the group average is also an unbiased estimate of the population mean $\hat{r}^2_{\text{ER}}$. We computed such group means for the single-unit and multi-unit V4 recordings (Figure 3.10B), and found that the model performed significantly better in predicting the responses of multi-unit activity (Welch's t-test p=0.005, MUA mean=0.57, SUA mean=0.35). If instead the naive $\hat{r}^2$ were used, this finding could have been dismissed as the result of MUA having higher SNR and thus naturally higher $\hat{r}^2$. As it stands, this interesting observation can be followed up to potentially gain insight about the structure of selectivity across multiple units recorded nearby in V4.

Finally, this V4 data set provides a good example of how using $\hat{r}^2_{\text{ER}}$ can allow testing a larger stimulus space, as predicted by simulations above in Figure 3.3B. Figure 3.11 shows that with $\hat{r}^2_{\text{ER}}$ (solid lines, on average two trials is enough to estimate the true correlation, whereas the naive estimator requires more repeats (higher n) to converge. For example, for recording 1 (red), $\hat{r}^2_{\text{ER}}$ (solid line) on average predicts the same quality of model fit for two or more stimulus repeats, whereas even after six repeats, the naive $\hat{r}^2$ has not converged.

3.3.8 Signal-to-noise ratio as recording quality metric

We have shown above that correcting for bias in r^2 is important, but it is also critical to recognize when recordings are so noisy that they are effectively useless for evaluating a model. Here we demonstrate the use of the signal-to-noise ratio (SNR) as a quality metric to help make this determination. We define the SNR for a neuronal tuning curve to be the ratio of the variation in the expected response across stimuli to the trial-to-trial variability across repeats:

$$\text{SNR} = \frac{\frac{1}{m}\sum_{i=1}^{m}(\mu_i - \bar{\mu})^2}{\sigma^2}, \tag{3.5}$$

where μ_i is the expected response to the ith stimulus and $\bar{\mu} = \frac{1}{m}\sum_{i=1}^{m}\mu_i$. For experimental data, we do not know μ_i in Eqn. 3.5, and rather than substituting sample estimates, Y_i, which would give an inflated estimate, we use an equation that corrects for trial-to-trial noise ($\widehat{\text{SNR}}$, Eqn. 3.16, Methods). If SNR is close enough to 0, then no reasonable number of unique stimuli (m) or repeats (n) will allow successful inference about any non-flat tuning of the neuron. SNR is not a function of n or m, rather it can vary across neurons, sets of stimuli and recording modalities, as we show below.

We examined a diverse collection of neural data sets (see Methods, Electrophysiological data) and found wide variation in $\widehat{\text{SNR}}$ both within and across the data sets (Figure 3.12A). At the low end, calcium imaging data from cortical neurons in area VISp of mouse responding to gratings (pink trace N=40,520 neuronal ROIs, [95]) had a median SNR of 0.01, while at the high end, MT neurons in response to dot motion [92] had a median $\widehat{\text{SNR}}$ of 3.5 (blue trace, N=162). A stimulus protocol nearly identical to that used for the VISp Ca^{2+} data (pink and gray traces for gratings and natural images, respectively) was used to collect the Allen Institute NeuroPixel electrode data [96] (purple and brown traces N=2,015); however, the Ca^{2+} data had a substantially lower $\widehat{\text{SNR}}$ (0.01 and 0.02) compared to the electrode data ($\widehat{\text{SNR}}$ 0.12 and 0.19), suggesting that this difference relates to the recording modality.

In the case of spiking neurons, SNR can be improved by increasing the stimulus duration and thus the spike counting window. Under the generally optimistic assumption that spike rate stays constant in the counting window, we can normalize

$\widehat{\text{SNR}}$ across the data-sets to what the $\widehat{\text{SNR}}$ would have been had all spike count windows been 1 second long (Figure 3.12B). This reduces the differences in $\widehat{\text{SNR}}$ across the spiking data-sets (the six right-most traces), thus the outstanding $\widehat{\text{SNR}}$ of the MT data-set could potentially have been achieved if spike count windows had been longer for the other experiments. Still, of the spiking data, the Allen Neuropixel data has the lowest medians, thus additional efforts to ameliorate low SNR (via number of trials or stimulus choice) could be utilized. Furthermore, the assumption of a constant spike rate will hold to different degrees: neural responses can peak shortly after stimulus onset and then return close to baseline. Thus, different experimental conditions call for different standards for number of trials and stimulus duration to adequately characterize a tuning curve.

To provide concrete meaning to $\widehat{\text{SNR}}$, we suggest interpreting it in terms of the number of trials (m and n) needed to reliably detect stimulus modulation in an F-test. Specifically, for a given m and n we computed the minimal SNR required to achieve a high probability ($\beta = 0.99$) of rejecting the null hypothesis that the mean response to all stimuli is the same (see Methods, SNR relation to F-test and number of trials, Eqn. 3.22). We plot a color map of this minimal SNR as a function of m and n (Figure 3.13), where the diagonal grey contour lines indicate fixed total number of trials (mn) for different $m : n$ ratios. In general, as the total number of trials increases (moving perpendicular to the grey diagonals toward the upper right), the SNR required for reliable tuning curve estimates decreases. The SNR threshold is lower when n is favored over m for the same number of total trials, i.e., the SNR threshold level iso-contours have steeper slopes than the grey diagonals.

On this map, we can locate points corresponding to the m and n, roughly, for data sets in Figure 3.12. The three V4 data sets have about the same number of stimuli and repeats (arrow marked "V4", Figure 3.13), and thus require SNR$\approx$0.1 or greater, implying that from 3% to 23% of the V4 data does not pass the criterion (Figure 3.12A, red and green traces, respectively, define endpoints). The MT data has the fewest number of total trials and thus has the highest threshold SNR$\approx$0.5, which leaves 10% of the neurons with poorly estimated tuning curves. If more stimuli had been used at the expense of fewer repeats, say $n = 2$ and $m = 40$, then only a quarter of the neurons would have exceeded the increased threshold of SNR$>$1. The Allen Ca^{2+} and spike data sets both had similar m and n. Relative to the other data sets they had far more total trials and a greater number of repeats, thus the SNR criterion is substantially lower (SNR$>$0.01). Still, for the Ca^{2+} data, $\sim$ 37% of the grating and $\sim$ 25% of the natural image data did not have reliable tuning (Figure 3.12A, pink and grey thin trace). The Allen spiking data on the other hand had much higher SNR, and thus more trials could have been spent on expanding the stimulus set and fewer on repeated presentation (Figure 3.12A, thin brown purple trace).

We have shown SNR can be employed as a simple metric with a concrete interpretation to judge data quality across different organisms, recording modalities and brain regions for the purpose of making comparative analyses and setting aside data that has little or no power. The expected distribution of SNR, based on prior data, can be taken into account when choosing n and m to achieve a criterion level of statistical power for an experiment. If SNR is high, recording time can be reduced by keeping n and m low, or a larger stimulus set can be explored by increasing m at the expense of n.

3.4 Discussion

3.4.1 Summary

We have investigated the estimation of the correlation between a model prediction and the expected response of a neuron. Although it has been long established that trial-to-trial variability will cause the classic estimator, Pearson's $\hat{r}^2$, to underestimate correlation, there has been no direct comparison of prior methods to account for this confound. We found that some methods grossly over estimate correlation in high noise conditions, and we built upon the best performing method to derive a more generally applicable estimator, $\hat{r}^2_{\text{ER}}$, that performs as well as or better than prior methods. We analytically validated $\hat{r}^2_{\text{ER}}$ by determining that it was a consistent estimator in the number of stimuli. We found in simulation that it had a small upward bias, but this was only appreciable at very high noise levels. None of the prior methods that we examined had validated confidence intervals, thus we developed confidence intervals for $\hat{r}^2_{\text{ER}}$. Motivated by the failure of generic bootstrap methods to achieve satisfactory confidence intervals, we developed a confidence interval method that outperformed them.

Applying our estimator to neural data, we demonstrated its essential value. In the case of MT recordings, it was able to unambiguously distinguish neurons for which a sinusoidal model was a good fit from those for which it was a poor fit specifically because of systematic deviation and not because of noise. The associated confidence intervals allowed the systematic identification of noisy recordings that served no practical use in assessing the fit of the model. Poor model fits caused by noise vs. those caused by systematic differences in selectivity have very different interpretations, yet the traditional $\hat{r}^2$ does not differentiate them while $\hat{r}^2_{\text{ER}}$ does.

Application of the estimator to the winning UW neural data challenge model, a deep neural network (DNN), provides the only validated assessment of state-of-the-art predictive model performance in V4. The estimator along with its CIs identified neurons that were challenging to the DNN and perhaps require a different modeling approach. It also validated the existence of single units that had nearly 50% of their variance explained, indicating that the DNN functionally captured a substantial part of what these units encode across natural images and thus could provide real insight into naturalistic V4 single unit encoding. On a practical level, we showed how the estimator allows for gains in trial efficiency since it converges more rapidly than $\hat{r}^2$ (Figure 3.3B and Figure 3.11). This is important when many stimuli are needed to validate models of high dimensional neural tuning.

Our tests on experimental data revealed that some neurons had confidence intervals covering the entire range of possible values, motivating us to propose the signal-to-noise ratio (SNR) as a metric of neural recording quality in the context of model fitting. We provide an unbiased estimator of SNR (Eqn. 3.16) and a practical interpretation: for a given number of stimuli and repeats, the SNR should be sufficient to reliably detect stimulus-driven response modulation on the basis of an F-test (Eqn. 3.22). Examining a variety of data sets, we found differences with respect to how the numbers of stimuli vs. repeats (m vs. n) were balanced, revealing how adjustments can be made on the basis of SNR to improve experimental efficiency. We also found large differences in SNR across data sets that are likely related to recording modality (e.g., Ca2+ imaging vs. electrode recording), which could be important for selecting appropriate experimental approaches and for guiding the refinement of current techniques to improve SNR.

3.4.2 Interpretation of r^2_{ER}

We have introduced an estimator and confidence intervals for the correlation between the true tuning curve of a neuron (its expected value across stimuli) and model predictions: $\hat{r}^2_{ER}$. In the context of sensory neurophysiology, we believe it is reasonable to think of r^2_{ER} as reflecting solely how well a model explains a sensory representation. We justify this by the fact that r^2_{ER} is solely a function of E[Response|Stimulus] thus solely a function of stimuli. We note two caveats: (1) non-sensory signals can influence sensory responses, e.g., eye movements which may be stimulus dependent and (2) E[Response|Stimulus] is not the only component of the sensory response, e.g., variability can also be stimulus dependent [17].

3.4.3 Relationship of $\hat{r}^2_{ER}$ to Y

We have taken a similar approach to Haefner and Cumming [65], though we provide an important generalization: their formula requires the calculation of the sample variance because their derivation relies on the F-distribution formed by taking the ratio of the sum of squares of model residual over the sample variance (see Methods, Y). This is problematic if stimuli are never repeated in an experiment (for example, in free viewing experiments), then one has to assume a priori the trial-to-trial variability either from previous experimental measurements or by asserting a theoretical mean-variance relationship (e.g., the square root of Poisson distributed spiking gives $\sigma^2 = \frac{1}{4}$).

Haefner and Cumming's estimator is more general than the $\hat{r}^2_{ER}$ we have presented: their Y estimates variance explained for a linear model, ours estimates Pearson's r^2. Ours can be seen as a special case of theirs, being identical when estimating the fit of a linear model with a slope and intercept. We also provide the more general version of our estimator for the case of variance explained by a linear model (Eqn. 3.23) with the advantage discussed in the previous paragraph.

3.4.4 SNR

We found that differences in SNR can be substantial and widely varying across neurons, data sets, and recording methods. Given the rise in large scale recordings and sharing of neuronal data, we believe unbiased estimates of SNR should be reported so that researchers can quickly judge whether a data set has sufficient statistical power or whether its power is on par with that of data sets from potentially comparable studies. We provide concrete criteria by which to interpret SNR: the statistical power to detect stimulus-driven response modulation. Strikingly, in our small sample of data sets, many neurons do not pass this criteria, suggesting that the adoption of a standard criterion for data quality, such as our SNR metric, could have a major impact in practice. Furthermore, guided by the metric the experimentalist can take steps to improve SNR by increasing stimulus duration and associated spike counting windows or by customizing stimuli to the preferences of a neuron. On the other hand, the deleterious effects of low SNR can be ameliorated by favoring repeats over number of stimuli (Figure 3.13).

One conceptual interpretation of the SNR metric we introduced is that it quantifies, for the time scale of the spike count window, the overall variance in the responses of the neuron attributable to the tuning curve of the neuron vs. trial-to-trial variability about that tuning curve. For example on the time scale of 1 second, a large fraction of spike-based recordings had SNR>1, indicating that more variance was caused by the stimulus than by other sources (Figure 3.12B blue, orange, green

traces median SNR>1). Still, an appreciable number of neurons were dominated by their trial-to-trial variability. Whether this is the result of stimulus choice and perhaps would be different in a more natural context is an open question. Recent theoretical and experimental work has argued that weakly tuned and untuned neurons can contribute to sensory encoding [97–99]. The corrected estimate of SNR we provide (Eqn. 3.16) along with naturalistic stimulation can help to identify such neurons.

3.4.5 Further work

Small improvements to our $\hat{r}^2_{\mathrm{ER}}$ estimator could be made by decreasing its bias in the case of very low SNR (see Methods, "Bias of $\hat{r}^2_{\mathrm{ER}}$"). In the case of very low SNR, a single neuronal recording has little inferential power, but across a population of neurons, estimates of the average correlation to a model's predictions can have low enough variance to provide useful inference. Yet, at very low SNR an appreciable bias begins to appear that will remain in the population average. We showed this bias is the result of the covariance between the numerator ($\hat{C}_{\mathrm{ER}_m}$) and inverse of the denominator ($\frac{1}{\hat{V}_{\mathrm{ER}_m}}$) of $\hat{r}^2_{\mathrm{ER}}$ and Jensen's gap where $\mathrm{E}[\frac{1}{\hat{V}_{\mathrm{ER}_m}}] > \frac{1}{\mathrm{E}[\hat{V}_{\mathrm{ER}_m}]}$ (Eqn. 3.17). The former covariance can be removed by using separate subsets of the data for estimation of the numerator and denominator. We find in simulation that using separate subsets actually increases bias, even accounting for the change in trial number. We explain this as follows: numerator and denominator covariance is typically negative because they share terms and the denominator is inverted. This negative covariance cancels the positive Jensen's gap. To reduce the influence of Jensen's gap, further work could attempt to directly estimate and correct for its value.

Here we have derived an estimator for the case where deterministic model predictions are correlated to a noisy signal. Often, one noisy signal is correlated to another, for example when judging the similarity of tuning curves from two neurons (termed signal correlation). We have extended the methods described here to the neuron-to-neuron case and will describe this in a forthcoming publication.

A subtle but important point about our estimator is that it assumes stimuli are fixed: it estimates the r^2_{ER} for the exact stimuli shown. An investigator may be interested in the quality of a model across a large corpus of natural images of which only a small fraction can be included in a recording session. In this case, one collects responses for a random sample of images, fits the model to some (training set) and tests the model on others (test set). The random test set will account for over-fitting and using $\hat{r}^2_{\mathrm{ER}}$ will account for noise in the neural responses in the evaluation on the test set. Crucially though, this does not account for the variability in the parameters of the model induced by the random training sample. Intuitively, estimated model parameters will vary across image sets even in the absence of trial-to-trial variability. The correct interpretation of $\hat{r}^2_{\mathrm{ER}}$ in this case is that it estimates how well a model can perform given finite noisy training data on noiseless test set data, and *not* as the best the model could possibly perform given infinite training data. Indeed, with more neural responses and less noise, model test set performance would improve. David and Gallant [89] explored this issue calling it 'estimation noise' and provided an extrapolation method for estimating the fit of a linear model given unlimited stimuli. The estimator was not evaluated in terms of its bias or variance, and no analytic solutions that directly remove the bias of finite training data have been proposed. Both are valuable directions to pursue: the former to build confidence in the current method and the latter for potential gains in trial efficiency. A data driven re-sampling approach may be unavoidable when evaluating complex

models where the relationship between the amount of training samples and model performance would be analytically intractable, such as a deep neural network or biophysical model.

3.5 Materials and Methods

3.5.1 Simulation procedure

To simulate model-to-neuron fits, the square root of neural responses, $r_{i,j}$, for the ith of m stimuli and the jth of n trials are modeled as independent normally distributed responses:

$$Y_{i,j} \sim N[\mu_i, \sigma^2], \tag{3.6}$$

where variance σ^2 is the same across all $Y_{i,j}$. The mean response of the neuron to the ith stimulus (tuning curve) is $\mu_i = a + b \sin(\frac{(i-1)2\pi}{m} + \theta)$ (Figure 1A, green trace solid dots) whose correlation to the model predictions $\mu_i = \sin(\frac{(i-1)2\pi}{m})$ (red trace solid dots) are estimated, and the true correlation is $r^2_{\mathrm{ER}} = \cos^2(\theta)$. The results of the simulation are only a function of the magnitude of the centered vector of expected responses d^2 the correlation between model prediction and tuning curves, m, n, and σ^2 thus the form of the model and true tuning curve is arbitrary. We choose a sinusoid for the simplicity of adjusting the phase, θ, to simulate different r^2_{ER}.

From this model we draw n responses for each of the m stimuli and apply our estimator to this sample. We repeat this across many IID simulations to accumulate reliable statistics.

3.5.2 Assumptions and terminology for derivation of unbiased estimators

Below we derive an unbiased estimator of the fraction of variance explained when a known signal is being fit to noisy neural responses. For this derivation, we assume the responses have undergone a variance stabilizing transform such that trial-to-trial variability is the same across all stimuli. For example, if the neural responses are Poisson distributed, $Y_{i,j} \sim P(\lambda_i)$, where $Y_{i,j}$ is the response to the jth repeat of the ith stimulus, which has expected response λ_i, then a variance stabilizing transform is the square root. In particular, if $Y^*_{i,j} = \sqrt{Y_{i,j}}$, then,

$$\mathrm{E}[Y^*_{i,j}] = \mathrm{E}[\sqrt{P(\lambda_i)}] \approx \sqrt{\lambda_i},$$

and

$$\mathrm{Var}[Y^*_{i,j}] = \mathrm{Var}[\sqrt{P(\lambda_i)}] \approx \frac{1}{4}.$$

The expected value of the transformed response, $Y^*_{i,j}$, still increases with λ_i, whereas the variance is now approximately constant. To improve the estimate of the mean response, n repeats of each stimulus are collected. Invoking the central limit theorem, we can make the approximation:

$$\frac{1}{n}\sum_{j=1}^{n} Y^*_{i,j} = \bar{Y}^*_i \sim N(\sqrt{\lambda_i}, \frac{1}{4n}),$$

where the average across the n repeats is approximately normally distributed with variance decreasing with n. The assumption of a Poisson distributed neural response is not always accurate. A more general mean-to-variance relationship,

$$\sigma^2(\mu) = a\mu^b,$$

can be approximately stabilized to 1 by,

$$f(x) = [\sqrt{a}(1 - \frac{1}{2}b)]^{-1}x^{1-\frac{1}{2}b}.$$

A square root will stabilize any linear mean-to-variance relationship ($b = 1$), but an unknown slope, a, requires that this parameter be estimated. In the case of the linear relationship, this simply requires taking a square root and then averaging the estimated variance, which is constant, across all stimuli. If it is not reasonable to assume a parametric mean-to-variance relationship and there are enough repeats, one can simply divide all responses to a given stimulus by their sample standard deviation to achieve $\sigma^2 \approx 1$. For the derivation below, we assume that variance-stabilized responses to n repeats have been averaged for each of m stimuli to yield the mean response to the ith stimulus: $Y_i \sim N(\mu_i, \frac{\sigma^2}{n})$, where σ^2 is the trial-to-trial variability and μ_i the ith expected value.

3.5.3 Unbiased estimation of r^2

Given a set of mean neural responses, Y_i, and model predictions normalized to unit length, ν_i, the naive estimator, $\hat{r}^2$, is calculated as follows:

$$\hat{r}^2 = \frac{\left(\sum_{i=1}^m (\nu_i - \bar{\nu})(Y_i - \bar{Y})\right)^2}{\sum_{i=1}^m (\nu_i - \bar{\nu})^2 \sum_{i=1}^m (Y_i - \bar{Y})^2}. \tag{3.7}$$

Our goal is to find an estimator such that,

$$\mathrm{E}[\hat{r}^2_{\mathrm{ER}}] = r^2_{\mathrm{ER}} = \frac{\left(\sum_{i=1}^m (\nu_i - \bar{\nu})(\mu_i - \bar{\mu})\right)^2}{\sum_{i=1}^m (\nu_i - \bar{\nu})^2 \sum_{i=1}^m (\mu_i - \bar{\mu})^2}, \tag{3.8}$$

where r^2_{ER} is the correlation in the absence of noise, i.e., the fraction of variance explained by the model prediction, ν, of the expected response (ER), μ_i, of the neuron. Our strategy will be to remove the bias in the numerator and denominator separately and then reform the ratio of these unbiased estimators for an approximately unbiased estimator.

Unbiased estimate of numerator.

The numerator of Eqn. 3.7, which we call $\hat{C}_m$, is a weighted sum of normal random variables that is then squared, thus it has a scaled non-central chi-squared distribution:

$$\hat{C}_m = \left(\sum_{i=1}^m (\nu_i - \bar{\nu})(Y_i - \bar{Y})\right)^2 \sim \frac{\sigma^2}{n}\sum_{i=1}^m (\nu_i - \bar{\nu})^2 \chi^2_1\left(\frac{\left(\sum_{i=1}^m (\nu_i - \bar{\nu})(\mu_{i=1} - \bar{\mu})\right)^2}{\frac{\sigma^2}{n}\sum_{i=1}^m (\nu_i - \bar{\nu})^2}\right), \tag{3.9}$$

and since $\mathrm{E}[\chi^2_m(\lambda)] = \lambda + m$ its expectation is:

$$\begin{aligned}
\mathrm{E}[\hat{C}_m] &= \frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2\mathrm{E}\left[\chi_1^2\left(\frac{\left(\sum_{i=1}^{m}(\nu_i-\bar{\nu})(\mu_i-\bar{\mu})\right)^2}{\frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2}\right)\right] \\
&= \frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2\left(\frac{\left(\sum_{i=1}^{m}(\nu_i-\bar{\nu})(\mu_i-\bar{\mu})\right)^2}{\frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2}+1\right) \\
&= \left(\sum_{i=1}^{m}(\nu_i-\bar{\nu})(\mu_i-\bar{\mu})\right)^2+\frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2. \qquad (3.10)
\end{aligned}$$

In the final line, the term on the left is the desired numerator and the term on the right the bias contributed by σ^2. To form our estimator, $\hat{C}_{\mathrm{ER}_m}$, for the numerator of Eqn. 3.15, we simply subtract an unbiased estimator of this bias term from the numerator of the naive estimator 4.2:

$$\hat{C}_{\mathrm{ER}_m} = (\sum_{i=1}^{m}(\nu_i-\bar{\nu})(Y_i-\bar{Y}))^2-\frac{\hat{\sigma}^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2, \qquad (3.11)$$

where $\hat{\sigma}^2$ is typically the sample variance, s^2, estimated from the data, but it can be any unbiased estimator, even an assumed constant. For example, if stimuli are not repeated (i.e., $n = 1$) and one is willing to assume that responses are Poisson distributed, then the square root of these responses will give $\sigma^2 = \frac{1}{4}$ and thus one can substitute $\hat{\sigma}^2 = \frac{1}{4}$. The case for the denominator is similar.

Unbiased estimate of denominator

. The denominator of Eqn. 3.7, which we call $\hat{V}_m$, is a weighted sum of squared normal random variables and thus also follows a scaled non-central chi-squared distribution:

$$\sum_{i}^{m}(\nu_i-\bar{\nu})^2\sum_{i}^{m}(Y_i-\bar{Y})^2 \sim \frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2\chi_{m-1}^2\left(\frac{\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2}{\frac{\sigma^2}{n}}\right), \qquad (3.12)$$

with expectation,

$$\begin{aligned}
\mathrm{E}[\hat{V}_m] &= \frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2\mathrm{E}\left[\chi_{m-1}^2\left(\frac{\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2}{\frac{\sigma^2}{n}}\right)\right] \\
&= \sum_{i=1}^{m}(\nu_i-\bar{\nu})^2\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2+(m-1)\frac{\sigma^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2. \qquad (3.13)
\end{aligned}$$

Similarly to the numerator, the first term is the desired denominator, and the second term is the bias. Thus, we subtract an unbiased estimate of this second term from the naive denominator:

$$\hat{V}_{\mathrm{ER}_m} = \sum_{i=1}^{m}(\nu_i-\bar{\nu})^2\sum_{i=1}^{m}(Y_i-\bar{Y})^2-(m-1)\frac{\hat{\sigma}^2}{n}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2. \qquad (3.14)$$

Taking the ratio of these two unbiased estimators (Eqns. 3.11 and 3.14) we have:

$$\hat{r}^2_{\mathrm{ER}} = \frac{\hat{C}_{\mathrm{ER_m}}}{\hat{V}_{\mathrm{ER_m}}} = \frac{(\sum_i^m (\nu_i - \bar{\nu})(Y_i - \bar{Y}))^2 - \frac{\hat{\sigma}^2}{n}\sum_{i=1}^m (\nu_i - \bar{\nu})^2}{\sum_{i=1}^m (\nu_i - \bar{\nu})^2 \sum_{i=1}^m (Y_i - \bar{Y})^2 - \frac{\hat{\sigma}^2}{n}\sum_{i=1}^m (\nu_i - \bar{\nu})^2 (m-1)}. \tag{3.15}$$

This equation can be further simplified by scaling the model predictions such that $\sum_{i=1}^m (\nu_i - \bar{\nu})^2 = 1$.

Estimators of correction terms.

Two important parameters, $d^2 = \frac{1}{m}\sum_{i=1}^m (\mu_i - \bar{\mu})^2$ and σ^2, are unknown. Below we provide unbiased estimators of each of these terms. An unbiased estimate of sample variance for trials of the ith stimulus is $s_i^2 = \frac{1}{n-1}\sum_{j=1}^n (Y_{i,j} - \bar{Y}_{i,\cdot})^2$, where the dot in the subscript of $\bar{Y}_{i,\cdot}$ indicates the mean over repeats. Assuming the variance is the same across stimuli, we can average over i for a global estimate:

$$s^2 = \frac{1}{m}\sum_{i=1}^m s_i^2.$$

Throughout the paper we use this as our estimate of trial-to-trial variability $\hat{\sigma}^2$.

For d^2 we have:

$$\mathrm{E}[\frac{1}{m}\sum_{i=1}^m (Y_i - \bar{Y})^2] = \frac{1}{m}\mathrm{E}\left[\frac{\sigma^2}{n}\chi^2_{m-1}\left(\frac{n}{\sigma^2}\sum_i^m (\mu_i - \bar{\mu})^2\right)\right] = \frac{1}{m}\left(\sum_i^m (\mu_i - \bar{\mu})^2 + (m-1)\frac{\sigma^2}{n}\right),$$

which would be inflated by trial-to-trial variability, so as an unbiased estimator we use,

$$\hat{d}^2_{\mathrm{ER}} = \frac{1}{m}\left(\sum_{i=1}^m (Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma^2}}{n}\right).$$

We use this estimator to correct the estimate of SNR (Eqn. 3.5) for trial-to-trial variability as follows:

$$\widehat{\mathrm{SNR}} = \frac{\hat{d}^2_{\mathrm{ER}}}{\hat{\sigma}^2}. \tag{3.16}$$

3.5.4 Bias of $\hat{r}^2_{\mathrm{ER}}$

To remove the bias of Pearson's $\hat{r}^2$, we follow the approach of subtracting off its effect in the numerator and denominator. Prior work has not examined the potential problem with this approach: the expectation of a non-linear transformation of a set of random variables is not necessarily the transformation of their expected values. In this particular case, the expectation of the ratio is not necessarily the ratio of the expectations: $\mathrm{E}[\hat{C}_{\mathrm{ER}_m}/\hat{V}_{\mathrm{ER}_m}] \neq \mathrm{E}[\hat{C}_{\mathrm{ER}_m}]/\mathrm{E}[\hat{V}_{\mathrm{ER}_m}]$. Thus even though we have removed the bias in the numerator and denominator, it does not imply their ratio is unbiased. Calculating the expectation of the ratio we see the conditions under which it will be unbiased:

$$\mathrm{E}[\hat{C}_{\mathrm{ER}_m}/\hat{V}_{\mathrm{ER}_m}] = \mathrm{E}[\hat{C}_{\mathrm{ER}_m}\frac{1}{\hat{V}_{\mathrm{ER}_m}}] = \mathrm{Cov}[\hat{C}_{\mathrm{ER}_m}, \frac{1}{\hat{V}_{\mathrm{ER}_m}}] + \mathrm{E}[\hat{C}_{\mathrm{ER}_m}]\mathrm{E}[\frac{1}{\hat{V}_{\mathrm{ER}_m}}]. \tag{3.17}$$

Thus, $\hat{r}^2_{\text{ER}}$ is unbiased if $\text{Cov}[\hat{C}_{\text{ER}_m}, \frac{1}{\hat{V}_{\text{ER}_m}}] = 0$ and $\text{E}[\frac{1}{\hat{V}_{\text{ER}_m}}] = \frac{1}{\text{E}[\hat{V}_{\text{ER}_m}]}$, but we find in simulation often $\text{Cov}[\hat{C}_{\text{ER}_m}, \frac{1}{\hat{V}_{\text{ER}_m}}] \neq 0$ and by Jensen's inequality $\text{E}[\frac{1}{\hat{V}_{\text{ER}_m}}] \geq \frac{1}{\text{E}[\hat{V}_{\text{ER}_m}]}$.

Thus if the estimator $\hat{r}^2_{\text{ER}}$ is not unbiased for r^2_{ER} what recommends it over the naive $\hat{r}^2$? While we mainly focused on how in simulation for typical ranges of parameters it has a lower bias (Figure 3.3) it also has a theoretical justification. As we saw in simulation, as the number of stimuli, m, increases, its bias diminishes while that of $\hat{r}^2$ does not (Figure 3.3C). Convergence to the parameter of interest, otherwise known as consistency, gives a theoretical justification for an estimator. Below we show that $\hat{r}^2_{\text{ER}}$ is consistent for r^2_{ER}while $\hat{r}^2$ is not.

We note that the covariance in Eqn. 3.17 can be removed by using separate subsets of the data for the estimation of $\hat{C}_{\text{ER}_m}$ and $\hat{V}_{\text{ER}_m}$. This leaves the inflation by Jensen's inequality $\text{E}[\frac{1}{\hat{V}_{\text{ER}_m}}] \geq \frac{1}{\text{E}[\hat{V}_{\text{ER}_m}]}$, which could be estimated and corrected for via a simulation-based method such as the parametric bootstrap (see Discussion, "Further work").

Consistency of $\hat{r}^2_{\text{ER}}$ in m.

We aim to show that $\hat{r}^2_{\text{ER}}$ is consistent for r^2_{ER} in m, more formally:

$$\hat{r}^2_{\text{ER}} \xrightarrow{p} r^2_{\text{ER}} \equiv \lim_{m\to\infty} P(|\hat{r}^2_{\text{ER}} - r^2_{\text{ER}}| \geq \epsilon) = 0.$$

We make use of the continuous mapping theorem that guarantees if a random vector $X_m \xrightarrow{p} \vec{c}$, then for a continuous transformation g, $g(X_m) \xrightarrow{p} g(\vec{c})$ where the random vector is almost surely different from any discontinuity points. Taking our random vector to be, $[\hat{C}_{\text{ER}_m}, \hat{V}_{\text{ER}_m}]^T$, and our continuous transformation to be, $g([\hat{C}_{\text{ER}_m}, \hat{V}_{\text{ER}_m}]^T) = \frac{\hat{C}_{\text{ER}_m}}{\hat{V}_{\text{ER}_m}} = \hat{r}^2_{\text{ER}}$ (assuming expectation of the denominator is non-zero), it then suffices to show that $\hat{C}_{\text{ER}_m}$ and $\hat{V}_{\text{ER}_m}$ themselves are consistent estimators for the numerator and denominator of r^2_{ER}.

First, we have already shown that $\hat{C}_{\text{ER}_m}$ and $\hat{V}_{\text{ER}_m}$ are unbiased estimators. Next, we must show that their variance is decreasing with m, and then via Chebyshev's inequality,

$$P(|X - \mu| \geq \epsilon) \leq \frac{\text{Var}[X]}{\epsilon^2},$$

we can show their convergence to their expectation. Here we consider the case where $\hat{\sigma}^2 = s^2$. Since the model predictions (ν_i) are fixed for the purpose of the proof, we assume the dot product between model predictions and neural responses is scaled linearly by m:

$$\frac{1}{m}(\sum_i^m (\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2 = c,$$

as is the dynamic range of the neuron:

$$\frac{1}{m}\sum_{i=1}^m (\mu_i - \bar{\mu})^2 = v,$$

and we scale the numerator and denominator of $\hat{r}^2_{\text{ER}}$ by $\frac{1}{m}$ which makes no change to their ratio.

The numerator, $\hat{C}_{\mathrm{ER}_m} = \frac{1}{m}[(\sum_i^m (\nu_i - \bar{\nu})(Y_i - \bar{Y}))^2 - \frac{s^2}{n}]$, has variance equal to the sum of the variance of its first and second term (since they are independent). Since $\mathrm{Var}[\chi^2_m(\lambda)] = 2m + 4\lambda$ the variances are, respectively,

$$\mathrm{Var}\left[\frac{1}{m}(\sum_{i=1}^m (\nu_i - \bar{\nu})(Y_i - \bar{Y}))^2\right] = \mathrm{Var}\left[\frac{\sigma^2}{nm}\chi^2_1\left(\frac{(\sum_{i=1}^m (\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2}{\sigma^2/n}\right)\right] = \frac{2\sigma^4}{n^2m^2} + \frac{4\sigma^2 c}{mn},$$

and

$$\mathrm{Var}\left[\frac{s^2}{nm}\right] = \frac{1}{n^2m^2}\frac{2\sigma^4}{mn-1},$$

thus

$$\mathrm{Var}[\hat{C}_{\mathrm{ER}_m}] = \frac{2\sigma^4}{n^2m^2} + \frac{4\sigma^2 c}{mn} + \frac{1}{n^2m^2}\frac{2\sigma^4}{mn-1}.$$

The denominator, $\hat{V}_{\mathrm{ER}_m} = \frac{1}{m}(\sum_{i=1}^m (Y_i - \bar{Y})^2 - (m-1)\frac{s^2}{n})$, also has variance equal to the sum of the variance of its first and second term (by independence). The variances are, respectively,

$$\mathrm{Var}\left[\frac{1}{m}\sum_{i=1}^m (Y_i - \bar{Y})^2\right] = \mathrm{Var}\left[\frac{\sigma^2}{nm}\chi^2_{m-1}\left(\frac{\sum_{i=1}^m (\mu_i - \bar{\mu})^2}{\frac{\sigma^2}{n}}\right)\right] = \frac{2\sigma^4(m-1)}{n^2m^2} + \frac{4\sigma^2 v}{mn},$$

and

$$\mathrm{Var}\left[\frac{(m-1)}{nm}s^2\right] = \frac{(m-1)}{nm}\frac{\sigma^4}{mn-1},$$

thus

$$\mathrm{Var}[\hat{V}_{\mathrm{ER}_m}] = \frac{2\sigma^4(m-1)}{n^2m^2} + \frac{4\sigma^2 v}{mn} + \frac{(m-1)}{nm}\frac{\sigma^4}{mn-1}.$$

For both $\mathrm{Var}[\hat{V}_{\mathrm{ER}_m}]$ and $\mathrm{Var}[\hat{C}_{\mathrm{ER}_m}]$, all but m is constant; therefore, we can find an m to scale variance below any given ϵ. So by Chebyshev's inequality we have:

$$P(|\hat{C}_{\mathrm{ER}_m} - c| \geq \epsilon) \leq \frac{\mathrm{Var}[\hat{C}_{\mathrm{ER}_m}]}{\epsilon^2}.$$

Since as $m \to \infty$, $\frac{\sigma^2_{\hat{C}_{\mathrm{ER}_m}}}{m\epsilon^2} \to 0$ we have that,

$$\lim_{m\to\infty} P(|\hat{C}_{\mathrm{ER}_m} - c| > \epsilon) = 0 \equiv \hat{C}_{\mathrm{ER}_m} \xrightarrow{p} c,$$

and similarly,

$$\hat{V}_{\mathrm{ER}_m} \xrightarrow{p} v.$$

Thus by the continuous mapping theorem:

$$\hat{r}^2_{\mathrm{ER}} = \frac{\hat{C}_{\mathrm{ER}_m}}{\hat{V}_{\mathrm{ER}_m}} \xrightarrow{p} \frac{c}{v} = r^2_{\mathrm{ER}}.$$

In contrast, we show below that the naive estimator is not consistent and provide insight into when the difference between $\hat{r}^2$ and $\hat{r}^2_{\text{ER}}$ is large.

Inconsistency of $\hat{r}^2$ in m.

Similarly to the previous derivation, we can take the numerator and denominator of $\hat{r}^2$ (Eqn. 3.9, 3.12), scale by $\frac{1}{m}$, find their expected values, and in turn find the asymptotic value of $\hat{r}^2$. Here, though, we simplify by setting the model to be unit length,

$$\hat{r}^2_m \xrightarrow{p} \frac{c}{v + \frac{\sigma^2}{n}} \leq \frac{c}{v}.$$

This result shows that $\hat{r}^2$ is not a consistent estimator in m of r^2_{ER}.

3.5.5 Confidence intervals

Here we develop and prove a method that provides α-level confidence intervals for the estimator $\hat{r}^2_{\text{ER}}$. We considered the typical parametric bootstrap and non-parametric bootstrap approaches, but found that they were not reliable for typical ranges of parameters (see Results, "Confidence intervals for $\hat{r}^2_{\text{ER}}$").

Our approach hinges upon finding the lowest r^2_{ER} whose distribution would give an estimate greater than the observed $\hat{r}^{2*}_{\text{ER}}$ with probability $\alpha/2$, calling this $r^2_{\text{ER}(l)}$, and finding the highest r^2_{ER} that would give an estimate less then the observed $\hat{r}^{2*}_{\text{ER}}$ with probability $\alpha/2$, calling this $r^2_{\text{ER(h)}}$. The interval $[r^2_{\text{ER}(l)}, r^2_{\text{ER}(h)}]$ then serves as our α-level confidence interval (see Figure 3.14 for graphical explanation). We use a Bayesian framework to sample from the probability distribution, $f(\hat{r}^2_{\text{ER}}|r^2_{\text{ER}})$, parameterized by the observed neural statistics s^2 and $\hat{d}^2$, allowing us to find $[r^2_{\text{ER(l)}}, r^2_{\text{ER(h)}}]$ under assumed uninformative priors on σ^2 and d^2 (see "Computing confidence intervals", below).

Proof of α-level confidence intervals.

Here, we justify this procedure for the case of $r^2_{\text{ER}(h)}$ ($r^2_{\text{ER}(l)}$ is similar). Our two main assumptions are that the cumulative distribution $\text{F}\{\hat{r}^2_{\text{ER}}|r^2_{\text{ER}}\}$ is stochastically increasing in r^2_{ER},

$$r^2_{\text{ER}} \leq r^{2'}_{\text{ER}} \Leftrightarrow F(\hat{r}^2_{\text{ER}}|r^2_{\text{ER}}) \geq F(\hat{r}^2_{\text{ER}}|r^{2'}_{\text{ER}}), \tag{3.18}$$

and that we can always find an r^2_{ER} such that for any observed $\hat{r}^{2*}_{\text{ER}}$,

$$F(\hat{r}^{2*}_{\text{ER}}|r^2_{\text{ER}}) = \alpha \in (0,1). \tag{3.19}$$

We now consider two mutually exclusive possibilities. First, with probability $1 - \alpha/2$, the observed $\hat{r}^{2*}_{\text{ER}}$ is large enough to satisfy:

$$F(\hat{r}^{2*}_{\text{ER}}|r^2_{\text{ER}}) \geq \alpha/2.$$

Then by the assumption in Eqn. 3.19 we can find a $r^2_{\text{ER}(h)}$ where,

$$F(\hat{r}^{2*}_{\text{ER}}|r^2_{\text{ER}(h)}) = \alpha/2,$$

and under our initial assumption (Eqn. 3.18), this implies

$$r^2_{\mathrm{ER}} \leq r^2_{\mathrm{ER}(h)},$$

because

$$F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}}) \geq \alpha/2 = F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}(h)}).$$

Second, if on the other hand $\hat{r}^{2*}_{\mathrm{ER}}$ is small enough such that

$$F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}}) < \alpha/2,$$

then

$$r^2_{\mathrm{ER}} > r^2_{\mathrm{ER}(h)}.$$

Thus, under repeated sampling, with the desired probability $\alpha/2$, the upper limit of our confidence interval, $r^2_{\mathrm{ER}(h)}$, does not contain r^2_{ER}. The proof for the lower end of the confidence interval $r^2_{\mathrm{ER}(l)}$ is similar. The probability of the mutually exclusive events that either $r^2_{\mathrm{ER}} > r^2_{\mathrm{ER}(h)}$or $r^2_{\mathrm{ER}} < r^2_{\mathrm{ER}(l)}$ is the sum of the probability of the two events, α. See Figure 3.14 for a graphical explanation of this proof.

For simplicity of the proof, we assumed that it was possible to find $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}(h)}) = \alpha/2$, which is not necessarily the case because $r^2_{\mathrm{ER}(h)} \in [0,1]$ is bounded but $\hat{r}^2_{\mathrm{ER}}$ is not. If $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 1) > \alpha/2$ or $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 0) < \alpha/2$, then there is no $r^2_{\mathrm{ER}(h)}$ that will achieve $\alpha/2$. Under the condition where $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 1) > \alpha/2$, we simply set $r^2_{\mathrm{ER}(h)} = 1$, and since $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 1) > \alpha/2$ implies $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} \in [0,1]) > \alpha/2$, the confidence interval will contain the true value.

Under the condition $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 0) < \alpha/2$, we set $r^2_{\mathrm{ER}(h)} = 0$, but we must set the confidence interval, though normally inclusive, to be non-inclusive. Intuitively, this is because if $r^2_{\mathrm{ER}} = 0$, then the upper end of the confidence interval would always contain the true value, and we would be restricted to $\alpha = 1$. Making the CI non-inclusive avoids this problem. Doing this does not cause a problem when the true $r^2_{\mathrm{ER}} > 0$, because $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 0) < \alpha/2$ implies $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} \in [0,1]) < \alpha/2$, the confidence interval should not contain the true r^2_{ER} and it does not because $r^2_{\mathrm{ER}} > 0 = r^2_{\mathrm{ER}(h)}$. The case for $r^2_{\mathrm{ER}(l)}$ is similar.

In summary, our confidence interval is defined to be $[r^2_{\mathrm{ER}(l)}, r^2_{\mathrm{ER}(h)}]$ when $r^2_{\mathrm{ER}(l)} < 1$ and $r^2_{\mathrm{ER}(h)} > 0$ but $\emptyset$ (the empty set) if $r^2_{\mathrm{ER}(l)} = 1$ or $r^2_{\mathrm{ER}(h)} = 0$. The lower bound, $r^2_{\mathrm{ER}(l)}$, satisfies $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}(l)}) = 1 - \alpha/2$, except if $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 1) > 1 - \alpha/2$ or $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 0) < 1 - \alpha/2$, then respectively $r^2_{\mathrm{ER}(l)} = 1$ or $r^2_{\mathrm{ER}(l)} = 0$. The upper bound, $r^2_{\mathrm{ER}(h)}$, satisfies $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}(h)}) = \alpha/2$, except if $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 1) > \alpha/2$ or $F(\hat{r}^{2*}_{\mathrm{ER}}|r^2_{\mathrm{ER}} = 0) < \alpha/2$, then respectively $r^2_{\mathrm{ER}(h)} = 1$ or $r^2_{\mathrm{ER}(h)} = 0$.

To sample from the conditional distribution $f(\hat{r}^2_{\mathrm{ER}}|s^2, \hat{d}^2, r^2_{\mathrm{ER}})$, we assume that σ^2 and d^2 follow an uninformative non-negative uniform prior ($U[0,\infty]$), and given their observed estimates s^2 and $\hat{d}^2$, we obtain samples from the posterior distribution of σ^2 and d^2 via the Metropolis-Hastings sampling method (for details see "Bayesian model and simulation"). For a chosen r^2_{ER} (e.g., a candidate for $r^2_{\mathrm{ER}(h)}$, we sample from $f(\hat{r}^2_{\mathrm{ER}}|s^2, \hat{d}^2, r^2_{\mathrm{ER}})$ by drawing samples of σ^2 and d^2 from the posterior distribution $f(\sigma^2, d^2|s^2, \hat{d}^2)$ while r^2_{ER} is fixed to the desired value. Thus for each sample we then draw observations Y and predictions ν_i from the model described in Eqn. 3.6 and finally calculate $\hat{r}^2_{\mathrm{ER}}$ for a sample from $f(\hat{r}^2_{\mathrm{ER}}|s^2, \hat{d}^2, r^2_{\mathrm{ER}})$.

Computing confidence intervals.

We use a simple iterative bracketing algorithm to narrow down the range of candidate values for the ends of our confidence interval. For example, to estimate $r^2_{\mathrm{ER}(h)}$ within $[0,1]$, we first evaluate the highest possible value: 1. We sample N=2,500 draws from $f(\hat{r}^2_{\mathrm{ER}}|s^2,\hat{d}^2,r^2_{\mathrm{ER(c)}}=1)$ to find the proportion, $\hat{p}$, of those less than or equal to $\hat{r}^{2*}_{\mathrm{ER}}$. We then calculate a z-statistic to test whether this is significantly different from the desired $\alpha/2$:

$$z=\frac{\hat{p}-\alpha/2}{\hat{p}(1-\hat{p})/N}.$$

At some desired significance level (here $p<0.01$), we either do not reject the null and accept $r^2_{\mathrm{ER(h)}}=r^2_{\mathrm{ER(c)}}$, or we reject the null. In the latter case, if z is positive we determine that $r^2_{\mathrm{ER(h)}}$ must be higher, whereas if z is negative it must be lower. In the case where $r^2_{\mathrm{ER(c)}}=1$ and z is positive, there are no higher possible values of $r^2_{\mathrm{ER(h)}}$ and thus we accept $r^2_{\mathrm{ER(h)}}=r^2_{\mathrm{ER(c)}}$. Otherwise, on the next step we choose a new candidate by sampling from $r^2_{\mathrm{ER(c)}}\sim U[0,1]$ then evaluating the result and if we reject the null and z is positive our new interval will be $[r^2_{\mathrm{ER(c)}},1]$ and if z is negative $[0,r^2_{\mathrm{ER(c)}}]$. Otherwise, if we do not reject the null we accept $r^2_{\mathrm{ER(h)}}=r^2_{\mathrm{ER(c)}}$. We continue this bracketing until we do not reject the null or a pre-determined number of splits has passed (here we use 100). Accuracy of this algorithm will increase with number of splits and simulation samples.

Confidence interval validation.

We used simulations to evaluate our confidence interval methods under the sampling distribution $f(\hat{r}^2_{\mathrm{ER}}|s^2,\ \hat{d}^2,\ r^2_{\mathrm{ER}})$. Conceptually, this is the distribution of $\hat{r}^2_{\mathrm{ER}}$ after data has been collected and sample variance and sample dynamic range calculated, and now we wish to calculate the data's fit to a model with unknown but fixed r^2_{ER}. To demonstrate that our method contains the unknown r^2_{ER} at the desired α, our procedure is as follows. For a chosen n, m, σ^2, d^2, and r^2_{ER}, sample an $n\times m$ data matrix (Y) and calculate its sample variance s^2 and dynamic range $\hat{d}^2$. Then using the Metropolis-Hastings algorithm (see Methods, "Bayesian model and simulation"), draw 5,000 samples from the posterior distribution $f(\sigma^2,d^2|s^2,\hat{d})$. Next, we simulate the distribution of $\hat{r}^2_{\mathrm{ER}}$ for each of these data samples by drawing from $f(\hat{r}^2_{\mathrm{ER}}|\sigma^2\ ,\ d^2,r^2{}_{\mathrm{ER}})$. For each of these draws, we construct confidence intervals, and then we calculate the proportion of times that the confidence intervals contain the true $r^2{}_{\mathrm{ER}}$. This proportion estimates the true α level of the confidence interval method.

Bayesian model and simulation.

We sample from the posterior of two parameters: σ^2 and $d^2=\frac{1}{m}\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2$. Their associated sufficient statistics are:

$$\hat{s}^2=\frac{1}{m(n-1)}\sum_{i=1}^{m}\sum_{j=1}^{n}(Y_{i,j}-\bar{Y}_{i,\cdot})^2$$

$$\hat{d}^2=\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2$$

and their distributions are:

$$\hat{s}^2 \sim \frac{\sigma^2}{m(n-1)}\chi^2_{m(n-1)} \tag{3.20}$$

$$\hat{d^2} \sim \frac{\sigma^2}{n(m-1)}\chi^2_{m-1}\left(\frac{\sum_{i=1}^{m}(\mu_i - \bar{\mu})^2}{\frac{\sigma^2}{n}}\right) \tag{3.21}$$

By Bayes theorem we have,

$$P(\sigma^2, d^2|\hat{s}^2, \hat{d^2}) \propto P(\hat{s}^2, \hat{d^2}|\sigma^2, d^2)P(\sigma^2, d^2) = P(\hat{s}^2|\sigma^2)P(\hat{d^2}|d^2)\mathbf{1}(\sigma^2)_{[0,\infty]}\mathbf{1}(d^2)_{[0,\infty]},$$

where the equality is derived by recognizing the sample variance ($\hat{s}^2$) and dynamic range ($\hat{d^2}$) are independent and setting the prior to be uniform non-negative. The estimates $\hat{s}^2$ and $\hat{d^2}$ are fixed, calculated from the data, and our goal is to look up the distribution of the parameters given these fixed values. We use the Metropolis-Hastings algorithm to draw from the desired distribution $P(\sigma^2, d^2|\hat{s}^2, \hat{d^2})$ and approximate it with the empirical distribution (a histogram). Our sampling procedure is as follows. We initialize our parameter samples σ^2, d^2 at their estimates $\hat{s}^2, \hat{d^2}$, and we then sample a new candidate from our proposal distribution: a truncated multivariate normal with means $\hat{s}^2, \hat{d^2}$ and diagonal variances equal to the variance of the distributions (Eqns. 3.20 and 3.21) where $\sigma^2 = \hat{s}^2, d^2 = \hat{d^2}$. We take the ratio of likelihoods,

$$a = P(\hat{s}^2, \hat{d^2}|\sigma^2_{\text{proposal}}, d^2_{\text{proposal}}) / P(\hat{s}^2, \hat{d^2}|\sigma^2_{\text{current}}, d^2_{\text{current}}).$$

If $a > 1$, we accept the candidates as our new current samples, but if $a < 1$, we then draw from $u \sim U[0,1]$. If $u < a$, we also accept the candidates but if not, we retain the current samples. Throughout the paper we run the chain for 5,000 iterations then randomly sample with replacement from it.

3.5.6 SNR relation to F-test and number of trials

Our goal is to be able find for a given SNR and number of repeats the number of stimuli needed to reliably detect tuning under an F-test. To calculate the F-statistic for testing whether there is variation in the expected responses across stimuli (i.e., stimulus selectivity), we form the ratio,

$$F = \frac{\frac{n}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2}{\frac{1}{n(m-1)}\sum_{i=1}^{m}\sum_{j}^{n}(Y_{i,j} - \bar{Y}_{i,\cdot})^2},$$

where for clarity we indicate dimensions averaged over with a dot. The numerator calculates the amount of variance explained by stimuli and the denominator calculates the amount of variance unexplained by stimuli. The numerator is a scaled non-central χ^2 distribution:

$$\frac{n}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2 \sim \frac{n}{m-1}\frac{\sigma^2}{n}\chi^2_{m-1}\left(\frac{\sum(\mu_i - \bar{\mu})^2}{\sigma^2/n}\right) = \frac{n}{m-1}\frac{\sigma^2}{n}\chi^2_{m-1}(mn\text{SNR}),$$

where the final equality comes from the definition of SNR (3.5). The denominator is a central χ^2 distribution:

$$\frac{1}{n(m-1)}\sum_{i=1}^{m}\sum_{j}^{n}(Y_{i,j}-\bar{Y}_{i,\cdot})^2 \sim \frac{\sigma^2}{n(m-1)}\chi^2_{m(n-1)}.$$

Thus taking the ratio we have a singly non-central F-distribution:

$$F_{m-1,m(n-1)}(mn(\mathrm{SNR})). \tag{3.22}$$

To test for significant tuning, we set an α-level criterion, $c_{F(\alpha)}$, under the null hypothesis that the observed F-statistic is from a central F-distribution:

$$P[F_{m-1,m(n-1)} \geq c_{F(\alpha)}] = \alpha.$$

Finally, given m and n, we can find the SNR where, for some high probability β,

$$P[F_{m-1,m(n-1)}(mn(\mathrm{SNR})) > c_{F(\alpha)}] = \beta.$$

We set $\beta = 0.99$ and $\alpha = 0.01$ and numerically solve for the SNR.

3.5.7 Electrophysiological data

We reanalyzed a variety of neuronal data from previous studies. This includes three experiments in area V4 and one in MT of the awake, fixating rhesus monkey (Macaca mulattta), as well as spiking and two-photon imaging in awake mouse VISp. Experimental protocols for all studies are described in detail in the original publications.

From Pasupathy and Connor [41], we examined responses of 109 V4 neurons to a set of 362 shapes. There were typically 3-5 repeats of each stimulus, but we used only the 96 cells that had at least 4 repeats for all stimuli. We used the spike count for each trial during the 500 ms stimulus presentation.

From Popovkina et al. [70], we examined responses of 43 V4 neurons (7 from one monkey, 36 from another) to filled stimuli (drawn from the same set of shapes used for the previous study) and to outline stimuli that were the same except the fill was set to be equivalent to background color. Stimulus color and luminance were customized to elicit a robust response from the recorded neuron. Spikes were counted over the 300 ms duration of each stimulus presentation.

From the 2019 UW V4 Neural Data Challenge, we examined single unit (SUA) and multi-unit (MUA) data from 7 V4 recordings. Up to 601 images were shown with between 3-20 repeats for each image. The images were drawn semi-randomly from the 2012 ILSVRC validation set of images [83] where an 80X80 pixel patch was sampled and had a soft window applied (circular Gaussian, SD 16 pixels, applied to the alpha channel). Images were shown for 300 ms with 250 ms in between images. The model we analyze was the winner of the Neural Data Challenge (out of 32 competitors) on held-out data from the 14 sets of V4 responses to natural images. For access to data see 'https://www.kaggle.com/c/uwndc19/'.

From Zohary et al. [92, 93], we examined responses from 81 pairs of MT neurons recorded from three awake rhesus monkey (Macaca mulatta) viewing dynamic random dots (stimuli described in Britten et al. [100]). Optimal speed of drifting dots was found for the one of the two neurons being recorded. Eight different directions of motion at 45° increments were repeated 10-20 times. Monkeys performed a two alternative forced choice task of motion direction discrimination during the experiment. Post-stimulus spikes were counted in the 2 s window of stimulus presentation. Experimenters were rigorous in only recording from pairs of neurons whose spike wave forms were strikingly different.

From the Allen Institute for Brain Science (AIBS) mouse database [95], we examined calcium fluorescence data. Fluorescence of mouse visual cortex neurons expressing GCaMP6f was measured via 2-photon imaging through a cranial window. We analyzed signals recorded in response to natural scenes and static gratings presented for 0.25 s each with no interval between with 50 repeats in random order. The natural scene stimulus consisted of 118 natural images from a variety of databases. The static grating stimulus consisted of a full field static sinusoidal grating at a single contrast (80%). Gratings were presented at 6 orientations, 5 spatial frequencies (0.02, 0.04, 0.08, 0.16, 0.32 cycles/degree), and 4 phases (0, 0.25, 0.5, 0.75). For every trial, $\Delta F/F$ was estimated using the average fluorescence of the preceding second (4 image presentations) as baseline. We analyzed the average change in fluorescence during the 1/2 s period after the start of the image presentation relative to 1 s before. We also examined SUA data from the AIBS mouse neuropixel data set [96], which was recorded in response to the same stimuli as the calcium data. Spike counting windows were 0.25 s.

3.5.8 Prior analytic methods for estimating r^2_{ER}

Our estimator, $\hat{r}^2_{\mathrm{ER}}$, is derived via the strategy of Haefner and Cumming [65] to unbias the numerator and denominator of the coefficient of determination. Haefner and Cumming in turn cite Sahani and Linden [87] as the predecessor to their method. Sahani and Linden constructed an unbiased estimator of the variation in the expected response of the neuron (i.e. tuning curve) they called this 'signal power'. They normalize an estimator of explained variation, unbiased with respect to noise under conditions they did not specify (1-parameter regression of model predictions, see Methods, "Derivation of Normalized Signal Power Explained ($\mathrm{SPE}_{\mathrm{norm}}$)"). Sahani and Linden, by not specifying how a given model prediction should be fit to the neural data before estimating the quality of the fit, introduced potential problems in their estimator. This was recognized by Schoppe et al. [90], who point out that the estimator was sensitive to differences between the mean and amplitude of the model predictions and the neural data. Consequently, the estimator could give large negative values because the squared error between model predictions and neural responses was unbounded. This criticism, while technically correct, is easily overcome by regressing (with intercept term) the given model predictions onto the neural data before using normalized SPE. Schoppe et al., motivated by the problems they found in SPE, focused on simplifying $\mathrm{CC}_{\mathrm{norm}}$ of Hsu et al. [88] to not require resampling by making use of the signal power estimator developed by Sahani and Linden. They derived a simple estimator whose square, termed here $\mathrm{CC}^2_{\mathrm{norm\text{-}SP}}$, we find is essentially numerically equivalent to $\mathrm{SPE}_{\mathrm{norm}}$ of Sahani and Linden in the case of one-parameter regression.

For the purpose of comparison, below we write out the exact formulas and approximate expected values of two prior methods [65, 87] that are closely related to $\hat{r}^2_{\mathrm{ER}}$ in the notation we use throughout our paper. For all estimators, we assume responses to m stimuli with n repeats where variance have been stabilized. The response to the ith stimulus, jth repeat, is $Y_{i,j} \sim N(\mu_i, \sigma^2)$ where σ^2 is the trial-to-trial variability and μ_i the ith expected value of response after variance stabilization. The predictions are fixed for the m stimuli and the ith predicted expected value of the data is ν_i and we assume they have been fit by a linear model with d degrees of freedom. When averaging data across trials our notation will be $\bar{Y}_{i,\cdot} = \frac{1}{n}\sum_{j=1}^{n} Y_{i,j}$ and across stimuli $\bar{Y}_{\cdot,j} = \frac{1}{m}\sum_{i=1}^{n} Y_{i,j}$.

Derivation of Normalized Signal Power Explained ($\mathbf{SPE_{norm}}$.)

The original description of SPE [87] did not specify, when calculating sample estimates of 'power' (better known as variance), whether to normalize by $m-1$ (the unbiased estimate) or m (the MLE). Nor was it specified whether an average across trials was used when calculating the difference between variance of the measured response and the residual. We thus use the formula as described by Schoppe et al. [90], who provide code to unambiguously calculate SPE, albeit in a manner that differs from their text (in particular, their TP= $(n-1)\sum_{j=1}^{n}\frac{1}{m-1}\sum_{i=1}^{m}(Y_{i,j}-\bar{Y}_{\cdot,j})^2)$ (similar to Eqn. 5 of Sahani and Linden), but in their code it is implemented as $\frac{1}{n}\sum_{j=1}^{n}\frac{1}{m-1}\sum_{i=1}^{m}(Y_{i,j}-\bar{Y}_{\cdot,j})^2)$). In the context of the quantities being estimated, the latter makes more sense (see calculation of expected value of denominator below). For comparison to the derivation in Schoppe et al., their notation equates to ours as follows: $N=n$, $T=m$, $R=Y_{i,j}$, $y=\bar{Y}_{i,\cdot}$, $\hat{y}=\hat{v}_i$, and $Var(y)=\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2$. Finally, in our notation their estimator is:

$$\mathrm{SPE}_{\mathrm{norm}}=\frac{\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2-\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\hat{v}_i)^2}{\frac{1}{n-1}(n\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2-\frac{1}{n}\sum_{j=1}^{n}\frac{1}{m-1}\sum_{i=1}^{m}(Y_{i,j}-\bar{Y}_{\cdot,j})^2)}.$$

Calculating the expectation of the numerator and the denominator for the fit of a linear model with d degrees of freedom, we can find the asymptotic expectation. Numerator:

$$\mathrm{E}[\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2-\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\hat{v}_i)^2]$$

$$=\frac{1}{m-1}(\mathrm{E}[\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2]-\mathrm{E}[\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\hat{v}_i)^2])$$

$$=\frac{1}{m-1}(\mathrm{E}[\frac{\sigma^2}{n}\chi^2_{m-1}(\frac{\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2}{\frac{\sigma^2}{n}})]-\mathrm{E}[\frac{\sigma^2}{n}\chi^2_{m-d}(\frac{\sum_{i=1}^{m}(\mu_i-\hat{v}_i)^2}{\frac{\sigma^2}{n}})])$$

$$=\frac{1}{m-1}((\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2+(m-1)\frac{\sigma^2}{n})-(\sum_{i=1}^{m}(\mu_i-\hat{v}_i)^2+(m-d)\frac{\sigma^2}{n}))$$

$$=\frac{1}{m-1}(\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2-\sum_{i=1}^{m}(\mu_i-\hat{v}_i)^2)+\frac{\sigma^2}{n}(d-1)).$$

Denominator:

$$\mathrm{E}[\frac{1}{n-1}(n\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-\bar{Y}_{\cdot,\cdot})^2-\frac{1}{n}\sum_{j=1}^{n}\frac{1}{m-1}\sum_{i=1}^{m}(Y_{i,j}-\bar{Y}_{\cdot,j})^2)]$$

$$=\frac{1}{n-1}(n\frac{1}{m-1}(\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2+(m-1)\frac{\sigma^2}{n})-\frac{1}{n}\sum_{j=1}^{n}\frac{1}{m-1}\mathrm{E}[\sigma^2\chi^2_{m-1}(\frac{\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2}{\sigma^2})])$$

$$=\frac{1}{n-1}(n\frac{1}{m-1}(\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2+(m-1)\frac{\sigma^2}{n})-\frac{1}{m-1}\mathrm{E}[\sigma^2\chi^2_{m-1}(\frac{\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2}{\sigma^2})])$$

$$=\frac{1}{n-1}(n\frac{1}{m-1}(\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2+(m-1)\frac{\sigma^2}{n})-\frac{1}{m-1}(\sum_{i=1}^{m}(\mu_i-\bar{\mu}_\cdot)^2+(m-1)\sigma^2)])$$

$$= \frac{1}{m-1}\frac{1}{n-1}(n-1)\sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2 = \frac{1}{m-1}\sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2.$$

Putting the expectations into the numerator and denominator we have:

$$\frac{\sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2 - \sum_{i=1}^{m}(\mu_i - \hat{v}_i)^2 + \frac{\sigma^2}{n}(d-1)}{\sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2}.$$

We note that only if $d = 1$ (i.e., the model has only 1 term) is the numerator unbiased. Below we describe how Haefner and Cumming developed an estimator that accounts for degrees of freedom more generally.

Derivation of Y.

For comparison to the original paper of Haefner and Cumming [65], we give their notation and its equivalent terms in our notation: $d_{i,j} = Y_{i,j}$, $d_i = \bar{Y}_{i,\cdot}$, $\bar{\bar{d}} = \bar{Y}_{\cdot,\cdot}$, $\Sigma^2 = \sigma^2$, $N = m$, $N_\sigma = m(n-1)$, $R = n$, $n = d$, $D_i = \mu_i$, $M_i = \nu_i$, $m_i = \hat{v}_i$, $\sigma^2 = \hat{\sigma^2}$, $\lambda_{DD} = \sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2/\sigma^2$, $\lambda_{DM} = \sum_{i=1}^{m}(\mu_i - \nu_i)^2$.

These authors explicitly attempt to remove the bias of the coefficient of determination:

$$r^2 \equiv 1 - \frac{\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \nu_i)^2}{\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2}.$$

Their unbiased estimator is derived by dividing the numerator and denominator by the sample trial-to-trial variability ($\hat{\sigma}^2 = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{n-1}\sum_{j=1}^{n}(Y_{i,j} - \bar{Y}_{\cdot,j})^2$) and their respective degrees of freedom (d below being the degrees of freedom of the linear model), noting the numerator and denominator become non-central F-distributions, then shifting and scaling these to provide unbiased estimates. Since $\mathrm{E}[F_{d_1,d_2}(\lambda)] = \frac{d_2(d_1+\lambda)}{d_1(d_2-2)}$, the expectation of the numerator is,

$$\mathrm{E}[\frac{1}{m-d}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \hat{v}_i)^2/\hat{\sigma}^2] = \mathrm{E}[F_{(m-d),m(n-1)}(\sum_{i=1}^{m}\frac{(\mu_i - \hat{v}_i)^2}{\sigma^2})] = \frac{m(n-1)(m-d+\sum_{i=1}^{m}(\mu_i - \hat{v}_i)^2/\sigma^2)}{(m-d)(m(n-1)-2))},$$

thus the unbiased estimate of the numerator is:

$$\mathrm{E}[\frac{(m-d)(m(n-1)-2)}{m(n-1)}\left(\frac{1}{m-d}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \hat{v}_i)^2)/\hat{\sigma}^2\right) - (m-d)] = \sum_{i=1}^{m}(\mu_i - \hat{v}_i)^2/\sigma^2.$$

The expectation of the denominator is:

$$\mathrm{E}[\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2/\hat{\sigma}^2] = \mathrm{E}[F_{(m-1),m(n-1)}(\sum_{i=1}^{m}(\frac{(\mu_i - \bar{\mu}_\cdot)^2}{\sigma^2}))] = \frac{m(n-1)(m-1+\sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2/\sigma^2)}{(m-1)(m(n-1)-2)},$$

thus the unbiased estimate of the denominator is:

$$\mathrm{E}[\frac{(m-1)(m(n-1)-2)}{m(n-1)}\left(\frac{1}{m-1}\sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2/\hat{\sigma}^2)\right) - (m-1)] = \sum_{i=1}^{m}(\mu_i - \bar{\mu}_\cdot)^2/\sigma^2.$$

Forming the ratio, we obtain the Heafner and Cumming estimator:

$$\mathrm{Y} = 1 - \frac{\frac{(m-d)(m(n-1)-2)}{m(n-1)} \left(\frac{1}{m-d} \sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \hat{v}_i)^2 / \hat{\sigma}^2 \right) - (m-d)}{\frac{(m-1)(m(n-1)-2)}{m(n-1)} \left(\frac{1}{m-1} \sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2 / \hat{\sigma}^2 \right) - (m-1)} =$$

$$1 - \frac{\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \hat{v}_i)^2 / \hat{\sigma}^2 - \frac{m(n-1)}{m(n-1)-2}(m-d)}{\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2 / \hat{\sigma}^2 - \frac{m(n-1)}{m(n-1)-2}(m-1)}.$$

3.5.9 Extension of $\hat{r}^2_{\mathrm{ER}}$ to fit of linear model

The derivation of Haefner and Cumming via the non-central F was not necessary: the expectation of the numerator and denominator are straightforward to calculate as non-central χ^2 random variables. While our $\hat{r}^2_{\mathrm{ER}}$ is explicitly meant to be the analogue of Pearson's r^2, we re-derive the Haefner and Cumming formula along the lines of $\hat{r}^2_{\mathrm{ER}}$ for measuring the fit of a linear model. We specifically avoid the non-central F-distribution so that it is unnecessary to estimate variance (if for example there is a strong prior for the variance and/or multiple trials were not collected). We assume, as did Haefner and Cumming that $\hat{v}_i$ were fit from a linear model via least squares with d coefficients. The expectation of the numerator is:

$$\mathrm{E}[\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \hat{v}_i)^2] = \mathrm{E}[\frac{\sigma^2}{n} \chi^2_{m-d} (\sum_{i=1}^{m} (\frac{(\mu_i - \hat{v}_i)^2}{\frac{\sigma^2}{n}})] = \sum_{i=1}^{m} (\mu_i - \hat{v}_i)^2 + (m-d)\frac{\sigma^2}{n},$$

thus its unbiased estimate is:

$$\mathrm{E}[\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \hat{v}_i)^2 - (m-d)\frac{s^2}{n}] = \sum_{i=1}^{m} (\mu_i - \hat{v}_i)^2.$$

The expectation of the denominator is,

$$\mathrm{E}[\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2] = \mathrm{E}[\frac{\sigma^2}{n} \chi^2_{m-1} (\sum_{i=1}^{m} \frac{(\mu_i - \bar{\mu})^2}{\frac{\sigma^2}{n}})] = \sum_{i=1}^{m} (\mu_i - \bar{\mu})^2 + (m-1)\frac{\sigma^2}{n},$$

thus its unbiased estimate is,

$$\mathrm{E}[\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2 - (m-1)\frac{s^2}{n}] = \sum_{i=1}^{m} (\mu_i - \bar{\mu}_\cdot)^2.$$

Thus their ratio forms:

$$\hat{r}^2_{\mathrm{ER}} = 1 - \frac{\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \hat{v}_i)^2 - (m-d)\frac{\hat{\sigma}^2}{n}}{\sum_{i=1}^{m} (\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2 - (m-1)\frac{\hat{\sigma}^2}{n}}. \tag{3.23}$$

3.6 Figures

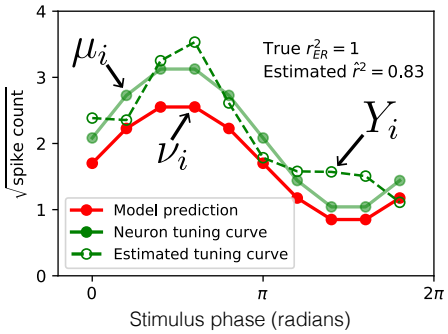


FIGURE 3.1: Sampling noise confounds estimation of the correlation between model prediction and neuronal tuning curve. The expected (true) spike counts in response to a set of 10 stimuli (solid green points) is perfectly correlated with a model (red points), yet owing to sampling error (neural trial-to-trial variability) the estimated tuning curve (green open circles) has correlation less than one with the model ($\hat{r}^2 = 0.83$).

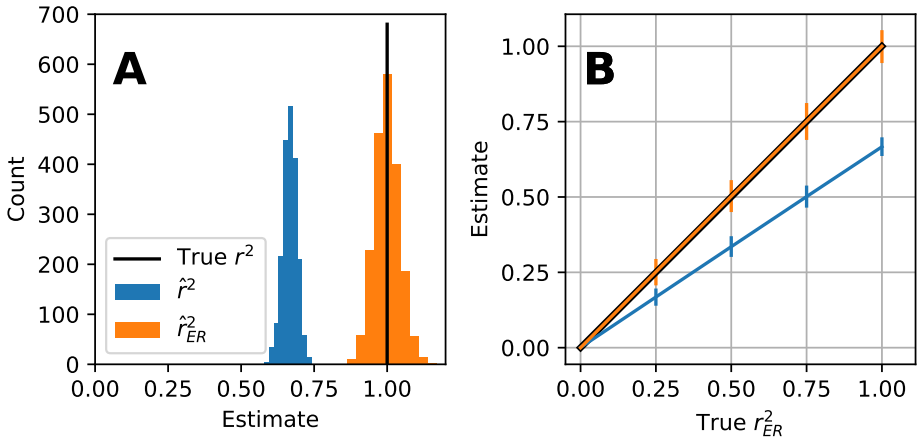


FIGURE 3.2: Simulation of the naive $\hat{r}^2$ and unbiased $\hat{r}^2_{\text{ER}}$ estimators for model-to-neuron fits at varying levels of r^2_{ER} where $m = 362$, $n = 4$, and $\sigma^2 = 0.25$. **(A)** For true $r^2 = 1$, at a moderately low SNR=0.5, $\hat{r}^2$ (blue) is on average 0.67 whereas $\hat{r}^2_{\text{ER}}$ (orange) is on average 1.00. The bias of r^2_{ER} (see Methods, "Bias of $\hat{r}^2_{\text{ER}}$") is small relative to its variability (90% quantile $= [0.93, 1.07]$ vertical bars) and to the bias of $\hat{r}^2$. **(B)** Same simulation as A but at five levels of r^2_{ER} (0, 0.25, 0.5, 0.75, 1). Lines show mean values of $\hat{r}^2$ (blue) and $\hat{r}^2_{\text{ER}}$ (orange). Black line (beneath orange) shows true r^2_{ER}; error bars show 90% quantile.

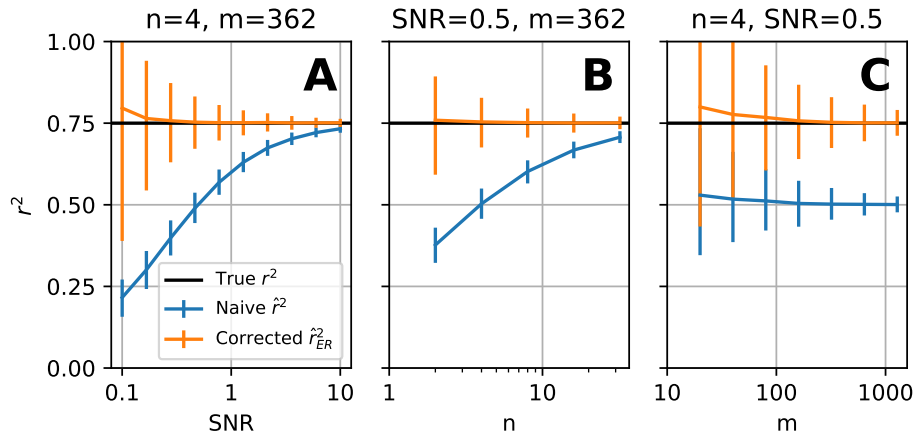


FIGURE 3.3: Comparison of $\hat{r}^2$ and $\hat{r}^2_{\text{ER}}$ for estimating model-to-neuron fit across broad, relevant ranges of SNR, n, and m . **(A)** Average performance of naive $\hat{r}^2$ (blue) and corrected $\hat{r}^2_{\text{ER}}$ (orange) as a function of SNR for a simulation where true $r^2_{\text{ER}} = 0.75$ (horizontal black line), $m = 362$, $n = 4$, and $\sigma^2 = 0.25$. Error bars indicate 90% quantiles. **(B)** Performance of estimators as a function of n, the number of repeats of each stimulus. Simulation like (A), except SNR=0.5 and n is varied. **(C)** Performance as a function of m, the number of unique stimuli, for a low number of repeats (n=4). Like (A), except SNR=0.5 and m is varied.

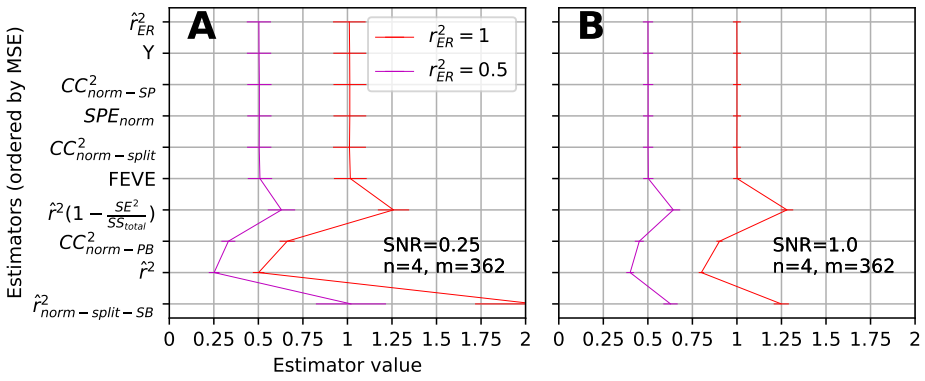


FIGURE 3.4: Comparison of $\hat{r}^2_{\text{ER}}$ with published estimators of r^2_{ER} on the basis of simulated data. **(A)** Low SNR (0.25) simulation where estimators on vertical axis are sorted from top to bottom by smallest MSE with respect to estimating $r^2_{\text{ER}} = 1$. Traces show mean and SD of each estimator. **(B)** Same simulation at higher SNR (1.0) but same m, n.

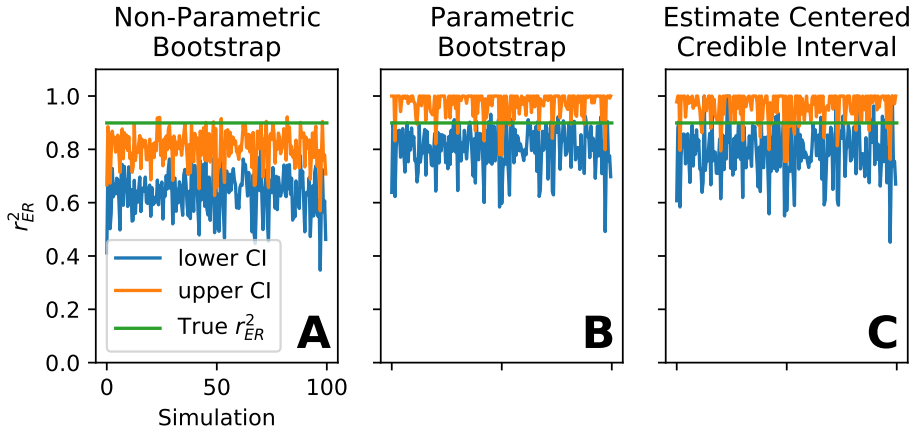


FIGURE 3.5: Validation of confidence interval (CI) methods by simulation—example CIs for three methods. Simulation parameters: $n = 4$, $m = 40$, true $r^2_{\text{ER}} = 0.91$, dynamic range $d^2 = 0.25$, trial-to-trial variability $\sigma^2 = 0.25$, and target confidence level $\alpha = 0.8$. Of 2000 independent simulations, CIs for the first 100 are plotted here for three different methods. CIs for all methods were calculated using the same set of randomly generated responses. **(A)** For the non-parametric bootstrap method, the upper end (orange) and lower end (blue) of the CI were almost always both below the true correlation value (0.91, green line), indicating an overwhelming failure to achieve 80% containment of the true value. **(B)** The parametric bootstrap method and **(C)** our ECCI method perform substantially better. Performance of all three methods over the full range of true r^2_{ER} is plotted in Figure 3.6.

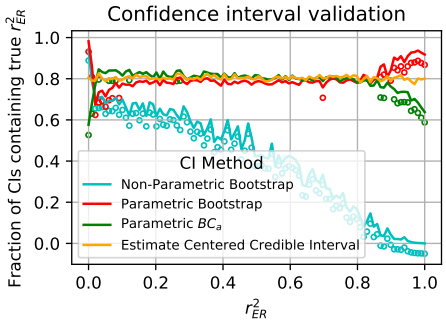


FIGURE 3.6: Comparison of four methods for computing confidence intervals for $\hat{r}^2_{\mathrm{ER}}$ spanning the full range of true correlation. The fraction of times the CI contained the true value is plotted for each method (see line style inset) as a function of the true correlation value, r^2_{ER}, at 100 values linearly spaced between 0 and 1. The target α-level was 0.8. Open circles indicate that the fraction deviated from 0.8 significantly (p<0.01, Bonferroni corrected).

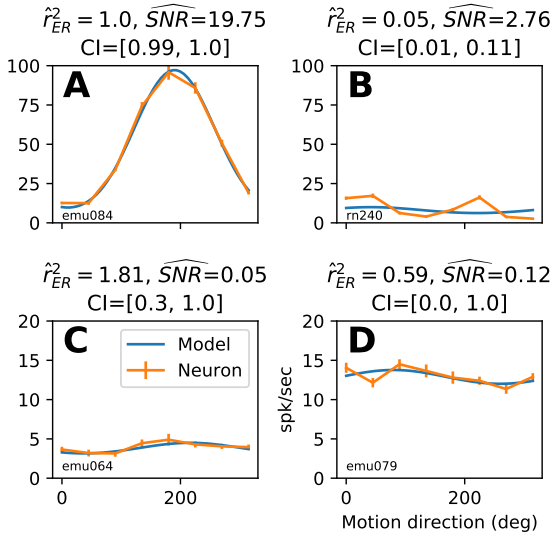


FIGURE 3.7: Applying our unbiased estimator with CIs to fit four example MT neuronal direction tuning curves to a sinusoidal model. **(A)** Example neuron tuning curve (orange trace with SEM bars) with excellent fit to sinusoidal model (blue trace, $\hat{r}^2_{\mathrm{ER}}$=1.0), high SNR and tight CI (parameters specified above plot panel). **(B)** Example neuron with poor fit to sinusoidal model but with a reasonable SNR and narrow CI that provide confidence that the neuronal tuning systematically deviates from the model. **(C)** Example neuron with poor SNR and wild estimate of $\hat{r}^2_{\mathrm{ER}}$, which is reflected in large CI=[0.3, 1], suggesting that no conclusion can be made about how well the model describes any actual tuning here. **(D)** Example neuron with a seemingly reasonable $\hat{r}^2_{\mathrm{ER}}$, but the low SNR and CI covering the entire interval [0,1] reveals that this fit cannot be trusted.

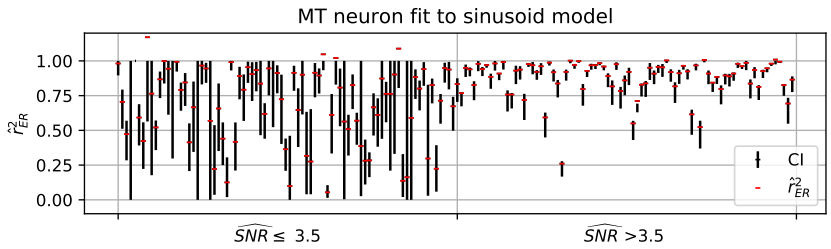


FIGURE 3.8: Confidence intervals (vertical black lines $\alpha = 90\%$) and point estimates (horizontal red lines) $\hat{r}^2_{\text{ER}}$ across all MT neurons fit to sinusoidal model. Estimates in first interval (left) had $\widehat{\text{SNR}}$ less than the median (3.5), whereas in the second interval greater than the median.

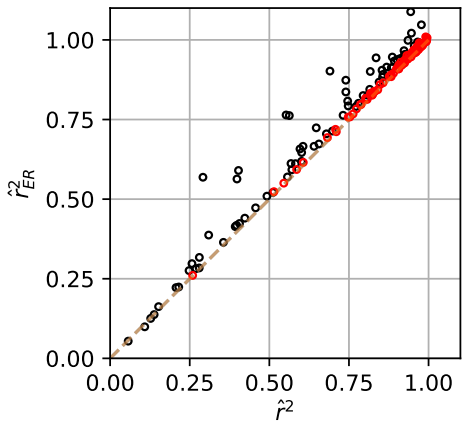


FIGURE 3.9: Relationship of naive $\hat{r}^2$ and corrected $\hat{r}^2_{\text{ER}}$ between fits of sinusoidal model to MT data. Units with $\widehat{\text{SNR}}$ greater than the median across the population ($\widehat{\text{SNR}} = 3.5$) are plotted in red and those less than or equal to in black.

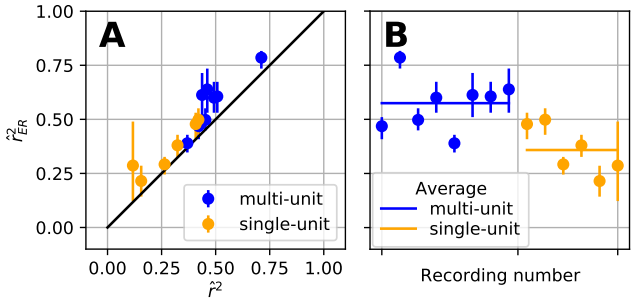


FIGURE 3.10: Applying $\hat{r}^2_{\text{ER}}$ to analyze performance of a deep neural network (DNN) in predicting V4 responses to natural images. **(A)** For single-unit (orange) and multi-unit (blue) recordings, $\hat{r}^2_{\text{ER}}$ is plotted against the naive $\hat{r}^2$. The relatively short $\alpha = 0.1$ CIs (vertical bars) suggest that most of these correlation values are trustworthy. **(B)** The mean $\hat{r}^2_{\text{ER}}$ value across multi-unit recordings (horizontal blue line) is significantly higher than that for the set of single-unit recordings (orange horizontal line; Welch's t-test t=3.7, p=0.005). Because individual estimates are asymptotically unbiased, the group average inherits this lack of bias.

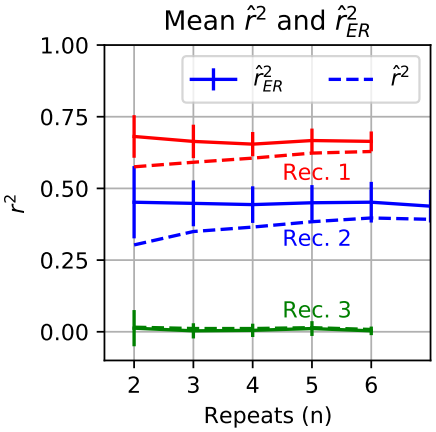


FIGURE 3.11: Relationship of naive $\hat{r}^2$ and corrected $\hat{r}^2_{\text{ER}}$ with n, the number of repeats for V4 data. Different colors indicate different recordings. Solid lines show the average $\hat{r}^2_{\text{ER}}$ estimate across random shuffling of trials (with replacement); vertical bars indicate SD. Dashed lines show average $\hat{r}^2$.

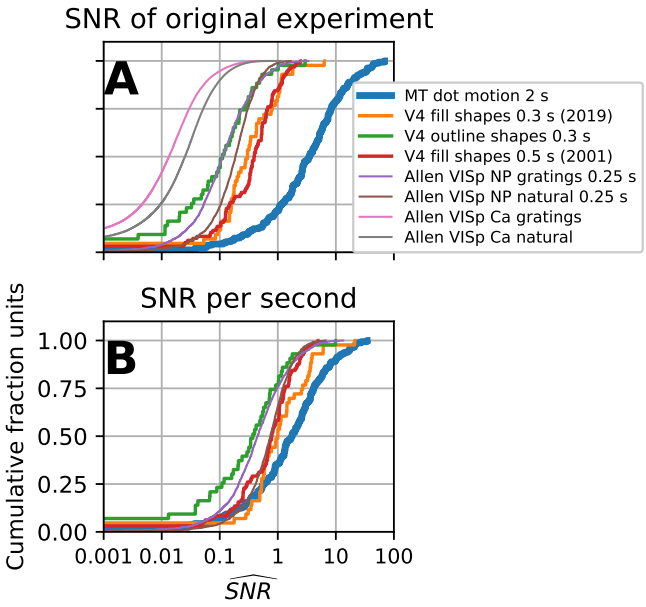


FIGURE 3.12: A comparison of our data quality metric, the signal-to-noise ratio estimator $\widehat{\text{SNR}}$ (Eqn. 3.16), across several datasets. **(A)** The cumulative distribution of $\widehat{\text{SNR}}$ under the original experimental protocols. Traces with the same line thickness have similar numbers of n and m. Thick line (blue): MT data has $n \approx 10$, $m = 8$. Medium lines (green, orange, red): V4 data has $n \approx 5$, $m \approx 350$. Thin lines: Allen Inst. data has $n \approx 50$, $m \approx 120$. The Allen Inst. data has two recording modalities: extracellular action potentials (spikes) on Neuropixel probes (NP) and two-photon calcium imaging (Ca). Both were recorded for the same stimuli: natural scenes and gratings (see Methods, "Electrophysiological data"). **(B)** Distribution of $\widehat{\text{SNR}}$ after normalization with respect to the duration of the spike counting window (traces for calcium signal are not included). The normalization assumes that the original average spike rate can be applied to a 1 s counting window. But, if firing rates tend to decay over time, this will produce overestimates for recordings shorter than 1 s and underestimates for recordings longer than 1 s.

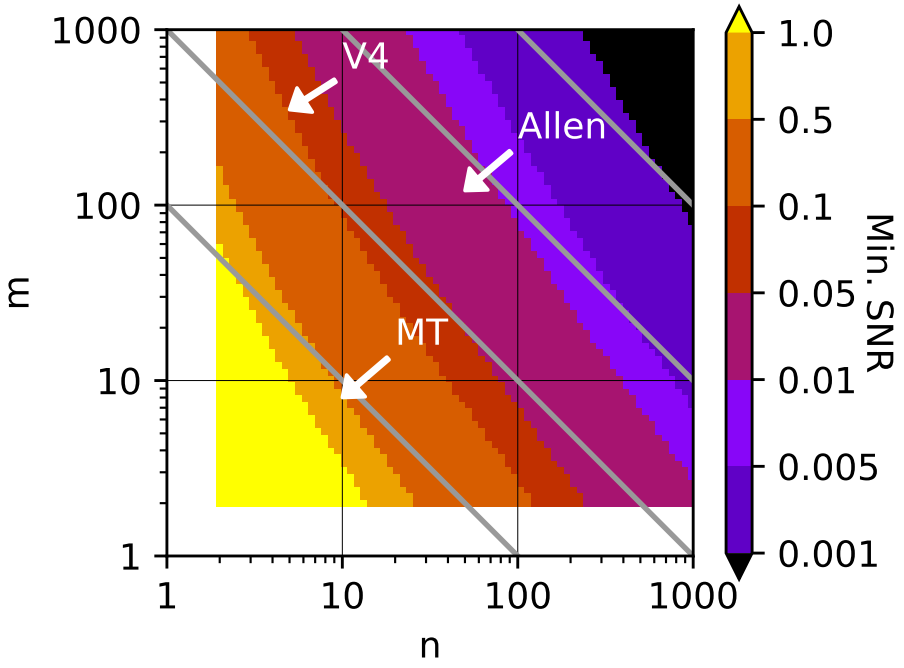


FIGURE 3.13: The minimal SNR needed to reliably detect tuning as a function of m, the number of unique stimuli, and n, the number of repeats of each stimulus. White arrows indicate the approximate location in (m, n) corresponding to the datasets used in Figure 3.12. Gray diagonal lines indicate constant number of total trials ($n \times m$).

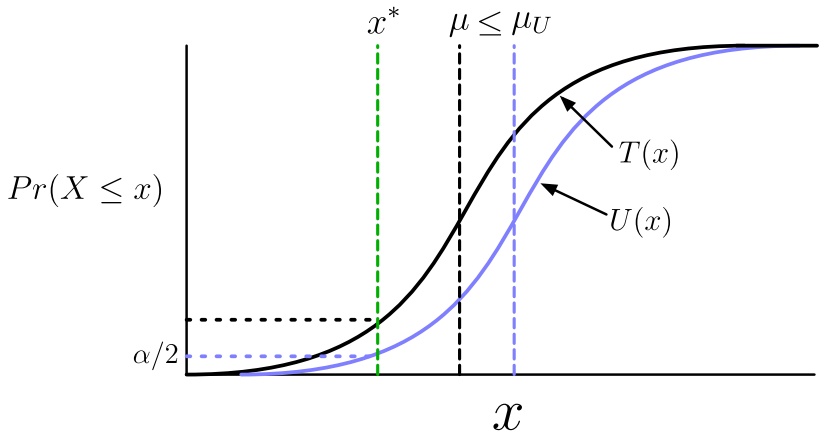


FIGURE 3.14: Illustrative schematic of confidence interval estimation. Given an observed estimate x^* (green dashed vertical) from the distribution of the estimator X with CDF $T(x)$ (solid black curve) associated with the parameter being estimated μ (black dashed vertical), the upper limit of the α-level confidence interval is the μ_U (purple vertical dashed) corresponding to the cumulative distribution of X_U, $U(x)$ (solid purple curve) that would generate values less than x^* with probability $\alpha/2$ (purple horizontal dashed). Thus $U(x)$ is defined by $U(x^*) = \alpha/2$. Under the assumption the family of CDFs of X are stochastically increasing in μ, the event that $T(x) \geq \alpha/2$ corresponds to the event that $\mu < \mu_U$, thus the upper limit of the confidence interval contains the true value of μ. In graphical terms, if the black horizontal dashed line is above the purple, then it is guaranteed that the purple vertical dashed is to the right of the black. Thus these two events have the same probability: $Pr(\mu \leq \mu_U) = Pr(\alpha/2 \leq T(X)) = 1 - \alpha/2$. Here we have used generic symbols for illustrative purposes, but for reference to the proof (see Methods, "Proof of α-level confidence intervals"), the notation used here correspond as follows: $X = \hat{r}^2_{\text{ER}}$, $x^* = \hat{r}^{2*}_{\text{ER}}$, $\mu = r^2_{\text{ER}}$, $\mu_U = r^2_{\text{ER(h)}}$, $T(x) = F(\hat{r}^2_{\text{ER}}|r^2_{\text{ER}})$, and $U(x) = F(\hat{r}^2_{\text{ER}}|r^2_{\text{ER(h)}})$.

Chapter 4

The unbiased estimation of r^2 between tuning curves

4.1 Summary

Pearson's correlation coefficient squared, r^2, is often used in the analysis of neural data to estimate the relationship between neural tuning curves. Yet it is biased by trial-to-trial variability: as trial-to-trial variability increases, measured correlation decreases. Even identically tuned neurons can appear to have tuning curves that are orthogonal. Major areas of research such as invariance and interneuronal signal correlation are confounded by the bias. Here we extend an estimator we developed, $\hat{r}^2_{\text{ER}}$, for estimating model-to-neuron fit to the neuron-to-neuron case. We compare the estimator to a prior method developed by Spearman commonly used in other fields and our method outperforms it. We then apply our estimator to the study of two forms of invariance and demonstrate how it avoids drastic confounds introduced by trial-to-trial variability.

4.2 Introduction

The measurement of correlation is ubiquitous in sensory neuroscience. The r^2 between two sets of neural mean responses is fundamental to many lines of research including the study of: invariance, the maintenance of a tuning curve within a neuron despite the transformation of stimuli [52, 66, 70], functional clustering [101, 102], and signal correlation the degree to which tuning curves between neurons correlate and its consequence for encoding and computation [17].

Yet the typical estimator, $\hat{r}^2$, is fundamentally confounded by the trial-to-trial variability of neural responses. Even if two neurons have identical tuning curves, their measured correlation to each other's responses will decrease as trial-to-trial variability increases.

One approach to the problem is to average over many repeated trials of the same stimulus in order to reduce the influence of trial-to-trial variability. This approach will never wholly remove the influence of noise and its confounding effect. Moreover, the collection of additional trials is expensive.

A more principled approach has been to account for trial-to-trial variability in the estimation of correlation. Correction for the attenuation of the correlation coefficients by measurement error has received considerable attention from fields outside of neuroscience [24–27]. The most popular correction is given by Spearman [23] which multiplies the correlation coefficient by the inverse of its estimated attenuation. Here we follow an approach developed in neuroscience where instead we

remove the bias from the sample covariance and variance separately then form their ratio for the corrected estimate.

Prior work in the neurosciences unbiasing the estimation of the correlation with respect to noise has focused on the problem of correlating models-to-neurons, and a variety of methods have been developed (for review see Pospisil and Bair, 2020 [103]). Currently, there has been no attempt in the neurosciences to address this fundamental problem in the neuron-to-neuron case. This is despite the fact that the major lines of research mentioned above rely on the estimation of the correlation between neurons. Here we address these gaps in methods providing an estimator that outperforms those in other fields.

We call our estimator $\hat{r}^2_{\text{ER}}$ as it estimates the fraction of variance explained between the expected responses (r^2_{ER}) of the neurons or equivalently their tuning curves. We validate this in simulation and show in neural data how it provides insights into two forms of invariance.

4.3 Results

Here we provide the essential intuition into the problem $\hat{r}^2_{\text{ER}}$ solves and its derivation. Consider a typical scenario in sensory neuroscience where the response of two neurons to m stimuli across n trials have been collected and the average of these responses are compared (Figure 4.1). Even if the m expected values of the neurons, μ_i and ν_i (solid green and red trace), perfectly correlated, the m sample averages (Y_i and X_i dashed green and red trace) will deviate from the expected value owing to trial-to-trial variability ($\frac{\sigma^2}{n}$ where σ^2 is trial-to-trial variability). Thus the reported r^2 can be appreciably less than 1 even though in fact the r^2 between the expected values of the neurons is 1 (solid red and green are identical up to a shift and scaling).

The quantity we attempt to estimate here is r^2 between the the expected values (μ_i and ν_i) of the two neuron: the tuning curves. We will call this quantity r^2_{ER} (fraction variance of the 'expected response' explained),

$$r^2_{\text{ER}} = \frac{(\sum_{i=1}^{m}(\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2}{\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 \sum_{i=1}^{m}(\mu_i - \bar{\mu})^2}. \tag{4.1}$$

If we estimate this quantity with the naive sample estimator

$$\hat{r}^2 = \frac{(\sum_{i=1}^{m}(X_i - \bar{X})(Y_i - \bar{Y}))^2}{\sum_{i=1}^{m}(X_i - \bar{X})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2}, \tag{4.2}$$

the expected value of the numerator and denominator of Eqn. 4.2 whose ratio is approximately the expected value of $\hat{r}^2$ (see Methods) is:

$$E[\hat{r}^2] \approx \frac{E[(\sum_{i=1}^{m}(X_i - \bar{X})(Y_i - \bar{Y}))^2}{E[\sum_{i=1}^{m}(X_i - \bar{X})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2]}$$

$$= \frac{\left((\sum_{i=1}^{m}(\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2\right) + \left(\frac{\sigma^2}{n}(\sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 + \sum_{i=1}^{m}(\mu_i - \bar{\mu})^2 + (m-1)\frac{\sigma^2}{n})\right)}{\left(\sum_{i=1}^{m}(\mu_i - \bar{\mu})^2 \sum_{i=1}^{m}(\nu_i - \bar{\nu})^2\right) + \left((m-1)\frac{\sigma^2}{n}(\sum_{i=1}^{m}(\mu_i - \bar{\mu})^2 + \sum_{i=1}^{m}(\nu_i - \bar{\nu})^2 + (m-1)\frac{\sigma^2}{n})\right)}.$$

While the bracketed terms on the left in the numerator and denominator are the same as r^2_{ER} the terms on the right, proportional to the trial-to-trial variability (σ^2) , cause $\hat{r}^2$ to deviate from r^2_{ER}. This is the essential problem: $\hat{r}^2$ is biased away from r^2_{ER} proportional to $\frac{\sigma^2}{n}$ the amount of variability in the average across trials. Even

as $m \to \infty$ the bias in the denominator remains. Thus $\hat{r}^2$ is asymptotically, in m, biased w.r.t r^2_{ER}. The strategy we take to solve this problem is straightforward, find unbiased estimators of these noise terms and subtract them from the numerator and denominator of $\hat{r}^2$:

$$\hat{r}^2_{\text{ER}} = \frac{(\sum_{i=1}^{m}(X_i - \bar{X})(Y_i - \bar{Y}))^2 - \frac{\hat{\sigma}^2}{n}(\sum_{i=1}^{m}(X_i - \bar{X})^2 + \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma}^2}{n})}{\sum_{i=1}^{m}(X_i - \bar{X})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma}^2}{n}(\sum_{i=1}^{m}(X_i - \bar{X})^2 + \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma}^2}{n})},$$

where s^2 is the unbiased estimator for sample variance. The numerator and denominator of the fraction $\hat{r}^2_{\text{ER}}$ are unbiased estimators of the numerator and denominator of r^2_{ER}. A more detailed derivation of this method is given in Methods, 'Unbiasing r^2 of neuron to neuron'.

The results of the paper are as follows. We first validate the method in simulation, then compare it to alternative methods, then validate our method in neural data across responses to the same stimuli on different trials finding the estimator does on average predict a perfect correlation. We then demonstrate how the method avoids meaningful confounds in two applications: the measurement of average population translation invariance, the measurement of single-neuron fill-outline invariance in V4. Finally, we discuss the interpretation of r^2_{ER}, further improvements that can be made to the estimator, and other areas of research in which it could be employed.

4.3.1 Validation of estimator by simulation

Here, in simulation, we demonstrate the effectiveness of the estimator in the neuron-to-neuron case. We assume both neurons have the same trial-to-trial variability (σ^2) and the number of trials but potentially different dynamic ranges.

In the case of estimating the correlation of two neurons, we find both dynamic ranges need to be high to have good estimates (see Methods). We first consider the case where one neuron has a high dynamic range while the other low. We find that the average value of $\hat{r}^2$ is 0.25 when $r^2_{\text{ER}} = 1$ (Figure 4.2A orange) while $\hat{r}^2_{\text{ER}}$ (average 1.02) is the less biased estimator of r^2_{ER} it has become more variable (blue 95% quantile bars are longer than orange).

When both neurons have a high SNR (Figure 4.2B) the difference between the average of $\hat{r}^2_{\text{ER}}$ and $\hat{r}^2$ shrinks commensurate with the increased SNR range.

4.3.2 Comparison to Spearman's correction

Spearman (1904) [23] recognized that the measured strength of a linear relationship between bivariate normal random variable by Pearson's correlation coefficient would be attenuated by measurement error. He provided the asymptotic formula of this attenuation (termed A) and suggested dividing the raw correlation by it to correct for attenuation. Thus if ρ_0 was the correlation between X and Y before the addition of measurement error and ρ the attenuated correlation after:

$$\rho_0 = \rho / A.$$

Spearman did not specify how to estimate the unknown variances in the equation. Adolph and Hardin [27] adopted the formula and took the approach of plugging in the sample variance for both d^2 and σ^2. For comparison to $\hat{r}^2_{\text{ER}}$ we square this estimator and call it $\hat{\rho}^2_0$. While s^2 calculated gives an unbiased estimate of σ^2, s^2 calculated

across trial averages is not an unbiased estimate of d^2. Thus we also provide an unbiased estimate of d^2 (Eqn.) plugging it into Spearman's estimate, we call it $\hat{\rho}^2_{0_{ER}}$ to see if it corrects the biases observed in $\hat{\rho}^2_0$.

We now compare the four estimators described on the basis of their bias in a simulation where we vary the SNR, equal for both neurons, and the number of stimuli m(Figure 4.3). For our initial simulation consider the case of a small number of stimuli ($m = 20$) and judge the bias of the estimators when $r^2_{ER} = 1$. As expected, the naive r^2 is well below 1 and remains below for all levels of SNR consistent with its asymptotic bias (blue trace Figure 4.3A), this bias remains constant across m (B, C). The estimator proposed by Adolph and Hardin $\hat{\rho}^2_0$ is less biased (red trace) but still consistently underestimates and does not improve with m thus is not consistent. The same estimator but with an unbiased estimate of neural dynamic range plugged in, $\hat{\rho}^2_{0_{ER}}$, shows an upward but smaller bias which converges to the truth with m (green). Finally, we see $\hat{r}^2_{ER}$ shows the least bias with small upward bias except in the case of the lower SNR values the estimator becomes unstable but for higher m this is not an issue. For very high m our correction to Spearman's method $\hat{\rho}^2_{0_{ER}}$ and our estimator $\hat{r}^2_{ER}$ are essentially identical (orange and green overlap Figure 4.3C).

4.3.3 Split Trial Validation

We have shown the method works well under simulations that follow the assumptions of its derivation. Neuronal data is not guaranteed to be well approximated by these assumptions. So it is important to test the method for real neuronal data. In most cases, we do not know r^2_{ER} so we cannot be sure whether the estimator is working. One case in which we do know r^2_{ER} is across independent trials of a neuron's responseto the same stimulus. Theoretically, the expected value across trials to the same stimuli is the same thus the pattern of means across stimuli will also be identical thus $r^2_{ER} = 1$ when correlating one subset of trials against another for the same neural recording. Here we fit the odd trials of a neuron's responsesto the even trials with the hope that the method correctly estimates $r^2_{ER} = 1$. We also employ our confidence interval method developed in a previous publication [103]. Our data is taken from Pasupathy and Connor (2001) [41] where 109 cells were shown 370 stimuli 3-5 times. We use 96 of those cells which were shown each stimulus at least 4 times then evaluate our estimators on the 1st and 3rd trial correlated to the 2nd and 4th trial.

We first measured $\hat{r}^2$ for all 96 cells (Figure 4.4A blue points) and plotted them as a function their estimated SNR (across all trials). Despite the theoretical value of $r^2_{ER} = 1$ the unit with the highest SNR (rightmost point) only achieves $\hat{r}^2 = 0.70$. Thus on real neural data $\hat{r}^2$ clearly underestimates r^2_{ER}. The points have a strong increasing relationship with SNR. In essence $\hat{r}^2_{ER}$ was constructed to correct all these points to be distributed about 1. Indeed, in Figure 4.4B the $\hat{r}^2_{ER}$ across the population of cells is approximately centered around 1 with an increasing upward bias for low SNR. The highest SNR unit (Figure 4.4B blue point far right) is estimated to have $r^2_{ER} = 1.01$.

The example plots of odd vs even trials (Figure 4.5) provides intuition into the metric. Inboth cases, all the residual variance in the scatters is attributable to the measured level of trial-to-trial variability. The estimator $\hat{r}^2_{ER}$, in essence, predicts that with more trials we could expect these points will settle onto a line. For lower SNR units the values become unstable many becoming far greater than 2 (cyan points top left) consistent with our simulations 4.3. We now apply $\hat{r}^2_{ER}$ to estimate translation invariance in area V4.

4.3.4 Measuring population translation invariance

Response invariance is the degree to which the pattern of responses of a neuron to a set of stimuli are correlated to the responses when that set is transformed. For example, a neuron with high translation invariance would have the same pattern of responses to a set of stimuli at one position in the receptive field (RF) as they would if the stimuli were shifted. This can be quantified by measuring the correlation between the responses at one position to those at the other. The naive $\hat{r}^2$ will suffer from the confounds we have already discussed such as dependence on SNR. To avoid these confounds we can apply $\hat{r}^2_{\text{ER}}$ estimating the correlation between two sets of neural responses where the responses come from the same neuron but to different stimuli. Here we apply this method to studying translation invariance in V4. We find using the naive estimator $\hat{r}^2$ V4 neurons appear very sensitive to the position of stimuli but $\hat{r}^2_{\text{ER}}$ shows that neurons responses are quite stable for a small shift. We demonstrate how the unbiasedness of the estimator with respect to noise allows us to include even noisy measurements in estimating population invariance.

We measure invariance as the correlation of the tuning curve of a neuron to a set of stimuli at one position to the same set of stimuli but shifted to a different position. More generally invariance involves maintaining the same tuning curve given some transformation of the stimuli. We seek to answer the question what is the average invariance of the V4 neurons in this study? Is tuning typically maintained across the RF or does it change rapidly with the position? The unbiasedness of the metric we have developed is crucial in answering this question: since each individual measurement is unbiased the average across a population will also be unbiased. If V4's tuning is perfectly maintained then we would expect observations of $\hat{r}^2_{\text{ER}}$ of the population to be centered around 1.

Here we reanalyze the data from El-Shamayleh and Pasupathy (2016), recordings of 80 cells shown a set of simple shapes presented at up to 4 positions within the RF of V4 neurons. One potential confound, in this case, is that the RFs of the neurons are diverse (Figure 4.6, grey traces) some shape responses quickly fall off whereas others are maintained. On average the neurons are at less than $\sim$ 80% of mean response for the furthest stimuli. Given our study of SNR it is quite possible that using $\hat{r}^2$ we might observe a fall of in correlation simply because SNR was lower further from the center of the RF (see Figure 4.3 increase in $\hat{r}^2$ (blue trace) with SNR). Indeed $\hat{r}^2$ drops off steeply from the center of the RF (Figure 4.6B avg=0.51 to left of center and 0.48 right of center) to the edge (avg=0.23 left of center). Because of the dependence of this measurement on SNR whether this steep fall off would be observed if more trials were collected remains ambiguous. Applying $\hat{r}^2_{\text{ER}}$ (Figure 4.6C) to each neurons correlation of responses to centered stimuli to shifted stimuli we then average across neurons weighting by the SNR of responses. We see that for the smallest shifts tuning remains quite similar on average (average $\hat{r}^2_{\text{ER}}$ = 0.92 right of center) thus this initial steep drop off was in fact likely just a result of trial-to-trial variability and for a small shift there is a small shift in tuning as would be expected. On the other hand, the steep drop off across the entire RF remains, thus on average for the positions tested tuning changes significantly across positions.

By choosing units with high SNR we can also find units which are truly invariant but also those which change tuning with the position. Examining single traces (Figure 4.7) we find neurons with high SNR, thus well-tuned for the stimuli, which are invariant (blue solid trace neuron 25) and others which are quite sensitive to position (cyan neuron 68). For the same fraction estimated RF shift (-0.15) one cell has near-perfect invariance, despite the RF dropping off by $\sim$ 11% (blue dotted) whereas

our second example cell shares only half the variance with the tuning at the center of the RF (cyan trace) and the RF drops $\sim$ 20%. The important contribution of the $\hat{r}^2_{ER}$ estimator is that it removes ambiguity about whether this difference is because of the lower firing rate of unit 68 at this position. Examining firing rates at this position (Figure 4.6B) for unit 25 there is a strong linear relationship with a residual attributable to trial-to-trial variability whereas for unit 25 there is a change in tuning, one cluster of stimuli lower right cyan halves in the response it evokes at the shifted position whereas another cluster doubles its response.

Thus the metric clearly indicates units with sensitivity, and insensitivity, to stimuli position in addition to giving unbiased averages across the population without needing to resort to removing all but the most well-tuned neurons.

4.3.5 Measuring correlation between tuning for shapes and their outlines

In the prior section, we measured average invariance and the invariance of individual example cells for translations of stimuli within the RF of a neuron. We focus on using CI to select a subset of cells for which invariance can be meaningfully measured. Here we estimate r^2_{ER} of neurons responses to a set of filled shapes and their outlines (Figure 4.8).The goal of the investigators in the original study [70] was to determine whether V4 responses are dictated by shape or if the interior fill is also important. If a neuron was tuned for shape, they hypothesized tuning would be maintained whether the surface of the shape was the same as the outline (fill) or the background (outline). The authors found that most often V4 neurons did not maintain tuning for fill and outline either because one of the stimuli sets did not modulate the cell or both stimuli sets drove the cell but tuning was uncorrelated. Naturally, there is some ambiguity in the second case in terms of how much of the inconsistency of tuning is the result of SNR. Here the distribution of r^2_{ER} is the object of interest thus variability in our estimator can be confounding as it will tend to smear out the distribution. To this end, we can make use of confidence intervals to only select cells whose $\hat{r}^2_{ER}$ are reliable. Thus for this subset, we can be more confident their empirical distribution reflects the true distribution across the population. This criteria naturally removes cells that were not tuned for both sets of stimuli thus focuses on the question of maintained tuning on those cells which were tuned for both fill and outline.

When we examined the distribution of confidence intervals for $\hat{r}^2_{ER}$ we found nearly 40 % of the cells confidence intervals were in $[0, 1]$ (Figure 4.8A) thus provide little inferential power about shared tuning this is in concordance with the investigator's observation that many neurons were not tuned for one of either fill or outline.

Observing the low values of the raw $\hat{r}^2$ across all neurons (Figure 4.6B blue trace) may lead to the conclusion that FO invariance is exceedingly rare. But this estimate does not differentiate between differences in the magnitude of tuning vs selectivity. By selecting $\hat{r}^2_{ER}$ with small confidence intervals the neurons which in fact change the pattern of selectivity across fill and outline are indicated by a low value. Selecting the subset for which $\hat{r}^2_{ER}$ is reliable the distribution (orange trace) is higher and more uniform (median=0.38). We provide example plots of cells with middling $\hat{r}^2$ but high $\hat{r}^2_{ER}$ (Figure 4.9A), middling $\hat{r}^2$ and middling $\hat{r}^2_{ER}$ (B), low $\hat{r}^2$ and low $\hat{r}^2_{ER}$ (C), and finally a confidence interval covering [0,1] because the neuron responded only to filled shapes (D).

Thus by focusing on estimates with small confidence intervals we naturally exclude cells that did not show tuning for one set or both stimuli (which can seriously

bias estimates of shared tuning downward) and cells from the population whose estimates can be meaningfully visualized.

4.4 Discussion

4.4.1 Summary

We have introduced a new estimator of r^2, r^2_{ER} which estimates the fraction of variance shared between the expected value of neural responses across stimuli. We show it has less bias than previous methods of accounting for the attenuation of correlation by measurement noise. We have demonstrated in two different neural data sets that it avoids ambiguity and confounds that can meaningfully change conclusions drawn from data in particular with respect to using the typical r^2. In invariance data we found $\hat{r}^2$ typically grossly underestimated invariance but $\hat{r}^2_{\text{ER}}$ did not. For translation invariance, we showed how the estimator could be used for estimating average population invariance. For fill outline data, we showed how the confidence intervals could be used as a criterion for including estimates. We hope that in future studies these confounds will be considered and this new metric can be used as a tool to address them.

4.4.2 Quantifying invariance

We quantified invariance as the correlation in the tuning curves of a neuron to a set of reference stimuli and that same set but transformed. In our case, the transformation was a translation and then the removal of all but the outline of a shape. This measure is explicitly insensitive to changes in the mean or amplitude of tuning curves. We consider this a 'weak' form of invariance since the spike count across the transformation for a given stimulus can be wildly different. 'Strong' invariance would be if the tuning curve does not change at all across a transformation. One condition in which weak invariance may become strong is if gain control mechanism can remove the difference between the two sets of interleaved stimuli when they are shown in separate blocks [15]. This is a testable hypothesis: do neurons with weak invariance when stimuli are interleaved, achieve strong invariance when those stimuli are blocked. The estimator we have introduced would be crucial to answering this question.

4.4.3 Further work

The correlation coefficient is the basis of several multivariate data analysis methods. Examples include canonical correlation analysis (CCA), representational similarity analysis (RSA), and principal component analysis when the variance is factored out (PCA). With respect to neuroscience the latter is the basis of a popular measure of neural dimensionality and the method of spike-triggered covariance. Further work can study the effect of noise (specifically in shrinking correlation) on the estimation of these quantities and if applying our corrected estimator can improve inference.

Here we have derived for the case of independent samples but often nearby neurons are recorded simultaneously and show correlation across trials, termed noise correlation [17]. Our estimator could be extended to account for the effect of correlated samples.

4.5 Methods and Materials

4.5.1 Simulation procedure

To simulate neuron-to-neuron fits the square root of neural responses for the ith of m stimuli and the jth of n trials ($r_{i,j}$) of the two neurons, termed neuron X and Y, are modeled as independent normally distributed responses:

$$r_{i,j} \sim N[\mu_i, \sigma^2]$$

where variance σ^2 is the same across all $r_{i,j}$. The mean response of the neuron X to the ith stimulus (tuning curve) is $\mu_i = a + b\sin(\frac{i2\pi}{m} + \theta)$ (Figure 4.2 red trace solid dots) whose correlation to neuron Y with mean $\nu_i = \sin(\frac{i2\pi}{m})$ (red trace) are estimated. The true correlation is $r^2_{\mathrm{ER}} = \cos^2(\theta)$.

From this model, we draw n responses for each of the m stimuli (green and red open dots) then apply our estimator to estimate the correlation between neuron X and Y.

4.5.2 Assumptions and terminology for the derivation of unbiased estimators

For this derivation, we assume the responses have undergone a variance stabilizing transform such that trial-to-trial variability is the same across all stimuli. For example, if the neural responses are Poisson distributed, $Y_{i,j} \sim P(\lambda_i)$, where $Y_{i,j}$ is the response to the jth repeat of the ith stimulus, which has expected response λ_i, then a variance stabilizing transform is the square root. In particular, if $Y^*_{i,j} = \sqrt{Y_{i,j}}$, then,

$$\mathrm{E}[Y^*_{i,j}] = \mathrm{E}[\sqrt{P(\lambda_i)}] \approx \sqrt{\lambda_i},$$

and

$$\mathrm{Var}[Y^*_{i,j}] = \mathrm{Var}[\sqrt{P(\lambda_i)}] \approx \frac{1}{4}.$$

The expected value of the transformed response, $Y^*_{i,j}$, still increases with λ_i, whereas the variance is now approximately constant. To improve the estimate of the mean response, n repeats of each stimulus are collected. Invoking the central limit theorem, we can make the approximation:

$$\frac{1}{n}\sum_{j=1}^{n} Y^*_{i,j} = \bar{Y}^*_i \sim N(\sqrt{\lambda_i}, \frac{1}{4n}),$$

where the average across the n repeats is approximately normally distributed with variance decreasing with n. The assumption of a Poisson distributed neural response is not always accurate. A more general mean-to-variance relationship,

$$\sigma^2(\mu) = a\mu^b,$$

can be approximately stabilized to 1 by,

$$f(x) = [\sqrt{a}(1 - \frac{1}{2}b)]^{-1} x^{1-\frac{1}{2}b}.$$

A square root will stabilize any linear mean-to-variance relationship ($b = 1$), but an unknown slope, a, requires that this parameter be estimated. In the case of the

linear relationship, this simply requires taking a square root and then averaging the estimated variance, which is constant, across all stimuli. If it is not reasonable to assume a parametric mean-to-variance relationship and there are enough repeats, one can simply divide all responses to a given stimulus by their sample standard deviation to achieve $\sigma^2 \approx 1$. For the derivation below, we assume that variance-stabilized responses to n repeats have been averaged for each of m stimuli to yield the mean response to the ith stimulus: $Y_i \sim N(\mu_i, \frac{\sigma^2}{n})$, where σ^2 is the trial-to-trial variability and μ_i the ith expected value.

4.5.3 Unbiasing r^2 of neuron to neuron

In the case where both X and Y are equal variance stochastic responses: $X_i \sim N(\nu_i, \frac{\sigma^2}{n})$ and $Y_i \sim N(\mu_i, \frac{\sigma^2}{n})$, we aim to unbias,

$$\hat{r}^2 = \frac{(\sum_{i=1}^m (X_i - \bar{X})(Y_i - \bar{Y}))^2}{\sum_{i=1}^m (X_i - \bar{X})^2 \sum_{i=1}^m (Y_i - \bar{Y})^2},$$

to achieve a corrected version $\hat{r}^2_{\text{ER}}$ such that,

$$E[\hat{r}^2_{\text{ER}}] = r^2_{\text{ER}} = \frac{(\sum_{i=1}^m (\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2}{\sum_{i=1}^m (\nu_i - \bar{\nu})^2 \sum_{i=1}^m (\mu_i - \bar{\mu})^2}.$$

In our approach, we will unbias the numerator and denominator separately.

Unbiased estimate of numerator

The expected value of the numerator is

$$E[(\sum_{i=1}^m (X_i - \bar{X})(Y_i - \bar{Y}))^2] = (\sum_{i=1}^m (\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2 + \frac{\sigma^2}{n}(\sum_{i=1}^m (\nu_i - \bar{\nu})^2 + \sum_{i=1}^m (\mu_i - \bar{\mu})^2 + (m-1)\frac{\sigma^2}{n}),$$

so to unbias the numerator,

$$E[(\sum_{i=1}^m (X_i - \bar{X})(Y_i - \bar{Y}))^2 - \frac{\sigma^2}{n}(\sum_{i=1}^m (\nu_i - \bar{\nu})^2 + \sum_{i=1}^m (\mu_i - \bar{\mu})^2 + (m-1)\frac{\sigma^2}{n})] \tag{4.3}$$

$$= (\sum_{i=1}^m (\nu_i - \bar{\nu})(\mu_i - \bar{\mu}))^2.$$

See Methods, 'Estimators of correction terms', for estimators of the term being subtracted off.

Unbiased estimate of denominator

The denominator is:

$$\sum_{i=1}^m (X_i - \bar{X})^2 \sum_{i=1}^m (Y_i - \bar{Y})^2,$$

and the two terms in the product are scaled non-central chi-squared distributions

$$\sum_{i=1}^m (X_i - \bar{X})^2 \sim \frac{\sigma^2}{n}\chi^2_{m-1}(\frac{n}{\sigma^2}\sum_{i=1}^m (\nu_i - \bar{\nu})^2),$$

and

$$\sum_{i=1}^{m}(Y_i-\bar{Y})^2 \sim \frac{\sigma^2}{n}\chi^2_{m-1}(\frac{n}{\sigma^2}\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2),$$

with expectations

$$E[\sum_{i=1}^{m}(X_i-\bar{X})^2] = E[\frac{\sigma^2}{n}\chi^2_{m-1}(\frac{n}{\sigma^2}\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2)] = \sum_{i=1}^{m}(\nu_i-\bar{\nu})^2+(m-1)\frac{\sigma^2}{n},$$

and

$$E[\sum_{i=1}^{m}(Y_i-\bar{Y})^2] = E[\frac{\sigma^2}{n}\chi^2_{m-1}(\frac{n}{\sigma^2}\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2)] = \sum_{i=1}^{m}(\mu_i-\bar{\mu})^2+(m-1)\frac{\sigma^2}{n}.$$

Since these two random variables are independent the expected value of their product is the product of their expected values,

$$E[\sum_{i=1}^{m}(X_i-\bar{X})^2\sum_{i=1}^{m}(Y_i-\bar{Y})^2] = (\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2+(m-1)\frac{\sigma^2}{n})(\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2+(m-1)\frac{\sigma^2}{n}) =$$

$$[\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2]+[(m-1)\frac{\sigma^2}{n}(\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2+\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2+(m-1)\frac{\sigma^2}{n})].$$

To remove bias from the denominator we subtract off the terms in

$$E[\sum_{i=1}^{m}(X_i-\bar{X})^2\sum_{i=1}^{m}(Y_i-\bar{Y})^2]$$

which are not

$$\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2.$$

Doing so gives,

$$E[\sum_{i=1}^{m}(X_i-\bar{X})^2\sum_{i=1}^{m}(Y_i-\bar{Y})^2-(m-1)\frac{\sigma^2}{n}(\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2+\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2+(m-1)\frac{\sigma^2}{n})]$$

$$=\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2$$

and see Methods, 'Estimators of correction terms', for estimators of the term being subtracted off

Estimators of correction terms

Three of the terms that we use to unbias these estimators: $d_x^2=\sum_{i=1}^{m}(\nu_i-\bar{\nu})^2$, $d_y^2=\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2$, and σ^2 are unknown. Below we provide unbiased estimators of these terms.

In the case of the dynamic range d^2 the naive sample estimator would give,

$$E[\sum_{i=1}^{m}(Y_i-\bar{Y})^2] = E[\frac{\sigma^2}{n}\chi^2_{m-1}(\frac{n}{\sigma^2}\sum_{i=1}^{m}(\mu_i-\bar{\mu})^2)] = \sum_{i=1}^{m}(\mu_i-\bar{\mu})^2+(m-1)\frac{\sigma^2}{n}$$

,

so an unbiased estimator is,

$$\hat{d^2}_{y_{\text{ER}}} = \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\sigma^2}{n}. \tag{4.4}$$

For the case of the sample variance if we have n repeated trials we can calculate sample variance over those trials, then since the variance is the same across stimuli and neurons we can average them for a global estimate,

$$\hat{\sigma}^2 = (\frac{1}{m}\sum_{i=1}^{m} s^2_{i,X} + \frac{1}{m}\sum_{i=1}^{m} s^2_{i,Y})/2.$$

Plugging these in we have the estimator of r^2_{ER} for neuron-to-neuron:

$$\frac{(\sum_{i=1}^{m}(X_i - \bar{X})(Y_i - \bar{Y}))^2 - \frac{\hat{\sigma}^2}{n}(\sum_{i=1}^{m}(X_i - \bar{X})^2 + \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma}^2}{n})}{\sum_{i=1}^{m}(X_i - \bar{X})^2 \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma}^2}{n}(\sum_{i=1}^{m}(X_i - \bar{X})^2 + \sum_{i=1}^{m}(Y_i - \bar{Y})^2 - (m-1)\frac{\hat{\sigma}^2}{n})}.$$

4.5.4 Spearmans correction for attenuation

Spearman (1904) notes that in the case of measurements of two quantities with some underlying 'true' correlation but with additive independent measurement error the measured correlation would tend to be less than the true correlation. He provided the analytic expression for the scaling of attenuation under a bivariate normal model:

$$A = \frac{1}{\sqrt{(1 + \frac{\sigma^2_x}{d^2_x})(1 + \frac{\sigma^2_y}{d^2_y})}},$$

such that the observed correlation was a scaling of the true correlation by A,

$$\rho = A\rho_0.$$

Where σ^2_x and σ^2_y are the variance of the additive measurement error and d^2_x and d^2_y are the variance of the underlying quantities (in our work these are the tuning curves). The method to reverse this attenuation is straightforward, multiply estimated correlation by the inverse of an estimated A.

Spearman does not specify estimators for the unknowns in the attenuation term. Adolph and Hardin [27] use the sample variance of trial-to-trial variability and dynamic range. Thus the estimator takes the form:

$$\hat{\rho}_0 = \sqrt{(1 + \frac{s^2_x/n}{\hat{d}^2_x})(1 + \frac{s^2_y/n}{\hat{d}^2_y})}\frac{\sum_{i=1}^{m}(\bar{X}_{i,\cdot} - \bar{X}_{\cdot\cdot})(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot\cdot})}{\sqrt{\sum_{i=1}^{m}(\bar{X}_{i,\cdot} - \bar{X}_{\cdot\cdot})^2 \sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot\cdot})^2}}$$

where,

$$s^2_x = \frac{1}{m}\sum_{i=1}^{m}\frac{1}{n-1}\sum_{j=1}^{n}(X_{i,j} - \bar{X}_{i,\cdot})^2,$$

and

$$\hat{d}_x^2 = \frac{1}{m-1}\sum_{i=1}^{m}(\bar{X}_{i,\cdot} - \bar{X}_{\cdot,\cdot})^2.$$

This latter term is a biased estimator as shown above thus for an improved estimator we use Eqn 4.4 scaled by $1/m$. We call this estimator $\hat{\rho}^2_{0_{\mathrm{ER}}}$.

4.5.5 Electrophysiological data

To demonstrate our metric we re-analyzed data from three previous single-unit, extracellular studies of parafoveal V4 neurons in the awake, fixating rhesus monkey (Macaca mulattta). Data from the first study, Pasupathy and Connor (2001) [41], consists of the responses of 109 V4 neurons to a set of 362 shapes. There were typically 3-5 repeats of each stimulus and we used only the 96 cells which had 4 repeats for all stimuli. We used the spike count for each trial during the 500 ms stimulus presentation. To estimate translation invariance, we used data from a second study, El-Shamayleh and Pasupathy (2016) [104]. The data from the second study consists of responses of 39 neurons tested for translation invariance. The stimuli were the same types of shapes as the first study, but where the position of the stimuli within the RF was also varied. Each neuron was tested with up to 56 shapes (some of which are rotations of others) presented at 3-5 positions within the RF. Each unique combination of stimulus and RF position was presented for 5-16 repeats, and spike counts were averaged over the 300 ms stimulus presentation. Experimental protocols for both studies are described in detail in the original publications.

4.6 Figures

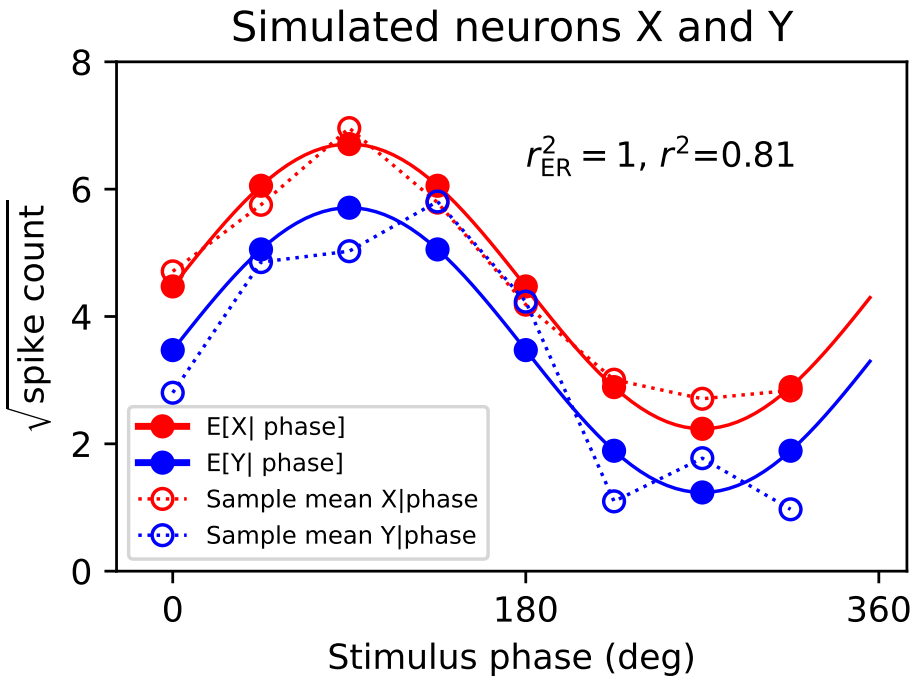


FIGURE 4.1: Simulation model of neuron-to-neuron fits. Here $r^2_{ER} = 1$ because tuning curves (solid trace) are identical up to a shift and scaling. The estimate of correlation from trial averages (open circles) is lower than 1.

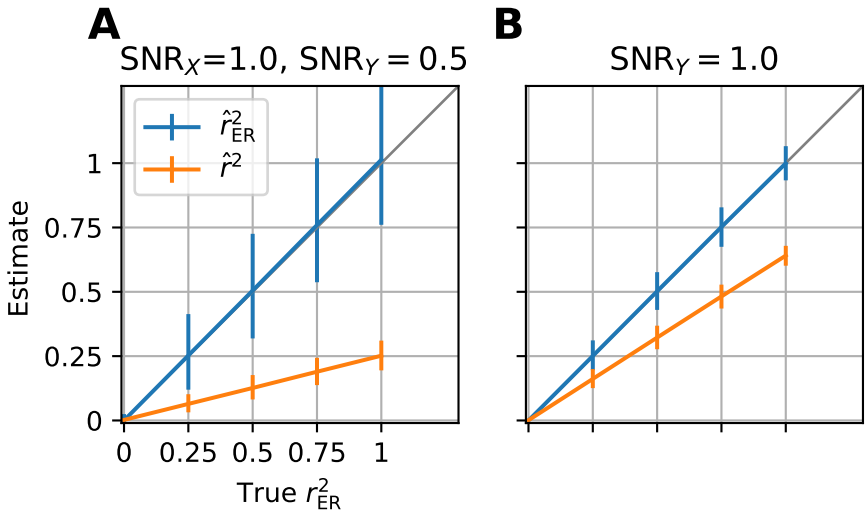


FIGURE 4.2: Simulation of $\hat{r}^2$ (orange) and $\hat{r}^2_{ER}$ (blue) for estimating fit of neuron to neuron at varying levels of r^2 and SNR of neuron y SNR_y while the other neurons $SNR_x = 1$ stays fixed. Vertical bars are 95 % quantiles. **(A)** Simulation at lower SNR, $SNR_y = 0.5$ and $SNR_x = 1$. **(B)** The same simulation as (A) where both neurons have SNR=1.

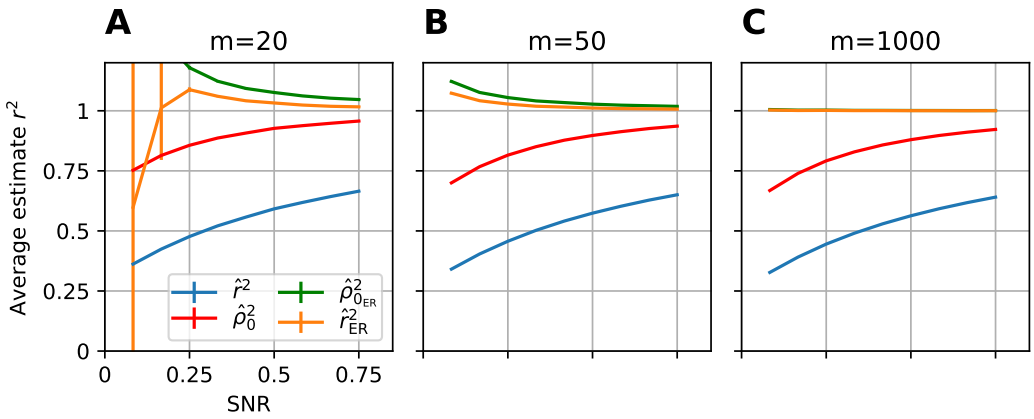


FIGURE 4.3: Comparison of $\hat{r}^2_{ER}$ to alternative methods. The mean and SE (vertical bar) of each estimator is calculated across a simulation of fitting two neurons (see Methods) with 10,000 simulations, $n = 4$. In blue is the naive estimator Pearson's r^2, in red is Spearman's estimator [23] corrected according to methods of Adolph and Hardin [27], in green is Spearman's estimator with unbiased estimator of dynamic range Eqn. 4.4, and in orange is the estimator we use through out the paper, $\hat{r}^2_{ER}$. **(A)** Low number of stimuli (m). **(B)** Intermediate number stimuli. **(C)** High number stimuli.

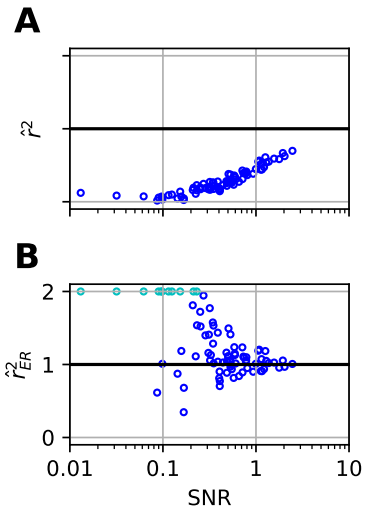


FIGURE 4.4: Motivation and validation of $\hat{r}^2_{\text{ER}}$ on split-half correlation for neural responses. **(A)** The raw $\hat{r}^2$ between the odd and even trials of a neurons responses to a set of 371 stimuli conditions, 96 neurons total. Plotted as a function of the SNR of the neurons calculated across all trials. Theoretically the signal correlation here is 1 (bold black horizontal line) yet even the neuron with the highest SNR is far from 1 with $\hat{r}^2 = 0.7$. **(B)** $\hat{r}^2_{\text{ER}}$ for split half responses (same neurons and responses as B) as function of SNR. Cyan points are estimates that either went above 2 (set to 2) or below 0 (set to 0, none here).

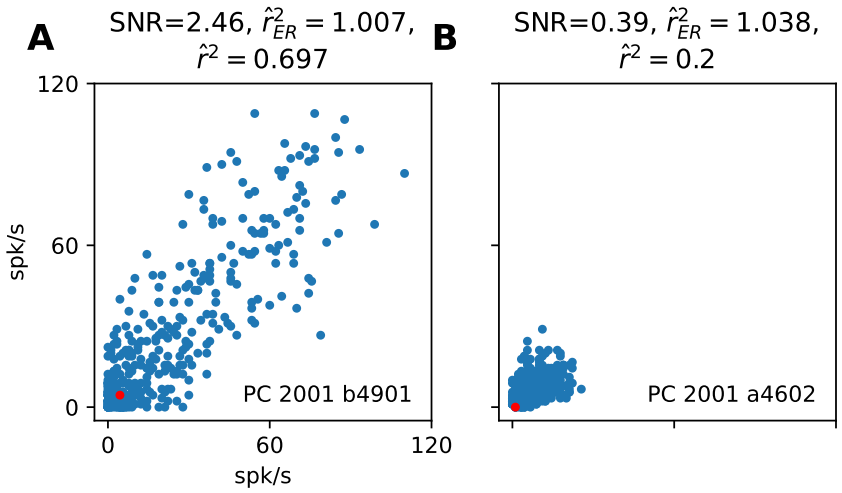


FIGURE 4.5: Example cells from split-trial correlation. **(A)** Example cell with high SNR. Each point is the average of the even (x-axis) and odd (y-axis) spike rate for a single stimulus. In red is the baseline firing rate. SNR is high because of large dynamic range in spike count. Despite having an $\hat{r}^2 = 0.7$ by taking into account its trial-to-trial variability the estimator accurately predicts that the pattern of mean responses with respect to the stimuli are the same $\hat{r}^2_c = 1.01$. **(B)** Example cell with smaller SNR despite $\hat{r}^2$ being lower in this case the estimator predicts that, similarly to (A), the neural responses have the same pattern of means.

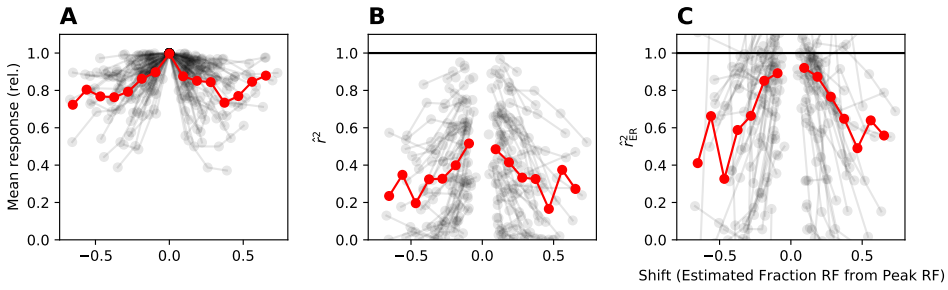


FIGURE 4.6: RFs and translation invariance across 80 cell from El-Shamayleh and Pasupathy (2016). **(A)** Average response across stimuli of each cell at each position of stimuli in RF (grey lines) normalized to peak average response defined to be center of RF. Plotted as function of shift of stimuli in units of fraction of RF. Average across population (red) points are the average across the population in 15 non-overlapping bins of 1/10 RF width, points are at center of bins. **(B)** $\hat{r}^2$ of all cells responses to shifted stimuli to responses to stimuli at center RF (grey). Center is excluded since it is by definition 1. Average across population with same binning as (B) in red. **(C)** $\hat{r}^2_{\text{ER}}$ of all cells (grey) and weighted binned average (red) where points with weights equal to SNR.

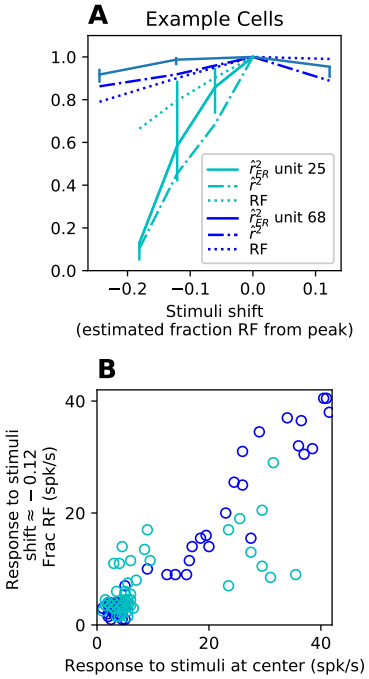


FIGURE 4.7: Example cells of correlation of responses as function of stimuli shift in RF. Examples were chosen for high SNR and high average TI and high SNR and low average TI with measurements at similar position in RF. **(A)** A cell with high TI (blue) maintains a high $\hat{r}^2$ (dash and dot line) across RF (dotted line) and r^2_{ER} is near 1 reflecting near perfect invariance as would be expected for a high $\hat{r}^2$. For a second example cell (cyan) r^2 drops off quickly and r^2_{ER} is similar thus the drop is not the result noisy responses. **(B)** Comparing the responses of the two example cells where the shift at 0.12 fraction of RF (2nd point from left on solid lines in (A)) is plotted against the responses at the center. Unit 25 (blue) shows a strong linear relationship reflecting its high $\hat{r}^2$ whereas for unit 68 one subset of stimuli evoke a higher at response at the shifted position (higher cyan points on left) then at the center and another subset (cyan lower right) evoke a lower response thus tuning is clearly changing with position.

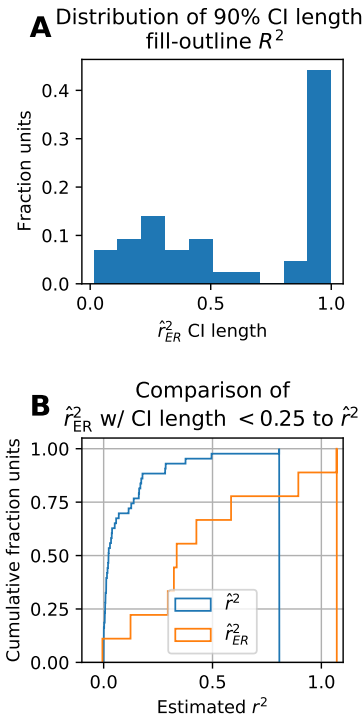


FIGURE 4.8: Comparison of $\hat{r}^2$ and $\hat{r}^2_{\text{ER}}$ for determining strength of correlation between fill and outline responses. **(A)** The distribution of length of 90 % confidence intervals. **(B)** Cumulative distribution of $\hat{r}^2$ (blue) and $\hat{r}^2_{\text{ER}}$ where only $\hat{r}^2_{\text{ER}}$ with an associated confidence interval with length less than 0.5 are included leaving 17 cells of the original 42. Whereas the bulk of the distribution of $\hat{r}^2$ is low (median=0.02) suggesting little to no shared tuning between fill and outline shapes when the large CI units are removed and corrected ($\hat{r}^2_{\text{ER}}$) shared tuning becomes common (median $\hat{r}^2_c = 0.42$) the overall distribution suggesting there is usually at least some shared tuning.

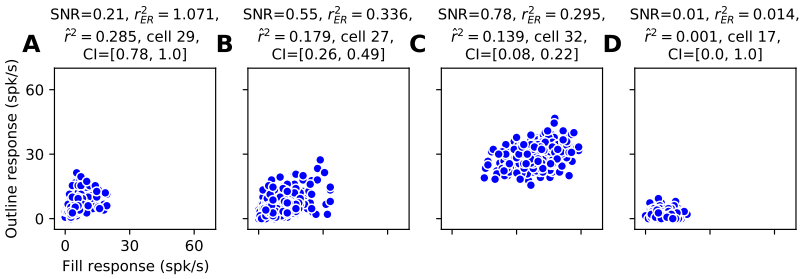


FIGURE 4.9: Example cells responses to fill and outline shape stimuli. **(A)** Cell with lower SNR but high fill-outline invariance. On the basis of $\hat{r}^2 = 0.29$ it might be assumed this has low invariance but given the amount of noise and low dynamic range this correlation is quite high and confidence intervals suggest the true invariance is upwards of 0.7. **(B)** Example of cell with high SNR but middling fill-outline invariance. **(C)** Example of cell with high SNR but little fill-outline invariance. **(D)** Example of cell with low SNR and confidence interval of length 1. Typically of units with long CI's the neuron only evokes strong modulation for one set of stimuli (in this case filled shapes).

Chapter 5

Accounting for biases in the estimation of neuronal signal correlation

5.1 Summary

Signal correlation (r_s) is commonly defined as the correlation between the tuning curves of two neurons and is widely used as a metric of tuning similarity. It is fundamental to how populations of neurons represent stimuli and has been central to many studies of neural coding. Yet the classic estimate, Pearson's correlation coefficient, $\hat{r}_s$, between the average responses to stimuli suffers from confounding biases. The estimate $\hat{r}_s$ can be downwardly biased by trial-to-trial variability and also upwardly biased by trial-to-trial correlation between neurons, and these biases can hide important aspects of neural coding. Here we provide analytic results on the source of these biases and explore them for ranges of parameters typical in electrophysiological experiments. We then provide corrections for these biases that we validate in simulation. We apply these corrected estimators to make a novel observation in area MT: pairs of nearby neurons that are strongly tuned for motion direction tend to have high signal correlation and pairs that are weakly tuned tend to have low signal correlation. We dismiss a trivial explanation for this and discuss potential consequences for encoding whereby the association of signal correlation and tuning strength naturally regularizes the dimensionality of downstream computations.

5.2 Introduction

Signal and noise correlation are fundamental metrics used to measure the relationship between the responses of pairs of neurons [17]. Signal correlation measures how similar the tuning of one neuron is to that of another across a set of stimuli [105], whereas noise correlation measures the relationship between trial-to-trial variability across two neurons. These metrics are often related—pairs of neurons with higher signal correlation tend to have higher noise correlation [93, 106–109]. Furthermore, the interaction between signal and noise correlation is important for determining whether correlation increases or decreases information when considering population codes [18–21], but see [8]. By itself, signal correlation is used for functional clustering [101, 102], measuring invariance [52, 70] , and as a metric of coding redundancy [105, 110, 111]. However, signal correlation has two types of bias, described below, that have not been examined and corrected for in the literature but that could lead to artifacts or obscure important relationships.

Firstly, signal correlation is biased toward zero by independent trial-to-trial noise. This bias, mentioned by Gawne and Richmond (1993) [105], arises when the experimental estimates of two true underlying tuning curves (Figure 5.1A, solid lines, see legend) are independently deformed by noise (dashed lines). Any relationship between the true tuning curves, for example a positive correlation as demonstrated in Figure 5.1B, will be weakened as the estimated means are spread separately along the two response axes (circles indicate independent spread). This bias is large when there are too few repeats of each stimulus or when recordings are excessively noisy. Secondly, signal correlation is biased toward the noise correlation. Noted by Rothschild et al. (2010) [112], this occurs when correlated noise induces spurious correlation between estimated tuning curves (Figure 5.1C, dashed lines). Here, the trial-to-trial noise imparts a tilted elliptical distribution (Figure 5.1D, ovals), strengthening the positive relationship between the estimated means. This bias can inflate the magnitude of signal correlation and spuriously create or inflate the frequently reported positive relationship between naive estimators for signal and noise correlation. Here, we derive an equation that unifies these two biases, determine under what conditions the biases substantially influence results and demonstrate how to avoid or correct the biases using a method for unbiasing Pearson's r^2 [103].

To demonstrate the advantages of using corrected estimators for signal correlation, we reanalyzed data from a study of interneuronal correlation in pairs of simultaneously recorded neurons in the cortical motion area MT [93]. Area MT contains a preponderance of direction selective neurons [113–116], and nearby neurons tend to prefer the same direction of motion [115]. Nevertheless, considerable unexplained diversity in signal correlation exists for neurons recorded at the same location in cortex [93]. Using our corrected estimator, we found a significant positive relationship between signal correlation and tuning curve modulation: neuron pairs with higher signal correlation tended to have tuning curves with higher signal-to-noise ratios and were better fit by a sinusoid compared to pairs with lower signal correlation. This relationship would have been trivial had it been observed using the downwardly noise-biased $\hat{r}^2$, but our corrected estimator avoids this artifact. Thus, diversity in MT direction tuning decreases among the most strongly modulated neurons. In Discussion, we describe other published results that could depend heavily on the ability to eliminate bias in signal correlation.

5.3 Materials and Methods

5.3.1 Stochastic model of neuronal responses

Here we describe our stochastic models of spike counts for pairs of neurons responding to ensembles of stimuli that we use for our derivations and simulations. Rather than considering raw spike count, we consider the square root of spike count, because this is a variance-stabilizing transformation for the Poisson distribution, which is a useful approximation to trial-to-trial variability in neuronal firing. In other words, a Poisson spike count at a high rate has a larger variance than one at a lower rate, but after taking the square root, the variance is approximately equal across different mean firing rates. We modeled the square roots of spike counts of neurons X and Y as $X_{i,j}$ and $Y_{i,j}$ for the presentation of the j^{th} repeat $(j = 1, .., n)$ of the i^{th} stimulus $(i = 1, .., m)$. Let these random variables be constructed as follows:

$$X_{i,j} = a_x + \delta_x T_{i,x}(r_s) + \sigma_x V_{i,j,x}(r_n), \tag{5.1}$$
$$Y_{i,j} = a_y + \delta_y T_{i,y}(r_s) + \sigma_y V_{i,j,y}(r_n), \tag{5.2}$$

where $T_{i,\cdot}$ is the component attributable to tuning curve modulation, i.e., the mean response to stimulus i relative to a baseline offset, a, and $V_{i,j,\cdot}$ is the trial-to-trial variability. The parameters δ and σ set the signal and noise standard deviation, respectively, and we define their squared ratio, δ^2/σ^2, to be the neuronal signal-to-noise ratio (SNR), which will be a critical factor in Results below. The tuning curve modulation is defined as follows,

$$T_{i,x}(r_s) = \sqrt{|r_s|}S_i + \sqrt{1-|r_s|}R_{i,x},$$
$$T_{i,y}(r_s) = \mathrm{sgn}(r_s)\sqrt{|r_s|}S_i + \sqrt{1-|r_s|}R_{i,y},$$

where S_i is the common signal shared between the neurons, $R_{i,x}$ and $R_{i,y}$ are the uncorrelated signal between neurons, and all are standard normal distributions ($N(0,1)$). The parameter r_s sets the level of signal correlation, and the sgn() function implements negative signal correlation by inverting the shared component of tuning. Similarly, the trial-to-trial variability is defined as,

$$V_{i,j,x}(r_n) = \sqrt{|r_n|}C_{i,j} + \sqrt{1-|r_n|}N_{i,j,x},$$
$$V_{i,j,y}(r_n) = \mathrm{sgn}(r_n)\sqrt{|r_n|}C_{i,j} + \sqrt{1-|r_n|}N_{i,j,y},$$

where $C_{i,j}$ is the common noise, $N_{i,j,y}$ and $N_{i,j,x}$ are the independent noise and all are standard normal. The parameter r_n sets the noise correlation.

In most cases, we will use a fixed, sinusoidal tuning curve model. In this case, the tuning curve for neuron X is no longer a function of r_s, it is simply a cosine,

$$t_{i,x} = T_{i,x} = \frac{\cos(\theta_i)}{\sqrt{\sum_{i=1}^{m}\cos(\theta_i)^2}}, \tag{5.3}$$

where the denominator normalizes the length of the tuning vector, and the tuning curve for neuron Y is simply phase shifted to achieve the desired value of r_s, as follows,

$$t_{i,y}(r_s) = T_{i,y}(r_s) = \frac{\cos(\theta_i + \arccos(r_s))}{\sqrt{\sum_{i=1}^{m}\cos(\theta_i + \arccos(r_s))^2}}, \tag{5.4}$$

where,

$$\theta_i = \frac{(i-1)}{m}2\pi.$$

5.3.2 Sample correlation values

Given two sets of responses for two neurons, $X_{i,j}$ and $Y_{i,j}$, for m stimuli and n repeats, the typical estimators, $\hat{r}_s$ and $\hat{r}_n$, for signal and noise correlation based on Pearson's r-value are:

$$\hat{r}_{\mathrm{s}} = \frac{\sum_{i=1}^{m}(\bar{X}_{i,\cdot} - \bar{X}_{\cdot,\cdot})(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})}{\sqrt{\sum_{i=1}^{m}(\bar{X}_{i,\cdot} - \bar{X}_{\cdot,\cdot})^2 \sum_{i=1}^{m}(\bar{Y}_{i,\cdot} - \bar{Y}_{\cdot,\cdot})^2}}, \tag{5.5}$$

$$\hat{r}_{\mathrm{n}} = \frac{1}{m}\sum_{i=1}^{m} \frac{\sum_{j=1}^{n}(X_{i,j} - \bar{X}_{\cdot,j})(Y_{i,j} - \bar{Y}_{\cdot,j})}{\sqrt{\sum_{j=1}^{n}(X_{i,j} - \bar{X}_{\cdot,j})^2 \sum_{j=1}^{n}(Y_{i,j} - \bar{Y}_{\cdot,j})^2}}, \tag{5.6}$$

where the over-bar indicates the mean of the indicated variable computed over all values of each subscript that has been replaced by a dot. For example, $\bar{X}_{\cdot,\cdot}$ is the grand average over all repeats of all stimuli for neuron X.

5.3.3 Simulation of correlation between signal and noise correlation

To model a relationship between signal and noise correlation, we assume that z_n and z_s, the Fisher-z transformations, $z = \frac{1}{2}\ln[(1+r)/(1-r)]$, of signal and noise correlation, respectively, are bi-variate normal random variables:

$$z_n \sim N(\mu_n, \sigma_n), \;\; z_s \sim N(\mu_s, \sigma_s)$$

$$\mathrm{Corr}(z_n, z_s) = r_{\mathrm{NS}}. \tag{5.7}$$

We use the inverse Fisher transformation to ensure the support of r_{n} and r_{s} is in $[-1, 1]$. The correlation of z_n and z_s, r_{NS}, will give similar values to correlating r_{n} and r_{s}, especially if the bulk of the distribution is in $[-0.7, 0.7]$. The transformed correlation z_n and z_s, since Fisher's z is a variance stabilizing transform, will have the advantage of asymptotically known and equal variance across different levels of correlation. The estimate $\hat{r}_{\mathrm{NS}}$ is Pearson's correlation coefficient between the Fisher transformed estimates of signal and noise correlation, respectively $\hat{z}_s$ and $\hat{z}_n$, across a sample of neurons.

5.3.4 Derivation of signal correlation estimate for fixed stimuli

The calculation of the asymptotic value of signal correlation in the case where stimuli are considered random is straightforward and given in Eqn. 5.9. The derivation in the case of fixed stimuli is more complex. We take the approach of separately calculating the expected value of the numerator and denominator of estimated signal correlation (Eqn. 5.5) then taking their ratio as an approximation to the expected value of estimated signal correlation. For reference, we note the relevant moments, which follow from the definitions above: $\mathrm{E}[\bar{X}_{i,\cdot} - a_x] = t_{i,x}$, $\mathrm{E}[\bar{Y}_{i,\cdot} - a_y] = t_{i,y}$, $\mathrm{Var}[\bar{Y}_{i,\cdot}] = \sigma_y^2/n$, $\mathrm{Var}[\bar{X}_{i,\cdot}] = \sigma_x^2/n$, $\mathrm{Corr}[\bar{Y}_{i,\cdot}, \bar{X}_{i,\cdot}] = r_{\mathrm{n}}$.

Now consider the numerator of estimated signal correlation, the sample covariance between $\bar{X}_{i,\cdot}$ and $\bar{Y}_{i,\cdot}$:

$$\mathrm{E}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot} - a_x)(\bar{Y}_{i,\cdot} - a_y)] = \sum_{i=1}^{m}\mathrm{E}[(\bar{X}_{i,\cdot} - a_x)(\bar{Y}_{i,\cdot} - a_y)]$$

$$= \sum_{i=1}^{m}\mathrm{Cov}[(\bar{X}_{i,\cdot} - a_x), (\bar{Y}_{i,\cdot} - a_y)] + \mathrm{E}[\bar{X}_{i,\cdot} - a_x]\mathrm{E}[\bar{Y}_{i,\cdot} - a_y]$$

$$= m\frac{\sigma_x\sigma_y r_{\mathrm{n}}}{n} + \sum_{i=1}^{m} t_{i,x}t_{i,y},$$

thus we see the numerator is increasing in r_n similarly to the random stimuli case (Eqn. 5.9).

For the denominator of the sample signal correlation, we consider the product of the sample variances of $\bar{X}_{i,\cdot}$ and $\bar{Y}_{i,\cdot}$:

$$\mathrm{E}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2]$$
$$=\mathrm{Cov}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2,\ \sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2]+\mathrm{E}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2]\mathrm{E}[\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2)].$$

We first find $\mathrm{E}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2]$ and $\mathrm{E}[\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2)]$,

$$\mathrm{E}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2]=\sum_{i=1}^{m}\mathrm{E}[(\bar{X}_{i,\cdot}-a_x)^2]=\sum_{i=1}^{m}\mathrm{E}[(\bar{X}_{i,\cdot}-a_x)]^2+\sum_{i=1}^{m}\mathrm{Var}[(\bar{X}_{i,\cdot}-a_x)]^2=\sum_{i=1}^{m}t_{i,x}^2+m\frac{\sigma_x^2}{n},$$

and similarly,

$$\mathrm{E}[\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2]=\sum_{i=1}^{m}t_{i,y}^2+m\frac{\sigma_y^2}{n}.$$

For $\mathrm{Cov}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2,\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2]$ we note only when $i=i'$ do the terms in the two sums depend on each other thus we can move the summation out of the covariance:

$$\mathrm{Cov}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2,\ \sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2)]=\sum_{i=1}^{m}\mathrm{Cov}[(\bar{X}_{i,\cdot}-a_x)^2,\ (\bar{Y}_{i,\cdot}-a_y)^2]$$

we then make use of the fact that for a bivariate normal,

$$\mathrm{Cov}[X^2,Y^2]=4\mu_x\mu_y\rho\sigma_x\sigma_y+2\rho^2\sigma_x^2\sigma_y^2.$$

Plugging the relevant moments in,

$$\sum_{i=1}^{m}\mathrm{Cov}[(\bar{X}_{i,\cdot}-a_x)^2,(\bar{Y}_{i,\cdot}-a_y)^2]=4r_\mathrm{n}\frac{\sigma_x\sigma_y}{n}\sum_{i=1}^{m}t_{i,x}t_{i,y}+2m(r_\mathrm{n}\frac{\sigma_x\sigma_y}{n})^2.$$

Summing the covariance and product of expected values together:

$$\mathrm{E}[\sum_{i=1}^{m}(\bar{X}_{i,\cdot}-a_x)^2\sum_{i=1}^{m}(\bar{Y}_{i,\cdot}-a_y)^2]$$
$$=4r_\mathrm{n}\frac{\sigma_x\sigma_y}{n}\sum_{i=1}^{m}t_{i,x}t_{i,y}+2m(r_\mathrm{n}\frac{\sigma_x\sigma_y}{n})^2+(\sum_{i=1}^{m}t_{i,x}^2+m\frac{\sigma_x^2}{n})(\sum_{i=1}^{m}t_{i,y}^2+m\frac{\sigma_y^2}{n}).$$

Thus our approximation of the expected value as the ratio of the expected values of the numerator and denominator of $\hat{r}_\mathrm{s}$ in the case of fixed stimuli is:

$$\mathrm{E}[\hat{r}_\mathrm{s}]\approx\frac{\sum_{i=1}^{m}t_{i,x}t_{i,y}+m\frac{\sigma_x\sigma_y r_\mathrm{n}}{n}}{\sqrt{4r_\mathrm{n}\frac{\sigma_x\sigma_y}{n}\sum_{i=1}^{m}t_{i,x}t_{i,y}+2m(r_\mathrm{n}\frac{\sigma_x\sigma_y}{n})^2+(\sum_{i=1}^{m}t_{i,x}^2+m\frac{\sigma_x^2}{n})(\sum_{i=1}^{m}t_{i,y}^2+m\frac{\sigma_y^2}{n})}},\tag{5.8}$$

where as a last approximation we take the expected value of the square root of the denominator to be the square root of the expected value of the denominator.

5.3.5 Electrophysiological data

We reanalyzed data taken from a previous electrophysiological study where pairs of neurons were isolated and recorded on a single electrode in two awake, fixating macaques [92, 93]. Specifically, we reexamine the data for eight-point direction tuning curves in response to coherently moving dots for 81 pairs of MT neurons. For a detailed description of the visual stimuli, electrophysiological methods and data set, see [93].

We also analyzed the publicly available data (http://dx.doi.org/10.6080/K0NC5Z4X) of Kohn and Smith (2016) [117] from V1 of anesthesized macaque monkeys. This data set includes the spiking activity from 3 monkeys (n=106, 88, and 112 single- and multi-unit recordings) in response to drifting sinusoidal gratings across 12 directions (1.28 s presentation).

In addition, we analyzed simultaneously recorded responses from the awake mouse, a detailed description of which is given in: Allen Brain Observatory (2016) [96]. In the analyses here, we examined single-unit spiking activity from 75 neurons recorded in VISp of mouse visual cortex in response to a set of 118 natural images. Each natural image was presented for 0.25 s in random order with no intervening blank to achieve ∼50 repeats. For details of electrodes and spike sorting, see the Allen Brain Observatory website.

5.4 Results

Our results are organized into five sections. First, we describe analytically the origin of the biases in signal correction; next we describe how to correct for these biases. Third, we examine how the correlation between signal and noise correlation can become inflated spuriously. Fourth, we demonstrate the presence of these confounds in the analysis of neuronal data. Finally, we describe a novel result obtained by using an unbiased estimator of signal correlation.

5.4.1 Signal correlation confounds

The de facto estimators of signal ($\hat{r}_s$) and noise correlation ($\hat{r}_n$) suffer from the following two confounds:

1. $\hat{r}_s$ can be downwardly biased by trial-to-trial variability (Figure 5.1, top row).
2. $\hat{r}_s$ can be upwardly biased by r_n (Figure 5.1, bottom row) and can create a spurious positive relationship between $\hat{r}_s$ and $\hat{r}_n$ (Eqns. 5.5 and 5.6).

Here we describe the analytic relationship between the typical estimator, $\hat{r}_s$, of signal correlation and three confounding factors: (1) noise correlation, r_n, (2) the number of repeats, n, of each stimulus and (3) the signal-to-noise ratio (SNR), which we define below. These results assume stimuli are randomly sampled but the fixed case is similar (see Methods: Derivation of signal correlation estimate for fixed stimuli). As m becomes large the estimator $\hat{r}_s$ converges to the correlation between $\bar{X}_{i,\cdot}$ and $\bar{Y}_{i,\cdot}$ (using the dot notation of Eqn. 5.5):

$$\hat{r}_s \xrightarrow{m} \frac{\delta_x \delta_y r_s + \sigma_x \sigma_y r_n / n}{\sqrt{(\delta_x^2 + \sigma_x^2/n)(\delta_y^2 + \sigma_y^2/n)}}.$$

Rather than converging to r_s, the estimator depends on other factors. With the simplifying assumption that the dynamic ranges and noise amplitudes are the same for each neuron, i.e., $\delta_x = \delta_y = \delta$ and $\sigma_x = \sigma_y = \sigma$, the above can be reduced to,

$$\hat{r}_s \xrightarrow{m} \frac{\delta^2 r_s + \sigma^2 r_n / n}{\delta^2 + \sigma^2 / n}.$$

This can be rewritten in terms of the product of n and the SNR, defined as δ^2/σ^2, by defining $\lambda = n\delta^2/\sigma^2$, so that,

$$\hat{r}_s \xrightarrow{m} \frac{r_s + r_n/\lambda}{1 + 1/\lambda}. \tag{5.9}$$

As λ increases, reflecting more signal in the data, the estimator approaches the true r_s. As λ decreases, the estimator is increasingly biased towards r_n, which can be an upward or downward bias depending on the relative values of r_s and r_n.

The nature of this bias is depicted in Figure 5.2, which plots $\hat{r}_s$ as a function of the true noise correlation, r_n, for several values of r_s (columns), number of repeats n (rows) and noise levels (higher noise purple). When $r_s = 0$ (upper left panel), the lines should be flat at 0, but instead are biased toward r_n (they have a positive slope) by an amount proportional to r_n. The higher the noise, the larger the bias (steeper slopes). Increasing n (going down the left column of Figure 5.2) decreases the slope, as does increasing the SNR (going from purple to yellow; see Eqn. 5.9). For higher values of r_s (middle and right columns), the same curves from the left column are simply translated so that their intersection (the only non-biased point) slides rightward up the main diagonal to $r_n = r_s$. When signal and noise correlation are equal, there is no bias at any noise level. In summary, $\hat{r}_s$ is biased away from the true r_s and toward r_n, and this bias becomes substantial for low numbers of repeats and when tuning curve modulation is weak relative to trial-to-trial noise. It cannot be overcome by increasing the number of stimuli, m. We now propose and validate a solution to these biases.

5.4.2 Corrected estimator of signal correlation

A simple strategy to remove the bias towards r_n is to compute r_s based on repeats that are not recorded simultaneously. After all, r_s is a measure of tuning curve similarity, and tuning curves for different neurons do not have to be acquired at the same time. Specifically: the estimated tuning curves from the odd repeats for one neuron and those from the even repeats for the other can be used to measure $\hat{r}_{s(1)}$ and vice-versa for $\hat{r}_{s(2)}$, then the two estimates can be averaged. This estimator has the benefit that one can now seek to determine whether signal and noise correlation are in fact related across neurons in the absence of this otherwise built-in bias toward noise correlation.

The above strategy means that r_n is now zero, thus leaving $\hat{r}_s$ (or $\hat{r}_{split}$) with a bias toward zero that increases with noise:

$$\hat{r}_s \xrightarrow{m} \frac{r_s}{1 + 2/\lambda},$$

where the "2" in the denominator reflects that n has been cut in half. To remove this bias toward 0, we use the estimator $\hat{r}^2_{\text{ER}}$ of Pospisil and Bair (2020), which estimates the correlation between the 'true' tuning curves of the two neurons (Figure 5.1A, solid blue and red traces). More formally, it estimates the r^2 between the expected values of the two neuronal responses, $t_{i,x}$ and $t_{i,y}$ (Eqns. 5.3 and 5.4). This quantity r^2_{ER}, is the explained fraction of variance of the expected response (ER):

$$r^2_{\text{ER}} = \frac{(\sum_{i=1}^{m}(t_{i,x} - \bar{t}_{\cdot,x})(t_{i,y} - \bar{t}_{\cdot,y}))^2}{\sum_{i=1}^{m}(t_{i,x} - \bar{t}_{\cdot,x})^2 \sum_{i=1}^{m}(t_{i,y} - \bar{t}_{\cdot,y})^2}.$$

Whereas the naive r^2 is heavily biased downwards by trial-to-trial variability, $\hat{r}^2_{\text{ER}}$ converges to the true r^2_{ER} as the number of stimuli $m \to \infty$. For finite m, $\hat{r}^2_{\text{ER}}$ has little bias relative to r^2 (Figure 5.3 shows validation on simulated data). We make a small modification to $\hat{r}^2_{\text{ER}}$ to account for noise correlation by performing the same splitting procedure proposed above and call this estimator $\hat{r}^2_{\text{ER}_{\text{split}}}$.

Correction for the attenuation of correlation coefficients by measurement error has received considerable attention from fields outside of neuroscience [24–27]. The most popular correction is given in Spearman, 1904 [23]. An additional correction for noise correlation is given in Saccenti et al. [118]. Rothschild et al. (2010) [112], to our knowledge, are the first to recognize the confound of noise correlation inflating $\hat{r}_{\text{NS}}$ and provide an estimate of signal correlation to account for this. The statistical properties of this potentially useful estimator are not well studied (e.g. bias, variance and asymptotic properties) and it is not clear how to extend it to account for the downward bias of trial-to-trial variability. In this paper we use trial-splitting to remove the effect of noise correlation for its simplicity and $\hat{r}^2_{\text{ER}}$ because we have studied its statistical properties [103] and have provided validated confidence intervals, which prior methods lack. We next validate our estimators across a range of typical experimental parameters.

In Figure 5.3 (upper left panel) we plot the estimated signal correlation squared as a function of the true signal correlation squared (r^2_{s}) in the case where there is no noise correlation ($r_{\text{n}} = 0$) and SNR is low (0.1). We evaluate the estimator at 5 levels of true signal correlation ($r^2_s = [0, 0.25, 0.5, 0.75, 1]$). The naive estimator (blue) shows a strong downward bias where for example when $r^2_s = 1$ the average $\hat{r}^2$ is less than 0.25. Thus two neurons with identical tuning would be reported as having little shared tuning. When the split-trial estimator, $\hat{r}_{\text{split}}$, is applied (orange) the downward bias is stronger because correlation is being estimated across fewer repeats. On the other hand, $\hat{r}^2_{\text{ER}}$ (green trace) on average estimates the true value (green trace lies on diagonal) though is highly variable because of low SNR (vertical bars indicate SD). For example when $r^2_s = 1$ the average $\hat{r}^2_{\text{ER}}$ is 1.01 though SD is 0.12. The slight over estimation is the result of the low SNR (see [103]). Estimates could be truncated to remove impossible values of correlation ($r_s > 1$), but this would introduce an even greater downward bias. The split-trial version of this estimator $\hat{r}^2_{\text{ER}_{\text{split}}}$ shows slightly worse performance where it overestimates (average $\hat{r}^2_{\text{ER}_{\text{split}}} = 1.1$ when $r^2_s = 1$) because the estimate over fewer repeats essentially has lower SNR.

When noise correlation is introduced (Figure 5.3 upper middle panel, $r_{\text{n}} = 0.25$) we observe the expected increase in $\hat{r}^2_{\text{s}}$ (blue trace is higher than in left column) as it is now biased toward 0.25^2 rather than 0. This naive estimator gets closer to the true value but only because it is being spuriously inflated. Importantly, $\hat{r}^2_{\text{split}}$ does not change across columns (orange traces remain identical), demonstrating its immunity to noise correlation. Applying $\hat{r}^2_{\text{ER}}$ without splitting trials creates a large upward bias (green trace above diagonal) because its underlying assumption of independent

trial-to-trial variability is violated when $r_n \neq 0$. Whereas, $\hat{r}^2_{ER_{split}}$ (red trace), like $\hat{r}^2_{split}$ is not influenced by r_n (red trace same across columns) because the assumption of independent noise is correct across split trials. The differences between estimators described for the top row in Figure 5.3 are similar, but progressively diminished as SNR increases for the middle (SNR=0.5) and lower (SNR=1.0) rows, as all curves move towards the diagonal. Overall, $\hat{r}^2_{ER_{split}}$ (red) shows the least bias, upward or downwards, of the estimators of signal correlation considered here.

A clear disadvantage of $r^2_{ER_{split}}$ is that it tends to be more variable than $\hat{r}_s$. This is one justification for using $\hat{r}_s$, split or not, when estimating r_{NS} as it reduces the downward bias associated with sampling variability (discussed below). Thus, in our results on estimation of r_{NS} we use $\hat{r}_s$.

5.4.3 Spurious correlation between signal and noise correlation

The inflation of $\hat{r}_s$ by r_n can lead to a spurious positive relationship between the estimators $\hat{r}_s$ and $\hat{r}_n$. A positive relationship between signal and noise correlation has been observed across many experiments and is important to the theory of neural population coding. Thus, it is critical to ensure it is not a spurious relationship. Here we perform a series of simulations where the correlation between $\hat{r}_s$ and $\hat{r}_n$ is measured across a population of neurons, evaluate the severity of the confound for the classic estimators and then validate that the split-trial estimator removes the confound.

An example of a simulation where a spurious relation between $\hat{r}_s$ and $\hat{r}_n$ is induced is shown in Figure 5.4. We begin by drawing the true signal and noise correlation between 50 pairs of neurons from a bivariate normal distribution such that r_s and r_n are independent (Figure 5.4A, see legend for simulation details). We set the SNR to be low (0.1), the condition under which r_n has the greatest influence on $\hat{r}_s$. Indeed, when we simulate the responses from the pairs of neurons and plot the naive signal correlation estimate against the true noise correlation (Figure 5.4B), there is a substantial correlation, 0.42, despite there being no relationship between the true r_s and r_n. This is because $\hat{r}_s$ is biased toward r_n (Eqn. 5.9); the average slope of this relationship (as calculated in Eqn. 5.8) is plotted for reference in Figure 5.4B. The relationship is not perfectly linear in part because $\hat{r}_s$ is a variable estimate. This variability is reflected in the relationship between the estimated $\hat{r}_s$ and the true r_s having a weak relationship further weakened by the perturbation of r_n (Figure 5.4C). The estimate of r_n, $\hat{r}_n$, is not confounded as $\hat{r}_s$ but is highly variable (Figure 5.4D weak relationship in scatter). Finally when plotting $\hat{r}_s$ and $\hat{r}_n$ (Figure 5.4E) we see there is a relationship between the estimators ($r_{NS} = 0.38$) that does not exist in the true parameters.

Alternatively, because the estimates $\hat{r}_s$ and $\hat{r}_n$ are themselves noisy, there can be an underestimation of r_{NS}. This is similar to the downward bias in $\hat{r}_s$. We present an example of this in Figure 5.5 where we set the true correlation between signal and noise correlation $r_{NS} = 0.8$ across the population, and we set SNR to be high so there is little influence of noise correlation on signal correlation. Thus, the relationship between $\hat{r}_s$ and r_n is strong, not because of the influence of r_n (Figure 5.5B, small slope of line implies weak relationship), but because there is a true underlying relationship. Yet both estimators are noisy (Figure 5.5C and D estimates scattered about diagonal) and thus the observed correlation between the estimators, $\hat{r}_{NS} = 0.54$ (Figure 5.5E), undershoots the true value, 0.8. Thus $\hat{r}_{NS}$ can suffer from both an upward

and downward bias. Below we outline the parameter ranges where these biases remain and show how a split-trial estimator removes the upward but not downward bias.

We performed simulations like those in Figures 4 and 5 over a wide range of the key parameters n, m, and SNR and plotted $\hat{r}_{NS}$ for each simulated population of pairs using the classical $\hat{r}_s$ (solid trace) and the split-trial estimator $\hat{r}_{s_{split}}$ (dashed trace) (Figure 5.6). First, we set SNR far higher than any observed in our neural data so that the confound of noise correlation on signal correlation is weak (Figure 5.6A). When $r_{NS} = 0$, the estimator is also close to zero (light blue trace) thus the confound of noise correlation inflating estimation of signal correlation is removed because of high SNR. Yet for all other levels of correlation, r_{NS} is underestimated inversely proportional to m (blue, purple, pink increasing with m to true values). This is because the variability of both $\hat{r}_s$ and $\hat{r}_n$ are inversely proportional to m and thus with higher m there is less sampling noise corrupting the correlation between $\hat{r}_s$ and $\hat{r}_n$ across the population. In the case where SNR is low (Figure 5.6B) $\hat{r}_{NS}$ is systematically biased for all levels of correlation including $r_{NS} = 0$. Here the low SNR allows r_n to inflate $\hat{r}_s$, and increasing m reduces the ability of sampling noise to attenuate this effect. Employing the split-trial estimator removes the effect of noise correlation and $\hat{r}_{NS}$ converges to the true value while the classic $\hat{r}_{NS}$ does not. Increasing n (Figure 5.6C) for lower r_{NS} (0, 0.25), the split-trial estimator converges more quickly to the true value (compare light blue and dark blue solid traces to dashed). For higher r_{NS} inflation by noise correlation balances out attenuation by variability thus $\hat{r}_{NS}$ is closer to the true value. The effect of increasing SNR (Figure 5.6D) is largely equivalent to that of increasing n because both shrink the effect of trial-to-trial variability. At all levels of correlation (separate traces) the estimated correlation between signal and noise correlation decrease as SNR increases. For $\hat{r}_{NS}$ using $\hat{r}_{s_{split}}$ there is no longer the decreasing relationship (dash lines rise instead of fall with SNR). Thus split-trial estimation of signal correlation removes the upward bias from $\hat{r}_{NS}$ which is worst for low n and SNR. High m reduces sampling variability of $\hat{r}_s$ and $\hat{r}_n$ thus can increase the spurious upward bias.

5.4.4 Demonstration of confounds in neural data

We have demonstrated the confounds of signal correlation both theoretically and in simulation. Here we demonstrate these confounds in neural data. We first examine the downward bias of signal correlation caused by trial-to-trial variability and then the inflation of $\hat{r}_{NS}$.

To demonstrate the effect of trial-to-trial variability in attenuating $\hat{r}^2$ and the efficacy of $\hat{r}^2_{ER}$ to correct this bias, we estimate the correlation between tuning curves computed from the odd stimulus repeats and those computed from the even repeats for the same neuron. In this case, the signal correlation is 1 by definition because the expected value of the response to each stimulus is the same across repeats. In Figure 5.7A we plot $\hat{r}^2$ between the even-repeat and odd-repeat direction tuning curves for 81 MT neurons as a function of SNR. The positive relationship (Spearman's rank-order r=0.91, p«0.001) is consistent with the theoretical curves in Figure 5.3 (left column across rows) when $r_s^2 = 1$ and $r_n = 0$ where the blue trace increases with SNR. Thus, even though signal correlation is 1, the classic estimator is well below 1 for the many MT neurons that have low SNR. When $\hat{r}^2_{ER}$ is applied to the same responses, it on average corrects for the bias (Figure 5.7B points clustered around horizontal line at 1; Spearman's r=-0.16, p=0.16). In summary, for units with low SNR in this data set, the signal correlation is underestimated and the estimator $\hat{r}^2_{ER}$ corrects this bias.

Next, we demonstrate the effect of noise correlation inflating $\hat{r}_{NS}$ in neural data and its relation to the number of repeats, n. To reveal the influence of noise correlation on $\hat{r}_{NS}$, we measure $\hat{r}_s$ using tuning curves computed from non-simultaneous trials ("split-trial" condition, see Figure 5.8 legend for details) and compare this to the "same-trial" condition where simultaneous trials are used. In the split-trial condition, noise correlation effectively has no influence because the trials are not simultaneous. Estimated noise correlation, $\hat{r}_n$, was computed across all trials. Figure 5.8A shows that same-trial estimates of $\hat{r}_{NS}$ (orange trace) are consistently higher than split-trial estimates (blue trace). The estimates converge as n increases (although the change in n is limited here), consistent with the simulations in Figure 5.6C. The inflation of $\hat{r}_{NS}$ is modest relative to the strength of the correlation itself, nevertheless this demonstrates that the effect occurs in neural data even when the SNR is relatively high. We next consider a lower SNR dataset.

In a previous comparison of SNRs across data sets [103], we observed that the SNR of the Allen Institutes Neuropixel recordings from area VISp of mouse visual cortex was substantially lower than that of the MT data analyzed here (respective median SNRs = 0.16 and 4.0). We repeated the above analysis with this lower SNR data set and again found that $\hat{r}_{NS}$ for the same-trial condition was higher than that for the split-trial condition (Figure 5.8B). Consistent with simulations in Figure 5.6D, the difference in the split- vs. same-trial conditions was much greater in the lower SNR case. Thus, we have shown in neural data that the influence of noise correlation on $\hat{r}_{NS}$, when it is not accounted for by using non-simultaneous trials to compute $\hat{r}_s$, can vary greatly depending on SNR. Comparing the two methods of estimation as we have done can help assess the magnitude of the confound.

In the Discussion, we cover a variety of examples where controlling for SNR could potentially make a difference in the scientific conclusions of published studies.

5.4.5 Novel relationship between tuning strength and signal correlation in area MT

Applying the improved estimators described above, we reanalyzed direction tuning data from a study of signal and noise correlation in simultaneously recorded pairs of well-isolated MT neurons in awake macaques [92, 93]. The data set from these studies showed wide variation in $\hat{r}_s$, which measures tuning curve similarity, across pairs of neurons recorded from a single electrode tip. This was the case in spite of the expectation that nearby MT neurons should have similar direction tuning [115, 116]. We used the estimators developed in the previous section to understand what factors might have contributed to the variation in signal correlation.

Variation in $\hat{r}_s$ across pairs of neurons could happen in several ways. We have shown it can vary spuriously as a function of SNR and noise correlation. With $\hat{r}^2_{ER_{split}}$, we can remove variation as a function of these confounds and focus on the nature of the systematic relationship between tuning curves. Even for well-estimated tuning curves, there are several ways signal correlation can vary. Many MT neurons have approximately sinusoidal direction tuning curves, thus pairs of tuning curves might be more or less correlated simply because of differences in the phase of the tuning curves (i.e., shifts in preferred direction as modeled in Figure 5.1A, solid lines). On the other hand, some MT neurons do not have classic sinusoidal tuning curves, raising an alternative possibility that more complex tuning curve profiles play a role in $\hat{r}_s$ variation. Below, we show that both cases exist and that they relate to the dynamic range of the neuronal tuning curves.

We begin by giving insight into our basic results on the basis of tuning curves for representative pairs of neurons. The direction tuning curves for a pair of MT neurons are plotted in the upper left panel of Figure 5.9. Both neurons were strongly modulated by the stimulus (geometric mean SNR = 18.66) and have the same preferred direction (orange and blue curves both peak near 270°) and similar classic unimodal tuning profiles. The similarity in tuning is reflected unambiguously by the high $\hat{r}^2_{ER} = 0.97$ with a narrow confidence interval (0.93, 0.97). This pair and the other examples in the top row of Figure 5.9 demonstrate a trend whereby strongly modulated pairs of neurons (joint SNR > 15 in all cases) tended to also have a high signal correlation with narrow confidence intervals and classical unimodal tuning curves. On the other hand, pairs that had low joint SNR (Figure 5.9, bottom row, geometric mean SNR < 6 in all cases) tended to have low $\hat{r}^2_{ER}$ values and each included at least one neuron that deviated substantially from unimodal tuning (blue curves indicate the more weakly tuned neuron). Importantly, the low signal correlation is not a trivial result of noisy data: $\hat{r}^2_{ER}$ takes into account the low SNR and the confidence intervals are narrow.

To examine the relationship between SNR and r^2_{ER} across the population of MT pairs, we plotted estimates of these values against each other for pairs with narrow confidence intervals (Figure 5.10, CI$<$ 0.25) and found a positive relationship between SNR and signal correlation (Spearman's rank-order r=0.40, p=0.005, two-tailed t-test). Thus, when one or both of the neurons in a pair are not well driven by the stimuli, the tuning curves of the pair tend to be less correlated. This relationship would have been confounded had we used the naive $\hat{r}^2$, because noise would trivially reduce correlation, but the corrected estimator, $\hat{r}^2_{ER}$, gives confidence that the relationship is intrinsic to the pattern of tuning of neurons simultaneously recorded on a single electrode. A notable exception to the relationship observed (Figure 5.10, blue point closest to upper left) is poorly tuned for direction but still has very high signal correlation. Such exceptions demonstrate that the relationship is not inevitable. Thus, consistent with the example pairs in Figure 5.9, it is the neurons that are not well tuned for the moving dot stimuli that produce low signal correlation values.

We also used $\hat{r}^2_{ER_{split}}$ but found its estimates were nearly identical to those in Figure 5.10 (Pearson's r=1.0) but the increase in length of CI's caused the inclusion of fewer pairs (29 vs the original 48). Spearman's r between $\hat{r}^2_{ER_{split}}$ and SNR was $r = 0.4$ but the p-value was lower, p=0.027, because of the lower number of pairs.

To demonstrate the implications of the SNR-r_s relationship for the tuning of a concrete population of nearby cortical neurons, and to show this relationship is not a trivial result of trial-to-trial variability, we carried out stochastic simulations. First, consider the simple scenario where all MT neurons have sinusoidal direction tuning curves and only the amplitude and phase (i.e., preferred direction) can vary. Here we do not induce noise correlation. Figure 5.11A (left column) shows a distribution of tuning curve amplitudes and preferred directions that are chosen independently, with preferred direction limited to a narrow range, consistent with cortical columnar organization [115]. We simulated noisy tuning curves for 100 pairs picked randomly from the parameter distribution where the noise arises from Poisson firing statistics for $n = 10$ repeats (for details see Methods: Stochastic model of neuronal responses). For such a simulated population, the observed relationship between the naive $\hat{r}^2_s$ estimate and SNR is shown in Figure 5.11C (left), and the distribution of Spearman's rank-order r-values across 200 such populations (Figure 5.11D, left) reveals a substantial spurious positive correlation (mean 0.55) caused by trial-to-trial variability

reducing the naive $\hat{r}_s$ when SNR is low. This spurious relationship disappears as expected when we use $\hat{r}^2_{ER}$ (Figure 5.11E and F, left), because, by construction, tuning curve preferred direction differences are independent from SNR. A similar result holds even if preferred direction is allowed to vary widely across the population (Figure 5.11, middle column). Thus, this scenario is not consistent with our findings, and it highlights the importance of accounting for bias in $\hat{r}_s$.

In a second scenario, we made a simple addition to reflect our observation that neurons with lower dynamic range often have more irregular tuning curves (as observed in Figure 5.9, bottom panels, blue curves). Specifically, we added constant standard deviation Gaussian noise to each simulated tuning curve, independent of the amplitude or phase of the tuning curves. An example of a pair of tuning curves with moderate SNR (Figure 5.11B, right) reveals that the underlying true tuning curves are no longer perfectly sinusoidal. Crucially, the deviation from sinusoidal is larger at lower SNR because the average magnitude of the deforming component is constant. Under this scenario, the correlation between SNR and signal correlation is positive (Figure 5.11, column 3) when measured using our corrected estimate, $\hat{r}^2_{ER}$ (Figure 5.11E and F, right). This simple model, in which lower amplitude tuning curves are less sinusoidal on average than higher amplitude curves, is sufficient to recapitulate the SNR-r_s relationship observed for the MT data.

If this simple model is correct, it predicts that sinusoidal fits to tuning curves should improve with SNR. We fit a sinusoidal model across all MT neurons and estimated the variance explained with $\hat{r}^2_{ER}$, thus again removing the trivial dependence of the r-value on SNR. We indeed found that higher SNR tuning curves were better fit by sinusoids (Figure 5.12, red circles). In particular, for SNR$>$ 10 nearly all neurons had $\hat{r}^2_{ER} > 0.8$, whereas for SNR$<$ 10, about half of neurons had $\hat{r}^2_{ER}$<0.80. As a control we plot the results of the same analysis but performed on the stochastic responses of units drawn from the simulation in Figure 5.11 where tuning curves are truly sinusoidal and only vary with respect to phase and amplitude. We find that these curves are consistently reported as having a near perfect correlation to the sinusoidal tuning curve model *regardless of SNR*. Thus, deviation of these MT units from the sinusoidal model is not the result of trial-to-trial variability but of systematic difference from the sinusoid model. Furthermore, these systematic differences are more pronounced for neurons that are not as well tuned for the stimuli (i.e., have lower SNR).

Here we have uncovered a relationship between SNR and signal correlation in area MT. If the relationship had been estimated using classic measures it would be confounded. We propose a simple model sufficient to explain this trend and in simulation demonstrate the corrected estimator both avoids confounds and reveals the relationship between SNR and signal correlation. To test whether this generalizes beyond this one data set, we carried out a similar analysis of V1 data (Kohn lab, publicly available, [117]) for three monkeys, but for orientation tuning rather than for direction tuning (see Methods). Figure 5.12B shows that a similar trend held for all three animals: recordings with higher SNR tended to be better fit by sinusoidal tuning curves, after factoring out the influence of noise.

5.5 Discussion

We have examined two biases in the estimation of signal correlation: attenuation by trial-to-trial variability and a bias toward the noise correlation. To address these biases, we measured signal correlation across trials that were not simultaneously

recorded ($\hat{r}_{s_{split}}$) and introduced $\hat{r}^2_{ER_{split}}$, which uses an estimate of trial-to-trial variability to correct for its correlation-attenuating effect. We also examined how the often-reported positive relationship between signal and noise correlation, $\hat{r}_{NS}$, could arise artifactually because of the bias of $\hat{r}_s$ towards $\hat{r}_n$. We found this positive bias was pronounced in cases of low SNR and few repeats, and we showed that the estimator $\hat{r}_{s_{split}}$ prevented this bias. Finally we demonstrated the utility of the estimator r^2_{ER} by showing a novel positive relationship between SNR and signal correlation within MT. We supported our results in a stochastic simulation and concluded that underlying the relationship is the tendency of MT neurons that are not driven strongly by uniform frontoparallel motion to have more diverse tuning curve shapes.

5.5.1 Practical advantages of $\hat{r}^2_{ER_{split}}$

Neurophysiologists can, and have, tempered the biases examined here by averaging across stimulus repeats, but this never wholly removes them and restricts the number of stimuli (m) that can be presented during an experiment. A major practical benefit of using the corrected estimator $\hat{r}^2_{ER_{split}}$ is that it provides more flexibility in choosing m and n. When SNR is high, the naive estimator will give similar results to $\hat{r}^2_{ER_{split}}$, but until $\hat{r}_s$ is compared to $\hat{r}^2_{ER_{split}}$, any conclusions about signal correlation will need to be couched in the untested assumption that trial-to-trial variability is not confounding them. Thus, our estimator is valuable even in the case where SNR is high.

5.5.2 Prior work sensitive to confounds

The confounds discussed above have rarely been addressed in the neuroscience literature. Below, we review four studies where they could potentially influence published results. Vinje and Gallant (2000) [111] compared signal correlation in V1 neurons responding to natural videos in two conditions: contained within small apertures matching the classical receptive field versus 4X larger apertures covering the surround. They reported higher signal correlation for the smaller stimuli, but noted that the larger stimuli tended to suppress responses, consistent with well-known V1 surround suppression [119]. A suppression of responses would likely decrease SNR, biasing estimated correlation downward. Thus, it would be important to correct for trial-to-trial variability in such a study.

Gawne et al. (1996) [110] studied signal correlation between adjacent V1 neurons using 128 Walsh patterns and 16 oriented bars. They found that signal correlation was higher for bars than for Walsh patterns and bars combined. However, if dynamic range was higher for bars, which are known to highly modulate many V1 neurons, than for Walsh patterns (as in their Figure 5.5), then their finding that signal correlation was lower for a more diverse stimulus set could have been influenced by a difference in SNR.

Averbeck and Lee (2003) [107], studying neurons in macaque supplementary motor area, reported signal correlation, noise correlation, and the correlation between the two as a function of spike counting bin width (5-200 ms). SNR will likely decrease with bin width, thus any relationship could be influenced by corresponding changes in SNR. For example, they report that signal correlation magnitude increases with bin width (their Figure 5.6D), consistent with the downward bias of signal correlation with SNR shown in our Eqn. 5.9. It would be important to know

how much of the observed effects could be explained by changes in SNR with bin width.

Solarana et al. (2019) [120] examined the effects of sensory deprivation on signal and noise correlation in mouse primary auditory cortex. Comparing tone-evoked Ca^{2+} signals in light-deprived vs normally reared mice, they reported a small but significant decrease in average signal correlation with respect to frequency selectivity with dark rearing (their Figure 5.6) and sharper, higher amplitude frequency tuning curves in the deprived state. Given the complex changes in tuning curves, an analysis of SNR would be required to determine how their results might be influenced by the bias of $\hat{r}_s$ with SNR.

These are just a few examples where there is great potential value in controlling for trial-to-trial variability when studying signal correlation. The most common potential confound involves comparing signal correlation across conditions where SNR could plausibly differ, which is often the case when one stimulus set drives weaker tuning-curve modulation than another.

The bias of signal correlation by trial-to-trial variability discussed above tends to be downward with an appreciable upward bias only for low SNR and atypically high noise correlation (Figure 5.3, blue trace). On the other hand, the bias of $\hat{r}_{NS}$ is more complex. For typical ranges of values seen in neural data, we found the bias can plausibly be upwards or downwards depending on a variety of factors. The upward bias could either inflate or create a relationship between signal and noise correlation and remains largely unaddressed [7, 93, 106–109, 121–123]. In the case of Bair et al. (2001) [93] we found a mild inflation; however, the upward confound could dominate in cases where r_{NS} is low, m is high, and SNR is low. While we have provided a simple effective solution to the upward bias, below we discuss a possible approach for addressing the downward bias of $\hat{r}_{NS}$.

Future work might address this downward bias by applying $\hat{r}^2_{ER}$ to the estimated correlation between a signed version of $r^2_{ER_{split}}$ and r_n. Using $\hat{r}^2_{ER_{split}}$ to measure signal correlation will remove variability from differences in SNR across neurons, and extending r^2_{ER} to estimate correlation between $r^2_{ER_{split}}$ and r_n will remove the downward bias resulting from sampling variability. Addressing this may in part explain differences in $\hat{r}_{NS}$ between studies and cortical areas (for example see [93] versus [105, 110]).

5.5.3 SNR-signal correlation relationship

We found a positive relationship between SNR and signal correlation in area MT for the robustly encoded stimulus dimension of motion direction. This raises the possibility of a potentially ubiquitous connection between signal correlation and stimulus choice. Specifically, nearby neurons robustly encoding the same stimulus dimensions will tend to have similar tuning along those dimensions and those that weakly encode those stimulus dimensions will not. A consequence is that signal correlation for one set of stimuli may not generalize to other sets. While variation in noise correlation as a function of the stimulus set has been well studied [17], such variation in signal correlation has not been explored.

We modeled this relationship as the result of fixed-amplitude noise added to tuning curves of varying amplitude. We do not specify the source of this tuning curve noise, but future theoretical work could focus on its relationship to the maintenance of synaptic connectivity [124] or ongoing plasticity in response to variable inputs [125, 126].

In terms of sensory encoding, our results can be interpreted with respect to the dimensionality of the population representation of stimuli. Our MT data suggests that tuning of neurons with higher SNR (more strongly modulated) can be well captured by a low dimensional model whereas lower SNR (poorly modulated) neurons cannot. Specifically, strongly modulated MT neurons were well fit by a sinusoidal model and thus were constrained to the 2D plane defined by amplitude and phase, whereas less modulated neurons diverged from this plane (Figure 5.12). This has potential consequences for downstream computation. Consider a downstream neuron building a new tuning curve as a linear function of these inputs: if inputs were only high SNR neurons in the 2D plane, then the resulting tuning of the neuron would remain in the 2D plane. However, more complex tuning would require lower SNR inputs with more diverse tuning curves, and achieving high SNR for this would require combining more of the low SNR inputs. Assuming some cost to synapse formation, the steeper the relation between SNR and signal correlation, the greater the cost to increase dimensionality. Low vs. high dimensional representations have different advantages. In low dimensional representations, functions can be learned with fewer samples and estimation of underlying input drive is more resistant to noise. Whereas high dimensional representations can be more flexible in the functions they compute. Further research is required to determine whether the form of dimensionality regularization for a given cortical region implied by a particular SNR-r_s relationship has normative advantages for cortical encoding.

5.6 Figures

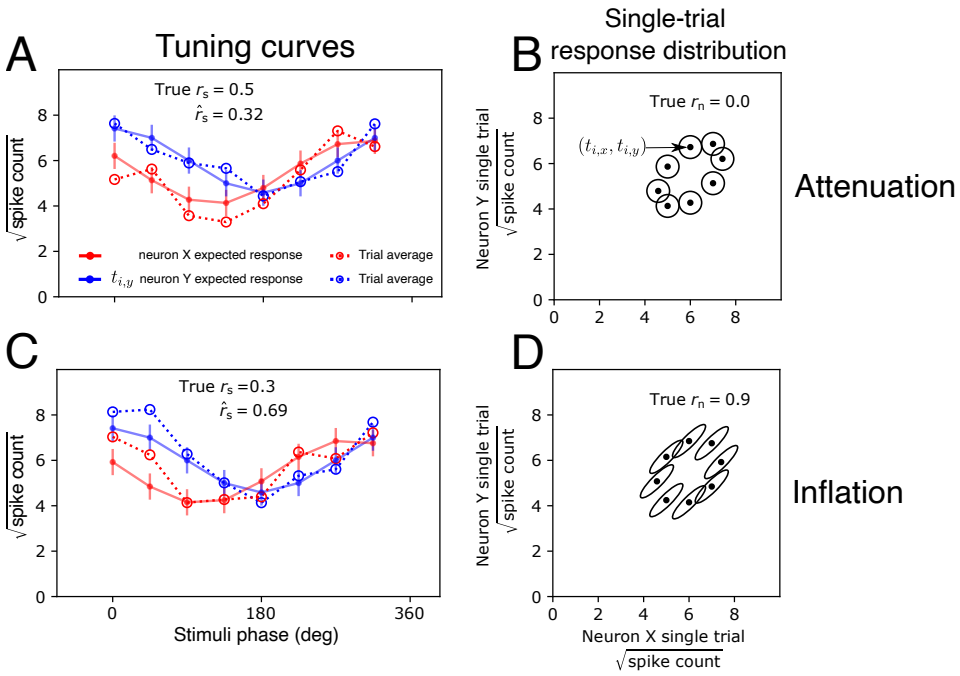


FIGURE 5.1: Simulations using sinusoidal tuning curves provide intuition into the attenuation and inflation of estimated signal, i.e., tuning curve, correlation ($\hat{r}_s$). **(A)** Sinusoidal tuning curves (solid lines, red and blue) show theoretical expected value ($t_{i,x}$ and $t_{i,y}$) for responses of two neurons to repeated stimulation. Their relative phase shift sets a fixed $r_s = 0.5$. Simulating an experiment, single trial responses are drawn and averaged for each neuron at each point on the tuning curves (dots and bars show expected value and variance, open circles indicate the sample mean). Signal correlation is typically estimated across such trial-averages (open circles), which often have a lower correlation (here $\hat{r}_s = 0.32$) than do the expected values (solid lines). Thus, sample signal correlation is downwardly biased compared to the correlation between the true tuning curves. **(B)** A schematic distribution of single-trial simultaneous responses of both neurons plotted against each other. The lack of noise correlation is reflected by the circular contour lines. **(C)** Simulation showing how signal correlation can be inflated by noise correlation. Here, deviations of the dotted lines around the means (solid lines) are correlated across the two neurons, i.e., at a given x-value, both dashed lines often lie above or below their means. **(D)** Noise correlation is indicated in the joint distribution of simultaneous single-trial responses by the tilted ellipsoids around expected values. This positive tilt inflates the estimated signal correlation.

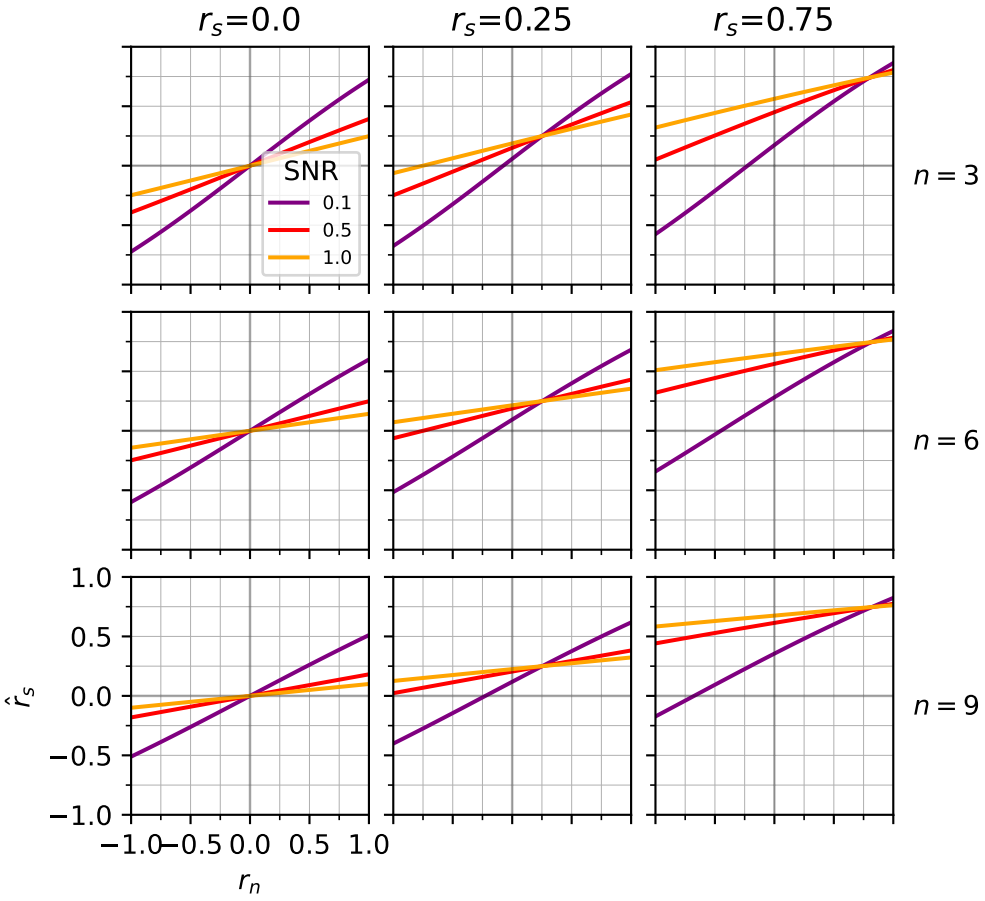


FIGURE 5.2: Analytically derived relationship between typical estimators of signal (r_s) and noise correlation (r_n) for electrophysiologically plausible parameters. The asymptotic value of $\hat{r}_s$ is plotted as a function of noise correlation, r_n (see Results: Signal correlation confounds, Eqn. 5.9). Number of stimulus repeats, n, increases from top to bottom. True signal correlation, r_s, increases from left to right. Line color indicates SNR level. Trial-to-trial variability is fixed to $\sigma^2 = 0.25$. Any deviation from a horizontal line (with intercept equal to r_s) indicates a confounded relationship between $\hat{r}_s$ and $\hat{r}_n$, and larger slopes indicate stronger biases.

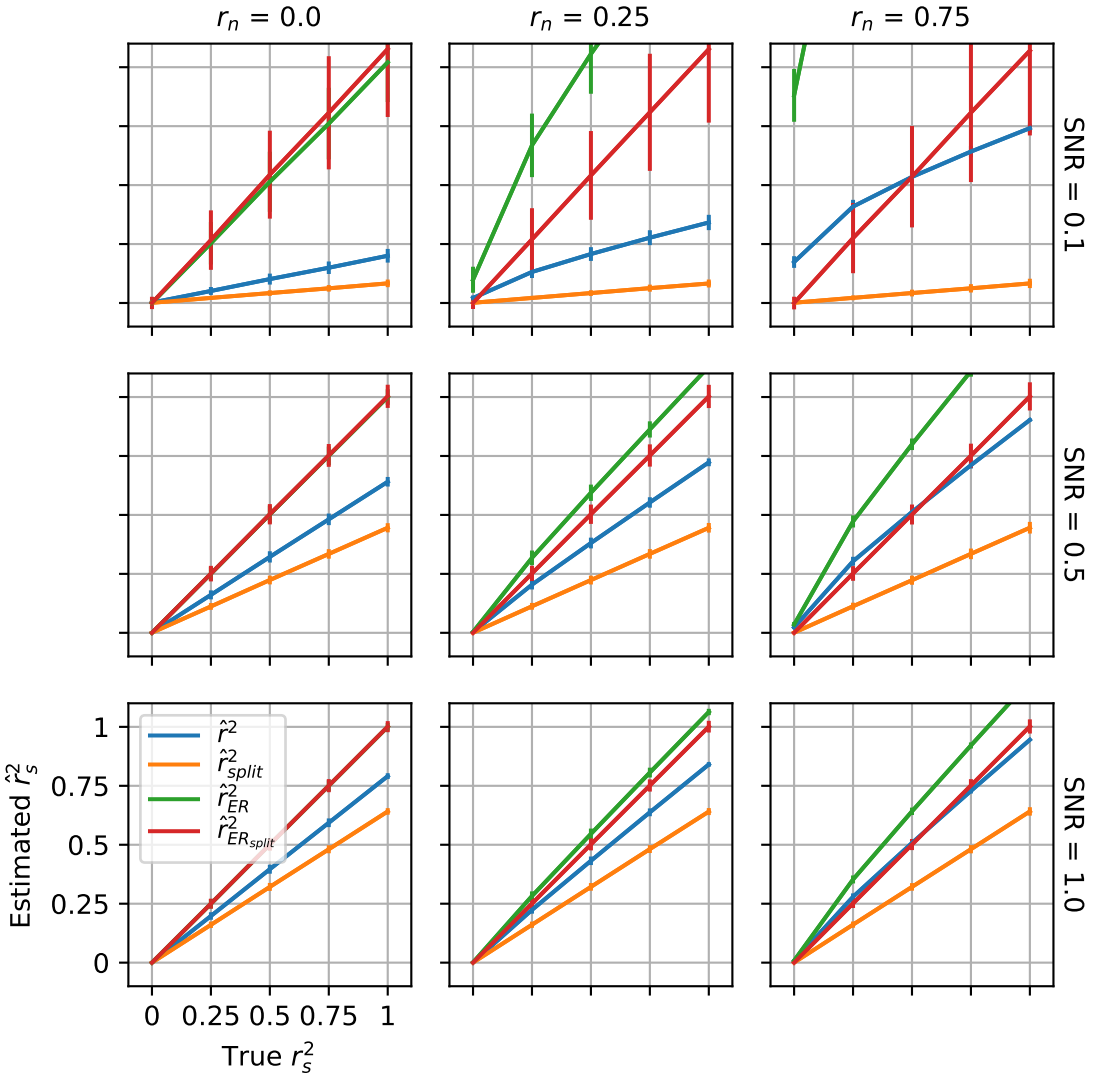


FIGURE 5.3: Simulation-based comparison of signal correlation estimators under varying degrees of SNR, noise correlation, and true signal correlation. The estimators are the naive measure of signal correlation $\hat{r}^2$ (blue), the split-trial version $\hat{r}^2_{\text{split}}$ (orange), the corrected version of $\hat{r}^2$ from Pospisil and Bair (2020): $\hat{r}^2_{\text{ER}}$ (green), and the split version of this estimator $\hat{r}^2_{\text{ER}_{\text{split}}}$ (red). There are $m = 500$ stimuli and $n = 8$ repeats. In general, positive noise correlation inflates naive estimators (blue and green), but split estimators remove this (orange and red); whereas noise attenuates naive estimators (blue and orange), but ER estimators overcome this (green and red). As SNR increases down each column, all estimators begin to approach the true signal correlation, and variability of the estimators is reduced.

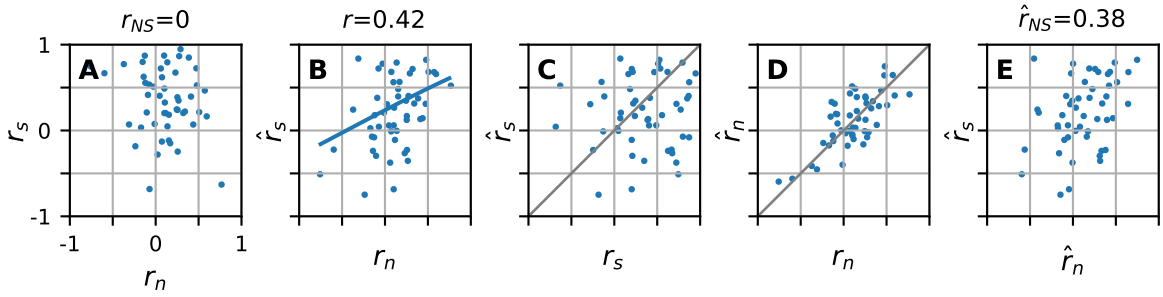


FIGURE 5.4: A simulation elucidating the inflating effect of noise correlation on the estimate of r_{NS} (Eqn. 5.7). **(A)** The true signal and noise correlation (z-transformed) for neuronal pairs are randomly sampled from a bivariate normal distribution with correlation 0. The moments of the z-transformed distribution of r_{s} are $\sigma = 0.5$ and $\mu = 0.5$. The moments of r_{n} are $\sigma = 0.3$, $\mu = 0.1$. The SNR of all neurons is low (SNR=0.1). **(B)** Estimates of signal correlation have a noisy linear relationship to r_{n} (Eqn. 5.8, corresponding roughly to top row, middle Figure 5.2) despite there being no relationship between the true parameters. **(C)** Estimated signal correlation (y-axis) has little relation to true signal correlation (x-axis) because of low SNR and the influence of noise correlation. **(D)** Noise correlation estimates (y-axis) are themselves noisy, thus there is an imperfect relationship with true noise correlation (x-axis) across the population and the estimate. **(E)** Overall, there is a significant correlation between the simulated experimental estimates of signal and noise correlation, despite there being no true correlation between the parameters being estimated (compare to A). The correlation is lower than that of $\hat{r}_{\mathrm{s}}$ and r_{n} (see B) because of variability in $\hat{r}_{\mathrm{n}}$.

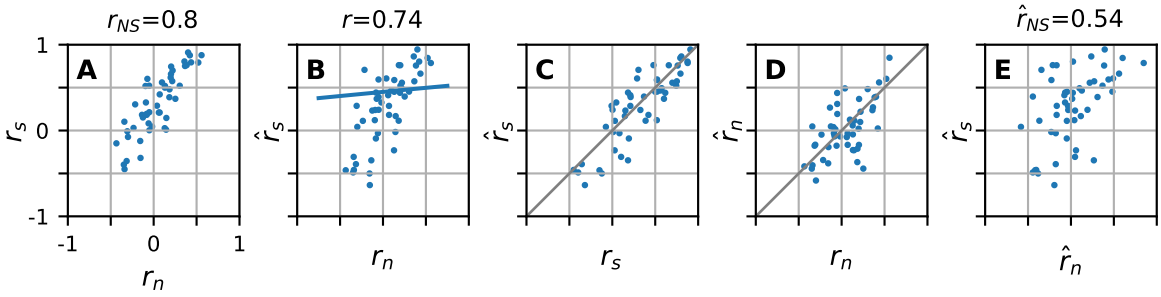


FIGURE 5.5: A simulation demonstrating the attenuating effect of noise on the estimate of r_{NS}. The moments of the z transformed distribution of r_{s} are $\sigma = 0.5$ and $\mu = 0.5$. The moments of r_{n} are $\sigma = 0.3$, $\mu = 0.1$. The SNR of the neurons are higher here (SNR=1) than in the previous figure. **(A)** Here the true r_{n} and r_{s} are jointly distributed to have a strong correlation (0.8). **(B)** Signal correlation estimates (y-axis) have a strong linear relationship to r_{n} (Eqn. 5.8, corresponding roughly to row 1 column 2 of Figure 5.2) yet this is not the result of the spurious correlation in Figure 5.3: the blue line shows the theoretically predicted linear relationship is weak compared to the relationship in the data points. **(C)** Signal correlation estimates have a strong relation to true signal correlation because of high SNR and corresponding lack of influence of noise correlation. **(D)** Estimates of noise correlation are as noisy as Figure 5.3 since SNR does not reduce variance in the estimate $\hat{r}_{\text{n}}$. **(E)** Overall, there is a significant correlation of the estimates, though far lower than the true correlation. Thus, in addition to inflation (Figure 5.4), there can also be attenuation in estimating r_{NS}.

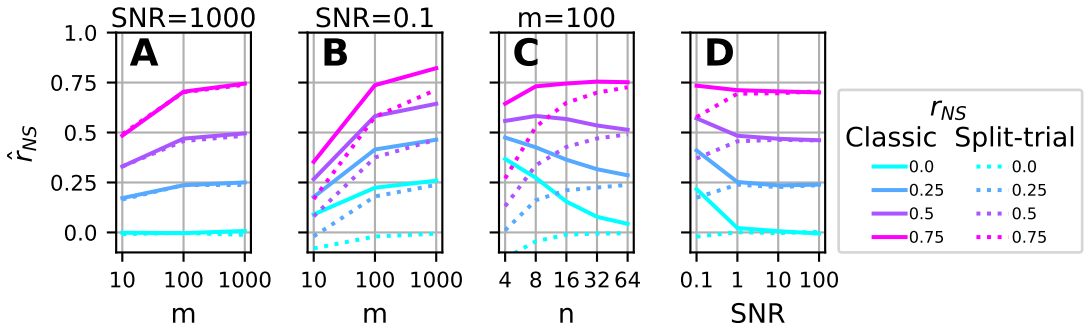


FIGURE 5.6: Simulation of experiment estimating r_{NS} (Eqn. 5.7) as a function of m, n, and SNR. Default values of simulation are: $m = 100$, $n = 10$, and SNR=0.1. Solid lines show $\hat{r}_{\text{NS}}$ using the naive estimator of signal correlation and dashed show the split-trial estimator. **(A)** Average $\hat{r}_{\text{NS}}$ as a function of number of stimuli (m). We set SNR high to reduce the bias of signal correlation by noise correlation. **(B)** Average $\hat{r}_{\text{NS}}$ as function of number of stimuli (m) when SNR is low. **(C)** Average $\hat{r}_{\text{NS}}$ as function of number of repeats (n). **(D)** Average $\hat{r}_{\text{NS}}$ as function of SNR.

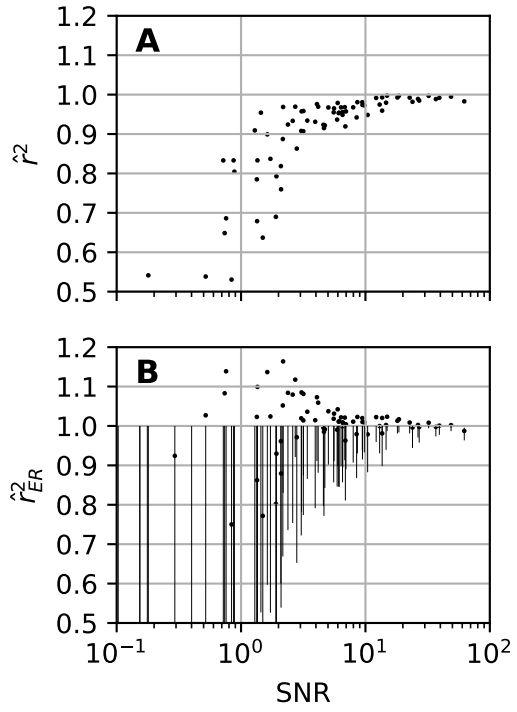


FIGURE 5.7: Demonstration of attenuation of signal correlation by trial-to-trial variability and its correction by r^2_{ER}. For each neuron, responses from odd trials are correlated with those from even trials so the theoretical true signal correlation is 1. **(A)** Naive r^2 estimate for 81 area MT neurons, 8 stimulus conditions, and 10 repeats in total, thus 5 repeats for the estimate of r^2 across odd-even trials. **(B)** Same as A except using estimator $\hat{r}^2_{\mathrm{ER}}$ and 90% confidence intervals shown as black vertical lines.

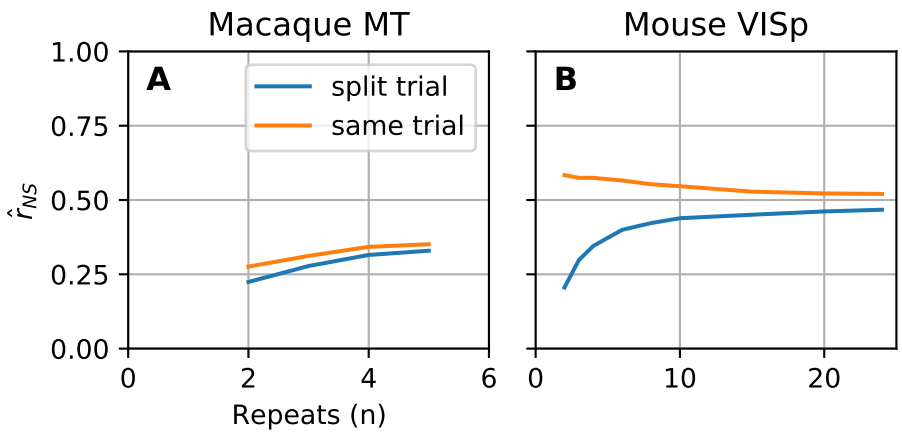


FIGURE 5.8: Demonstration of inflation of $\hat{r}_{\text{NS}}$ by noise correlation in neural data. Each point on a trace is an average of 1000 simulations where $2 \times n$ trials (x-axis) are randomly sampled without replacement from the original data. In the split-trial simulations (blue), signal correlation is estimated by correlating the odd trials from the 1st neuron to the even trials of the second then vice versa, taking an average of the two estimates, and finally averaging across all simulations. Same trial estimates (orange trace) are similar but even and odd trials from the 1st neuron are respectively correlated to the even and odd trials from the second neuron. For both, noise correlation is estimated using all n randomly sampled trials. **(A)** Simulations from MT data. **(B)** Simulations from Allen Brain Observatory VISp data.

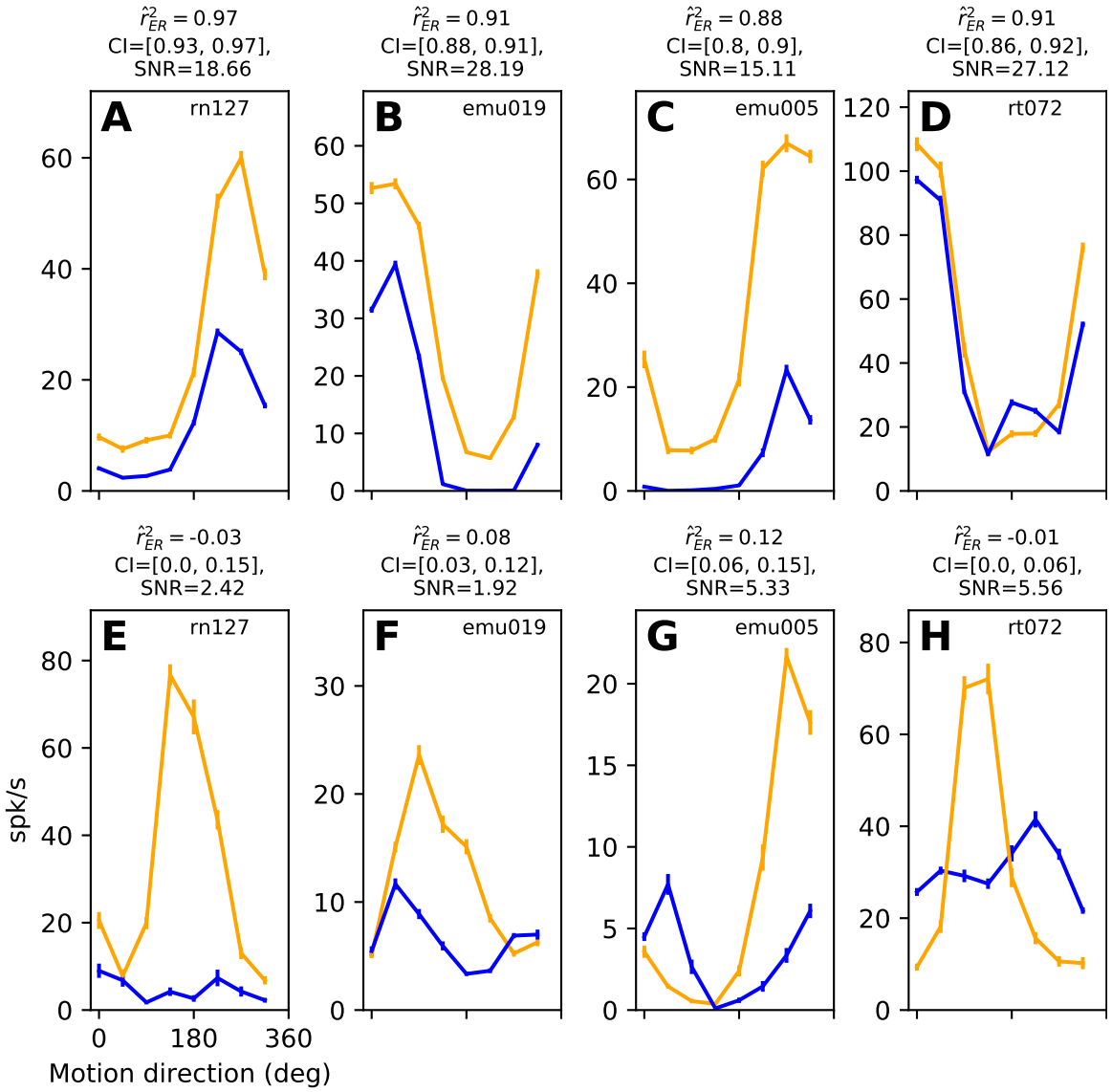


FIGURE 5.9: Examples of estimated direction tuning curves for pairs of MT neurons (error bars show SEM; Zohary et al., 1994). **(A)-(D)** Examples of pairs of tuning curves that have high joint SNR (geometric mean), large $\hat{r}^2_{\text{ER}}$ and visibly similar shapes. **(E)-(H)** Example pairs of tuning curves with relatively low signal correlation, low SNR and visibly dissimilar shapes. The tuning curve for the more weakly modulated neuron in each pair is shown in blue.

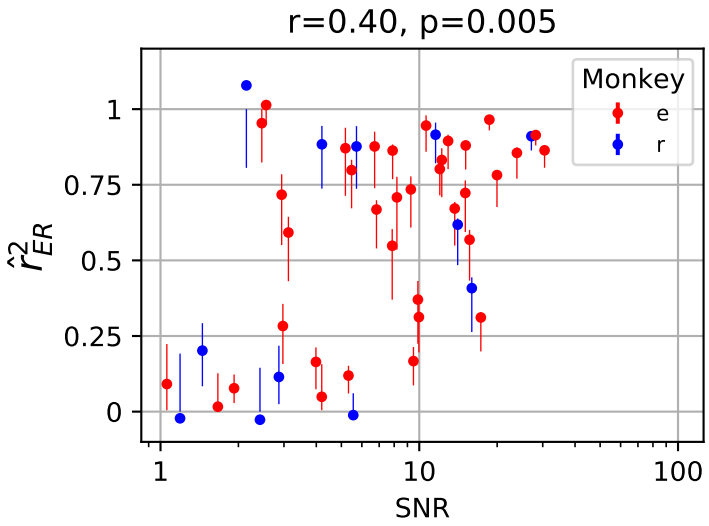


FIGURE 5.10: A positive correlation between signal correlation and SNR. Our unbiased estimate, $\hat{r}^2_{\text{ER}}$, of signal correlation is plotted against the geometric mean of SNR for each pair of neurons having CI length for $\hat{r}^2_{\text{ER}}$ less than 0.25 (n=48). Color indicates animal ID.

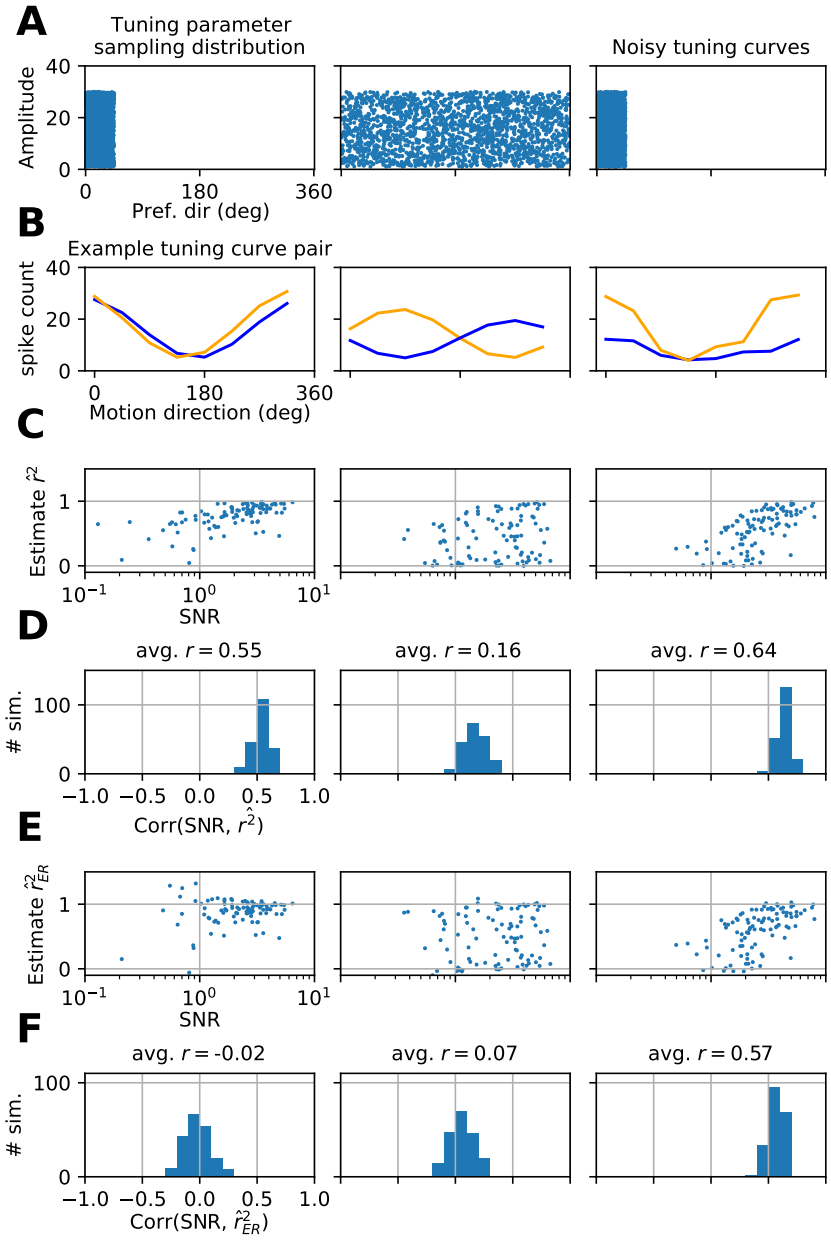


FIGURE 5.11: Simulation of MT neuronal data under different assumptions of tuning statistics across a population. **(A)** Joint distribution of phase and amplitude of simulated sinusoidal neural tuning curves. In column 1, phase, which sets preferred direction, is uniformly distributed between 0 and 45 degrees, amplitude is uniformly distributed from 1-30 spikes and the distributions are independent. In column 2, the range of preferred directions is wider. In column 3, the range of preferred directions is again narrow, but these tuning curves have noise added to them ($\sigma^2 = 2$). **(B)** Plots of example tuning curves. Spike counts are drawn from Poisson distributions with means set by tuning curves. In column 3, tuning curves are not smooth since noise is added. **(C)** Results of simulation of an experiment where there are 8 directions of motion, 10 repeats, and 100 pairs of neurons. Spearman's ranked correlation is measured between the geometric mean of SNRs and $\hat{r}^2$ (naive estimator). **(D)** Distribution of correlations between SNR and $\hat{r}^2$ across 200 simulated experiments. **(E)** Identical to (C) but corrected $\hat{r}^2_{\text{ER}}$ used instead of naive $\hat{r}^2$. **(F)** Distribution of correlations between SNR and $\hat{r}^2_{\text{ER}}$ across 200 simulated experiments.

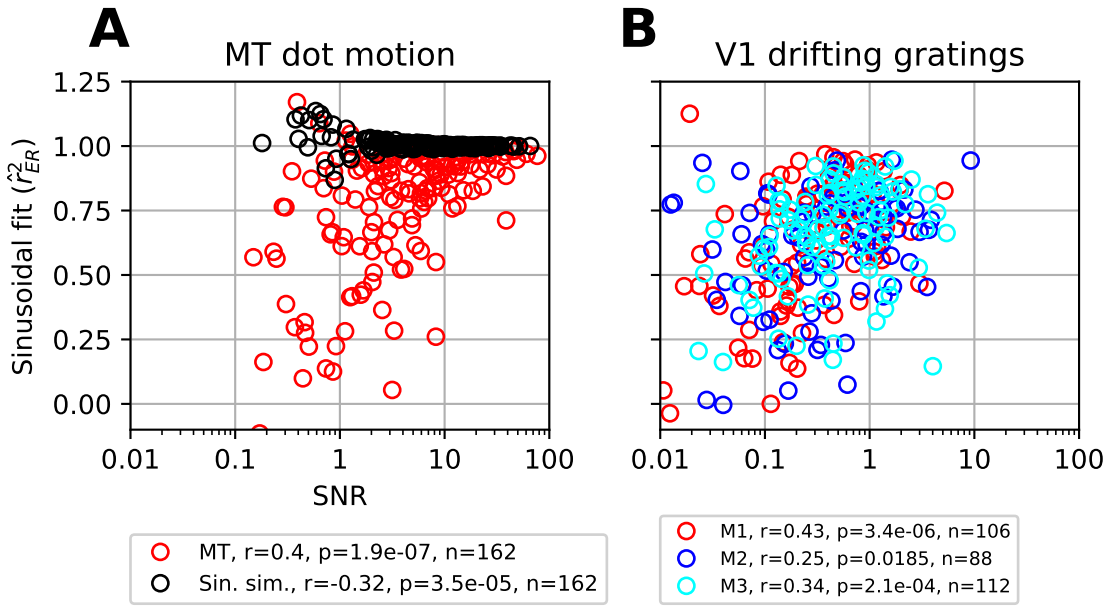


FIGURE 5.12: Relationship between SNR and sinusoid model fit quality. **(A)** Fit quality ($\hat{r}^2_{\text{ER}}$) of MT tuning curve to sinusoid, $a\cos(\theta) + b\sin(\theta)$, where θ is the orientation of motion, as a function of SNR. For all 81 pairs of neurons (162 neurons total) we estimated the fraction of variance in the tuning curve explained by the fit of a sinusoid. As a control, we also fit the sinusoid model to the simulation of neurons with sinusoidal tuning curves from Figure 5.11 column 1 (black). **(B)** Same analysis as A except for V1 orientation tuning curves in response to drifting gratings. Here, the sinusoidal model tuning curve frequency is doubled to account for the difference between orientation vs. direction tuning.

Chapter 6

DNN response property attribution through the hierarchical decomposition of response covariance

6.1 Summary

Deep neural networks (DNNs) have shown impressive performance in predicting cortical responses to naturalistic stimuli. Yet, it can be difficult to gain insight from these models. A variety of methods have been developed to provide insight into the computations underlying selectivity of units in a network. Broadly this work has either focused on the input space (visualization) or attribution of a given unit's response properties to hidden layers of the network. Visualization involves transforming inputs such that what features in the input led to a response or a decision become clear upon examination. Attribution methods, more generally, ask how was a response generated or at least what was the degree of participation by different elements of the network. Here we develop an attribution method from the perspective of systems sensory neuroscience based on the covariance of population responses, the transformation of that covariance by network weights and how the covariance is scaled by nonlinearities.. We provide two examples of this method, demonstrating attribution in the case of selectivity in a model of V4 responses to natural images and in the case of invariance of a DNN to shape surface and background luminance.

6.2 Introduction

A central goal in the study of the primate visual cortex is characterizing its selectivity. Deep neural networks (DNNs) are currently the best predictive model of intermediate primate cortex [29, 33, 34]. Yet, these models typically have huge numbers of parameters, are highly nonlinear, and it is not immediately apparent how to go from their impressive predictive performance to real insight. To tackle this problem a variety of attribution methods have been proposed. Most focus on the output or prediction of a network but could, or have been, adapted to an internal representation. For example, DeepLIFT chooses reference activations then assigns scores to units based on their differences from this reference [127]. Layer-wise relevance propagation is essentially the same as DeepLIFT except the reference is set to 0 [128]. The LIME method makes a local approximation around a given response that is more interpretable typically a sparse linear model [129]. Expectation Shapley (ES) values are a distribution of credit according to assumptions derived in game theory [130]. The

integrated gradient approach takes an average of derivatives between a reference activation and the activation of interest [131].

Here we propose an attribution method that is developed from the perspective of systems sensory neuroscience. It does not ask how particular predictions were made but instead solely seeks to attribute variation and covariance of a unit's tuning curve with respect to that of its inputs. It factorizes this attribution into categories of interest to systems neuroscience: a linear and nonlinear stage [28], individual tuning of units, and signal correlation. The method solely requires responses to a set of images and the calculation of covariance matrices from those responses. In the context of a feedforward DNN the method is applied recursively, starting from a given hidden unit, attribution is assigned to all its inputs from the previous layer, and then to all their inputs, and so on down to the input space.

We first develop the method and then give two examples of how the attribution of covariance can give insight into selectivity and then invariance in a DNN.

6.3 Methods

Our method is focused on the case of linear-nonlinear cascades where the network is solely composed of linear weightings of inputs and then the application of a static nonlinearity (such as a sigmoid or in our case rectification). For attribution in the case of the linear stage, we can take advantage of the fact that the covariance of a random vector after a linear transformation is a simple quadratic function, and in the case of the nonlinearity, we can simply form the ratio of the resulting covariance with linear covariance and cast the nonlinearity as the scaling of covariance. Thus if the property of interest of a unit can be cast as a transformation of covariance, this decomposition can be useful in understanding how the property arose. In the case of selectivity, variance of the tuning curve suffices and in the case of invariance, the ratio of variance to reference stimuli and the covariance to transformed stimuli suffices.

6.3.1 Linear network covariance

To develop covariance attribution we first consider the case of a linear transformation of a random vector, which is equivalent to the first layer of a feedforward neural network before the application of the nonlinearity:

$$\vec{x}_1 = W_1 \vec{x}_0$$

where W_1 is a fixed $n_1 \times n_0$ weight matrix, n_1 is the number of output units and n_0 is the number of inputs (or input units from a previous layer) and $\vec{x}_0$ is the random vector of input activations, $\vec{x}_1$ outputs. Covariance would be unchanged by an added bias vector.

Now consider the task of taking a single output unit or covariance between two output units and attributing it across its inputs. We make use of the fact that we can write the output covariance matrix as a linear function of the input covariance matrix and weight matrix,

$$\Sigma_1 = W_1 \Sigma_0 W_1^T, \tag{6.1}$$

where Σ_1 is the $n_1 \times n_1$ covariance matrix of the random vector $\vec{x}_1$and Σ_0 is the $n_0 \times n_0$ covariance matrix of the random vector $\vec{x}_0$ (the input). So for the case of a single entry in the output covariance matrix,

$$\mathrm{Cov}(x_{1,i}, x_{1,j}) = \vec{w}_i \Sigma_0 \vec{w}_j^T,$$

where $\mathrm{Cov}(x_{1,i}, x_{1,j})$ is the covariance of the i and jth output units, $\vec{w}_i$ is the ith row vector of the weight matrix W_1 or equivalently the weights of the ith output unit on its inputs. We can rewrite this equation:

$$\mathrm{Cov}(x_{1,i}, x_{1,j}) = \vec{e}^T(\vec{w}_i \vec{w}_j^T \odot \Sigma_0)\vec{e} = \sum_{k,q}^{n_0} w_{i,k} w_{j,q} \Sigma_{0_{k,q}}$$

where the output covariance entry is written as the element-wise product ($\odot$, Hadamard product) of the input covariance matrix with the outer product of the weight vectors, then summing all elements with the vectors of 1's, $\vec{e}$. The tensor in the parentheses after the first equality will be useful when we perform further factorization below (see 'Decomposition of covariance'). We can further factorize this equation by pulling out the variances, let $\vec{\sigma}_i = \sqrt{\Sigma_{i,i}}$ the square root of the diagonal of the covariance matrix and R_0 be the matrix of correlation coefficients,

$$\mathrm{Cov}(x_{1,i}, x_{1,j}) = \vec{e}^T(\vec{w}_i \vec{w}_j^T \odot \vec{\sigma}\vec{\sigma}^T \odot R_0)\vec{e},$$

thus we can think of the variances as similar to the weights in that they change the magnitude but not direction of the covariance components. Furthermore since variance is positive they only change the magnitude but not the sign as the weights can.

6.3.2 Linear-nonlinear covariance

The results in the previous section are a consequence of the well-known relationship between linear transformations and covariance. Typically the computations of interest in a DNN are the result of the nonlinear stages. Just as the nonlinearity is a transformation of the linear response there is a corresponding element-wise transformation of the covariance,

$$\Sigma \underset{\mathrm{g}}{\rightarrow} g(\Sigma),$$

where implicit in the function are the properties of the bivariate distributions, other than covariance, which interact with the nonlinearity to transform the covariance. For a detailed analysis of the normal and t-distribution (see Result: Rectification's effect on covariance). Here we represent this transformation as the element-wise multiplication that gives the transformation

$$g'_{i,j} = \frac{g(\Sigma_{i,j})}{\Sigma_{i,j}},$$

so that,

$$g(\Sigma) = g' \odot \Sigma. \tag{6.2}$$

Thus we can write the covariance of the output units after the nonlinearity as:

$$g(\Sigma_1) = g'_1 \odot W_1 \Sigma_0 W_1^T.$$

Again, this representation hides the dependence of the transformation of covariance on the distribution pre-nonlinearity. It simply reports the end result of this potentially complex relationship in terms of a ratio of the covariance before and after the

transformation.

We can now write the covariance of the output units as:

$$g(\Sigma_1)_{i,j} = g'_{1,i,j}\vec{e}^T(\vec{w}_i\vec{w}_j^T \odot \Sigma_0)\vec{e} \tag{6.3}$$

$$= g'_{1,i,j}\sum_{k,q}^{n_0} w_{i,k}w_{j,q}\sigma_{0,k,q} \tag{6.4}$$

Thus we have factored the covariance of the i,jth output units into the product of the covariance of the inputs (Σ_0), the weights on the inputs ($\vec{w}_i\vec{w}_j^T$), and the ratio of the covariance after the nonlinearity to the covariance before (g'_1).

We use the representation in Eqn. 6.4 to take a highly fine-grained approach by ranking the k, q terms in the sum to focus investigation of a unit onto the input units that have the most influence on the variance (or covariance) with the additional information of how the nonlinearity transformed the covariance.

This representation can be extended across N stages in a linear-nonlinear cascade setting the indices of the ith layer to k_i, q_i:

$$\Sigma_{0,k_0,q_0}[w_{1,k_0,k_1}w_{1,q_0,q_1}g'_{1,k_1,q_1}][w_{2,k_1,k_2},w_{2,q_1,q_2},g'_{2,k_2,q_2}]\dots$$

so we have a tensor with dimensions: $(n_0 \times n_0) \times (n_1 \times n_1) \times (n_2 \times n_2)\dots$ and summing over indices $1:i$ gives the covariance of entry k_{i+1}, q_{i+1}. Graphically we can think of this as a nested series of covariance matrices where the smallest is the input covariance and all others are transformations of it (Figure 6.1).

This formulation gives the first indication of the potential complexity of attribution based on covariance deep within a feedforward network as the number of terms is $\prod_{i=0}^{N} n_i^2$, a number which grows quickly with the layer depth (N).

Decomposition of covariance

While the individual contributions described above provide the finest grained description of how the variance of a unit relates to its inputs, weights, and nonlinearity there are natural gross distinctions that can be useful. Here we describe a nested series of three decompositions: diagonal and off-diagonal terms, mean and residual of these terms, and a decomposition of the dot product between weights and covariance into their magnitudes and correlation.

Diagonal and off-diagonal

Neuroscientists have extensively studied signal correlation (or the overlap between tuning curves) and speculated as to its function [17]. Separating out the diagonal and off-diagonal of input covariance matrices distinguishes signal covariance from variance. First we note that we can separate the diagonals and off-diagonals of the covariance and their contributions can be calculated separately before adding them:

$$\vec{w}^T\Sigma\vec{w} = \vec{w}^T[\text{diag}(\Sigma) + (\Sigma - \text{diag}(\Sigma))]\vec{w} = \vec{w}^T\text{diag}(\Sigma)\vec{w} + \vec{w}^T(\Sigma - \text{diag}(\Sigma))\vec{w} \tag{6.5}$$

thus we can separately study the contribution of the variance of each individual input unit (the diagonal) and the contribution of the correlation between these units. These terms precisely quantify the degree to which a unit's response variance is the result of the correlation between its inputs vs individual variances of the inputs.

From hereon for convenience, we will refer to the diagonal as D and the off-diagonal as O.

Mean and variance of covariance entries

We can further decompose the diagonal and off-diagonal in terms of their mean and variance (see Figure!6.2 for example in the case of the diagonal:

$$D = \bar{D} + (D - \bar{D}).$$

The product with the weights is simplified by the identity:

$$\vec{x} \cdot \vec{y} = (\vec{x} - \overline{x})(\vec{y} - \overline{y}) + n\overline{x}\overline{y} \tag{6.6}$$

plugging in:

$$wDw^T = \sum_{i=1}^{n} (D - \bar{D})(w_i^2 - \bar{w_i^2}) + n\bar{D}\bar{w^2}.$$

Furthermore by the decomposition of a dot-product into magnitudes and correlation,

$$(\vec{x} - \overline{x}) \cdot (\vec{y} - \overline{y}) = |\vec{x} - \overline{x}||\vec{y} - \overline{y}| \frac{(\vec{x} - \overline{x}) \cdot (\vec{y} - \overline{y})}{|\vec{x} - \overline{x}||\vec{y} - \overline{y}|} = |\vec{x} - \overline{x}||\vec{y} - \overline{y}| r[\vec{x}, \vec{y}],$$

where $r[\vec{x}, \vec{y}]$ is the Pearson's product moment correlation between the two vectors. Plugging in,

$$\vec{w}D\vec{w}^T = |(D - \bar{D})||(\vec{w}^2 - \bar{w^2})| r[\vec{w}, D] + n\bar{D}\bar{w^2}.$$

The off diagonal is similar and thus we have the decomposition:

$$\begin{aligned}
&\vec{w}^T \Sigma \vec{w} = \\
&\quad n\overline{D}\overline{w^2} && (6.7)\\
&+(n^2 - n)\overline{O}\overline{ww^T} && (6.8)\\
&+|D - \overline{D}||\vec{w}^2 - \overline{w^2}| r[D, \vec{w}^2] && (6.9)\\
&+|O - \overline{O}||w\vec{w}^T - \overline{ww^T}| r[O, w\vec{w}^T] && (6.10)
\end{aligned}$$

where ww^T is implied to be the off-diagonals of the outer product of the weights.

Decomposition of invariance

Here we apply the representation of response covariance developed above to understanding how invariance is attained. We measure invariance as the correlation between the response of a unit to a reference set of stimuli and the same set but transformed (e.g., rotated). This is simply the ratio of the covariance between responses to reference and transformed images divided by the product of the variance to reference and transformed images. We then study the interpretation of this ratio with respect to our decomposition.

We define invariance as the correlation between the responses of a unit to a reference set of images x_r and the same set of images after a transformation x_t: $\rho_{r,t}$. For the purposes of ease of analytic computation, we assume that the reference and transformed set of images contain the exact same images. The transformed set of

images is 'transformed' simply through reindexing. For an example see Figure 6.34. Thus responses to the reference and transformed image set will have the same covariance matrices. This allows for a single term in the denominator of $\rho_{r,t}$ as opposed to a product between reference and transformed variance.

This can be theoretically justified if transformed images can be assumed as coming from the same distribution as reference images. For example, the set of natural images is closed under translation thus a set of natural images and their translations are from the same population.

We define the input (we will call the individual input units subunits) distribution of responses to the reference and the transformed as s_r and s_t and their full covariance matrix:

$$\text{Cov}\begin{bmatrix} s_r \\ s_t \end{bmatrix} = \begin{bmatrix} \Sigma_r & \Sigma_t \\ \Sigma_t & \Sigma_r \end{bmatrix}.$$

The term Σ_r is simply the covariance of the subunits' responses to the reference stimulus set, which because of our assumed stimuli structure, is the same as that to the transformed. Of more interest is Σ_t which is the cross-covariance between the reference and transformed responses. So the diagonal of Σ_t is the covariance of each subunit to its response in the transformed case, and the off-diagonals are the covariances of different subunits across the transformation.

Letting $\vec{w}$ be the output unit's weights on the subunits, we can write the correlation in terms of the subunits' covariance (Eqn. 6.1) as:

$$\rho_{r,t} = \text{Corr}(x_r, x_t) = \frac{\text{Cov}(x_r, x_t)}{\text{Var}(x_r)} = \frac{\vec{w}g(\Sigma_t)\vec{w}^T}{\vec{w}g(\Sigma_r)\vec{w}^T}.$$

We can now further decompose the numerator and denominator into the mean and residual of the diagonal and off-diagonal components (Eqn. 6.5 and 6.6) of the nonlinear covariance:

$$\rho_{r,t} = \frac{\bar{D}_t \sum_i^n w_i^2 + \bar{O}_t \sum_{i\neq j}^n w_i w_j + \sum_i^n (D_t - \bar{D}_t) w_i^2 + \sum_{i\neq j}^n (O_t - \bar{O}_t) w_i w_j}{\bar{D}_r \sum_i^n w_i^2 + \bar{O}_r \sum_{i\neq j}^n w_i w_j + \sum_i^n (D_r - \bar{D}_r) w_i^2 + \sum_{i\neq j}^n (O_r - \bar{O}_r) w_i w_j}.$$

We now consider this decomposition from a geometric perspective (Figure 6.3), where the matching covariance term pairs in the numerator and denominator (terms vertically aligned) can be thought of as points in the space of $[\Sigma_t, \Sigma_r] \in R^2$. The application of $\vec{w}$ then transforms the length of the mean components and the length and potentially sign of the residual components. The slopes of each component weighted by their lengths give the invariance of the output unit. We now consider the interpretation of these 4 components.

The first term, $[\bar{D}_t \sum_i^n w_i^2, \bar{D}_r \sum_i^n w_i^2]$ we can think of as representing the 'typical' input invariance or the hypothetical invariance if we simply took an average of all inputs and ignored correlations between units both within the reference stimulus and across the transformation. Intuitively this captures the idea that at some layer input units may have achieved some level of typical invariance which the output unit can inherit.

The second term $[\bar{O}_t \sum_{i\neq j}^n w_i w_j, \bar{O}_r \sum_{i\neq j}^n w_i w_j]$ is the mean of the off-diagonal covariance terms and can be thought of as the typical correlation between subunits across the transformation. If pairs of subunits typically correlate across the transformation it contributes to invariance. An intuitive example is if the transformation

is a rotation of an image then a filter and a version of the filter rotated by the same amount as the image will perfectly correlate across the transformation. Adding the output of the two filters will give invariance to that transformation.

These two terms give the first order of approximation of the relationship between the weights and covariance structure of inputs which we will explore in the results.

The third term $[\sum_i^n (D_t - \bar{D}_t) w_i^2, \sum_i^n (D_r - \bar{D}_r) w_i^2]$ takes into account that there is variation in invariance across input units and by weighting the more invariant units over the less invariant units we will get a more invariant output.

Finally the fourth term: $\sum_{i \neq j}^n (O_t - \bar{O}_t) w_i w_j, \sum_{i \neq j}^n (O_r - \bar{O}_r) w_i w_j$ recognizes variation in correlation across subunits across the transformation. Some subunits may strongly correlate across the transformation and others may not and depending on $\vec{w}$ some pairs will be weighted over others.

6.3.3 Network

We use the PyTorch [132] implementation of AlexNet [57] pretrained to categorize the standard 1000 categories of ImageNet [83]. We make a small modification to the architecture in the case of the multi-unit activity (MUA) model (see below, 'Fitting V4 responses to natural images'). Visualization was performed via guided back propagation [133] using the Python package of [134].

6.3.4 Electrophysiological data

Responses to natural images in V4 came from the V4 UW Neural Data Challenge (UWNDC) 2019. Up to 601 images were shown with 3-20 repeats for each image. These images were drawn semi-randomly from the 2012 ILSVRC validation set of images where an 80×80 pixel patch was sampled then had a soft window applied (circular Gaussian with a standard deviation of 16 pixels). There are two basic categories of recordings: MUA where the number of spikes from a small population of neurons is recorded, and single unit, where, through careful post-processing it was insured the spikes recorded were from a single neuron. MUA is essentially all of the detected spikes that did not come from the isolated single neuron. Images were shown for 300 ms with 250 ms in between images. The model we analyze was the winner of the Neural Data Challenge (out of 32 competitors) on held-out data from the 14 sets of V4 responses to natural images. For access to data see 'https://www.kaggle.com/c/uwndc19/'.

6.3.5 Fitting V4 responses to natural images

We fit the responses described above to the outputs of Conv2 units in a pre-trained AlexNet. We made one modification to the network where instead of a max-pooling layer after the rectification of Conv1 we had an average pooling layer with the same kernel size and stride. This made the results more easily interpretable since averaging is a linear operation, unlike the max function.

Our fitting and validation procedure was as follows. We separated the up to 601 responses (one blank stimulus) to stimuli into a training (up to 551 responses) and test set (50 responses). We applied a square root variance stabilizing transform to all spike counts. Using Lasso regression with 10 fold cross-validation we determined optimal regularization of the regression of the 192 Conv2 units onto the training set of V4 responses averaged across trials. We then measured the correlation of this model on the test set of responses. We took the best fit recording (unit:"11-06-18-0-0",

$\hat{r}^2_{\text{ER}} = 0.51$, CI=$[0.43, 0.57]$) and its associated model to perform further analysis. We note that for the same recording the winning model of the UWNDC gave $\hat{r}^2_{\text{ER}} = 0.42$, CI=$[0.41, 0.51]$ thus the performance was comparable but required far fewer layers (Conv2 of AlexnNet vs 6 layers of winning DNN).

6.3.6 Max-pooling mask

In our study of invariance, the network responses pass through a max-pooling layer. We cast max-pooling as a scalar nonlinearity followed by a sum ($g(x_1) + ... + g(x_n)$) instead of a pooling nonlinearity ($g(x_1, ..., x_n)$). We do so by having a stage where the max responses in the pooling layer are kept to their original value but all others are set to 0. While, implicitly, all entries of this max pool 'mask' are a function of all spatial pooling positions it allows us to take our covariance attribution approach in determining which positions contributed the most variance.

6.4 Results

We begin by studying the influence of rectification on the moments of a bivariate normal to give insight into how covariance will be transformed by rectification in a DNN. We briefly examine the effect of rectification on the correlation of a bivariate t-distribution as we find some of the effects of rectification in the DNN are inconsistent with normally distributed inputs and the t-distribution can account for some of these inconsistencies specifically rectification transforming negative correlation into positive. We then explicate the decomposition developed above by applying it to a model of V4 MUA and provide insight into its selectivity. Turning to invariance, we show that the decomposition provides insight into how the network achieves invariance to image contrast reversal. Finally, we discuss these results in terms of predictions of MUA selectivity and theoretical consequences for the study of invariance.

6.4.1 The transformation of covariance by rectification

Here we briefly look at the relationship between the mean, variance, and correlation of a bivariate distribution before and after rectification. We will look at two distributions: normal and t-distribution. The main point for the normal is that negative correlation is attenuated more strongly than positive. For the t-distribution, when degrees of freedom are low, rectification has a weaker influence on positive correlation and can switch non-positive correlation to positive correlation. For increasing degrees of freedom, the t-distribution converges to normal.

In higher dimensions covariance between any two entries and its subsequent transformation by a nonlinearity is solely a function of the bivariate distribution. Thus the relationships learned below apply to the transformation of covariance of any dimensionality.

Rectified bivariate normal

We simulated draws from a bivariate normal where we varied the mean μ (same for both random variables), variance σ (same for both), and correlation ρ. We then rectified these random variables and measured the resulting moments: $g(\mu)$, $g(\sigma)$, $g(\rho)$ thus the threshold location in terms of quantiles was determined by $z = \mu/\sigma$ thus the results are invariant to changes in μ and σ that keep their ratio constant. In

Figure 6.4A, plotting $g(\rho)$, where the original ρ can be determined by the rightward asymptotes of the traces, we see for negative ρ the magnitude of $g(\rho)$ decays more rapidly then for positive ρ, $g'(\rho)$ shows the scaling down of correlation increases as correlation decreases (B), the variance monotonically decreases (C), and the rectified mean is equal or greater than the unrectified mean (D).

The intuition into the results on correlation is relatively simple. With a high mean not much of the distribution is rectified (Figure 6.5A), when the mean is low more of the points are rectified cutting off co-variation between two units while the average residual variance of the linear relationship remains about the same ($\mathrm{Var}(x_2|x_1)$) thus correlation is decreased (B), and finally if the correlation is negative both ends of the distribution are cut off and correlation drops precipitously (C). Similar issues for the normal distribution have been explored before [135, 136], below we briefly touch on the same analysis for a t-distribution which to our knowledge has not been studied despite its importance to natural image statistics and their consequences for normative models of encoding [137].

Rectified bivariate t

We repeated our analysis on the effect of rectification on the correlation in the case of a bivariate t-distribution. The t-distribution converges to normality with degrees of freedom thus we study it for a low number of degrees of freedom (df=3). We find two keys ways in which the effect of rectification is different from the normal case. Firstly positive correlation is resistant to the effects of rectification (Figure 6.6 red traces remain horizontal). Secondly, a negative correlation also decays more rapidly than a positive, but intermediate values can actually switch to a positive correlation. We will see this pattern is common in the DNN covariance after rectification.

Further work can characterize why the multivariate t-distribution is affected differently by rectification. While we do not give a quantitative justification, we note the 'bow-tie' shape of its distribution, where the variance increases with deviation from the mean, may explain the increase. Rectification would induce a positive slope in $\mathrm{E}[x_2|x_1]$ because $\mathrm{Var}[x_2|x_1]$ is increasing with x_1. Further work is needed to quantitatively characterize this effect.

6.4.2 Attribution of V4 selectivity to natural images

Here we demonstrate the use of the covariance decomposition in the case of a model of V4 responses to natural image patches.

The multi-unit activity (MUA) we examine here showed clear tuning across the natural image patches (Figure 6.7). Consistent with this strong tuning, the images which drove the highest responses vs lowest are qualitatively easily discriminated (Figure 6.8). The MUA recording has half of its test variance explained by a linear combination of units in the second layer of AlexNet (Figure 6.9 orange points, see Methods, 'Fitting V4 responses to natural images'). Ranking the stimuli with respect to the response of the model (Figure 6.10), we see the top 10 patches contain a few included in the top 10 for the neuron (e.g., upper left Figure Figure 6.8 is same as fifth from the left in Figure 6.10). To investigate the properties of this model we recorded its responses to 7,290,000 image patches drawn from 10,000 ImageNet validation images [83].

The most excitatory images across all patches were often, though not always, largely achromatic with multiple vertically oriented contours (Figure 6.11, top row). Suppressing images were typically full-field blue. We use a visualization technique

(see Methods, 'Network') on these images which finds the gradients of the image with respect to a positive or negative output of the unit, respectively for the excitatory and suppressive images. Conceptually this can be thought of as a perturbation to the image which locally would maximally increase excitation (Figure 6.12 top row) or suppression (bottom row). Qualitatively, the visualization technique tends to exaggerate components of the image driving the response though can also add elements that were not present in the image but would have driven the unit. Commensurate with Figure 6.11, chromatic content is de-emphasized and generally, contours are emphasized but on the bottom row, a local patch of green is emphasized with some diagonal contour to the bottom left. Most of the excitatory images lack a leftward slanting diagonal in the left corner of the patch while suppressing images include them.

Though maximally preferred images can be appealing it is important to remember that they can be misleading. Firstly these images are atypical in the sense that they don't reflect the typical variation of the unit. To achieve the highest response across 7,290,000 image patches places tight requirements on these images which in turn creates an appealing consistency across example images. Later results will show this unit is in part typified by permissiveness where the images which drive typical response deviations (e.g. 1 SD) can appear far more diverse.

To go beyond the raw output of this model unit we must now consider how all of its inputs contributed to its responses. The output of this unit is a linear combination of its inputs and thus its variance is a simple function of the inputs covariance: $\vec{w}^T \Sigma \vec{w}$. As described in methods this transformation can be broken into three components: the population covariance, the outer product of the variance of these units, and the outer product of the weights placed on them (Figure 6.13A-C). The sum of the elements of the product of these matrices is the variance of the output unit (D). We can ask for this final 'weighted covariance' matrix what terms contribute the most to the variation in the response i.e. the tuning curve. We can start at a gross level and simply ask which contributes more, the independent contributions of each unit or interactions between those units. This decomposition is discussed in methods (Eqn. 6.5) but we find that the diagonal accounts for half the variance. Thus half of the variation can be understood with respect to the response properties of units independent of each other but for the other half, we must consider their interaction. Examining the weighted covariance it is clear most variation is the result of unit 18 and its interactions with several of the other 12 units. Unit 18's central role is in a large part owing to its much higher dynamic range relative to the other units as its correlation and weights are not much different from several other units (compare Figure 6.13compare B to A and C).

We now consider the top five contributing units in terms of independent variance (Figure 6.14 and visualization 6.15) where their responses are calculated after applying the weights of the V4 model. Unit 18 shows strong excitation for achromatic rectilinear textures while suppression for full-field green and purple. The next highest independent contributor, unit 191, is selective for high spatial frequency repeating dots and is inhibited by low spatial frequency either with or without chromatic content. The next three units are more selective for contours. For example unit 27 appears to be selective for rightward tilting repeating contours but inhibited by those at a 90-degree rotation thus selective for one orientation at the exclusion of another unlike unit 18 which is driven well by the conjunction of orthogonal contours.

With regards to interactions, all units are positively correlated with unit 18 except for unit 89. The preference for green and red contours may be the basis of this anti-correlation but further work would be needed to confirm it.

We now consider in more detail how these individual response properties contribute to the final output. Calculated covariance is not invertible thus provides no indication of how specific samples contribute to the responses. We can simply examine the sum of squares leading to covariance to find such specifics. This is identical to looking at the lowest and highest responses from the V4 model since they have the greatest squared deviation from the mean for their given sign. We plot the joint distribution of responses of the top 5 input units in Figure 6.16. In red are a sampling of the lowest responses from the V4 model, in green the highest, and in cyan the entire distribution. Focusing on the first column we can consider how interactions with unit 18 contribute to the overall response. By selecting the top responses from the unit and indicating them in the joint distribution of the input (green) we are essentially viewing the distribution of the inputs conditioning on a high response from the output. Not surprisingly the responses of unit 18 are high when output is high (green vertical line in upper left distribution indicates mean of top responses). Examining the joint distribution we note that all but unit 89 also have high mean responses conditioning on a high response in the output, thus the strongest responses result from the co-occurrence of these features. Given the difference in scale of variance though, the responses from units other than 18 could be seen as simply modulating the response of 18. A high response from 18 is required for a high response in the model output, but the addition of the other units also having a high response (made more likely by correlation) will give an especially high response.

It is primarily unit 18 and 89 which, conditional on a low response in the output, show a low response in their own distributions (Figure 6.16 red dashes to the left of cyan in distributions on diagonal) thus these two units suppressive responses are driving the suppression of the output.

We now decompose the response properties of unit 18 in terms of its inputs. We choose unit 18 because it is the largest contributor to variance (Figure 6.13). We noticed that the most excitatory image patches to this unit appeared to be spatially homogeneous (top row Figure 6.14). This suggested the unit may be performing a similar computation across space and would greatly simplify the analysis. We checked whether the typical correlation between the weights on the 64 inputs coming from Conv1 at each of the $5 \times 5 = 25$ positions was high by calculating normalized weight covariance:

$$\mathrm{WC} = \frac{\sum_{i \neq j} \vec{w}_i \cdot \vec{w}_j}{\sum_{i \neq j} |\vec{w}_i||\vec{w}_j|} \tag{6.11}$$

where $\vec{w}_i$ is the ith vector of the 25 spatial position of inputs. We find that unit 18 has an atypically high weight covariance (Figure 6.17) consistent with its selectivity for homogeneous image patches. Thus the selectivity of one spatial subunit could suffice for understanding the selectivity of the other 24. To choose a unit for analysis we can determine the total variation each subunit contributes (Figure 6.18). The highest fraction covariance contributed by one unit is 6 percent, close to that of uniform $\frac{1}{25} = 0.04$. We choose this spatial subunit, at the 2nd row and 1st column for further examination.

Qualitatively its selectivity is consistent with the selectivity of the unit on the whole except flipped because the weights on this unit by the V4 model are negative (Figure 6.19. and visualized 6.20). We see the unit is driven best by uniform blue patches and suppressed by achromatic contours.

Descending another level we examine the covariance structure of this spatial subunit (Conv2, unit 18 row 2, column 1) and its inputs from Conv1. We skip the average pooling stage because it weights its 3×3 spatial pool equally. The linear input

covariance structure (Figure 6.21A) is heterogeneous with both strong negative and positive correlation, similarly, there are substantial differences in variance between units (B). The nonlinear covariance has distinctly less negative correlation (C less blue than A) and by eye, the variance clearly tends to decrease (D) both consistent with simulations in Figure 6.4. The outer product of the weight matrix is predominantly positive (E mostly red) implying most weights have the same sign. Finally, the weighted covariance appears quite sparse with two units having an outsize influence. We will find later that the weak correlations on the off-diagonal ($N = 64^2 - 64$) cumulatively also have a large effect.

Plotting nonlinear covariance for the same data but as the ratio of linear to nonlinear covariance (Eqn. 6.2), gives further insights (Figure 6.22). The motivation is to focus on the transformation by the nonlinearity. We first give the matrix of sign transformations whose range is three values instead of the usual two of the sign function (Figure 6.22A). A value of -1 (blue) indicates the sign went from positive to negative, 0 (white) indicates there was no change in the sign, and +1 (red) that the sign went from negative to positive. Here it is clear that often the sign of correlation flipped (red often and blue sometimes). This is surprising in the context of our results regarding the effect of rectification on the covariance between bivariate normal RVs. Rectification only affected the magnitude but not the sign of correlation. Crucially these results were based on an assumption of normality. We note that in our analysis, the multivariate *t*-distribution could have its correlation sign flipped by rectification.

We represent the ratio of magnitudes of correlation as log since it is a fraction and the range is large. Many correlation values are scaled, up or down, by a factor of 10,000 (Figure 6.22). The increase is a testament to the huge difference between the effect of rectification on the correlation of a bivariate normal and those of the DNN.

The scaling of variance by rectification is closer to what we would have expected under a normal distribution as it tends to only decrease (6.22C). It is not homogeneously scaled, some variances are scaled down by a factor of a 1000 (dark blue) but others barely at all (white). This shaping of variance we expect will largely be a function of the location of the threshold set by the bias of the unit and the mean of the distribution. Further work can examine the computational consequences of nonlinear variance shaping: for what purpose are some units highly rectified and others not? There was only a weak negative correlation between the nonlinear scaling of variance and the original variance thus it wouldn't likely be explained by a drive towards homoscedasticity (equal variance across channels).

Before going another level down, to the input space, we now consider the decomposition of the off-diagonal covariance in terms of their mean and variance (Eqn. 6.8 and 6.10) which show the weights align with the nonlinear transformation of covariance and that the mean component of covariance and weights are relatively high.

The relevant quantities are plotted but normalized to unit length since their relative contributions are maintained (Figure 6.23). The correlation between the off-diagonal covariance pre- and post-rectification is moderate implying a fairly strong transformation of covariance structure by the nonlinearity (A). One consequence is an increase in the mean (orange point over 0 on the x-axis but above it on the y-axis). While the weights are weakly negatively correlated to the linear covariance (B, r=-0.05) they are moderately positively correlated to the nonlinearly transformed covariance (C, r=0.29).

Another change to the nonlinear covariance is an increase in the average because of the reduction of negative values (compare B, C orange point on the y-axis). The

raw variance contributed by the product of means is 0.05 whereas the correlation between the weights and covariance is 0.06. So roughly half of the variance is not explained by any individual unit but by the fact that on average the weights tend to be the same sign and the covariance between units tends to be positive.

We will delve more deeply into this when studying invariance but this is a signature of a unit sensitive to input magnitude. Summing linearly anti-correlated but rectified units approximates squaring. Thus one way to think of this unit is as two units: one unit with weights that are a constant across the inputs and another with mean-centered weights that provide selectivity. The contribution of this 'second' unit's variance is the result of the weights aligning with the input covariance structure. We now examine this selectivity.

The weighted covariance of Conv1 (Figure 6.21F) indicated a handful of units and interactions had a much higher contribution to this spatial subunits variance than the others. Ranking these units in terms of their independent variance (the diagonal of the covariance matrix) we see the top two are maximally excited by a vertical achromatic contour (Figure 6.24) consistent with several of the most suppressive images of the spatial subunit (Figure 6.19 bottom row). But with fresh eyes, we note that many of the images have horizontal contours in addition to vertical and some have chromatic content and this is of the most driving patches across millions. Thus our initial hypothesis of spatial subunits selective for rectilinear achromatic images should be re-calibrated, selectivity seems to be broader and more invariant. We test this idea by considering whether the weights of this spatial subunit are predominantly chromatic or achromatic filters.

We use a simple chromaticity metric to rank Conv1 filters in terms of the fraction of their variance that can be accounted for by the chromatic plane (see Figure 6.25 and legend). Here we are taking advantage of linearity and the fact that these filters operate linearly on the input space. Ranking filters by this metric we see it captures what we would expect by visual inspection (Figure 6.26). When we examine the chromaticity of filters as a function of the weights on them by the spatial subunit we see no strong relationship just the two moderately achromatic filters (Figure 6.24 left most two points) with higher absolute weights but no systematic relationship across the rest. Thus there is not a strong preference for achromatic image patches.

To support this conclusion qualitatively we can examine images that evoked more typical deviations from the mean. Here we can define typical in terms of the fraction contribution of variation. Whereas per image the image that evokes the response deviating furthest from the mean (maximum or minimum) contributes the most to variance, taken as a whole the low probability of these images causes them to not contribute much to the variance of the unit. This reflects the fact that variance is weighted by probability $\mathrm{Var}(X) = \int (x-\mu)^2 p(x)\,dx$. In the case of the distribution of responses from unit 18 and its subunit at row 2, column 1 most of the variance is accounted for by values lower than the maximally suppressive (Figures 6.28 red trace). Images that evoked responses at more typical deviations from the mean appear far more diverse (Figure 6.29 and visualized 6.30).

While we have not examined all the descending covariance structure of unit 18 based on the high weight-covariance we hypothesize it also is driven in part by the average covariance. Indeed we will find in the invariance section below that this unit has the highest mean, relative to its variance, across weights of all 192 Conv2 units (Figure 6.41). Plotting its response distribution (Figures 6.31) we see variance is largely contributed by more typical deviations (red trace). Examining images which evoked these more typical deviations (Figure 6.32 and visualized 6.33) we see far more chromaticity and diversity than would have been expected based solely on the

preference to the typical images. This could reflect variation resulting from averaging Conv1 inputs as discussed above.

We focused on a single input to our V4 MUA model (which accounted for most of the variance). This unit is a texture unit but where half of the 'texture statistic' is magnitude in the input space of Conv1 owing to high negative correlation (opponent filters) that are rectified and then a high mean relative to the variance of the weights on these inputs. The other half is some preference for vertical and horizontal middling spatial frequency achromatic contours (thus extended contours) where it is invariant to contrast sign. If forced to name this unit we would call it the 'in-focus surface with contours' detector.

6.4.3 Attribution of invariance

Here we demonstrate insight into invariance by the method of covariance decomposition attribution for the second layer of a deep neural network (AlexNet) with respect to ON-OFF invariance. Our goal is to be able to rank the contribution of all inputs and their interactions to an output unit in Conv2 in terms of how much they contribute to invariance. Recursively applying this technique we can reach the point of interpretability: the input linear filters in Conv1.

We first examine the invariance of all Conv2 units at the center of their feature map (x=13,y=13). We use the stimuli from the Pasupathy and Connor shape set (Figure 2.3) but have an ON set where the shape surface is set to 1 and the background to 0 and an OFF set where the surface is set to 0 and the background to 1 Figure 6.34). Invariance is measured as the correlation between the responses to the ON and the OFF stimuli to the OFF and the ON stimuli respectively. Many units are invariant (Figure 6.35, median $r = 0.69$) to the ON-OFF transformation. What underlies this invariance?

As an example, we take the most invariant unit (Conv2 channel 83) with $r = 1.00$. Plotting the response to the reference and transformed stimuli we see the responses are predominantly negative and nearly identical across the transform (Figure 6.36).

As discussed in Methods the invariance is a function of the weights applied to the correlation structure of the inputs to the Conv2 unit. In the case of ON-OFF stimuli, we plot this correlation structure (Figure 6.37) where on the x-axis is the correlation of input units with respect to the reference stimuli and on the y-axis correlation with respect to reference-transformed stimuli. There are $(5 \text{ rows} \times 5 \text{ columns} \times 64 \text{ feature channels})^2 = 2,560,000$ terms. We decompose the correlation terms into the diagonal correlation terms (red, note that they all are aligned above 1 since the correlation of a unit with itself is 1) and several categories of off-diagonal terms: different units at the same position (orange), different units at different positions (blue), the same unit at different positions (green). Points above 0 on the y-axis are units that are correlated across the transformation and those below are anti-correlated across the transform.

The invariance of a unit is determined by how its' weights, in addition to the variances of inputs, scale these correlations. In the case of unit 83 its weights transform Figure 6.37 into Figure 6.38. The weighted covariance structure of this unit shows a few outlier contributors. Highlighted in purple is the highest contribution to this unit's invariance, the point highest on the y-axis. It is an interaction between two different units at the same position (Conv1 channel 5 and 47). We can examine the element-wise interaction and see, in terms of a linear interaction, what drove this invariance.

The weighted joint distribution between input channel 5 and 47 is plotted in row A of Figure 6.39. In the left column are the responses of these two units for the reference stimuli and on the right the responses to the reference for channel 47 and the transformed for channel 5. The correlation for the reference case is near 0 and the correlation across the transform is high. This is consistent with Figure 6.38 (purple highlighted orange point high on y-axis but near 0 on the x-axis). We can now trace this response property backward. Pre-weighting (row B) these two units had positive responses (consistent with rectification) thus the weighting applied negative weights to these two inputs maintaining the sign of the correlation. Interestingly it is the invariance of a suppressive contribution which most contributes to invariance in the output. One layer down is the max-pooling mask layer (see Methods, 'Max-pooling mask') and we select the position in the 3x3 max-pooling window which had the highest transform covariance in the upper left corner of the pooling window (sx=0, sy=0). Note most responses are either low or high as intermediate responses get set to 0 by other positions with higher responses. This position corresponds to a rectified Conv1 unit at position x=26 and y=26 in the Relu1 feature map (row D) which in turn corresponds to the same position in the Conv1 layer (row E). We see that rectification took a strong negative correlation and brought it to 0 while largely maintaining the positive correlation.

Finally, we can examine the weights of these units and the stimuli which most drove this strong correlation (orange point row 5). In Figure 6.40 row A is the stimuli which best drove invariance (Figure 6.38 orange point, row E, column 2) we see that it is the boundary of a shape. The weights of units 47 and 5 (row B) are flipped versions of each other which align with the dark vs light boundary of the reference and transformed stimuli. Thus in achieving invariance the interaction between these two units is essentially to 'follow' the transformation across units. Zooming out in row C see the shape had a boundary that ran through the center of the RF.

Thus above we have provided clear insight into how some fraction of invariance was achieved using a decomposition of covariance. This approach has been highly fine-grained thus it is natural to ask how many of the 2,560,000 potential input interactions will we need to examine before we have a complete picture of how one of the 192 Conv2 units achieved invariance? Thus while the path may be clear, examining all interactions which contribute to invariance, may be time-consuming.

This motivates examining aggregate statistics of how invariance is achieved. We found the main contribution to unit 83 was highly correlated across the transform which might suggest invariance is achieved by adding up transformations of input filters equivariant with transformations of the input (see [59] for review of equivariance). Under this notion correlation of inputs contributing to invariance should typically be very high. To test this we take an average of all positive correlations weighted by their variance contribution thus giving a 'typical' substantive contributor to invariance: $\bar{r}_c = 0.43$ thus on average it is moderate correlations across the transform producing invariance in this case.

Mean approximation of invariance

Some variation in invariance can be explained on the basis of a simpler approximation. We approximate invariance by only considering the average of the diagonal and off-diagonal of Σ_r and Σ_t (Eqn. 6.7 and 6.8)

$$\rho_{r,t} \approx \frac{\bar{D}_t \sum_i^n w_i^2 + \bar{O}_t \sum_{i \neq j}^n w_i w_j}{\bar{D}_r \sum_i^n w_i^2 + \bar{O}_r \sum_{i \neq j}^n w_i w_j}.$$

There is a simple interpretation of the quantity $\sum_{i\neq j}^{n} w_i w_j$ if we scale $\vec{w}$ to unit length and call the result m_p:

$$m_p = \frac{\sum_{i\neq j} w_i w_j}{\sum_{i=1}^{n} w_i^2} = \frac{\sum w_i w_j}{\sum_{i=1}^{n} w_i^2} - \frac{\sum_{i=1}^{n} w_i^2}{\sum_{i=1}^{n} w_i^2}$$

$$= \frac{(\sum_{i=1}^{n} w_i)^2}{\sum (w_i - \overline{w})^2 + \frac{1}{n}(\sum_{i=1}^{n} w_i)^2} - 1 = \frac{n^2 \overline{w}^2}{n s_w^2 + n\overline{w}^2} - 1$$

$$= n\frac{\overline{w}^2}{s_w^2 + \overline{w}^2} - 1 \in [n-1, -1] \tag{6.12}$$

where $\bar{w}^2 = (\frac{1}{n}\sum_i^n w_i)^2$ and $s_w^2 = \frac{1}{n}\sum_i^n (w_i - \bar{w})^2$. Thus scaling the numerator and denominator we can write:

$$\rho_{r,t} \approx \frac{\bar{D}_t + \bar{O}_t m_p}{\bar{D}_r + \bar{O}_r m_p}. \tag{6.13}$$

In general, m_p is increasing with the absolute value of the mean and decreasing with the magnitude of the residual of $\vec{w}$. This is closely related to the mean to variance ratio of the weights. So as the weights approach a constant the diagonal term is favored and as the mean of the weights goes to zero the off-diagonals are favored. The term m_p also grows with n the number of subunits essentially because the number of covariance terms increases as the square of the number of variance terms. So even weak element-wise correlations can have a large influence if the subunit population is large.

Our approximated invariance approaches $\frac{\bar{O}_t}{\bar{O}_r}$ as the weights become constant and $\frac{\bar{D}_t}{\bar{D}_r}$ as the mean of the weights goes to zero. This predicts a relationship where if $\frac{\bar{D}_t}{\bar{D}_r} < \frac{\bar{O}_t}{\bar{O}_r}$ then ρ_μ increases with $\bar{D}_r + \bar{O}_r m_p$ the response variance. Empirically we find for Conv2 $\frac{\bar{O}_t}{\bar{O}_r} = 0.96 > \frac{\bar{D}_t}{\bar{D}_r} = 0.25$ thus we expect invariance to increase with m_p.

Plotting invariance as a function of m_p (Figure 6.41) we do indeed see a modest but significant relationship for the Conv2 units (blue), and an even stronger relationship for the individual spatial subunits (orange). Thus theoretically, and here observed empirically, simply uniformly summing over inputs tends to increase invariance. The average off-diagonal covariance $\bar{O}_t = 0.014$ is far smaller than that of the diagonal $\bar{D}_t = 0.20$ but the numerous $(n^2 - n)$ weak interactions cumulatively can overwhelm the, on average, low invariance of the individual units.

As the weights converge to a constant the flexibility in tuning decreases thus this is an example of a trade-off between selectivity and invariance.

6.5 Discussion

6.5.1 Summary

We have introduced a method of attribution on the basis of covariance to characterize both selectivity and invariance of multi-layer neural networks. We provide factorization of each linear nonlinear stage in terms of linear input covariance, nonlinear scaling of covariance, and linear weighting by the output unit. The method was demonstrated by giving insight into the selectivity of a model of V4 MUA and the invariance of the second layer of a DNN to the ON-OFF transformation of stimuli.

6.5.2 V4 MUA model prediction

We analyzed a model of V4 MUA. We now consider an explanation for the broad tuning we found by considering a few simple factors involved in which spikes are counted in MUA. MUA are spikes that crossed the threshold that were not from the single unit being recorded. This count likely reflects action potentials from more than one neuron. Assuming a symmetrical decay of spike amplitudes with distance from the electrode tip a nearby neuron would be more likely to be counted than a far one. Though, given a uniform density of neurons, the number of neurons with a given distance grows cubically. Thus it seems plausible MUA would reflect the spike of many neurons. Finally, if tuning curves between neurons are correlated, the resulting variation across stimuli will be higher than if neurons were uncorrelated. Thus the MUA activity will over-represent tuning which is common amongst neurons.

Loosely speaking, if the population was large enough then it seems plausible that the common tuning between most neurons in V4 are features that tend to drive V1 neurons, one of their major inputs, potentially explaining the large fraction of variance explainable by mean correlation and weight of Conv1 filters tuned to spatial frequency and orientation for the MUA we analyzed.

The rest of the variation could be explained by the smaller more local population of neurons and thus their individual tuning properties do not get averaged out into their shared inputs.

A hypothesis to test this proposal is as follows: vary the spike detection threshold and determine whether the mean to variance ratio of weights which best fits the V4 unit co-varies. If the threshold is low then we predict the mean of the weights goes up but if the threshold is high the mean goes down.

6.5.3 DNN invariance analysis

The strongest individual driver to the invariance of the unit we studied was the result of filters which were equivariant [59] to the transformation of the stimuli. This was not typical, more often it was many weak correlations which taken together contributed to invariance. This challenges the notion that invariance is constructed by 'tiling' the transformations of stimuli. For example a convolutional architecture where filters tile space, is often associated with translation invariance [9–13]. Instead, weak distributed population correlation may be a more efficient method of achieving invariance while potentially accounting for the classic problem of a combinatorial explosion when trying to tile all possible transformation of an input.

6.5.4 Further work

The analyses done here were largely for the purpose of explication of the attribution method. The method can be further refined by a more thorough analysis of DNN models of the nervous system. Recent work has found significant increases in variance explained of V1 by DNN's [33, 34] but little insight into this tuning is given except for synthesized visualizations which give a qualitative sense of tuning [34]. This method could help assess what tuning has been captured that is fundamentally different from what is known about V1 tuning.

This method can also be applied to temporal selectivity as time can be treated as just another axis of the input like space was for the DNN studied here. Work on the naturalistic temporal selectivity of intermediate regions of the ventral stream is sparse and could be bolstered by DNN model predictions.

6.6 Figures

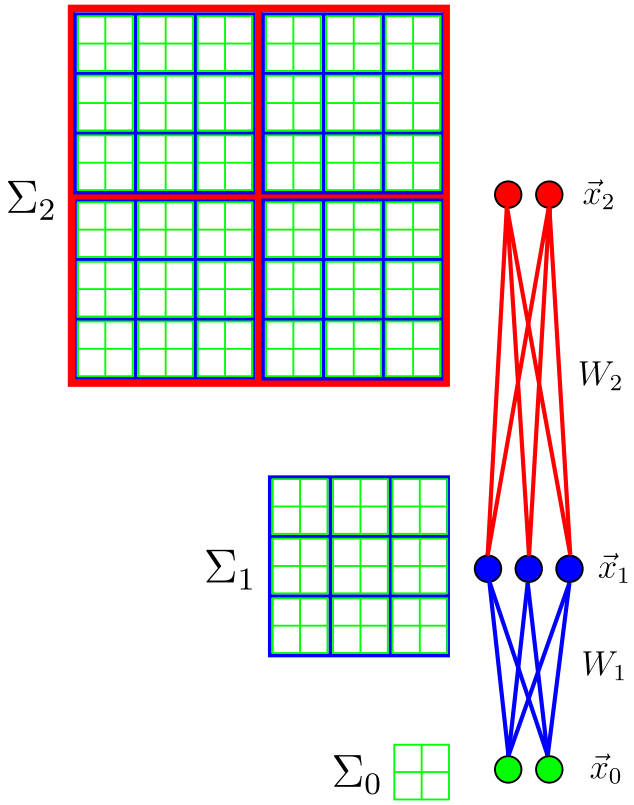


FIGURE 6.1: Schematic of DNN covariance decomposition. This represents the decomposition of covariance of a 2-layer network with 2 inputs ($\vec{x}_0$ green circles) whose covariance is given by the 2×2 covariance matrix Σ_0. The three hidden layer units ($\vec{x}_1$ blue circles) covariance, before the nonlinearity, are determined by the weight matrix's (W_1) transformation of Σ_0 such that each entry (blue square) is a weighted sum of the input covariance 2×2 (green grid, within blue squares). The nonlinearity then scales this covariance and the two output units ($\vec{x}_2$ red circles) covariance, in turn, is a weighted sum of the covariance of their inputs covariance (Σ_1 blue grid nested within red Σ_2). Thus variance and covariance of the outputs can be recursively attributed to linear weighting and nonlinear scaling of input covariance. Factorization of these covariance matrices in terms of the linearity, nonlinear scaling, variance, and correlation are discussed in Methods.)

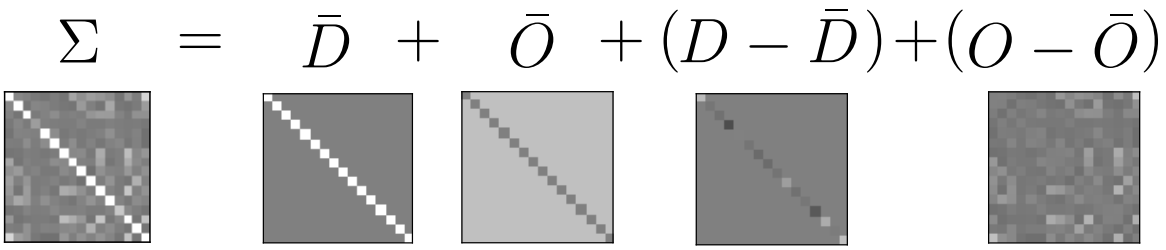


FIGURE 6.2: Decomposition of covariance into mean/residual and variance/covariance. The covariance matrix (Σ) can be decomposed into the sum of the average of the variance ($\bar{D}$ white diagonal), average of the covariance ($\bar{O}$ light grey off-diagonal) , residual of the variance ($D - \bar{D}$, pattern on diagonal), and residual of the covariance ($O - \bar{O}$ pattern on off-diagonal). This allows a clear attribution of covariance to population averages ($\bar{D}$ and $\bar{O}$) versus heterogeneous patterns of tuning ($D - \bar{D}$ and $0 - \bar{0}$) and individual tuning (D) versus interactions between tuning of units (O).

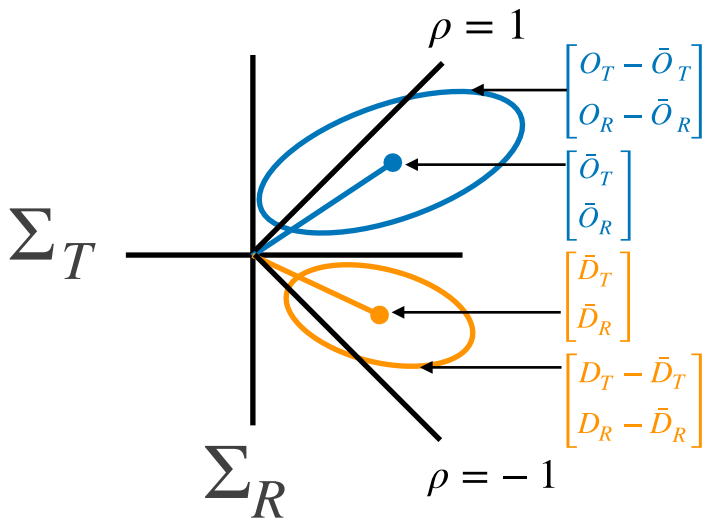


FIGURE 6.3: Geometric interpretation of covariance and invariance. Invariance is attributed to the relationship between covariance matrices with regards to reference stimuli (indicated with Σ_R) and reference stimuli covariance to transformed stimuli (indicated with Σ_T). The slope of the weighted sum of these matrices gives the invariance (ρ). The average of their respective diagonals (the orange point at end of line $[\bar{D}_T, \bar{D}_R]$) reflects the typical invariance of individual units across the transform, the residual (distribution indicated by orange elliptical $[D_T - \bar{D}_T, D_R - \bar{D}_R]$) reflects variation in the invariance of individual units and thus how weighting more invariant units can increase invariance (the portion of elliptical above average). The average of the off-diagonals (the blue point at end of line $[\bar{O}_T, \bar{O}_R]$) reflects the average covariance of units to each other across the transform which if it is positive can support invariance (blue line sloped upwards) and finally the residual off-diagonal reflects variation in covariance with and across the transformation (distribution indicated by blue elliptical $[O_T - \bar{O}_T, O_R - \bar{O}_R]$).

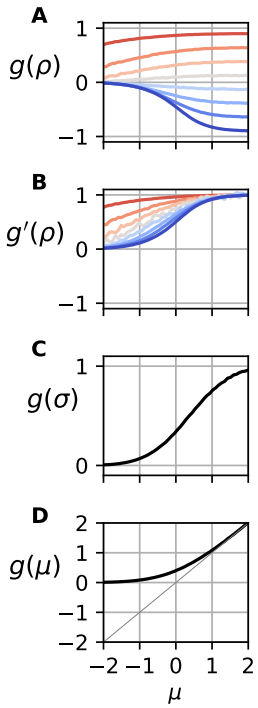


FIGURE 6.4: Effect of rectification on bivariate normal moments. Simulation of the moments of a bivariate normal after rectification as a function of the mean μ (same for both units) for a variety of correlations. **(A)** Rectified correlation decreases with μ, beginning value indicated by where traces asymptote on right. Negative correlation (blue) decreases more rapidly than positive (red). **(B)** Ratio of rectified to unrectified correlation casts $g'(\rho)$ in the bivariate normal case as a function of μ and ρ where for high correlation there is little nonlinear attenuation of correlation (red traces have less slope) but for negative a large scaling (blue traces rapidly decay). **(C)** The marginal rectified variance of both units invariably decays with μ. **(D)** The rectified mean (y-axis) is greater than or equal to the unrectified (x-axis).

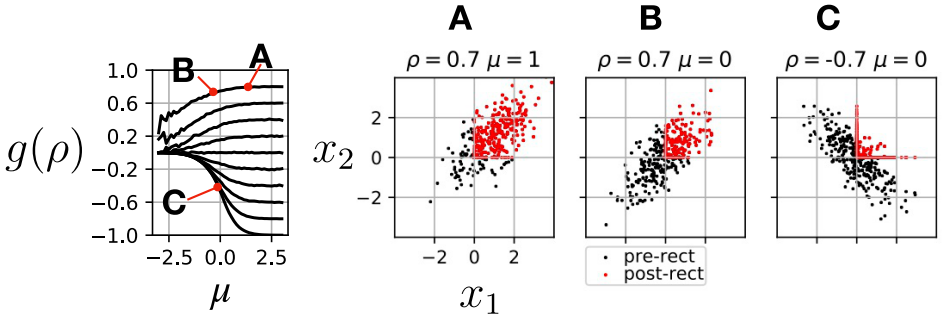


FIGURE 6.5: Effect of rectification on bivariate normal correlation. Simulation of the correlation of a bivariate normal's correlation after rectification as a function of the mean μ (same for both units) shows positive correlation decay with a lower mean (compare A to B) but less rapidly than negative correlation (compare B to C). **(A)** The unrectified responses (black points) correlation is not reduced much when rectified (red). **(B)** If the mean is decreased, then correlation drops more intuitively because more of the tail is cut-off. **(C)** With the same mean but negative correlation magnitude drops drastically. Intuitively, because both tails are cut-off.

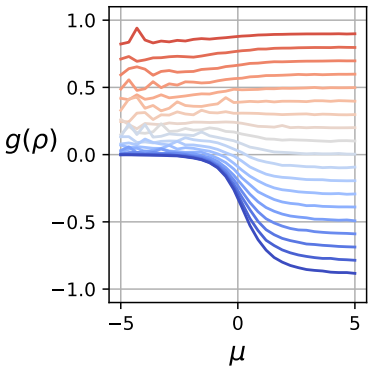


FIGURE 6.6: Effect of rectification on bivariate t-distribution correlation. Same simulation as previous figure except for a t-distribution with 3 degree of freedom. Increasing degree of freedom the t-distribution converges to a normal. We find t-distribution positive correlation is less attenuated by rectification and negative correlation (blue) can be switched to positive correlation (blue traces cross 0).

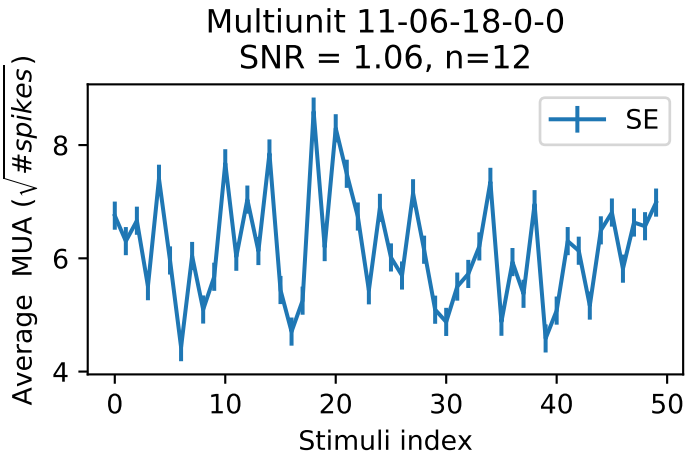


FIGURE 6.7: The average square root of the spike count in response to 50 stimuli with associated SE.

FIGURE 6.8: Stimuli ranked by excitatory and suppressive drive on MUA. The top row from left to right are the stimuli that evoked the highest average MUA and on the bottom row stimuli which evoked the lowest from left to right.

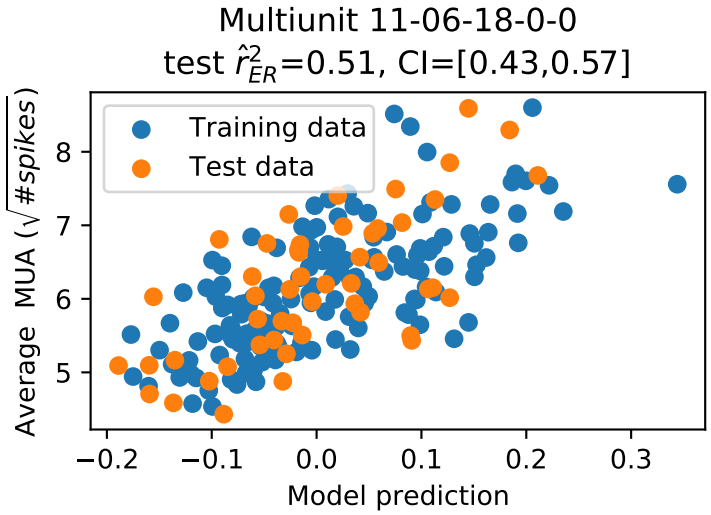


FIGURE 6.9: Scatter of DNN model predictions and MUA. In blue are the responses from training data and in orange from the test data which the model was not trained on.

FIGURE 6.10: Stimuli ranked by excitatory and suppressive drive on model prediction. The top row from left to right are the stimuli that evoked the highest model response, and on the bottom row stimuli that evoked the lowest from left to right.

FIGURE 6.11: Out of 7,290,000 image patches drawn from ImageNet [83] the most excitatory and suppressive drivers of the model fit to V4.

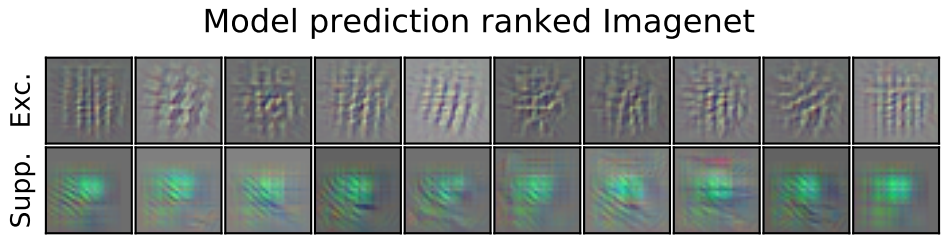


FIGURE 6.12: The same images as Figure 6.11 but visualized via guided back-propagation [133] see Methods.

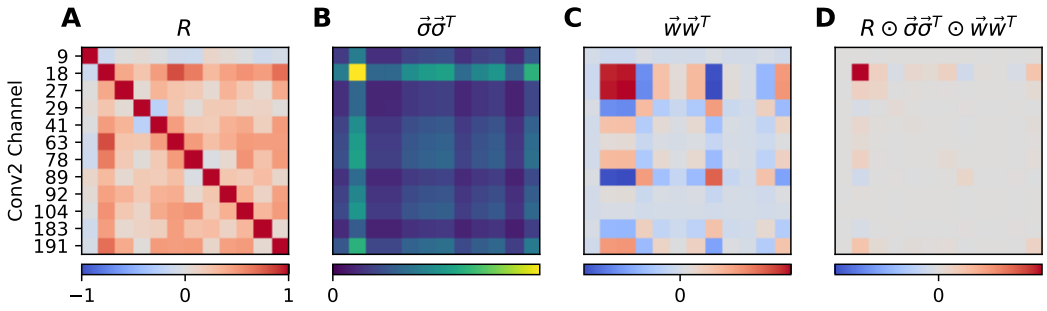


FIGURE 6.13: The factors whose product gives the weighted covariance of inputs to V4 model. **(A)** Correlation between 12 rectified Conv2 input to V4 model. **(B)** Outer product of variance of 12 input units, element-wise product with correlation (A) gives covariance.**(C)** Outer product of weights on inputs. **(D)** The element-wise product of A-C gives weighted covariance whose sum is the variance of the V4 MUA model.

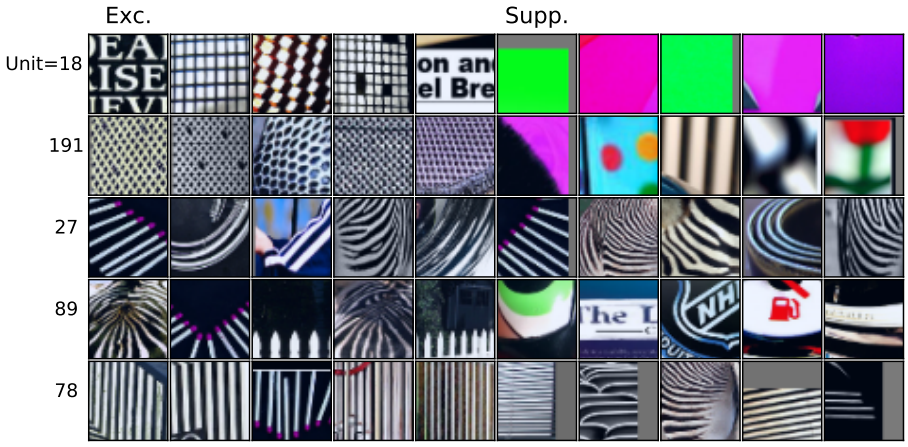


FIGURE 6.14: The most excitatory and suppressive drivers of the model fit to V4 inputs. Rows are input units ranked by greatest variance contribution, first 5 columns gave highest response of unit, next 5 lowest response of unit across ImageNet patches.

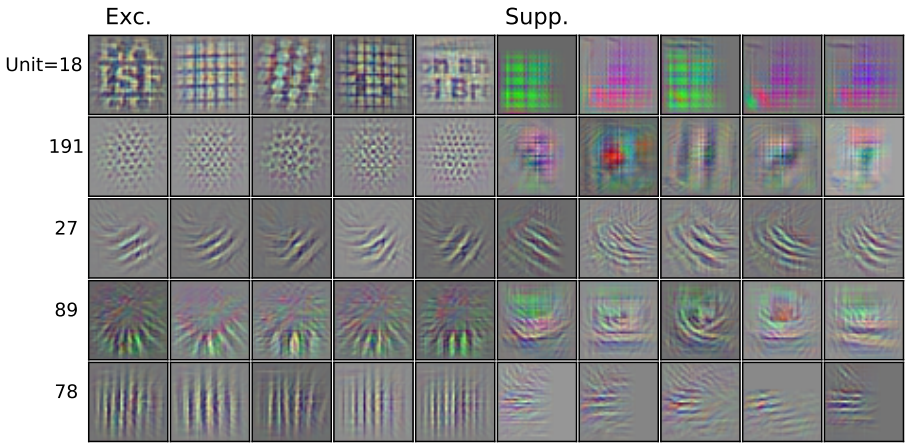


FIGURE 6.15: Same images as Figure 6.14 but visualized via guided backpropagation.

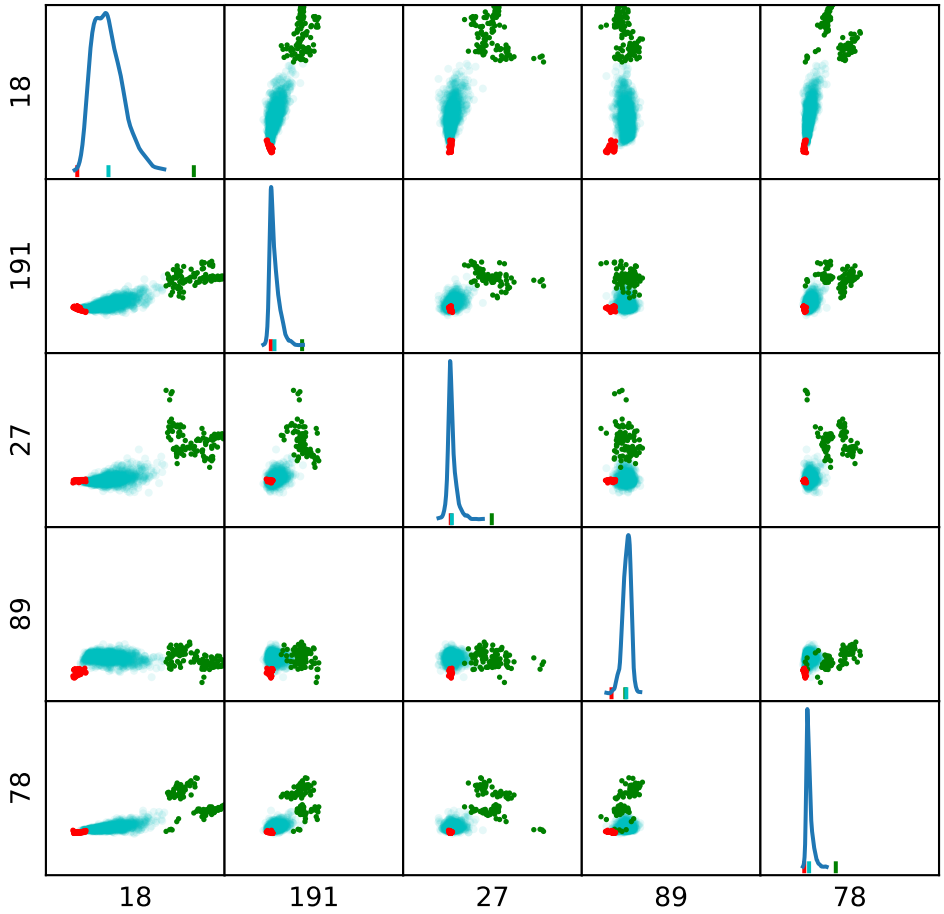


FIGURE 6.16: Pairwise relationships of top contributors to response variance of V4 model. In cyan is the sub-sampled scatter of all image patches, in green is the top 100 image patches for the V4 model, and red bottom 100. The KDE of marginal distributions are shown along diagonal with means of the three distribution subsets indicated with vertical ticks at the bottom.

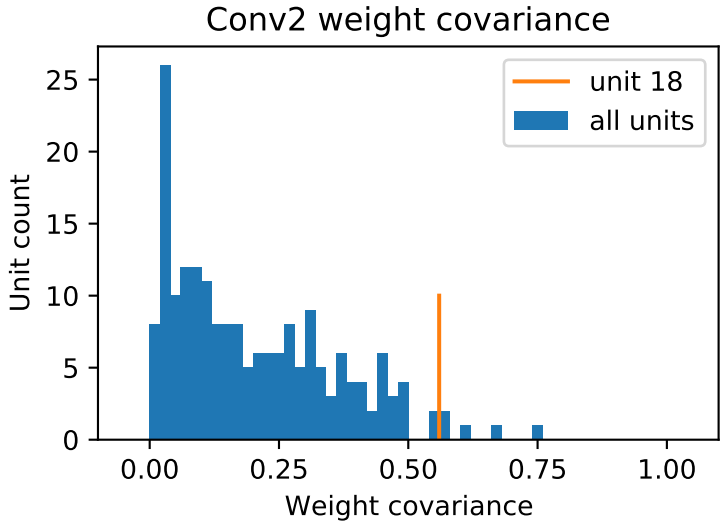


FIGURE 6.17: Conv2 normalized weight covariance distribution (Eqn. 6.11) of the 192 units. A low value indicates a low correlation between the $5 \times 5 = 25$ weight vectors on the 64 input feature channels at each spatial position and a value of 1 indicates all weight vectors are identical up to a shift and scaling. Orange vertical indicates weight covariance of the largest contributor to model variance thus this unit is more spatial homogenous in its selectivity than typical.

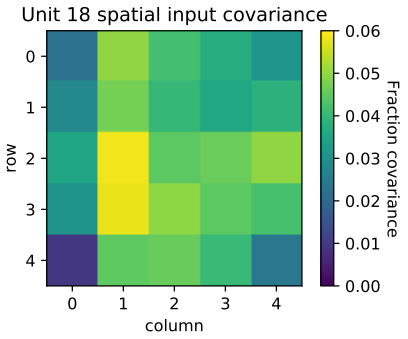


FIGURE 6.18: Conv2 unit 18 fraction contributed variation across spatial subunits.

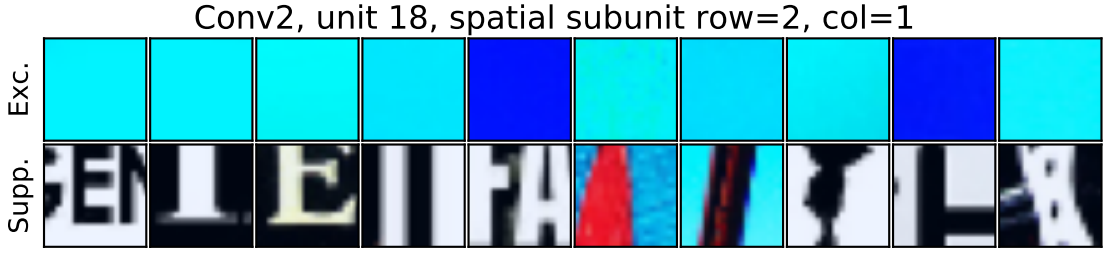


FIGURE 6.19: Conv2 unit 18 spatial subunit row 2 column 1 ranked images.

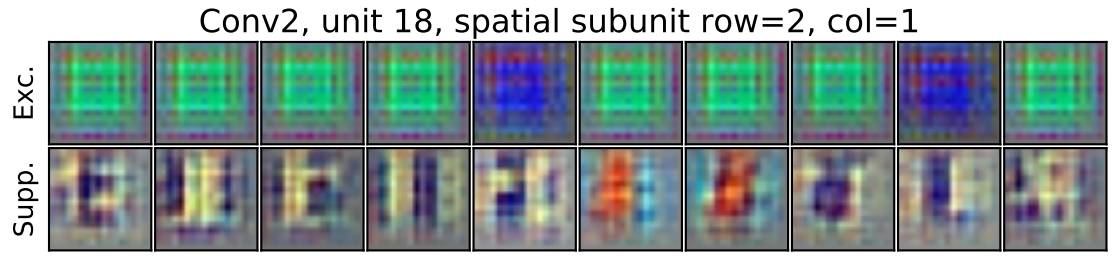


FIGURE 6.20: Conv2 unit 18 spatial subunit row 2 column 1 ranked images visualized.

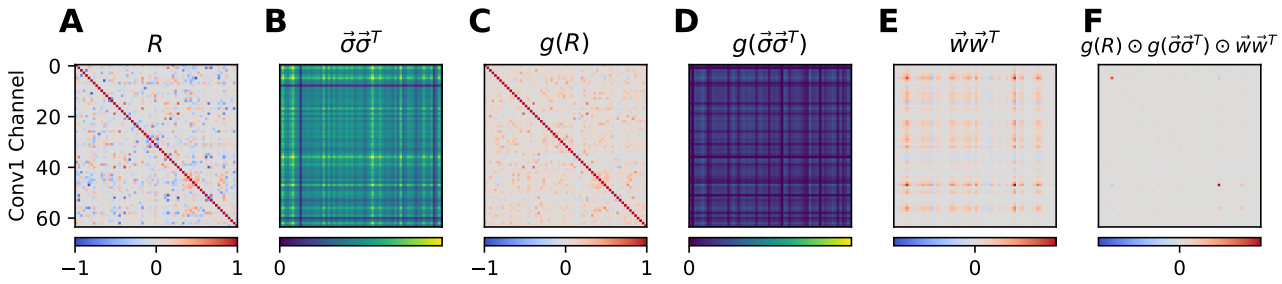


FIGURE 6.21: Conv1 feature channel (64) covariance weighted by unit 18 row 2 column 1 weights. **(A)** Correlation before rectification between input units. **(B)** Outer product of variance before rectification. **(C)** Correlation after rectification. **(D)** Outer product of variance after rectification. **(E)** Outer product of weights. **(F)** element-wise product (C-E) whose sum is the variance of unit 18 row 2 column 1 at a single position within average pooling window.

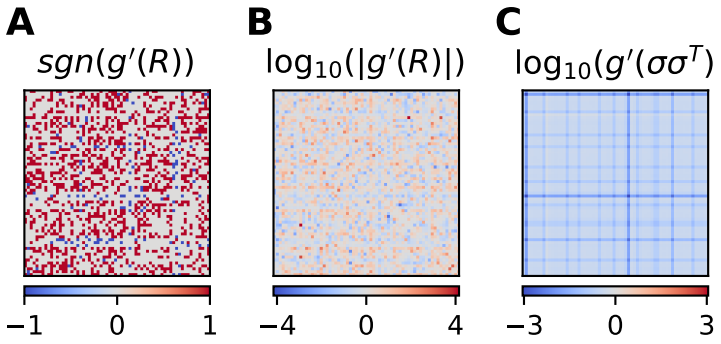


FIGURE 6.22: Conv1 channel covariance scaled by nonlinearity (g'). **(A)** Transition of sign of correlation from linear to nonlinear correlation where red indicates negative to positive, blue positive to negative, and grey no change in sign. **(B)** The logarithm of the scaling of the magnitude of correlation by rectification. Grey ($\log(1) = 0$) indicates no scaling, blue a decrease, red an increase. Many entries go from negative to positive consistent with a t-distribution Figure 6.6. **(C)** Outer product of scaling of linear variance by rectification. Grey ($\log(1) = 0$) indicates no scaling, blue a decrease, red an increase. All entries decrease, consistent with Figure 6.4C, but not uniformly.

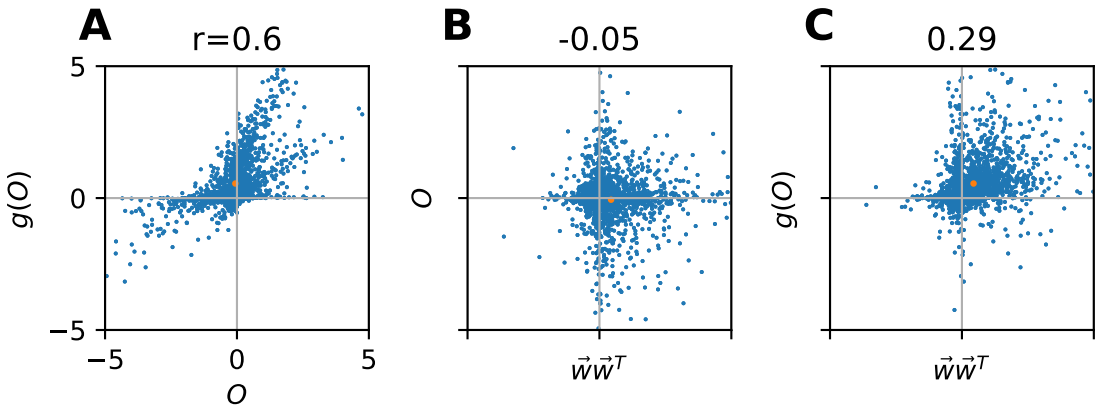


FIGURE 6.23: Relationship between Conv1 off-diagonal linear, nonlinear covariance, and outer product of weights. **(A)** Scatter of off-diagonal covariance before (x-axis) and after rectification (y-axis), mean indicated in orange. **(B)** Scatter between outer product of weights of spatial subunit on linear input covariance, where correlation is weak between the two. This correlation between off-diagonal weights and nonlinear covariance scales influence of off-diagonal see Eqn. 6.8. **(C)** Scatter between outer product of weights of spatial subunit on nonlinear input covariance, where correlation is stronger then (B) thus weights are aligned with the influence of nonlinearity on covariance.

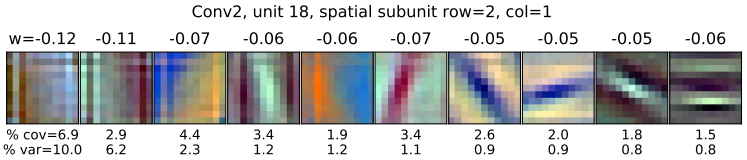


FIGURE 6.24: The top 10 Conv1 filters ranked by the percent of variation they contribute (values given in bottom row). The covariance contributed i.e. resulting from off-diagonals is given above that, and above the filter visualization are the weights applied to the filters.

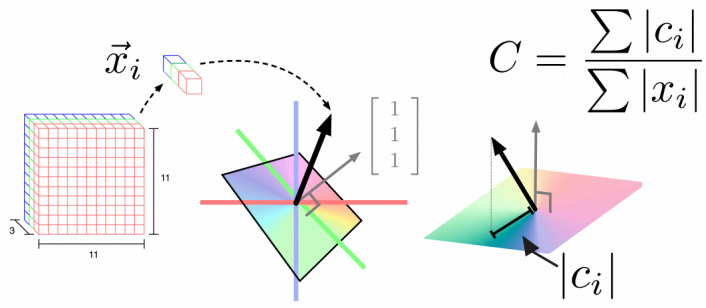


FIGURE 6.25: Chromaticity index of Conv1 filters denominator takes each 3×1 RGB pixels of the 11×11 filter and finding its vector length in R^3 (black arrow length) then summing all these lengths across pixels. The numerator is the sum of lengths of these same vectors projected into the chromatic plane (colorful plane orthogonal to grey luminance vector of ones). This ratio gives a number between 0 and 1 where 0 indicates variation solely along the vector of ones and 1 indicates variation solely within the chromatic plane.

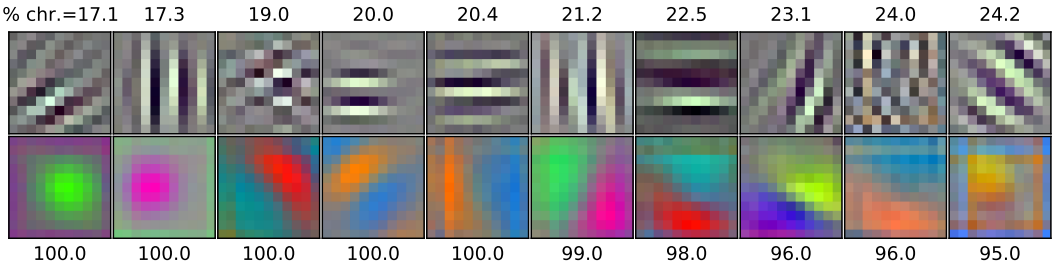


FIGURE 6.26: Conv1 filters ranked by chromaticity. Top row gives most achromatic with titles indicating percent RGB pixel variance in chromatic plane. Bottom row gives most chromatic filters with bottom labels indicating percent RGB pixel variance in chromatic plane.

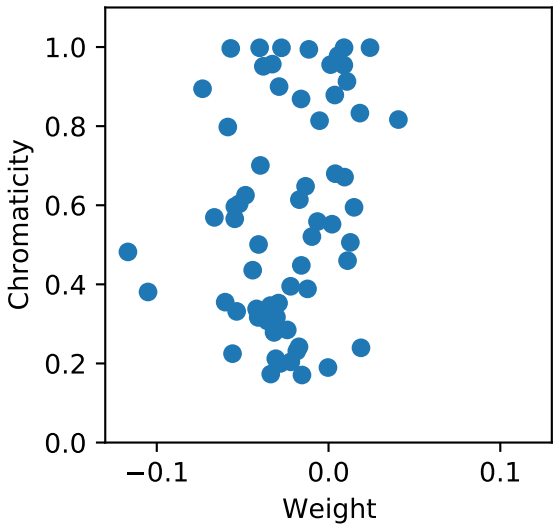


FIGURE 6.27: Chromaticity vs weight on Conv1 by Conv2 unit 18 row 2, column 1 spatial subunit. Top two contributing units are shown on far left same as the two filters show in Figure 6.24. Note overall mean of weights is negative and overall there is not a strong relationship between weights and chromaticity.

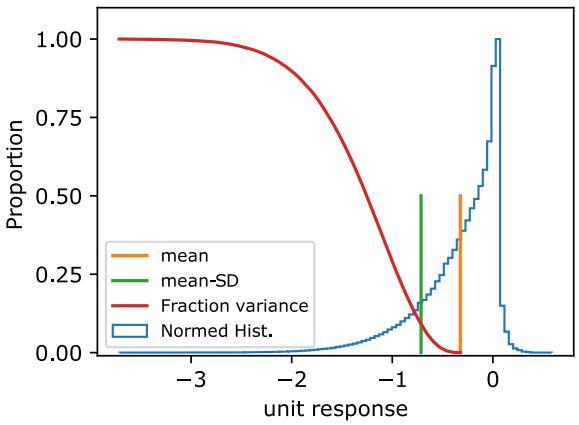


FIGURE 6.28: Response distribution of spatial subunit at row 2 column 1 of unit 18 in Conv2 (blue). Fraction variance is the cumulative sum of squared deviations from the mean for all samples less than the mean divided by the total (red). One standard deviation below mean is indicated by vertical green line.

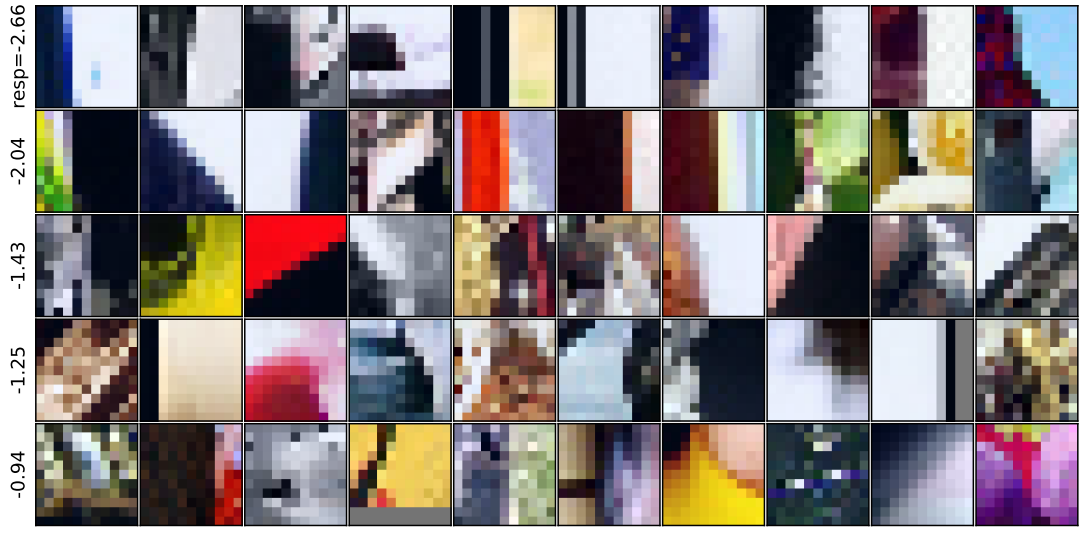


FIGURE 6.29: Ranked image patches of spatial subunit for more typical response levels. Each row of image patches gave approximately the same response. For reference to original distribution compare to Figure 6.28.

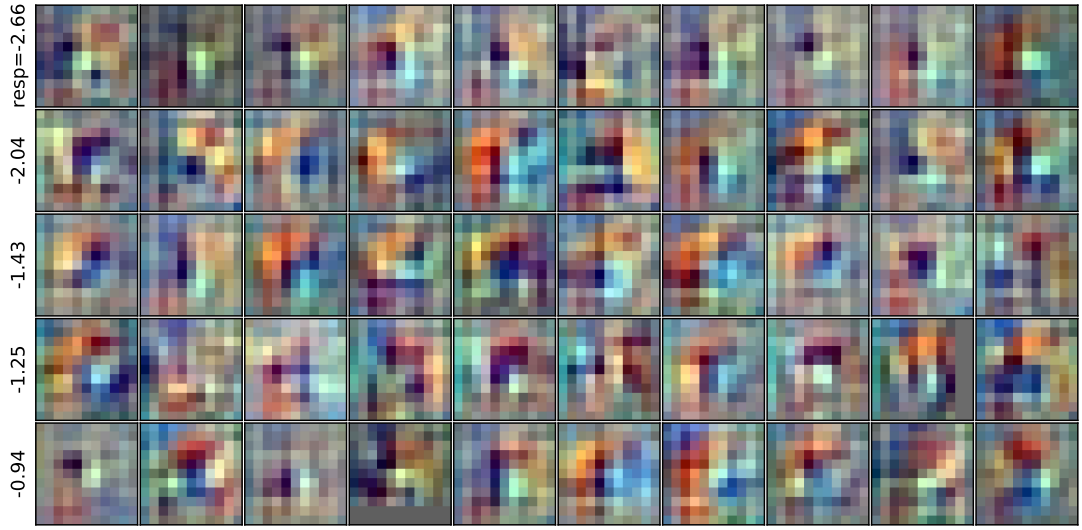


FIGURE 6.30: Same as Figure 6.29 but visualized by guided backpropagation.

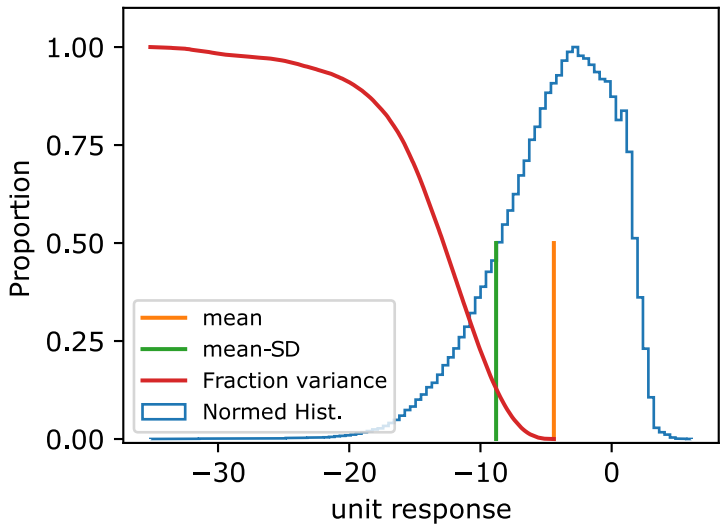


FIGURE 6.31: Response distribution of Conv2 unit 18. Same plots as Figure 6.28 except for Conv2 unit 18.

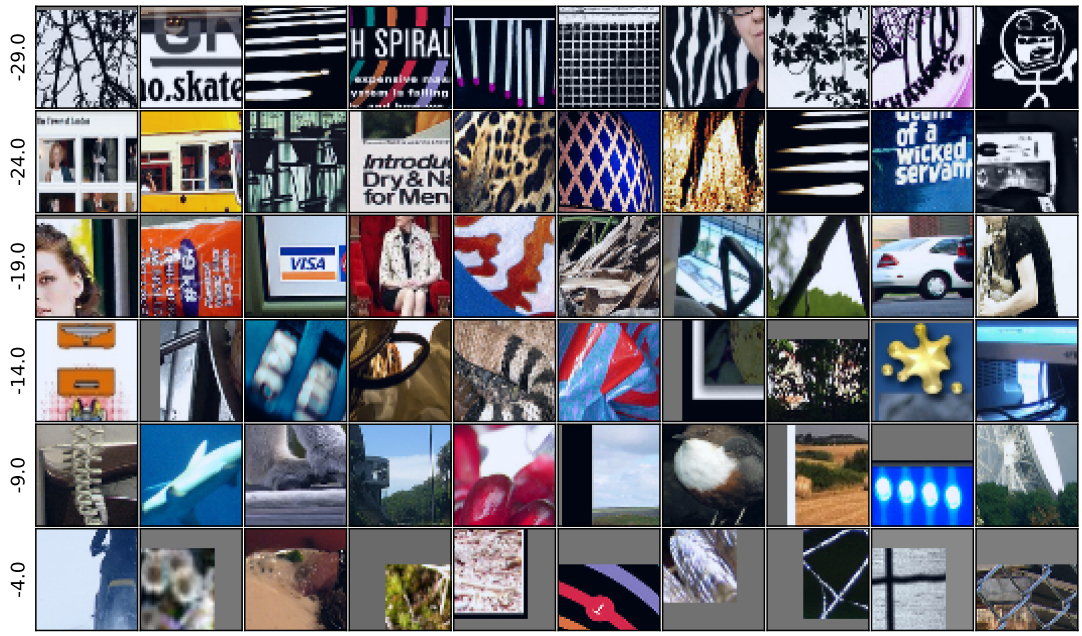


FIGURE 6.32: Ranked image patches of Conv2 unit 18 for more typical response levels. Same as Figure 6.29 except for Conv2 unit 18.

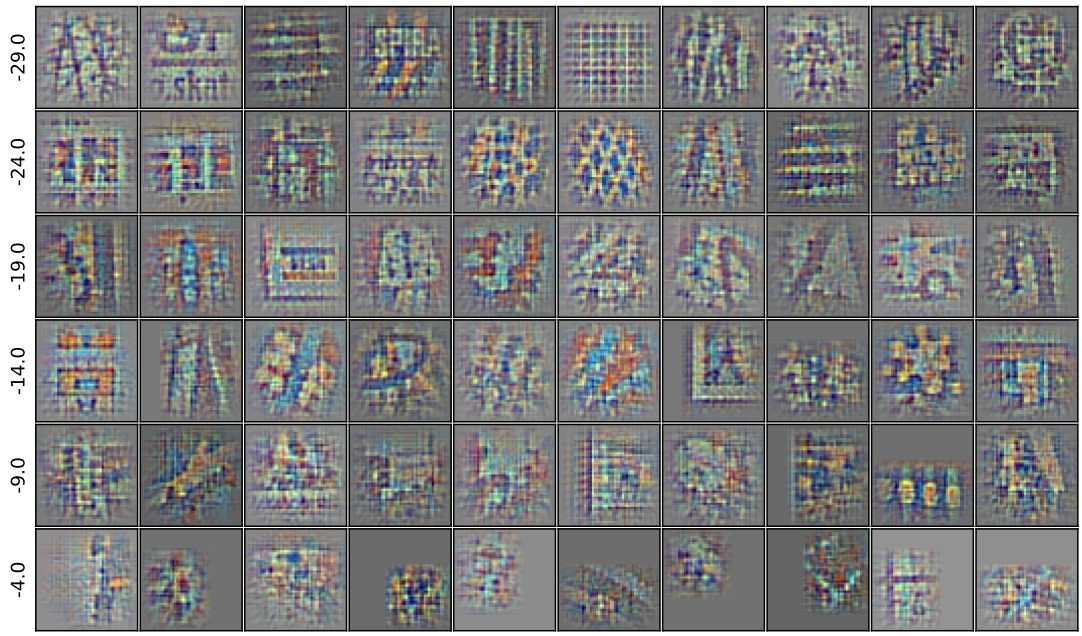


FIGURE 6.33: Same as Figure 6.32 except visualized by guided backpropagation.

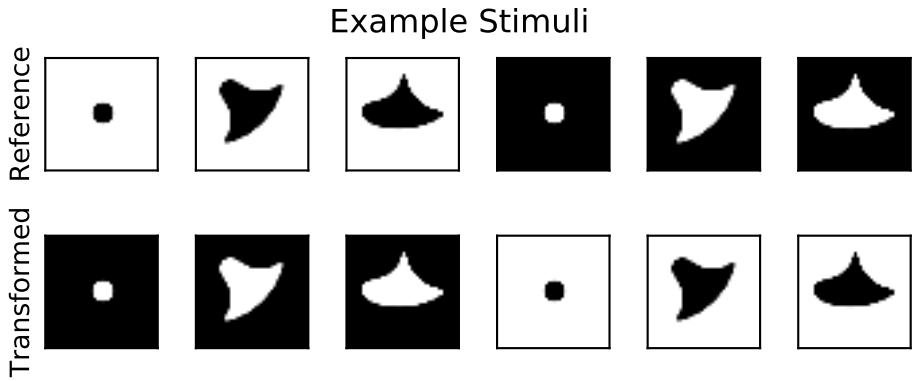


FIGURE 6.34: Example ON-OFF stimuli. Stimuli shapes are drawn from Figure 2.3. In the top row are reference stimuli, the first 3 are OFF stimuli where the background is set to 1 and foreground 0 and vice-versa for next three stimuli. The bottom row is the transformed stimuli where ON stimuli become OFF and ON become OFF.

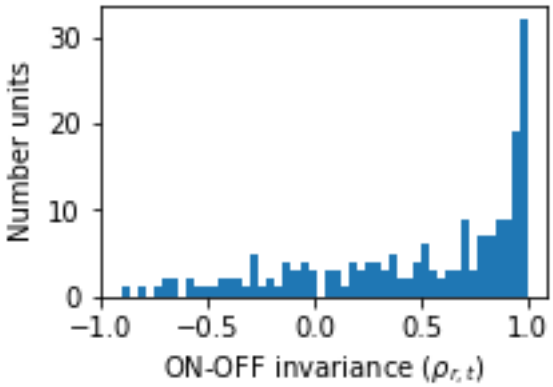


FIGURE 6.35: Correlation between responses of centered Conv2 units to ON and OFF stimuli plotted as a histogram.

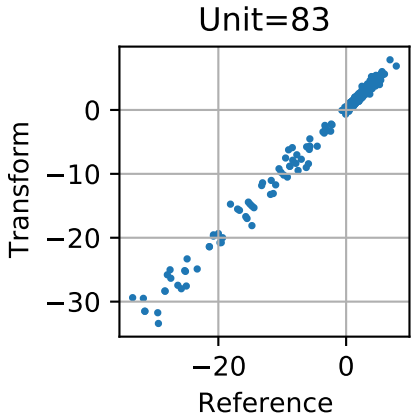


FIGURE 6.36: Correlation between responses of centered Conv2 units to ON and OFF stimuli plotted as a histogram.

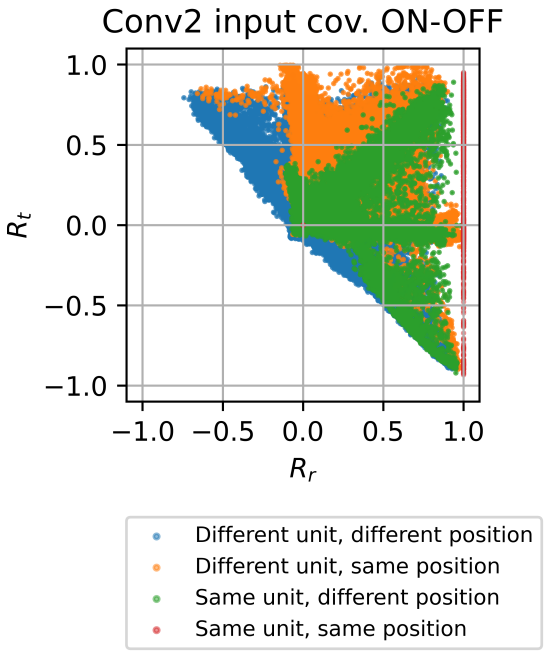


FIGURE 6.37: The correlation between all input units in the $5 \times 5 \times 64$ region, of the Maxpool1 layer, over which a Conv2 unit weights its input is for the reference stimuli (x-axis) and transformed (y-axis). In blue are the correlations between different feature channels (2 of the 64) across different positions (2 of the $25 = 5 \times 5$). In orange different feature channels at the same spatial position. In green different spatial positions of the same feature channel. In red the correlation of units with themselves, note all read points aligned above 1 because correlation to themselves is 1.

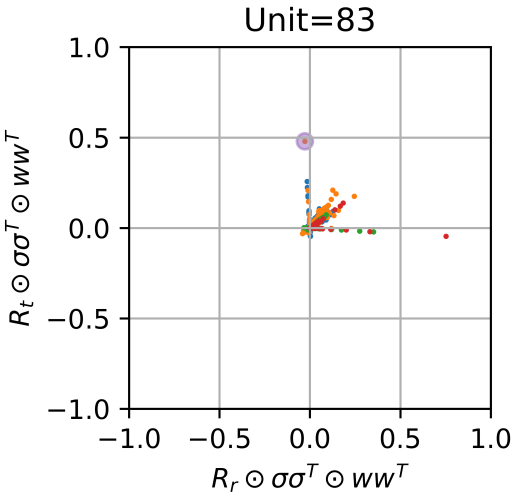


FIGURE 6.38: Covariance structure of inputs to Conv2 unit 83 after weighting. Colors indicate same categories as Figure 6.37. Highlighted unit responses shown in Figure 6.39 top row.

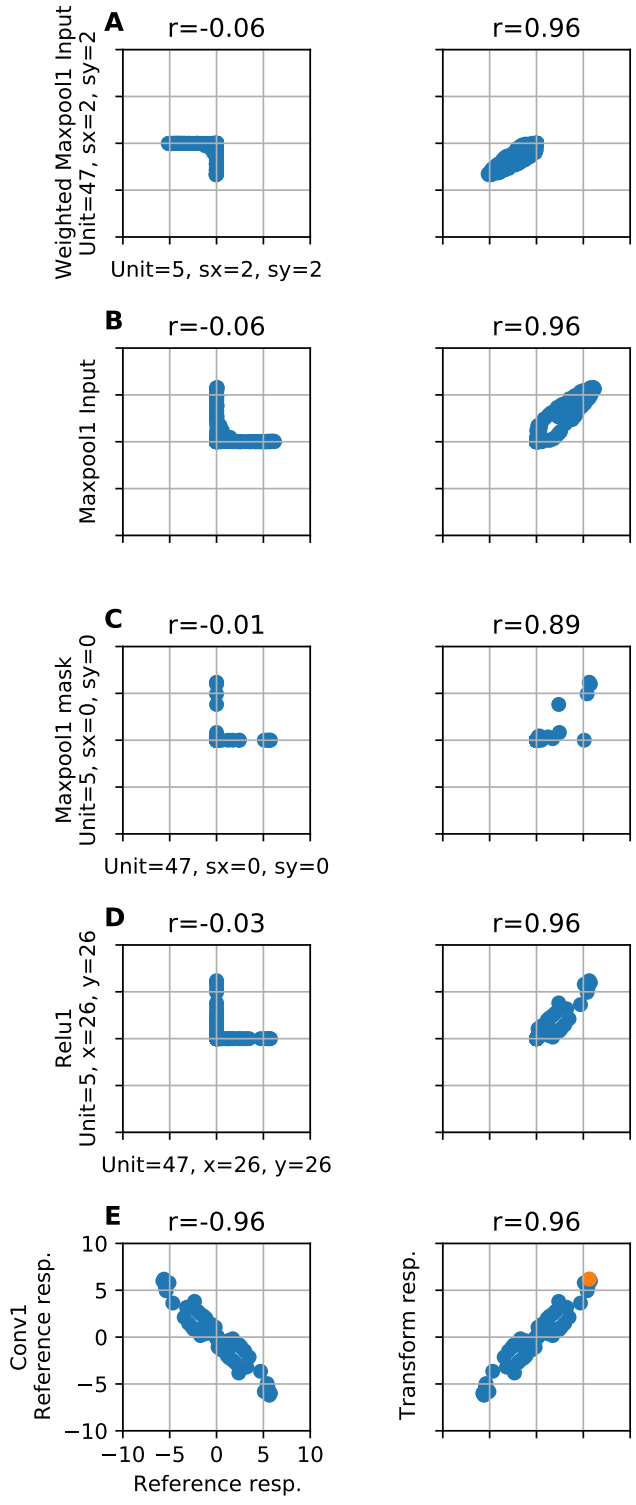


FIGURE 6.39: Conv2 unit 83's input units 5 and 47 at the same position (x=2, y=2) contributed most to invariance (see Figure 6.38). **(A)** Their weighted joint distribution (the response of these two units with the weights of unit 83 on them) is plotted for within transform correlation in the left column and across transform in the right. **(B)** Unweighted inputs. **(C)** Joint distribution of top Maxpool1 mask contributing to Maxpool1 invariance is in upper left of maxpool kernel (sx=0, sy=0). **(D)** Relu1 input to max mask layer corresponding to Maxpool1 mask input above. **(E)** Conv1 input corresponding to Relu1 above with sample contributing most to invariance highlighted in orange (see Figure 6.40).

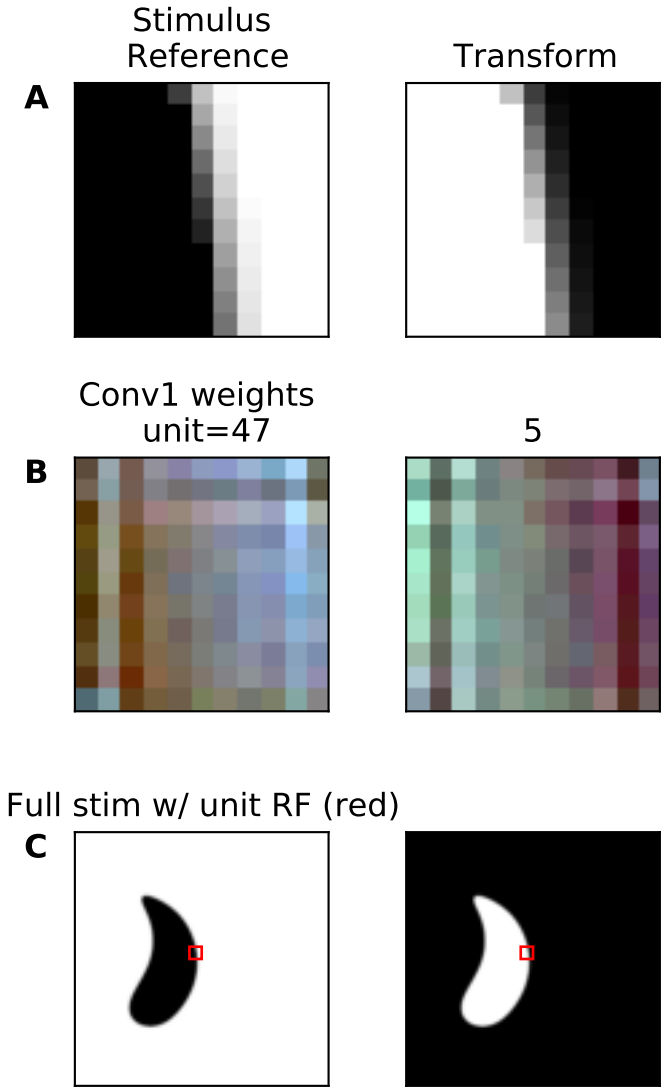


FIGURE 6.40: Inputs corresponding to greatest contribution to correlation in overlapping Conv1 units 47 and 5 (see orange point Figure 6.39). **(A)** The stimuli in the RF of the Conv1 units for OFF and ON in the left and right column respectively. **(B)** The visualized weights of unit 47 and 5 (min=0, max=1). **(C)** The full field stimulus with the RF of the Conv1 units delineated by red box.

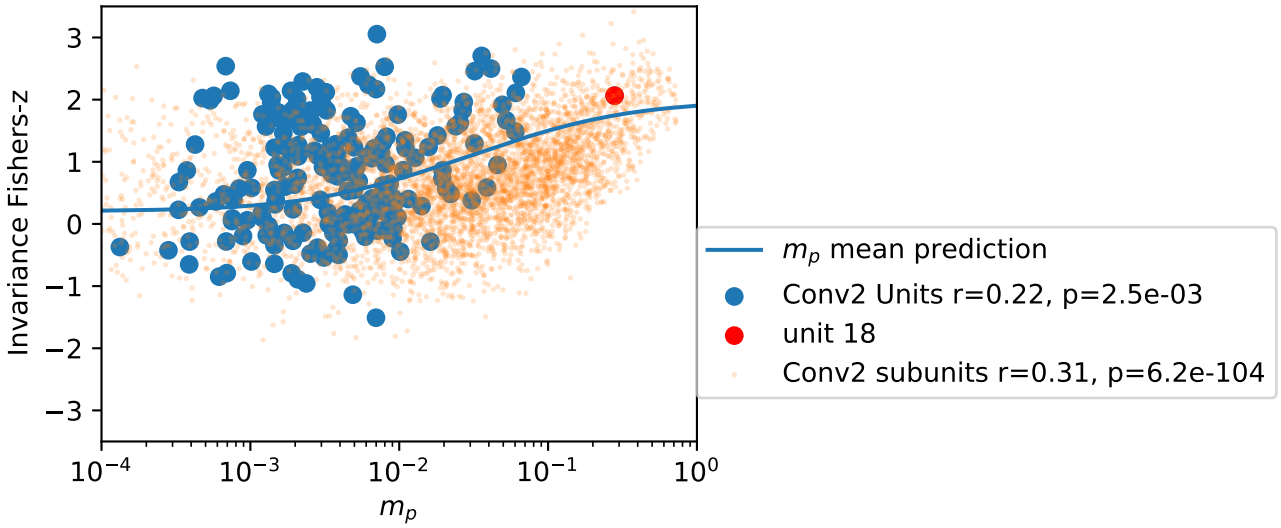


FIGURE 6.41: Invariance, Fisher-z transformed, of Conv2 units (blue $N = 192$), and spatial subunits (orange, $N = 192 \times 5 \times 5 = 4,800$), to ON-OFF shape stimuli as function of $\frac{1}{n-1} m_p$ (Eqn. 6.12) which increases with mean and decreases with the variance of the weights. The prediction of invariance under approximation of Eqn. 6.13 is plotted with blue trace. Conv2 unit 18 which was the primary contributor to the V4 MUA model is indicated in red.

Chapter 7

Discussion and Future Directions

In Chapter 2, I took a single-neuron approach to fit DNN models to visual cortical responses. I was confronted by the confound of attenuation of correlation by trial-to-trial variability in estimating the key properties in this study: translation invariance and correlation to a model of shape selectivity. The successes and challenges of this first major study motivated me to develop an estimator of correlation with less bias than all previous methods. I first developed an estimator for correlating noisy neural tuning curves to noiseless model predictions and then extended it to the correlation between pairs of noisy neural tuning curves required for estimating invariance. I applied this method to the estimation of the well-studied property of neuronal signal correlation, showed that it avoids confounds that prior studies had suffered from, and demonstrated a novel relationship between SNR and signal correlation. Also, motivated by observations in my original DNN to neuron comparative study, to better understand single-unit coding, I developed a method to track down variance through multi-layered networks, which is key to attributing responses to inputs and understanding how certain invariances are built.

7.1 Artiphysiology

I introduced the concept of 'artiphysiology' where the methods of a cortical sensory electro-*physiologist* are applied to a putative *arti*-ficially intelligent model of the nervous system that performs a concrete, plausibly intelligent, computation (in this case object recognition). This approach has great value because it brings the challenges faced by the cortical electrophysiologist, in applying their methods to a naturally intelligent nervous system, into a far more experimentally tractable, but still computationally relevant, artificial system. I argued this correspondence can aid both in sensory systems neuroscience research and in the advancement of artificial intelligence.

Questions about the nervous system, and the approach used to answer them, can be refined in the ideal experimental set-up: an unlimited number of experimental conditions, recording from every neuron, and the complete knowledge of the neural architecture. Any aggregate experimental measure can be explained in detail with regards to the computation being performed by the model, thus leading to the generation of hypotheses and refinement of conceptual approaches. Going the other direction, sensory electrophysiologists have spent decades refining their techniques for detailed, and importantly well-controlled, characterization of sensory representation. While interpretability techniques developed in the field of machine learning focus on explaining the outputs of a model, the electrophysiologist and artiphysiologist are focused on the internal representation that supports these outputs (or behavior). These two perspectives overlap and are fertile ground for further productive dialogue.

7.2 Accounting for trial-to-trial variability

My analytical work led me to account for trial-to-trial variability in the estimation of a handful of tuning curve statistics. The tuning curve is a theoretical quantity that the nervous system has no direct access to. But, it places strong constraints on the spiking activities of neurons, factors out unobserved processes generating trial-to-trial variability, and is experimentally tractable (i.e. can be well estimated with relatively few trials). Improved estimators can assist in achieving the important goal of disentangling trial-to-trial variability from tuning curves, thereby clarifying the relationship between these two quantities.

There are likely many statistics that suffer confounds from trial-to-trial variability. Statistics of the tuning curve that involves the calculation of a moment greater than or equal to two are candidate victims of trial-to-trial variability's confounding effect. For example, kurtosis, skew, and correlation-based analyses (e.g., CCA, PCA, RSA, and correlation-based clustering). In the case of statistics that estimate tuning curve variance, the estimators that I have developed could be simply plugged in. Indeed, the SNR metric, my improvement of Spearman's method $\hat{\rho}_{0_{\text{ER}}}$, and $\hat{r}^2_{\text{ER}}$, use the same unbiased estimates of tuning curve variance that I developed. Simulation and asymptotic analysis will be the final judge of whether, and under what conditions, the estimators behave reasonably.

In terms of model validation, accounting for trial-to-trial variability is valuable in part because it gives a theoretically achievable target for the sensory neuroscientist, $r^2_{\text{ER}} = 1$, and the means to accurately quantify how close they are to it, $\hat{r}^2_{\text{ER}}$. Taking an average of this estimate across a population then gives an estimate of how much total variation across an entire brain region could be explained. While a model with predictive power is not sufficient to improve understanding of the nervous system, it is necessary. The estimation of a relationship between resources (time spent recording) and a measure of scientific progress (explained variation) is appealing. It is appealing because typically this relationship, between resources and scientific progress, is uncertain and the uncertainty is difficult to quantify.

Further efforts to estimate potential gains in explained variation would be well worth developing. The model fit estimators that I developed can help with extrapolating explained variation with an increased number of trials (by setting its ceiling) or the number of neurons (by taking population averages) but not the number of stimuli. Model predictions will typically improve with the number of stimuli since their parameters are better constrained. For a linear model, estimating this increase seems analytically tractable, but for more complex models a simulation-based approach may be necessary.

7.3 Limitations of the tuning curve

The tuning curve of a single neuron is a theoretical scalar function that gives the expected response of that neuron for all stimuli. A subtle but important point is that the tuning curve is a look up table, not a computation. Computation is treated as a black box, and the expected value of the neurons response is solely given in terms of an association between inputs and outputs. In the example of the deterministic spiking network developed in the introduction of this thesis, if the stimulus window is long enough, then any neuron in the network will show no trial-to-trial variability and be described perfectly by its tuning curve. Yet a description of a neuron with a look up table of the response to every possible sequence of inputs over the course

of an organism's lifespan would be unwieldy. A more compact representation is the form of the computations performed by a neuron. A computations form is not a function of every possible input but the tuning curve is. The tuning curve better serves analysis of a neural process where computation is limited, and responses that are a function of short time scales naturally limit computation. For extended computations, such as those involving long term memory, like developing a PhD thesis, a description in terms of computation would be preferable to a tuning curve, whose window of input would need to extend across many long years. Sensory systems appear to operate on short time scales, consistent with the veridical encoding of rapid changes in the environment, and thus are particularly suited to the tuning curve.

Bibliography

1. Legg, S. & Hutter, M. A Collection of Definitions of Intelligence. *arXiv:0706.3639 [cs].* arXiv: 0706.3639 (June 2007).
2. Rodriguez-Molina, V. M., Aertsen, A. & Heck, D. H. Spike timing and reliability in cortical pyramidal neurons: effects of EPSC kinetics, input synchronization and background noise on spike timing. eng. *PloS One* **2,** e319. ISSN: 1932-6203 (Mar. 2007).
3. Mainen, Z. F. & Sejnowski, T. J. Reliability of spike timing in neocortical neurons. en. *Science* **268.** Publisher: American Association for the Advancement of Science Section: Reports, 1503–1506. ISSN: 0036-8075, 1095-9203 (June 1995).
4. Luck, S. J., Chelazzi, L., Hillyard, S. A. & Desimone, R. Neural mechanisms of spatial selective attention in areas V1, V2, and V4 of macaque visual cortex. eng. *Journal of Neurophysiology* **77,** 24–42. ISSN: 0022-3077 (Jan. 1997).
5. Van Vreeswijk, C. & Sompolinsky, H. Chaos in neuronal networks with balanced excitatory and inhibitory activity. eng. *Science (New York, N.Y.)* **274,** 1724–1726. ISSN: 0036-8075 (Dec. 1996).
6. Rabinovich, M. I & Abarbanel, H. D. I. The role of chaos in neural systems. en. *Neuroscience* **87,** 5–14. ISSN: 0306-4522 (June 1998).
7. Kohn, A. & Smith, M. A. Stimulus Dependence of Neuronal Correlation in Primary Visual Cortex of the Macaque. en. *Journal of Neuroscience* **25.** Publisher: Society for Neuroscience Section: Behavioral/Systems/Cognitive, 3661–3673. ISSN: 0270-6474, 1529-2401 (Apr. 2005).
8. Moreno-Bote, R. *et al.* Information-limiting correlations. *Nature neuroscience* **17,** 1410–1417. ISSN: 1097-6256 (Oct. 2014).
9. Fukushima, K. Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. ENG. *Biological Cybernetics* **36,** 193–202. ISSN: 0340-1200 (1980).
10. Rumelhart, D. E., Hinton, G. E. & Williams, R. J. Learning representations by back-propagating errors. en. *Nature* **323.** Number: 6088 Publisher: Nature Publishing Group, 533–536. ISSN: 1476-4687 (Oct. 1986).
11. Riesenhuber, M. & Poggio, T. Hierarchical models of object recognition in cortex. en. *Nature Neuroscience* **2,** 1019–1025. ISSN: 1097-6256 (Nov. 1999).
12. Serre, T., Wolf, L. & Poggio, T. *Object recognition with features inspired by visual cortex* in *2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR'05)* **2** (June 2005), 994–1000 vol. 2.
13. Cadieu, C. *et al.* A model of V4 shape selectivity and invariance. ENG. *Journal of Neurophysiology* **98,** 1733–1750. ISSN: 0022-3077 (Sept. 2007).
14. Anselmi, F. *et al.* Unsupervised learning of invariant representations. en. *Theoretical Computer Science. Biologically Inspired Processes in Neural Computation* **633,** 112–121. ISSN: 0304-3975 (June 2016).

15. Ohzawa, I., Sclar, G. & Freeman, R. D. Contrast gain control in the cat's visual system. en. *Journal of Neurophysiology* **54,** 651–667. ISSN: 0022-3077, 1522-1598 (Sept. 1985).

16. Tenenbaum, J. & Freeman, W. *Separating Style and Content* in *NIPS* (1996).

17. Cohen, M. R. & Kohn, A. Measuring and interpreting neuronal correlations. en. *Nature Neuroscience* **14.** Number: 7 Publisher: Nature Publishing Group, 811–819. ISSN: 1546-1726 (July 2011).

18. Oram, M. W., Földiák, P., Perrett, D. I., Oram, M. W. & Sengpiel, F. The 'Ideal Homunculus': decoding neural population signals. en. *Trends in Neurosciences* **21,** 259–265. ISSN: 0166-2236 (June 1998).

19. Panzeri, S, Schultz, S. R., Treves, A & Rolls, E. T. Correlations and the encoding of information in the nervous system. *Proceedings of the Royal Society B: Biological Sciences* **266,** 1001–1012. ISSN: 0962-8452 (May 1999).

20. Averbeck, B. B., Latham, P. E. & Pouget, A. Neural correlations, population coding and computation. en. *Nature Reviews Neuroscience* **7.** Number: 5 Publisher: Nature Publishing Group, 358–366. ISSN: 1471-0048 (May 2006).

21. Lyamzin, D. R. *et al.* Nonlinear Transfer of Signal and Noise Correlations in Cortical Networks. en. *Journal of Neuroscience* **35.** Publisher: Society for Neuroscience Section: Articles, 8065–8080. ISSN: 0270-6474, 1529-2401 (May 2015).

22. Fuller, W. A. *Measurement Error Models* en. Google-Books-ID: Nalc0DkAJRYC. ISBN: 978-0-470-31733-4 (John Wiley & Sons, Sept. 2009).

23. Spearman, C. The proof and measurement of association between two things. *The American Journal of Psychology* **15.** Place: US Publisher: Univ of Illinois Press, 72–101. ISSN: 1939-8298(Electronic),0002-9556(Print) (1904).

24. Thouless, R. H. The Effects of Errors of Measurement on Correlation Coefficients. English. *British Journal of Psychology. General Section; London, etc.* **29.** Num Pages: 21 Place: London, etc., United Kingdom, London, etc. Publisher: Cambridge University Press, 383–403. ISSN: 0373-2460 (Apr. 1939).

25. Beaton, G. H. *et al.* Sources of variance in 24-hour dietary recall data: implications for nutrition study design and interpretation. en. *The American Journal of Clinical Nutrition* **32.** Publisher: Oxford Academic, 2546–2559. ISSN: 0002-9165 (Dec. 1979).

26. Rosner, B. & Willett, W. C. Interval estimates for correlation coefficients corrected for within-person variation: implications for study design and hypothesis testing. eng. *American Journal of Epidemiology* **127,** 377–386. ISSN: 0002-9262 (Feb. 1988).

27. Adolph, S. C. & Hardin, J. S. Estimating Phenotypic Correlations: Correcting for Bias Due to Intraindividual Variability. *Functional Ecology* **21.** Publisher: [British Ecological Society, Wiley], 178–184. ISSN: 0269-8463 (2007).

28. Schwartz, O., Pillow, J. W., Rust, N. C. & Simoncelli, E. P. Spike-triggered neural characterization. eng. *Journal of Vision* **6,** 484–507. ISSN: 1534-7362 (July 2006).

29. Yamins, D. L. K. *et al.* Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences of the United States of America* **111,** 8619–8624. ISSN: 0027-8424 (June 2014).

30. Khaligh-Razavi, S.-M. & Kriegeskorte, N. Deep supervised, but not unsupervised, models may explain IT cortical representation. eng. *PLoS computational biology* **10,** e1003915. ISSN: 1553-7358 (Nov. 2014).

31. Kriegeskorte, N. Deep Neural Networks: A New Framework for Modeling Biological Vision and Brain Information Processing. *Annual Review of Vision Science* **1,** 417–446 (2015).

32. Pospisil, D. A., Pasupathy, A. & Bair, W. 'Artiphysiology' reveals V4-like shape tuning in a deep network trained for image classification. *eLife* **7** (eds Vaadia, E. & Gold, J. I.) Publisher: eLife Sciences Publications, Ltd, e38242. ISSN: 2050-084X (Dec. 2018).

33. Cadena, S. A. *et al.* Deep convolutional models improve predictions of macaque V1 responses to natural images. en. *PLOS Computational Biology* **15.** Publisher: Public Library of Science, e1006897. ISSN: 1553-7358 (Apr. 2019).

34. Kindel, W. F., Christensen, E. D. & Zylberberg, J. Using deep learning to probe the neural code for images in primary visual cortex. *Journal of Vision* **19.** ISSN: 1534-7362 (Apr. 2019).

35. Atick, J. J. Could information theory provide an ecological theory of sensory processing? eng. *Network (Bristol, England)* **22,** 4–44. ISSN: 1361-6536 (2011).

36. Desimone, R. & Schein, S. J. Visual properties of neurons in area V4 of the macaque: sensitivity to stimulus form. eng. *Journal of Neurophysiology* **57,** 835–868. ISSN: 0022-3077 (Mar. 1987).

37. Bushnell, B. N., Harding, P. J., Kosai, Y., Bair, W. & Pasupathy, A. Equiluminance Cells in Visual Cortical Area V4. *The Journal of Neuroscience* **31,** 12398–12412. ISSN: 0270-6474 (Aug. 2011).

38. Okazawa, G., Tajima, S. & Komatsu, H. Image statistics underlying natural texture selectivity of neurons in macaque V4. *Proceedings of the National Academy of Sciences of the United States of America* **112,** E351–360. ISSN: 1091-6490 (Jan. 2015).

39. Oleskiw, T. D., Nowack, A. & Pasupathy, A. Joint coding of shape and blur in area V4. eng. *Nature Communications* **9,** 466. ISSN: 2041-1723 (2018).

40. Popovkina, D. V., Bair, W. & Pasupathy, A. Modeling diverse responses to filled and outline shapes in macaque V4. eng. *Journal of Neurophysiology* **121,** 1059–1077. ISSN: 1522-1598 (2019).

41. Pasupathy, A. & Connor, C. E. Shape Representation in Area V4: Position-Specific Tuning for Boundary Conformation. *Journal of Neurophysiology* **86,** 2505–2519. ISSN: 0022-3077 (Nov. 2001).

42. Hubel, D. H. & Wiesel, T. N. Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *The Journal of Physiology* **160,** 106–154.2. ISSN: 0022-3751 (Jan. 1962).

43. LeCun, Y., Bengio, Y. & Hinton, G. Deep learning. en. *Nature* **521,** 436–444. ISSN: 0028-0836 (May 2015).

44. Zeiler, M. D. & Fergus, R. Visualizing and Understanding Convolutional Networks. *arXiv:1311.2901 [cs].* arXiv: 1311.2901 (Nov. 2013).

45. Mahendran, A. & Vedaldi, A. Understanding Deep Image Representations by Inverting Them. *arXiv:1412.0035 [cs].* arXiv: 1412.0035 (Nov. 2014).

46. Güçlü, U. & Gerven, M. A. J. v. Deep Neural Networks Reveal a Gradient in the Complexity of Neural Representations across the Ventral Stream. en. *Journal of Neuroscience* **35,** 10005–10014. ISSN: 0270-6474, 1529-2401 (July 2015).
47. Felleman, D. J. & Van Essen, D. C. Distributed hierarchical processing in the primate cerebral cortex. eng. *Cerebral Cortex (New York, N.Y.: 1991)* **1,** 1–47. ISSN: 1047-3211 (1991).
48. Pasupathy, A. & Connor, C. E. Responses to contour features in macaque area V4. eng. *Journal of Neurophysiology* **82,** 2490–2502. ISSN: 0022-3077 (Nov. 1999).
49. Gallant, J. L., Connor, C. E., Rakshit, S., Lewis, J. W. & Van Essen, D. C. Neural responses to polar, hyperbolic, and Cartesian gratings in area V4 of the macaque monkey. eng. *Journal of Neurophysiology* **76,** 2718–2739. ISSN: 0022-3077 (Oct. 1996).
50. Rust, N. C. & DiCarlo, J. J. Selectivity and tolerance ("invariance") both increase as visual information propagates from cortical area V4 to IT. *The Journal of neuroscience : the official journal of the Society for Neuroscience* **30,** 12978–12995. ISSN: 0270-6474 (Sept. 2010).
51. Rust, N. C. & DiCarlo, J. J. Balanced Increases in Selectivity and Tolerance Produce Constant Sparseness along the Ventral Visual Stream. en. *Journal of Neuroscience* **32,** 10170–10182. ISSN: 0270-6474, 1529-2401 (July 2012).
52. Nandy, A. S., Sharpee, T. O., Reynolds, J. H. & Mitchell, J. F. The Fine Structure of Shape Tuning in Area V4. English. *Neuron* **78,** 1102–1115. ISSN: 0896-6273 (June 2013).
53. Sharpee, T. O., Kouh, M. & Reynolds, J. H. Trade-off between curvature tuning and position invariance in visual area V4. en. *Proceedings of the National Academy of Sciences* **110,** 11618–11623. ISSN: 0027-8424, 1091-6490 (July 2013).
54. Pasupathy, A. & Connor, C. E. Population coding of shape in area V4. en. *Nature Neuroscience* **5,** 1332–1338. ISSN: 1546-1726 (Dec. 2002).
55. Murphy, T. M. & Finkel, L. H. Shape representation by a network of V4-like cells. eng. *Neural Networks: The Official Journal of the International Neural Network Society* **20,** 851–867. ISSN: 0893-6080 (Oct. 2007).
56. Kubilius, J., Bracci, S. & Beeck, H. P. O. d. Deep Neural Networks as a Computational Model for Human Shape Sensitivity. en. *PLOS Computational Biology* **12.** Publisher: Public Library of Science, e1004896. ISSN: 1553-7358 (Apr. 2016).
57. Krizhevsky, A., Sutskever, I. & Hinton, G. E. in *Advances in Neural Information Processing Systems 25* (eds Pereira, F., Burges, C. J. C., Bottou, L. & Weinberger, K. Q.) 1097–1105 (Curran Associates, Inc., 2012).
58. Yosinski, J., Clune, J., Nguyen, A., Fuchs, T. & Lipson, H. Understanding Neural Networks Through Deep Visualization. *arXiv:1506.06579 [cs].* arXiv: 1506.06579 (June 2015).
59. Lenc, K. & Vedaldi, A. Understanding image representations by measuring their equivariance and equivalence. *arXiv:1411.5908 [cs].* arXiv: 1411.5908 (Nov. 2014).
60. Donahue, J. *et al.* DeCAF: A Deep Convolutional Activation Feature for Generic Visual Recognition. *arXiv:1310.1531 [cs].* arXiv: 1310.1531 (Oct. 2013).
61. Szegedy, C. *et al.* Intriguing properties of neural networks. *arXiv:1312.6199 [cs].* arXiv: 1312.6199 (Dec. 2013).

62. Bau, D., Zhou, B., Oliva, A. & Torralba, A. Network Dissection: Quantifying Interpretability of Deep Visual Representations. *arXiv:1704.05796 [cs].* arXiv: 1704.05796 (Apr. 2017).

63. Tang, H. *et al.* Recurrent computations for visual pattern completion. *arXiv:1706.02240 [cs, q-bio].* arXiv: 1706.02240 (June 2017).

64. Flachot, A. & Gegenfurtner, K. R. Processing of chromatic information in a deep convolutional neural network. EN. *JOSA A* **35.** Publisher: Optical Society of America, B334–B346. ISSN: 1520-8532 (Apr. 2018).

65. Haefner, R. M. & Cumming, B. G. An improved estimator of Variance Explained in the presence of noise. *Advances in neural information processing systems* **2008,** 585–592. ISSN: 1049-5258 (2008).

66. El-Shamayleh, Y. & Pasupathy, A. Contour Curvature As an Invariant Code for Objects in Visual Area V4. eng. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* **36,** 5532–5543. ISSN: 1529-2401 (May 2016).

67. Goodfellow, I., Lee, H., Le, Q. V., Saxe, A. & Ng, A. Y. in *Advances in Neural Information Processing Systems 22* (eds Bengio, Y., Schuurmans, D., Lafferty, J. D., Williams, C. K. I. & Culotta, A.) 646–654 (Curran Associates, Inc., 2009).

68. David, S. V., Hayden, B. Y. & Gallant, J. L. Spectral receptive field properties explain shape selectivity in area V4. eng. *Journal of Neurophysiology* **96,** 3492–3505. ISSN: 0022-3077 (Dec. 2006).

69. Oleskiw, T. D., Pasupathy, A. & Bair, W. Spectral receptive fields do not explain tuning for boundary curvature in V4. eng. *Journal of Neurophysiology* **112,** 2114–2122. ISSN: 1522-1598 (Nov. 2014).

70. Popovkina, D. V., Bair, W. & Pasupathy, A. Modeling diverse responses to filled and outline shapes in macaque V4. eng. *Journal of Neurophysiology* **121,** 1059–1077. ISSN: 1522-1598 (2019).

71. Ranzato, M., Huang, F. J., Boureau, Y. L. & LeCun, Y. *Unsupervised Learning of Invariant Feature Hierarchies with Applications to Object Recognition* in *2007 IEEE Conference on Computer Vision and Pattern Recognition* (June 2007), 1–8.

72. Fawzi, A. & Frossard, P. Manitest: Are classifiers really invariant? *arXiv:1507.06535 [cs, stat].* arXiv: 1507.06535 (July 2015).

73. Shang, W., Sohn, K., Almeida, D. & Lee, H. Understanding and Improving Convolutional Neural Networks via Concatenated Rectified Linear Units. *arXiv:1603.05201 [cs].* arXiv: 1603.05201 (Mar. 2016).

74. Shen, X., Tian, X., He, A., Sun, S. & Tao, D. *Transform-Invariant Convolutional Neural Networks for Image Classification and Search* in *Proceedings of the 2016 ACM on Multimedia Conference* (ACM, New York, NY, USA, 2016), 1345–1354. ISBN: 978-1-4503-3603-1.

75. Tsai, C.-Y. & Cox, D. D. Measuring and Understanding Sensory Representations within Deep Networks Using a Numerical Optimization Framework. *arXiv:1502.04972 [cs, q-bio].* arXiv: 1502.04972 (Feb. 2015).

76. Rust, N. C. & Movshon, J. A. In praise of artifice. eng. *Nature Neuroscience* **8,** 1647–1650. ISSN: 1097-6256 (Dec. 2005).

77. Adelson, E. H. & Bergen, J. R. in *Computational models of visual processing* 3–20 (The MIT Press, Cambridge, MA, US, 1991). ISBN: 978-0-262-12155-2.

78. Movshon, J. A. & Simoncelli, E. P. Representation of Naturalistic Image Structure in the Primate Visual Cortex. en. *Cold Spring Harbor Symposia on Quantitative Biology* **79.** Publisher: Cold Spring Harbor Laboratory Press, 115–122. ISSN: 0091-7451, 1943-4456 (Jan. 2014).

79. Ziemba, C. M. & Freeman, J. Representing "stuff" in visual cortex. eng. *Proceedings of the National Academy of Sciences of the United States of America* **112,** 942–943. ISSN: 1091-6490 (Jan. 2015).

80. Bushnell, B. N., Harding, P. J., Kosai, Y. & Pasupathy, A. Partial occlusion modulates contour-based shape encoding in primate area V4. eng. *The Journal of Neuroscience: The Official Journal of the Society for Neuroscience* **31,** 4012–4024. ISSN: 1529-2401 (Mar. 2011).

81. Kosai, Y., El-Shamayleh, Y., Fyall, A. M. & Pasupathy, A. The Role of Visual Area V4 in the Discrimination of Partially Occluded Shapes. en. *Journal of Neuroscience* **34.** Publisher: Society for Neuroscience Section: Articles, 8570–8584. ISSN: 0270-6474, 1529-2401 (June 2014).

82. Jia, Y. *et al.* Caffe: Convolutional Architecture for Fast Feature Embedding. *arXiv:1408.5093 [cs].* arXiv: 1408.5093 (June 2014).

83. Deng, J. *et al. ImageNet: A large-scale hierarchical image database* in *2009 IEEE Conference on Computer Vision and Pattern Recognition* ISSN: 1063-6919 (June 2009), 248–255.

84. Mazer, J. A., Vinje, W. E., McDermott, J., Schiller, P. H. & Gallant, J. L. Spatial frequency and orientation tuning dynamics in area V1. en. *Proceedings of the National Academy of Sciences* **99,** 1645–1650. ISSN: 0027-8424, 1091-6490 (Feb. 2002).

85. Hinkle, D. A. & Connor, C. E. Three-dimensional orientation tuning in macaque area V4. eng. *Nature Neuroscience* **5,** 665–670. ISSN: 1097-6256 (July 2002).

86. Roddey, J. C., Girish, B. & Miller, J. P. Assessing the Performance of Neural Encoding Models in the Presence of Noise. en. *Journal of Computational Neuroscience* **8,** 95–112. ISSN: 1573-6873 (Mar. 2000).

87. Sahani, M. & Linden, J. F. in *Advances in Neural Information Processing Systems 15* (eds Becker, S., Thrun, S. & Obermayer, K.) 125–132 (MIT Press, 2003).

88. Hsu, A., Borst, A. & Theunissen, F. E. Quantifying variability in neural responses and its application for the validation of model predictions. *Network: Computation in Neural Systems* **15,** 91–109. ISSN: 0954-898X (Jan. 2004).

89. David, S. V. & Gallant, J. L. Predicting neuronal responses during natural vision. eng. *Network (Bristol, England)* **16,** 239–260. ISSN: 0954-898X (Sept. 2005).

90. Schoppe, O., Harper, N. S., Willmore, B. D. B., King, A. J. & Schnupp, J. W. H. Measuring the Performance of Neural Models. *Frontiers in Computational Neuroscience* **10.** ISSN: 1662-5188 (Feb. 2016).

91. Efron, B. & Tibshirani, R. J. *An Introduction to the Bootstrap* en. Google-Books-ID: MWC1DwAAQBAJ. ISBN: 978-1-00-006498-8 (CRC Press, May 1994).

92. Zohary, E., Shadlen, M. N. & Newsome, W. T. Correlated neuronal discharge rate and its implications for psychophysical performance. en. *Nature* **370.** Number: 6485 Publisher: Nature Publishing Group, 140–143. ISSN: 1476-4687 (July 1994).

93. Bair, W., Zohary, E. & Newsome, W. T. Correlated Firing in Macaque Visual Area MT: Time Scales and Relationship to Behavior. en. *Journal of Neuroscience* **21,** 1676–1697. ISSN: 0270-6474, 1529-2401 (Mar. 2001).

94. Olshausen, B. A. & Field, D. J. How close are we to understanding v1? eng. *Neural Computation* **17,** 1665–1699. ISSN: 0899-7667 (Aug. 2005).

95. De Vries, S. E. J. *et al.* A large-scale standardized physiological survey reveals functional organization of the mouse visual cortex. en. *Nature Neuroscience* **23.** Number: 1 Publisher: Nature Publishing Group, 138–151. ISSN: 1546-1726 (Jan. 2020).

96. Siegle, J. H. *et al.* A survey of spiking activity reveals a functional hierarchy of mouse corticothalamic visual areas. en. *bioRxiv.* Publisher: Cold Spring Harbor Laboratory Section: New Results, 805010 (Oct. 2019).

97. Leavitt, M. L., Pieper, F., Sachs, A. J. & Martinez-Trujillo, J. C. Correlated variability modifies working memory fidelity in primate prefrontal neuronal ensembles. *Proceedings of the National Academy of Sciences of the United States of America* **114,** E2494–E2503. ISSN: 0027-8424 (Mar. 2017).

98. Insanally, M. N. *et al.* Nominally non-responsive frontal and sensory cortical cells encode task-relevant variables via ensemble consensus-building. en. *bioRxiv.* Publisher: Cold Spring Harbor Laboratory Section: New Results, 347617 (June 2018).

99. Zylberberg, J. The role of untuned neurons in sensory information coding. en. *bioRxiv.* Publisher: Cold Spring Harbor Laboratory Section: New Results, 134379 (May 2018).

100. Britten, K. H., Shadlen, M. N., Newsome, W. T. & Movshon, J. A. Responses of neurons in macaque MT to stochastic motion signals. eng. *Visual Neuroscience* **10,** 1157–1169. ISSN: 0952-5238 (Dec. 1993).

101. Kiani, R. *et al.* Natural grouping of neural responses reveals spatially segregated clusters in prearcuate cortex. eng. *Neuron* **85,** 1359–1373. ISSN: 1097-4199 (Mar. 2015).

102. Power, J. D. *et al.* Functional network organization of the human brain. *Neuron* **72,** 665–678. ISSN: 0896-6273 (Nov. 2011).

103. Pospisil, D. A. & Bair, W. The unbiased estimation of the fraction of variance explained by a model. en. *bioRxiv.* Publisher: Cold Spring Harbor Laboratory Section: New Results, 2020.10.30.361253 (Nov. 2020).

104. El-Shamayleh, Y. & Pasupathy, A. Contour Curvature As an Invariant Code for Objects in Visual Area V4. en. *Journal of Neuroscience* **36,** 5532–5543. ISSN: 0270-6474, 1529-2401 (May 2016).

105. Gawne, T. J. & Richmond, B. J. How independent are the messages carried by adjacent inferior temporal cortical neurons? en. *Journal of Neuroscience* **13.** Publisher: Society for Neuroscience Section: Articles, 2758–2771. ISSN: 0270-6474, 1529-2401 (July 1993).

106. Lee, D., Port, N. L., Kruse, W. & Georgopoulos, A. P. Variability and Correlated Noise in the Discharge of Neurons in Motor and Parietal Areas of the Primate Cortex. en. *Journal of Neuroscience* **18.** Publisher: Society for Neuroscience Section: ARTICLE, 1161–1170. ISSN: 0270-6474, 1529-2401 (Feb. 1998).

107. Averbeck, B. B. & Lee, D. Neural Noise and Movement-Related Codes in the Macaque Supplementary Motor Area. *The Journal of Neuroscience* **23,** 7630–7641. ISSN: 0270-6474 (Aug. 2003).

108. Cohen, M. R. & Maunsell, J. H. R. Attention improves performance primarily by reducing interneuronal correlations. en. *Nature Neuroscience* **12.** Number: 12 Publisher: Nature Publishing Group, 1594–1600. ISSN: 1546-1726 (Dec. 2009).

109. Ecker, A. S. *et al.* State Dependence of Noise Correlations in Macaque Primary Visual Cortex. English. *Neuron* **82.** Publisher: Elsevier, 235–248. ISSN: 0896-6273 (Apr. 2014).

110. Gawne, T. J., Kjaer, T. W., Hertz, J. A. & Richmond, B. J. Adjacent Visual Cortical Complex Cells Share About 20% of Their Stimulus-Related Information. en. *Cerebral Cortex* **6.** Publisher: Oxford Academic, 482–489. ISSN: 1047-3211 (May 1996).

111. Vinje, W. E. & Gallant, J. L. Sparse Coding and Decorrelation in Primary Visual Cortex During Natural Vision. en. *Science* **287.** Publisher: American Association for the Advancement of Science Section: Report, 1273–1276. ISSN: 0036-8075, 1095-9203 (Feb. 2000).

112. Rothschild, G., Nelken, I. & Mizrahi, A. Functional organization and population dynamics in the mouse primary auditory cortex. en. *Nature Neuroscience* **13.** Number: 3 Publisher: Nature Publishing Group, 353–360. ISSN: 1546-1726 (Mar. 2010).

113. Zeki, S. M. Functional organization of a visual area in the posterior bank of the superior temporal sulcus of the rhesus monkey. *The Journal of Physiology* **236,** 549–573. ISSN: 0022-3751 (Feb. 1974).

114. Maunsell, J. H. & Van Essen, D. C. Functional properties of neurons in middle temporal visual area of the macaque monkey. I. Selectivity for stimulus direction, speed, and orientation. eng. *Journal of Neurophysiology* **49,** 1127–1147. ISSN: 0022-3077 (May 1983).

115. Albright, T. D., Desimone, R. & Gross, C. G. Columnar organization of directionally selective cells in visual area MT of the macaque. eng. *Journal of Neurophysiology* **51,** 16–31. ISSN: 0022-3077 (Jan. 1984).

116. Born, R. T. & Bradley, D. C. Structure and function of visual area MT. eng. *Annual Review of Neuroscience* **28,** 157–189. ISSN: 0147-006X (2005).

117. Kohn, A. & Smith, M. *Utah array extracellular recordings of spontaneous and visually evoked activity from anesthetized macaque primary visual cortex (V1).* en. Artwork Size: 180 MB Medium: application/matlab Pages: 180 MB type: dataset. 2016.

118. Saccenti, E., Hendriks, M. H. W. B. & Smilde, A. K. Corruption of the Pearson correlation coefficient by measurement error and its estimation, bias, and correction under different error models. en. *Scientific Reports* **10.** Number: 1 Publisher: Nature Publishing Group, 438. ISSN: 2045-2322 (Jan. 2020).

119. Angelucci, A. *et al.* Circuits and Mechanisms for Surround Modulation in Visual Cortex. eng. *Annual Review of Neuroscience* **40,** 425–451. ISSN: 1545-4126 (2017).

120. Solarana, K., Liu, J., Bowen, Z., Lee, H.-K. & Kanold, P. O. Temporary Visual Deprivation Causes Decorrelation of Spatiotemporal Population Responses in Adult Mouse Auditory Cortex. en. *eNeuro* **6.** Publisher: Society for Neuroscience Section: New Research. ISSN: 2373-2822 (Nov. 2019).

121. Ecker, A. S. *et al.* Decorrelated Neuronal Firing in Cortical Microcircuits. en. *Science* **327.** Publisher: American Association for the Advancement of Science Section: Report, 584–587. ISSN: 0036-8075, 1095-9203 (Jan. 2010).

122. Martin, K. A. C. & Schröder, S. Functional Heterogeneity in Neighboring Neurons of Cat Primary Visual Cortex in Response to Both Artificial and Natural Stimuli. en. *Journal of Neuroscience* **33.** Publisher: Society for Neuroscience Section: Articles, 7325–7344. ISSN: 0270-6474, 1529-2401 (Apr. 2013).

123. Smith, M. A. & Sommer, M. A. Spatial and Temporal Scales of Neuronal Correlation in Visual Area V4. en. *Journal of Neuroscience* **33.** Publisher: Society for Neuroscience Section: Articles, 5422–5432. ISSN: 0270-6474, 1529-2401 (Mar. 2013).

124. Lin, Y.-C. & Koleske, A. J. Mechanisms of Synapse and Dendrite Maintenance and Their Disruption in Psychiatric and Neurodegenerative Disorders. *Annual review of neuroscience* **33,** 349–378. ISSN: 0147-006X (2010).

125. Rokni, U., Richardson, A. G., Bizzi, E. & Seung, H. S. Motor Learning with Unstable Neural Representations. en. *Neuron* **54,** 653–666. ISSN: 0896-6273 (May 2007).

126. Duffy, A., Abe, E., Perkel, D. J. & Fairhall, A. L. Variation in sequence dynamics improves maintenance of stereotyped behavior in an example from bird song. en. *Proceedings of the National Academy of Sciences* **116.** Publisher: National Academy of Sciences Section: Biological Sciences, 9592–9597. ISSN: 0027-8424, 1091-6490 (May 2019).

127. Shrikumar, A., Greenside, P. & Kundaje, A. Learning Important Features Through Propagating Activation Differences. *arXiv:1704.02685 [cs].* arXiv: 1704.02685 (Oct. 2019).

128. Bach, S. *et al.* On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. en. *PLOS ONE* **10.** Publisher: Public Library of Science, e0130140. ISSN: 1932-6203 (July 2015).

129. Ribeiro, M. T., Singh, S. & Guestrin, C. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. *arXiv:1602.04938 [cs, stat].* arXiv: 1602.04938 (Aug. 2016).

130. Lundberg, S. & Lee, S.-I. An unexpected unity among methods for interpreting model predictions. *arXiv:1611.07478 [cs].* arXiv: 1611.07478 (Dec. 2016).

131. Sundararajan, M., Taly, A. & Yan, Q. *Axiomatic attribution for deep networks* in *Proceedings of the 34th International Conference on Machine Learning - Volume 70* (JMLR.org, Sydney, NSW, Australia, Aug. 2017), 3319–3328.

132. Paszke, A. *et al.* PyTorch: An Imperative Style, High-Performance Deep Learning Library. en. *Advances in Neural Information Processing Systems* **32,** 8026–8037 (2019).

133. Springenberg, J. T., Dosovitskiy, A., Brox, T. & Riedmiller, M. Striving for Simplicity: The All Convolutional Net. *arXiv:1412.6806 [cs].* arXiv: 1412.6806 (Apr. 2015).

134. Ozbulak, U. *PyTorch CNN Visualizations* Publication Title: GitHub repository (GitHub, 2019).

135. De la Rocha, J., Doiron, B., Shea-Brown, E., Josić, K. & Reyes, A. Correlation between neural spike trains increases with firing rate. en. *Nature* **448,** 802–806. ISSN: 1476-4687 (Aug. 2007).

136. Bedenbaugh, P. & Gerstein, G. L. Rectification of correlation by a sigmoid nonlinearity. en. *Biological Cybernetics* **70,** 219–225. ISSN: 1432-0770 (Jan. 1994).

137. Wainwright, M. J., Schwartz, O. & Simoncelli, E. Natural image statistics and divisive normalization: Modeling nonlinearity and adaptation in cortical neurons. English (US). *Probabilistic models of the brain: Perception and neural function.* Publisher: MIT Press, 203–222 (2002).