UNIVERSITY OF CALIFORNIA, SAN DIEGO

Decoding the computations of sensory neurons

A dissertation submitted in partial satisfaction of the
requirements for the degree
Doctor of Philosophy

in

Physics

by

Joel Thomas Kaardal

Committee in charge:

Professor Henry D. I. Abarbanel, Co-Chair
Professor Tatyana O. Sharpee, Co-Chair
Professor Timothy Q. Gentner
Professor Julius Kuti
Professor Terrence J. Sejnowski

2017

The dissertation of Joel Thomas Kaardal is approved, and it is acceptable in quality and form for publication on microfilm and electronically:

_____

_____

_____

_____
Co-Chair

_____
Co-Chair

University of California, San Diego

2017

TABLE OF CONTENTS

Signature Page iii

Table of Contents iv

List of Figures vii

Acknowledgements ix

Vita x

Abstract of the Dissertation xi

Chapter 1 Introduction and survey of neural coding methods 1
1.1 Linear-nonlinear model 2
1.2 Dimensionality reduction 3
1.3 Spike-triggered methods 3
1.3.1 Spike-triggered average 3
1.3.2 Spike-triggered covariance 4
1.3.3 Statistical limitations 5
1.4 Maximally informative dimensions 5
1.4.1 The curse of dimensionality 6
1.5 Maximum noise entropy 7
1.5.1 Overfitting 8
1.6 Statement of the problem 9

Chapter 2 Identifying functional bases of neural computations 10
2.1 Functional bases 11
2.2 Modeling the functional basis with Boolean operations . . . 14
2.2.1 Optimization procedure 15
2.2.2 Taking advantage of dimensionality reduction 16
2.3 Demonstration on synthetic neurons 17
2.4 Functional neural circuitry of retinal ganglion cells 24
2.5 Discussion and extensions 35

Chapter 3 Low-rank minimal models for multicomponent computations . . . 39
3.1 The low-rank MNE model 42
3.1.1 Constraint formulations 43
3.1.2 Nuclear-norm regularization 46
3.2 The optimization problem 52
3.2.1 Optimality conditions 53
3.2.2 Locally and globally optimal regularization domains 57
3.3 Optimizing the model weights 63

3.4 Hyperparameter optimization problems 66
3.4.1 Regularization parameters 66
3.4.2 Rank parameters 68

Chapter 4 Low-rank minimal models in practice: applications to the auditory system 71
4.1 Optimization algorithms 71
4.1.1 Grid search 74
4.1.2 Bayesian optimization 77
4.2 Validation on synthetic neurons 81
4.2.1 "Auditory" neuron 81
4.2.2 "Vision" neurons 87
4.3 Neural coding in the auditory system 96
4.3.1 The dataset and preprocessing procedure 97
4.3.2 Multicomponent receptive fields of high-level auditory neurons 99
4.3.3 Functional neural circuitry of high-level auditory neurons 103
4.4 Discussion 108

Appendix A Bayesian interpretation of the nuclear-norm 112

Appendix B Convergence of the block coordinate descent algorithm 114

Appendix C Resolving the optimal rank of second-order MNE models 122

Bibliography 125

LIST OF ABBREVIATIONS

LN	linear-nonlinear
STA	spike-triggered average
STC	spike-triggered covariance
MID	maximally informative dimensions
MNE	maximum noise entropy
RGC	retinal ganglion cell
CM	caudal mesopallium

LIST OF FIGURES

Figure 2.1: Illustration of two neurons computing Boolean operations in a two-dimensional stimulus space 13
Figure 2.2: The functional basis method applied to a synthetic translationally invariant vision neuron 20
Figure 2.3: The functional basis method applied to a synthetic rotationally invariant neuron 22
Figure 2.4: The functional input components recovered with undercomplete and overcomplete functional bases 23
Figure 2.5: The functional basis method applied to a synthetic retinal ganglion cell (RGC) 24
Figure 2.6: Example showing the spatial and temporal separability of an RGC receptive field 26
Figure 2.7: Receptive fields and functional bases from two RGCs 28
Figure 2.8: Comparing the prediction error of empirical nonlinearities across the population of RGCs 31
Figure 2.9: Comparison of the prediction error between Boolean functional bases and empirical nonlinearities and the distribution of functional basis set sizes across RGCs 32
Figure 2.10: Empirical nonlinearities of an example RGC projected into the receptive field and functional bases 34
Figure 2.11: Illustration of quadratic functional inputs 37

Figure 3.1: It is possible for suboptimal local minima to exist in the low-rank MNE problem 58

Figure 4.1: Relaxation from a globally optimal approximation to a locally optimal solution does not guarantee global optimality 73
Figure 4.2: Constructing the receptive field of a synthetic auditory neuron . . . 82
Figure 4.3: The receptive fields of the synthetic auditory neuron recovered using STC, full-rank MNE, and low-rank MNE methods 84
Figure 4.4: Diagram showing how the data is divided into training, cross-validation, and test sets 85
Figure 4.5: A globally optimal compression and expansion of the synthetic auditory neuron's receptive field using the low-rank MNE method . 87
Figure 4.6: Constructing the synthetic vision neurons' receptive field 88
Figure 4.7: Receptive fields recovered using STC, full-rank MNE, and the empirical model selection approach to the low-rank MNE method for two synthetic vision neurons 89
Figure 4.8: Selecting the best model among STC, full-rank MNE, and the empirically selected low-rank MNE models for two synthetic vision neurons 92

Figure 4.9: Globally optimal approximations to the receptive field of two synthetic vision neurons 93
Figure 4.10: Model selection between globally optimal approximations of different rank is unreliable 94
Figure 4.11: Eigenvalue spectra of the globally optimal approximations to the receptive fields of the two synthetic vision neurons 94
Figure 4.12: Receptive field components of two synthetic vision neurons recovered using Bayesian optimization 96
Figure 4.13: Multicomponent receptive fields recovered from two field L neurons 102
Figure 4.14: Selecting the best model among STC and first-order, full-rank, and low-rank MNE methods across the population of auditory neurons . 104
Figure 4.15: Functional bases of two field L neurons 107
Figure 4.16: The logical AND functional basis is the dominant description of the auditory neurons and is correlated with an imbalance of excitation and suppression 108

ACKNOWLEDGEMENTS

I would like to thank Dr. Tatyana Sharpee and the members of the CNL-T research group at the Salk Institute for Biological Studies for their guidance, insight, and feedback over the course of my journey through graduate school. I wish them all the best in their future endeavors. I would also like to thank all of those with whom I have collaborated, including those with which I have coauthored publications.

Chapter 2 contains material that was published in Kaardal, Fitzgerald, Berry, and Sharpee, *Neural Computation* (2013). The dissertation author was the primary investigator and author of this paper.

Chapters 3 and 4 and appendices B and C contain work that was published in Kaardal, Theunissen, and Sharpee, *Frontiers in Computational Neuroscience* (2017). The dissertation author was the primary investigator and author of the paper.

VITA

2010	B. S. in Chemistry *with distinction*, University of Minnesota, Minneapolis
2010	B. S. in Physics *with distinction*, University of Minnesota, Minneapolis
2013	M. S. in Physics, University of California, San Diego
2017	Ph. D. in Physics, University of California, San Diego

PUBLICATIONS

Kaardal, J.; Fitzgerald, J.D.; Berry, M.J.; Sharpee, T.O.; "Identifying functional bases for multidimensional neural computations", *Neural Computation*, 25(7), 1870-1890, 2013.

Kaardal, J.T.; Theunissen, F.E.; Sharpee, T.O.; "A low-rank method for characterizing high-level neural computations", *Frontiers in Computational Neuroscience*, 11, 68, 2017.

ABSTRACT OF THE DISSERTATION

Decoding the computations of sensory neurons

by

Joel Thomas Kaardal

Doctor of Philosophy in Physics

University of California, San Diego, 2017

Professor Henry D. I. Abarbanel, Co-Chair
Professor Tatyana O. Sharpee, Co-Chair

The nervous system encodes information about external stimuli through sophisticated computations performed by vast networks of sensory neurons. Since the space of all possible stimuli is much larger than the space of those that are ultimately meaningful, dimensionality reduction techniques were developed to identify the subspace of stimulus space relevant to neural activity. However, dimensionality reduction methods provide limited insight into the nonlinear functions that build the nervous system's internal model of the world. In Chapter 2, the *functional basis* is introduced that transforms the relevant subspace to a basis that describes the computational function of the subunits that make

up the neural circuitry. This functional basis is used to uncover novel insights about the computations performed by neurons in low-level vision and, later on, high-level auditory circuitry. For the latter, significant barriers are found in the capability of current dimensionality reduction methods to recover the relevant subspaces of high-level sensory neurons. This barrier is caused by the relative difficulty of stimulating high-level sensory neurons, which are often unresponsive to noise stimuli, while still maintaining a thorough exploration of the stimulus distribution. In response, a new approach to dimensionality reduction is formulated in Chapter 3 called the *low-rank maximum noise entropy method* that makes it possible to overcome challenges presented by high-level sensory systems. In Chapter 4, functional bases derived from the relevant subspaces recovered by the low-rank maximum noise entropy method are employed to study the neural computations performed by high-level auditory neurons.

Chapter 1

Introduction and survey of neural coding methods

Through the collective action of vast, hierarchical networks of neurons, the brain is capable of representing abstract concepts and making sophisticated decisions. Understanding the neural computations that lead to such complex phenomena is an important area of research that can be considered on multiple scales from the peripheral nervous system, to neural circuits, down to the fundamental computational unit: the neuron. This dissertation focuses on methods for decoding neural computations on the scale of single sensory neurons.

Sensory neurons are excitable cells that have roughly binary "on" and "off" states that are stimulated by external sensory signals, analogous in some ways to digital logic gates. A fundamental description of those stimuli that cause a neuron to reach its "on" state, responding with an action potential or spike (so-called due to its transient nature), is termed the *receptive field*. Put simply, the receptive field can be defined as the set of all stimuli that modulate the neural response. In reality, however, the sets of stimuli that do and do not modulate a neuron's response are not generally disjoint sets. Rather it is

possible on repeated presentations of the same stimulus that a neuron will sometimes produce a spike and sometimes not. If one were to take the average of these responses, one could compute the probability of a spike given that stimulus. Done over all possible stimuli, one can compute a mapping between a stimulus and the probability that a neuron will spike. Approximating aspects of this input-output mapping is the concern of the field called *neural coding*. By approximating the extent of the receptive field and mapping the input-output function of each neuron in a circuit, one can determine what role each of the neurons plays in the phenomena that emerge from the neural circuitry. In the following sections, a survey of methods used to decode the receptive fields of neurons from stimulus and response pairs is provided.

1.1 Linear-nonlinear model

The linear-nonlinear (LN) model [1–5] is a general computational framework that can be used to approximate the receptive field and input-output function. The model is composed of the two namesake stages (i) a *linear* projection of a stimulus vector, $\mathbf{s} \in \mathbb{R}^D$, onto a set of vectors, $\{\boldsymbol{\omega}_k\} \equiv \{\boldsymbol{\omega}_k \in \mathbb{R}^D | \forall k \in \{1, \cdots, r\}\}$ where r is the number of vectors in the set, and (ii) a *nonlinear* stage that approximates the input-output mapping as a function of the linear projections, $g(\boldsymbol{\Omega}^{\mathrm{T}}\mathbf{s}) = P(y = 1|\mathbf{s})$, where $y \in \{0, 1\}$ corresponds to the neural response state and $\boldsymbol{\Omega} = [\boldsymbol{\omega}_1, \boldsymbol{\omega}_2, \cdots, \boldsymbol{\omega}_3] \in \mathbb{R}^{D \times r}$. Note that $\boldsymbol{\Omega}$ is invariant to rotations that preserve the range space, $\boldsymbol{\mathcal{R}}(\boldsymbol{\Omega})$. That is, when $\boldsymbol{\Omega}' = \mathbf{R}_\theta \boldsymbol{\Omega}$ where $\mathbf{R}_\theta \in \mathbb{R}^{r \times D}$ such that $\boldsymbol{\mathcal{R}}(\boldsymbol{\Omega}') = \boldsymbol{\mathcal{R}}(\boldsymbol{\Omega})$, both $\boldsymbol{\Omega}$ and $\boldsymbol{\Omega}'$ are equivalent linear models. Since the input-output function (at least as it is defined here) is the probability of a spike, defined as state $y = 1$, given some stimulus, the input-output function is inherently nonlinear and is also referred to as the *nonlinearity*. In this model, the $\{\boldsymbol{\omega}_k\}$ vectors approximately span the receptive field and the nonlinearity is a function of the products

$\{\omega_k \cdot \mathbf{s}\}$.

There are several different methods available for estimating $\{\omega_k\}$ and the nonlinearity. The most common approaches are the following spike-triggered methods and information-theoretic methods.

1.2 Dimensionality reduction

Ultimately, since a neuron is selective for particular objects or patterns in the environment, the receptive field is expected to be spanned by a lower number of components than those that could describe the environment in its entirety. In other words, $r \ll D$ provided the stimulus space is comprehensive. The act of finding a lower-dimensional space that describes the activity of the neuron is called *dimensionality reduction*. If r is the number of components that significantly contribute to the neural response, then the receptive field can be estimated by the stimuli projected into the subspace of stimulus space defined by the r significant components of $\mathbf{\Omega}$.

1.3 Spike-triggered methods

Spike-triggered methods use spike-weighted moments of the stimulus distribution to compute a nonlinearity-independent estimate of the receptive field. These methods are widely used because they are quick to compute and work well under certain assumptions about the stimulus distribution.

1.3.1 Spike-triggered average

The spike-triggered average (STA) [2,4] is among the simplest methods for reconstructing the receptive field. The STA performs an average of the stimulus distribution

weighted by the measured response distribution,

$$\boldsymbol{\omega}^{(\mathrm{STA})} = \frac{1}{N_{\mathrm{spk}}} \sum_{t=1}^{N_{\mathrm{samp}}} y_t \mathbf{s}_t \tag{1.1}$$

where $\mathbf{s}_t$ and y_t are the tth sample from the stimulus-response distribution, N_{samp} is the total number of samples, and N_{spk} is the total number of spikes ($N_{\mathrm{spk}} = \sum_{t=1}^{N_{\mathrm{samp}}} y_t$).

1.3.2 Spike-triggered covariance

One of the downsides of the STA is that it can only reconstruct a single component of the receptive field. It has been shown that many neurons are selective for more than one component; the so-called *multidimensional receptive field* or *multicomponent receptive field* [6–19]. The discovery of multicomponent receptive fields has led to the development of quadratic methods where the feature space is expanded to include pairwise products, $s_i s_j$, of the stimulus vector [4, 20–25]. An example of a quadratic method is the spike-triggered covariance (STC) which is an extension of the STA from a spike-weighted first-order moment of the stimulus distribution to a spike-weighted second-order moment of the stimulus distribution. The STC is calculated by first forming the spike-triggered stimulus covariance matrix,

$$\mathbf{C}^{(\mathrm{spk})} = \frac{1}{N_{\mathrm{spk}}} \sum_{t=1}^{N_{\mathrm{samp}}} y_t \mathbf{s}_t \mathbf{s}_t^{\mathrm{T}} \tag{1.2}$$

and the stimulus covariance matrix,

$$\mathbf{C}^{(\mathrm{stim})} = \frac{1}{N_{\mathrm{samp}}} \sum_{t=1}^{N_{\mathrm{samp}}} \mathbf{s}_t \mathbf{s}_t^{\mathrm{T}}. \tag{1.3}$$

Then, the difference matrix, $\mathbf{C}^{(\mathrm{diff})} = \mathbf{C}^{(\mathrm{spk})} - \mathbf{C}^{(\mathrm{stim})}$, is computed. The estimated vectors $\boldsymbol{\Omega}$ are then exposed via eigendecomposition of $\mathbf{C}^{(\mathrm{diff})}$; i.e. $\mathbf{C}^{(\mathrm{diff})} = \boldsymbol{\Omega}\boldsymbol{\Lambda}\boldsymbol{\Omega}^{\mathrm{T}}$ where $\boldsymbol{\Omega}$

are the eigenvectors and $\mathbf{\Lambda}$ are the eigenvalues. A variation of STC subtracts the STA, $\boldsymbol{\omega}^{(\mathrm{STA})}$, from each stimulus vector in $\mathbf{C}^{(\mathrm{spk})}$ and then later includes $\boldsymbol{\omega}^{(\mathrm{STA})}$ in $\mathbf{\Omega}$ [4]. The form in Eq 1.2 without this subtraction will be used throughout this volume.

1.3.3 Statistical limitations

The main limitation of spike-triggered methods is that the estimate of the receptive field is biased unless the stimuli are drawn from a limited number of distributions. For the STA, an unbiased estimate of the receptive field can be obtained if the stimulus distribution is zero-centered and spherically symmetric [4]. For STC, the stimuli must be drawn from an uncorrelated, zero-centered Gaussian distribution [4]. This limitation can be a major hindrance for receptive field recovery, especially in the case of high-level sensory neurons that are not easily stimulated by noise. This limitation led to the development of the following information-theoretic methods.

1.4 Maximally informative dimensions

Maximally informative dimensions (MID) [26] is a nonlinearity-independent dimensionality reduction technique that can be applied to non-Gaussian distributed stimulus distributions without the biases intrinsic to the aforementioned spike-triggered methods. MID searches for the set of vectors $\{\boldsymbol{\omega}_k\}$ that maximize the mutual information per spike,

$$I\left(y=1;\{\boldsymbol{\omega}_k\}\right)=\int \mathrm{d}\mathbf{x}\,P(\mathbf{x}|y=1)\log\left[\frac{P(\mathbf{x}|y=1)}{P(\mathbf{x})}\right] \tag{1.4}$$

where $\mathbf{x} = \mathbf{\Omega}^{\mathrm{T}}\mathbf{s}$. This is done by making empirical estimates of the posterior distribution,

$$P(\mathbf{x}|y=1) \approx \left\langle y \prod_{k=1}^{r} \delta(\mathbf{x} - \boldsymbol{\omega}_k \cdot \mathbf{s}) \right\rangle_{y,\mathbf{s}}, \tag{1.5}$$

and prior distribution,

$$P(\mathbf{x}) \approx \left\langle \prod_{k=1}^{r} \delta(\mathbf{x} - \boldsymbol{\omega}_k \cdot \mathbf{s}) \right\rangle_{\mathbf{s}} \tag{1.6}$$

where $\delta(\cdot)$ is the Dirac delta function. This is typically achieved by binning the responses in $\mathbf{x}$ space. The mutual information is maximized using a global optimization heuristic since the mutual information is nonconcave [26].

1.4.1 The curse of dimensionality

The binning procedure used to compute the posterior and prior distributions require that the data samples be divided among n_{bins} bins along r components leading to a total of $r^{n_{\text{bins}}}$ total bins among which a given sample uniquely belongs to only one. This means that the average number of samples per bin is $N_{\text{samp}}/r^{n_{\text{bins}}}$ which gets exponentially smaller as r increases eventually leading to poor sampling of the distributions if N_{samp} is not sufficiently large. This is known as the curse of dimensionality where typically only up to three or four components can be reliably optimized. However, second-order extensions of MID have been proposed as possible solutions to avoid the curse of dimensionality where the subspace projection is changed to $x = \mathbf{h}^{\mathrm{T}}\mathbf{s} + \mathbf{s}^{\mathrm{T}}\mathbf{J}\mathbf{s}$ and the optimization is performed over $\mathbf{h}$ and $\mathbf{J}$ instead [27, 28]. The receptive field is then recovered by diagonalizing $\mathbf{J}$.

1.5 Maximum noise entropy

Maximum noise entropy (MNE) [28, 29] is one of the more recently developed dimensionality reduction methods that is meant to be an answer to both the issues spike-triggered methods have with correlated stimuli and the curse of dimensionality experienced by MID. Along with MID, MNE is an information-theoretic approach to dimensionality reduction. However, MNE minimizes rather than maximizes the mutual information between the response and the stimuli in an effort to limit bias from arbitrary stimulus distributions. The mutual information written in terms of the response entropy, $H_{\text{resp}} = -\int \mathrm{d}y\, P(y) \log\big(P(y)\big)$, and noise entropy, $H_{\text{noise}} = -\int \mathrm{d}y\, \mathrm{d}\mathbf{s}\, P(y|\mathbf{s}) P(\mathbf{s}) \log\big(P(y|\mathbf{s})\big)$, is $I(y;\mathbf{s}) = H_{\text{resp}} - H_{\text{noise}}$ [28–31]. Since $P(y)$ is already known in virtue of being simply calculated by averaging the response, H_{resp} is fixed and $I(y;\mathbf{s})$ is maximized when $I(y;\mathbf{s}) = H_{\text{resp}}$. Therefore, minimizing the mutual information is equivalent to maximizing the noise entropy with respect to the conditional probability $P(y|\mathbf{s})$.

Of course, $P(y|\mathbf{s})$ is ambiguous in the absence of further constraints being placed on the noise entropy maximization. In an attempt to introduce as little bias as possible into the model, only empirical constraints derived from the data and normalization are imposed. For instance, the statistics of the model can be constrained to match the response-weighted moments of the stimulus distribution $\langle y \rangle_y$, $\langle y\mathbf{s} \rangle_{y,\mathbf{s}}$, and $\langle y\mathbf{s}\mathbf{s}^{\mathrm{T}} \rangle_{y,\mathbf{s}}$, etc. Furthermore, since the responses can be assumed to be binary, the probability of of being in states $P(y=1|\mathbf{s})$ and $P(y=0|\mathbf{s})$ must add up to one. Under these constraints, the maximum noise entropy nonlinearity has the analytic form of a logistic function,

$$P(y=1|\mathbf{s}) = \frac{1}{1+e^{-z(\mathbf{s})}}, \tag{1.7}$$

where $z(\mathbf{s}) = a + \mathbf{h}^{\mathrm{T}}\mathbf{s} + \mathbf{s}^{\mathrm{T}}\mathbf{J}\mathbf{s} + \cdots$ and a, $\mathbf{h}$, $\mathbf{J}$, etc. are the Lagrange multipliers from the constrained noise entropy maximization, which shall be referred to as the "weights"

from here on, that must be optimized to finish computing the entropy maximization. This makes MNE a nonlinearity-dependent method. Despite this, the nonlinearity adheres to the principle of maximum entropy and is designed to be minimally biased in an information-theoretic sense towards unseen data.

The unknown weights in the nonlinearity are found by minimizing the negative log-likelihood,

$$L(a,\mathbf{h},\mathbf{J}) = -\frac{1}{N_{\text{samp}}}\sum_t \left[y_t \log\left(P(y=1|\mathbf{s}_t)\right) + (1-y_t)\log\left(1-P(y=1|\mathbf{s}_t)\right)\right], \quad (1.8)$$

where it is defined here as an average over samples and the model is truncated to second-order. Conveniently, this optimization happens to be convex and is therefore guaranteed to converge to a global minimum in polynomial time.

1.5.1 Overfitting

Since the argument of the nonlinearity, $z(\mathbf{s})$, is a polynomial of arbitrary order, the number of terms that appear must be truncated because each additional order m of weights included in the polynomial add an additional D^m weights that must be optimized. This explosion of dimensionality has led to the models typically being truncated to either first-order for single component receptive fields or second-order for multicomponent receptive fields where $\mathbf{J}$ is diagonalized to recover the receptive field components [28].

For the first-order and second-order MNE models, the results can be improved by including early stopping as a mild form of regularization to prevent overfitting [28]. The model is optimized on a training set and the prediction error of the model is measured on a separate cross-validation set over the course of the optimization. After several failures of the optimization to find a solution with lower prediction error, the optimization returns those weights that minimize the prediction error on the cross-validation set.

1.6 Statement of the problem

While the dimensionality reduction methods summarized above are able to reconstruct a subspace of stimulus space relevant to a neural response, they provide only a partial view of the underlying neural computations from which they are derived that must be subjected to further inspection. In Chapter 2, a method called the functional basis will be proposed that seeks to reveal more about neural computations by modeling the underlying functional neural circuitry. Applications of the functional basis method to early vision neurons and later in Chapter 4 to high-level auditory neurons provide novel conclusions about the type of computations that are performed by populations of neurons in these regions and provide evidence of a potential difference in how sensory information is processed in portions of the visual and auditory systems.

At high-levels of sensory processing, dimensionality reduction methods have so far had limited success in recovering multicomponent receptive fields. Resolving multicomponent receptive fields of high-level auditory neurons was found to be consistent with these past methodological struggles. In order to make progress on the study of the functional neural circuitry of high-level auditory neurons, it became necessary during the intermediate part of the analysis to expand upon the second-order MNE method to handle the harsher realities that accompany analysis of high-level sensory neurons. This method, called the low-rank MNE method, is the subject of Chapter 3 where the theoretical argument is presented and part of Chapter 4 where the practical application of the method is discussed before being applied to data recorded from high-level auditory neurons.

Chapter 2

Identifying functional bases of neural computations

In Chapter 1, several methods were introduced to reconstruct the receptive field, Ω, of a neuron. Each of these methods define a set of basis vectors that are relevant to the neural response. However, if one were to test all of these methods on the same data set for a neuron with a multicomponent receptive field (where $r > 1$) subject to stimuli that satisfy the necessary statistics for STC analysis, one is likely to find that each method produces different basis vectors. Of course, this is not surprising given the different measure by which extracted components are considered optimal. For instance, the information-theoretic approaches reconstruct components that either minimize or maximize the mutual information between the stimuli and spikes [26, 28, 29] while STC returns directions of largest absolute variance from the spike-weighted stimulus covariance matrix [4, 20–25]. On the other hand, it is expected that each of these methods will span the same subspace of stimulus space as discussed in regard to the LN model. While these methods have meaning in terms of their respective measures and strong theoretical backing for decoding the receptive field, these bases lack apparent biological

interpretability in terms of a description of the functional neural circuitry. This motivates the introduction of a *functional basis* that explicitly attempts to reconstruct components of the receptive field reflective of the underlying functional neural circuitry.

In this context, to decode the functional neural circuitry is to understand what decision is being made by the neuron based on external sensory input. Note that treating external sensory stimuli as input is defined here as the functional neural circuitry to distinguish it from the local electrical/chemical input that neurons receive at their dendrites. While the the number of anatomical inputs a single neuron might receive number on the order of $10^3 - 10^4$ [32], the number of functional inputs are in practice much fewer (e.g. ~ 10 in early vision layers).

In this chapter, noisy Boolean operations corresponding to logical OR and logical AND functions are proposed to study the computations performed by neurons early in the vision neural circuitry. It is thought that these logical functions can describe the computations that lead to translation invariance [33, 34], motion selectivity [35], coincidence detection [36], and constructing selectivity for sophisticated sensory structures [37]. Here the functional basis method is described in detail and the proposed logical operations are shown to yield biologically interpretable functional bases that are consistent with established knowledge about early visual processing.

2.1 Functional bases

At its core, the functional basis method computes a LN model (Section 1.1) where the nonlinear function $g(\mathbf{C}^{\mathrm{T}}\mathbf{s}) = P(y = 1|\mathbf{s})$ is some hypothesis about the underlying neural computation where $\mathbf{C} \in \mathbb{R}^{D\times n}$ is a matrix of n functional basis vectors, $\left\{\mathbf{c}_k \in \mathbb{R}^D | \forall k \in \{1, \cdots, n\}\right\}$. The notation uses n to distinguish the number of the functional basis vectors from the number of components spanning the receptive field, r,

since the functional basis can be undercomplete or overcomplete. Undercomplete and overcomplete bases are defined as basis set sizes where $n < r$ and $n > r$, respectively. Since the receptive field captures a full description of the subspace relevant to a neural response, it can be concluded that $\mathbf{C}$ is a subspace of $\boldsymbol{\Omega}$ (i.e. $\boldsymbol{\mathcal{R}}(\mathbf{C}) \subseteq \boldsymbol{\mathcal{R}}(\boldsymbol{\Omega})$). Therefore, in the overcomplete case, at least one column of $\mathbf{C}$ is linearly dependent on the other columns.

From a theoretical point of view, a neuron's functional basis is expected to be composed non-orthogonal components. In fact, it would be surprising if the functional basis is orthogonal given that objects in the sensory environment are not identified by orthogonal features. As an intuitive example, consider a neuron that is selective for the colors purple or blue (i.e. the neuron will spike to either purple or blue light but nothing else). The receptive field of the neuron can be completely defined by an orthogonal basis where one axis is blue and the other is red (and green can be safely ignored). Purple, however, projects onto both the red and blue axes. In this example, the neuron would be more accurately described as being selective for blue and purple rather than blue and red. Similar observations are made in the literature, where retinal ganglion cells (RGCs) are known to receive signals from bipolar cells as input whose receptive fields often spatially overlap and are therefore unlikely to be orthogonal [38, 39]. The bipolar cells have relatively simple receptive fields, often composed of a single component that is sensitive to roughly circular patches of light intensity [39]. Yet, if one were to apply the STC or second-order MNE dimensionality reduction methods on a multicomponent RGC, these methods by construction produce orthogonal receptive fields that are linear combinations of the incident bipolar cell receptive fields.

This scenario is illustrated in Fig 2.1 highlighting the misalignment of the receptive field components obtained through STC and a more interpretable functional basis. In this figure, the responses of two sensory neurons to stimuli drawn from a Gaussian white

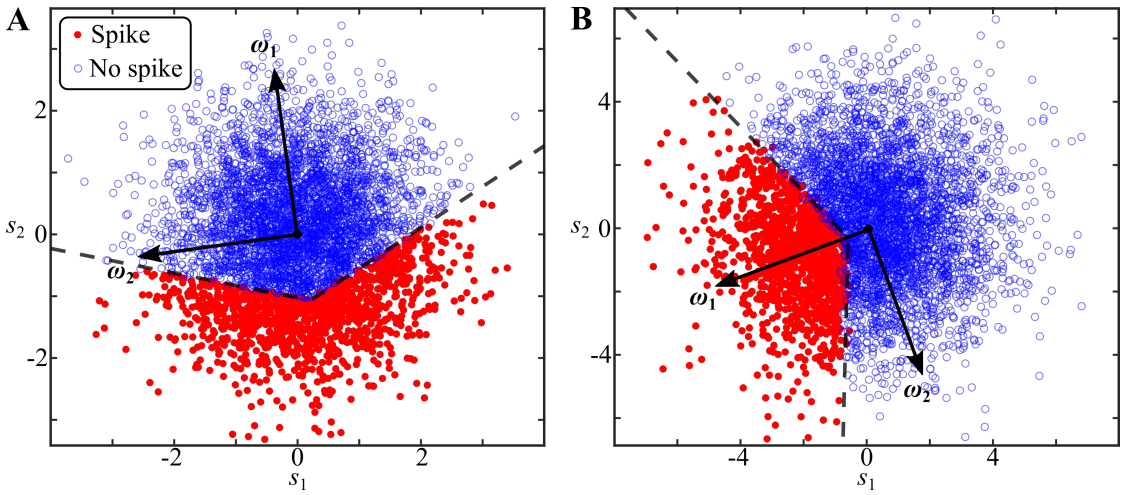

Figure 2.1: **Illustration of two neurons computing Boolean operations in a two-dimensional stimulus space.** The responses of two sensory neurons to stimuli defined in a two dimensional stimulus space follow a noisy logical OR operation (**A**) and a logical AND operation (**B**). The functional input thresholds (dashed lines) demarcate a transition between spiking and silent responses. The STC components are marked with arrows.

noise distribution are plotted as filled and open circles corresponding to spiking ($y = 1$) and silent ($y = 0$) responses. In Fig 2.1**A**, a noisy $n = 2$ logical OR neuron is generated via

$$g(\mathbf{C}^{\mathrm{T}}\mathbf{s}_t) = \begin{cases} 1, & \text{if } \exists k,\ \mathbf{c}_k^{\mathrm{T}}\mathbf{s}_t + \eta_k \geq \theta_k \\ 0, & \text{otherwise} \end{cases} \tag{2.1}$$

where η_k is a normally distributed random number quantifying the input noise and θ_k is the activation threshold of the kth input. The noisy $n = 2$ logical AND neuron that appears in Fig 2.1**B** was generated by changing the spiking condition in Eq 2.1 from existence, $\exists k$, to for all, $\forall k$.

Overcompleteness of the functional basis, where $n > r$, can occur if the neuron receives linearly dependent inputs. However, such a case does not seem likely to occur in practice given how unlikely it is that multiple presynaptic neurons would have perfectly

redundant receptive fields. For instance, in the case of RGCs, even if the presynaptic bipolar cells encode roughly the same information about the stimulus, it is unlikely their receptive fields would perfectly overlap. A more likely scenario is a practical failure of the chosen dimensionality reduction technique to recover the entire subspace spanning the receptive field.

2.2 Modeling the functional basis with Boolean operations

From the prior discussion, it is easy to see the intuition motivating the use of Boolean operations to describe the neural computations of sensory neurons. If a neuron is selective for some specific property of the stimulus, it would make sense that the neuron would want to make a provisional binary decision about the presence or absence of that property. This suggests that the selectivity for such properties could hypothetically be manifested as a binary decision with a threshold defined as a $(D-1)$-dimensional hyperplane normal to a property vector, $\mathbf{c}$, in stimulus space. The final binary decision of the neuron to produce a spike or not is then based on combined state of these provisional binary decisions about the inputs analogous to a truth table in digital logic.

The simplest Boolean operations are logical OR,

$$P_{\mathrm{OR}}(y=1|\mathbf{s}) = 1 - \prod_{k=1}^{n} \left[1 - \sigma_k\left(\mathbf{c}_k^{\mathrm{T}}\mathbf{s}\right)\right], \tag{2.2}$$

and logical AND,

$$P_{\mathrm{AND}}(y=1|\mathbf{s}) = \prod_{k=1}^{n} \sigma_k\left(\mathbf{c}_k^{\mathrm{T}}\mathbf{s}\right) \tag{2.3}$$

where σ_k is a function bounded between 0 and 1 that corresponds to the input nonlinearity or input activation function. In digital logic, where the signal-to-noise ratio of the nonlinearity is approximately infinite at the scale of typical applications, the input nonlinearities are deterministic where σ_k can be modeled by a Heaviside step function. Neurons, on the other hand, exhibit substantial noise in the region of stimulus space that transitions from silent to spiking responses. Therefore, σ_k is modeled as a logistic function

$$\sigma_k\left(\mathbf{c}_k^{\mathrm{T}}\mathbf{s}\right) = \frac{1}{1+e^{-b_k-\mathbf{c}_k^{\mathrm{T}}\mathbf{s}}} \tag{2.4}$$

where b_k is a scalar threshold. A couple alternatives that could also be used are $\sigma_k(\cdot) = \frac{1}{2}[1+\tanh(\cdot)]$ or $\sigma_k(\cdot) = \frac{1}{2}[1+\mathrm{erf}(\cdot)]$ (where $\mathrm{erf}(\cdot)$ is the Gauss error function). The logistic function is chosen because it is a continuously differentiable function and it is the the function that maximizes the noise entropy between the input and the provisional decision in similar fashion to Section 1.5 [28] (where y in this case is the provisional decision rather than a spike) which may limit the bias in the estimate of the functional basis. If the noise around the threshold is Gaussian distributed, a good argument could be made for using the error function input nonlinearity instead. However, the logistic distribution has a similar shape to the Gaussian distribution and has worked well in practice.

2.2.1 Optimization procedure

The Boolean operations are optimized by searching for the maximum likelihood estimate of $P(y=1|\mathbf{s})$ with respect to the thresholds and functional basis vectors. This

can be achieved by minimizing the mean negative log-likelihood cost function,

$$L(\mathbf{b},\mathbf{C}) = -\frac{1}{N_{\text{samp}}}\sum_{t=1}^{N_{\text{samp}}}\Big[y_t \log\big(P(y=1|\mathbf{s}_t)\big) + (1-y_t)\log\big(1-P(y=1|\mathbf{s}_t)\big)\Big], \tag{2.5}$$

where $\mathbf{b}^{\mathrm{T}} = [b_1, \cdots, b_n]$ collects the input thresholds into a vector. When $n = 1$, both the logical AND and logical OR models are equivalent to first-order MNE (Chapter 1.5) and therefore any local minimum of the $n = 1$ model is globally optimal. When $n > 1$, however, the cost function is nonconvex and may possess suboptimal local minima or saddle points. Since the cost function is continuously differentiable, gradient based methods such as conjugate gradient descent or Newton-type methods may be used to optimize the functional basis to local optimality [40, 41]. Global optimization heuristics such as running the optimization many times with several different initializations or stochastic gradient descent [42] may be used if the global minimum is desirable. Deterministic global optimization techniques that attain certifiable global optimality (to some precision), such as branch and bound [43], may be attempted but are intractable for large D or n since they converge at worst in exponential time.

2.2.2 Taking advantage of dimensionality reduction

Since $\boldsymbol{\mathcal{R}}(\mathbf{C}) \subseteq \boldsymbol{\mathcal{R}}(\boldsymbol{\Omega})$, it is not necessary for one to optimize $\mathbf{C}$ in the full D-dimensional stimulus space but rather within a reduced subspace defined by the receptive field estimate. A reduced subspace of stimulus space can be defined by projecting $\mathbf{s}$ into the basis recovered from a dimensionality reduction method such as STC where the reduced subspace is then $\mathbf{s}^* = \boldsymbol{\Omega}^{\mathrm{T}}\mathbf{s}$ and the rank of $\boldsymbol{\Omega}$ is assumed to be less than D. One can then find the maximum likelihood estimate of $P(y=1|\mathbf{s})$ in terms of a (potentially) much smaller set of reduced functional basis vectors, $\boldsymbol{\Phi} \in \mathbb{R}^{r\times n}$. The nonlinearities would then be some combination of the input nonlinearities defined in the reduced subspace,

$\sigma_k\left(\phi_k^{\mathrm{T}}\mathbf{s}^*\right)$ where ϕ_k is the kth column of $\mathbf{\Phi}$. After finding the optimal $\mathbf{\Phi}$, the functional basis can be transformed back into the full stimulus space by a backwards transformation $\mathbf{C} = \mathbf{\Omega\Phi}$.

It is recommended to take advantage of dimensionality reduction when applying the functional basis method for several practical reasons. With the exception of MID, the other dimensionality reduction techniques suggested in the introduction provide globally optimal estimates of the receptive field in polynomial time. Optimizing the functional basis in the reduced subspace can then lead to overall improvement in the computation time required to reach a globally optimal solution relative to optimizing in the full stimulus space due to a reduction in size of the search space. In particularly low-r cases, deterministic global optimization techniques may be practically employed. Related to this, the reduced subspace is likely to have a much smaller number of local minima and saddle points than the full stimulus space. Lastly, since undercompleteness and overcompleteness are expected to be rare, the number of components that make up the receptive field can act as a guide in setting the functional basis set size. The matter of determining the optimal number of components that make up the functional basis set is a topic that is discussed in a practical context in the following section.

2.3 Demonstration on synthetic neurons

The functional basis method was tested on three synthetic vision neurons to observe whether the maximum likelihood estimate (Eq 2.5) could recover the functional basis. For each neuron, 200,000 stimuli were drawn from a Gaussian white noise distribution. All of the synthetic neurons had multicomponent receptive fields and were designed to be integrative, spiking if the projection of a stimulus on any functional input component was above its respective spiking threshold. This corresponds to the logical

OR function. Noisy data was generated according to Eq 2.1 and a threshold common to the all input components was adjusted until the mean probability of a spike was between 0.2 and 0.4.

One key property of interest in neural coding is the ability of models to capture invariances in the inputs; e.g., translation, rotation, and scale invariance. Invariances in this context means that the neuron will respond in kind to the same stimulus subject to changes in either the location of the object of interest in the field of view (translation), the two or three-dimensional orientation of an object in space (rotation), or the various sizes the same object might take (scale). To test the ability of the functional basis method to capture invariant input components, a translationally and rotationally invariant neuron were generated.

The four input components of the translationally invariant neuron appear in Fig 2.2**A**. Each component features the same center-surround structure shifted to the four different corners of the image. At the center of the images, the input components overlap and are apparently non-orthogonal. As such, dimensionality reduction techniques that yield orthogonal components of the receptive field can be expected to fail to recover the functional input components. Since the synthetic neuron was stimulated by Gaussian white noise stimuli, STC was the model of choice for recovering the receptive field. Indeed, the STC components behave as expected in Fig 2.2**B** where the resulting components are linear combinations of the functional inputs in Fig 2.2**A**. This is not to say, however, that STC has failed in its more general purpose of reconstructing the receptive field. An overlap metric [44, 45] defined as

$$O(\mathbf{A},\mathbf{B}) = \frac{\left|\operatorname{Det}\left(\mathbf{A}^{\mathrm{T}}\mathbf{B}\mathbf{B}^{\mathrm{T}}\mathbf{A}\right)\right|^{\frac{1}{2r_{\min}}}}{\left|\operatorname{Det}\left(\mathbf{A}^{\mathrm{T}}\mathbf{A}\right)\right|^{\frac{1}{2r_{\mathbf{A}}}}\left|\operatorname{Det}\left(\mathbf{B}^{\mathrm{T}}\mathbf{B}\right)\right|^{\frac{1}{2r_{\mathbf{B}}}}} \tag{2.6}$$

can be used to measure to what degree matrix $\mathbf{A}$ defines a subspace of matrix $\mathbf{B}$ or vice

versa where $r_{\mathbf{A}} = \text{rank}(\mathbf{A})$, $r_{\mathbf{B}} = \text{rank}(\mathbf{B})$, and $r_{\min} = \min(r_{\mathbf{A}}, r_{\mathbf{B}})$. The overlap is bound between 0 and 1. If the subspaces are disjoint (i.e. $\mathcal{R}(\mathbf{A}) \cap \mathcal{R}(\mathbf{B}) = \emptyset$), then $O = 0$. If the subspaces have an overlap of $O = 1$, at least one of the matrices defines a subspace of the other (e.g. $\mathcal{R}(\mathbf{A}) \subseteq \mathcal{R}(\mathbf{B})$). Comparing the overlap of the STC receptive field with the ground truth functional inputs yields an overlap of 0.992 indicating that STC is performing triumphantly in its stated purpose.

Exploiting prior knowledge from the construction of the synthetic neuron, an $n = 4$ logical OR model was fit to the pairs of reduced stimuli and responses. The logical OR functional basis in the full stimulus space is presented in Fig 2.2**C** where the underlying translation invariance is recovered as hoped. To show the significance of the logical OR model, the logical AND model was also fit, recovering the functional basis that appears in Fig 2.2**D**. The logical AND functional basis illustrates how important the choice of nonlinearity is to functional basis recovery since the logical AND model roughly recovers four repeats of the STA. The predicted probabilities of a spike appears in Fig 2.2**E** for a 100 frame selection of stimuli drawn from the test set are shown alongside the ground truth spiking response where, by eye, the logical OR model can be seen to make better predictions. This is verified quantitatively by comparing the ability of each of the models to predict the spikes in the test sets. Logical OR was found to be a better predictor with a prediction error (negative log-likelihood) of $L = 0.1099 \pm 0.0007$ (correlation coefficient: $R_c = 0.9119 \pm 0.0001$) averaged across the test sets compared to logical AND where $L = 0.340 \pm 0.005$ ($R_c = 0.6791 \pm 0.0001$).

Of course, to make this simulation of the analysis more realistic, one cannot generally assume omniscience of which model is most appropriate and how many input components, n, should enter the functional basis set. While it is likely that n will be equal to the dimensionality of the receptive field, it would be useful to have a procedure to show this to be the case empirically to rule out (or discover) the however uncommon possibility

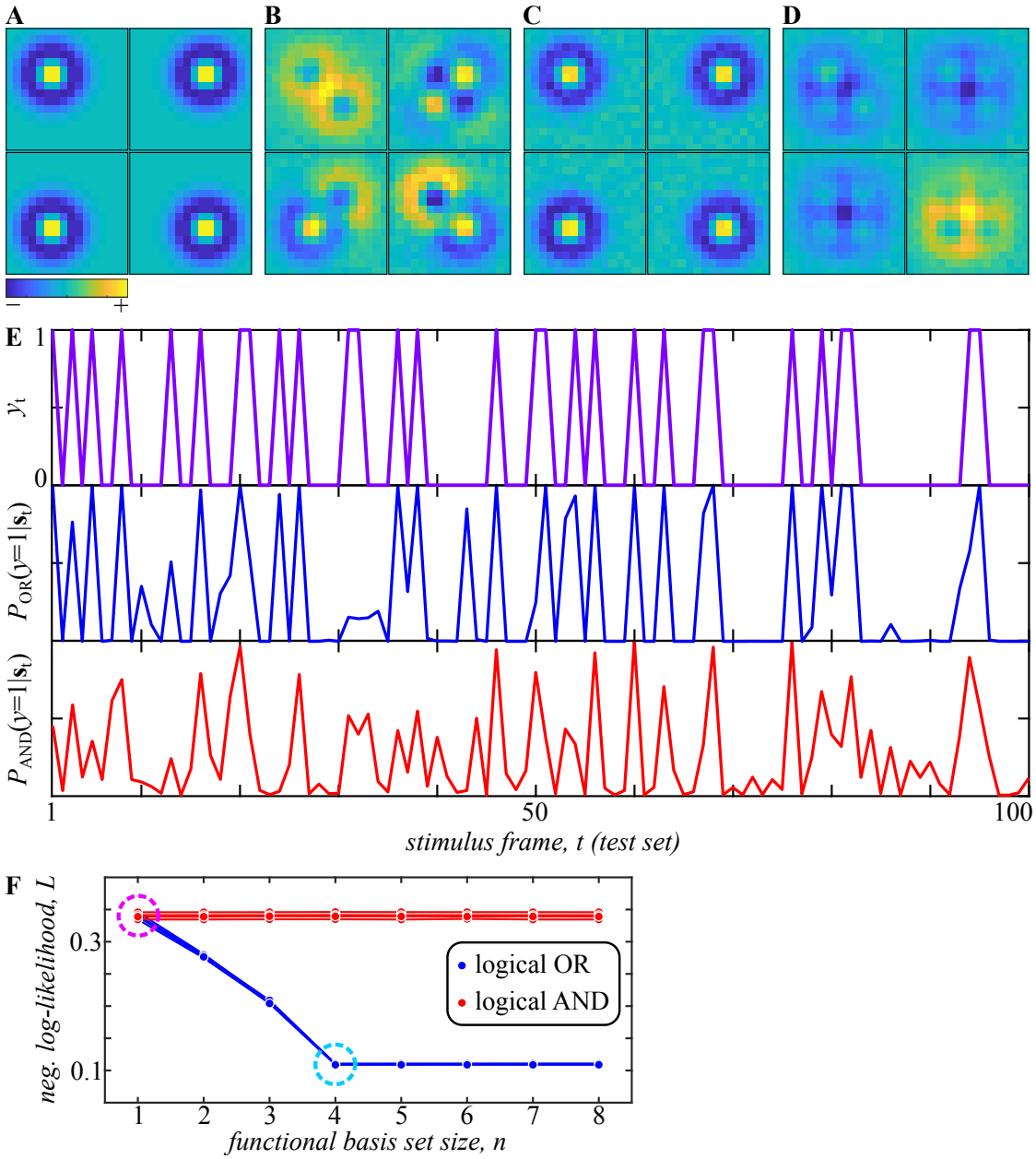

Figure 2.2: **The functional basis method applied to a synthetic translationally invariant vision neuron.** (**A**) The functional inputs are composed of four translationally invariant components. (**B**) STC recovers four orthonormal components. (**C**) A maximum likelihood estimated logical OR model recovers the functional input components. (**D**) The maximum likelihood estimated of a logical AND model recovers components resembling the STA. (**E**) The first 100 frames of the test set are plotted (top) alongside predictions from the logical OR (middle) and logical AND models (bottom). (**F**) The prediction error (negative log-likelihood) evaluated on four cross-validation sets saturates at four components for the logical OR model and one component for the logical AND model.

of undercompleteness or overcompleteness of the functional basis. The prescribed approach is to perform empirical model selection where the negative log-likelihood is evaluated on a cross-validation set. An example of the model selection procedure appears in Fig 2.2**F** where both logical OR and logical AND models are fit on four different training sets for all $n \in \{1, \cdots, 8\}$ and then a curve is plotted of the negative log-likelihood evaluated on four unique cross-validation sets. For a given model, the best fit basis set size is the minimal n where the log-likelihood plateaus or, more loosely, when the curve is deemed to provide diminishing returns (similar to the "elbow method" used in clustering [46]). For logical OR, the best model is $n = 4$, the same number of components as the ground truth. Logical AND, by contrast, saturates immediately at $n = 1$ which reflects the fact that the logical AND functional basis only recovers repeats of what resembles the STA. Once the best functional basis set sizes are known for each model, the best from each model type are then compared through evaluation of the prediction error on the cross-validation set and the model with the lowest prediction error is selected. Clearly, in Fig 2.2**F**, the logical OR model with $n = 4$ is the best fit.

The rotationally invariant synthetic neuron features a single curved component rotated eight times at $k\pi/4$ radians where $k \in \{1, \ldots, 8\}$ in a logical OR configuration is shown in Fig 2.3**A**. STC is again employed to estimate the receptive field and appears in Fig 2.3**B** where, once again, the STC components bear little resemblance to the functional input components. In fact, seeing the three largest absolute variance components (the three left-most components of Fig 2.3**B**), one may mistakenly conclude that the functional inputs are more localized to the center of the image. On the other hand, the lower absolute variance components (at the right end of Fig 2.3**B**) may be erroneously assumed to be largely a product of noise. However, these lower absolute variance components contain important structure and if one were then to decide to exclude them, it is possible that important ingredients necessary to recover the functional input space could be missing.

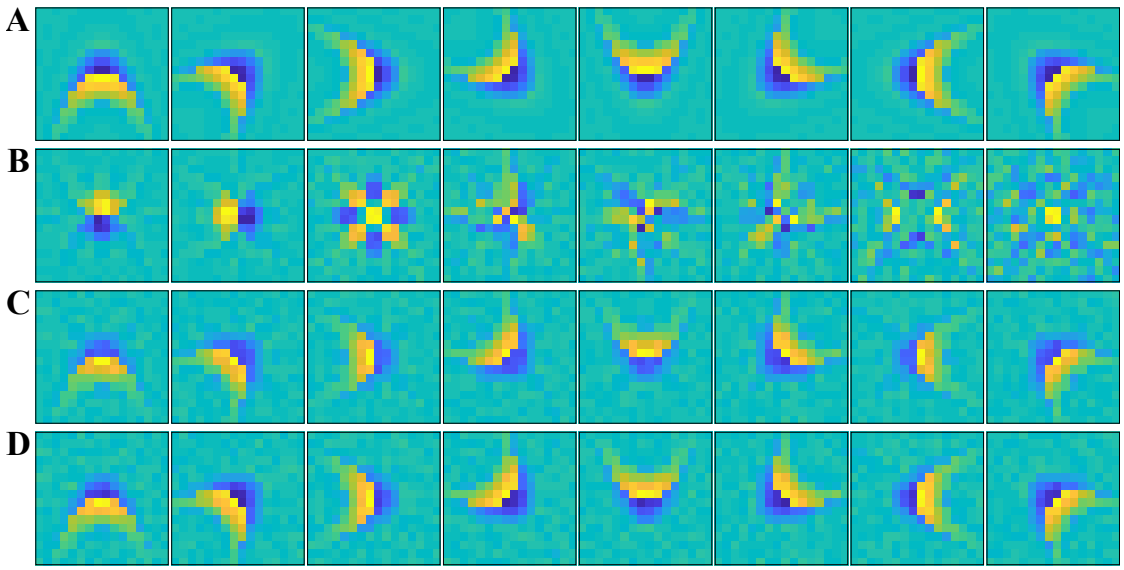

Figure 2.3: **The functional basis method applied to a synthetic rotationally invariant neuron.** (**A**) A rotationally invariant synthetic neuron is constructed from eight functional input components. (**B**) An orthonormal set of components spanning the receptive field is recovered using STC. (**C**) A receptive field where the right-most component from **B** is dropped was fit with an overcomplete logical OR model displaying subtlety diminished central amplitude. (**D**) Using the full receptive field estimate, the functional input components are reconstructed using a logical OR model.

For instance, in Fig 2.3**C** the right-most component of the receptive field is dropped such that $r = 7$ and an overcomplete $n = 8$ logical OR model is computed. Though it is subtle, there is a decrease in amplitude near the center of the components in Fig 2.3**C** compared to the ground truth. The impact of this subtle difference is a statistically significant increase in prediction error evaluated on the test sets where the overcomplete model's prediction error is $L = 0.241 \pm 0.002$ ($R_c = 0.8020 \pm 0.0003$) vs. $L = 0.187 \pm 0.002$ ($R_c = 0.8522 \pm 0.0002$) for the $r = 8$ and $n = 8$ logical OR model featured in Fig 2.3**D**.

What impact does varying n have on the functional basis reconstruction of the rotationally invariant neuron? The result of fitting an undercomplete logical OR model is shown in Fig 2.4**A** for $n = 7$ (while $r = 8$) where it can be seen that the first component of the reconstruction recovers a linear combination of two of the functional input components leading to an increase in the prediction error to $L = 0.242 \pm 0.001$ (and

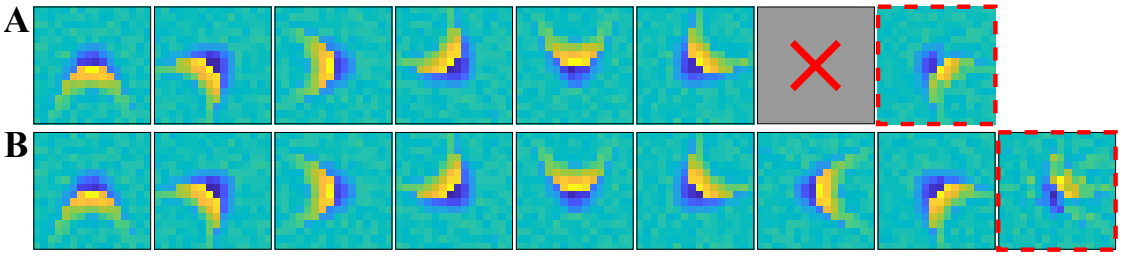

Figure 2.4: **The functional input components recovered with undercomplete and overcomplete functional bases.** (**A**) An undercomplete logical OR model recovers the first six functional input components but fails to recover the seventh (marked with a red 'x' and corresponding to the missing component below) and produces a compromised eighth component (boxed by a red dashed line). (**B**) The overcomplete logical OR model recovers all of the functional input components but also fits a novel component at the right end (boxed by a red dashed line).

decrease in the correlation coefficient: $R_c = 0.805 \pm 0.001$). An overcomplete model where $n = 9$ is also fit with a logical OR model recovering the components in Fig 2.4**B** without significant change in prediction error of the models ($L = 0.187 \pm 0.002$ and $R_c = 0.8523 \pm 0.0003$) relative to the complete $n = 8$ model. The overcomplete model recovers the rotationally invariant receptive field but also includes a novel component that marginally overfits to the noise in the data. Though it is not the case here, overcomplete models have also been known to occasionally yield repeated components.

The final simulated analysis is of a synthetic logical OR RGC to examine the ability of the functional basis method to identify bipolar cell inputs (Fig 2.5**A**) in preparation for the forthcoming application to authentic data recorded from the retina. It is worth mentioning here that this synthetic RGC was created and analyzed before attempting to apply the functional basis method to the retinal data. Its structure was therefore not defined after the fact but was rather hypothesized based on prior knowledge [39] and can act as a form of validation later on. Unlike the other model neurons, the synthetic RGC cannot be clearly demarcated as either translationally or rotationally invariant but rather could be argued to exhibit an inexact form of translation invariance. The STC basis in Fig 2.5**B** is especially difficult to interpret because the components reveal no apparent

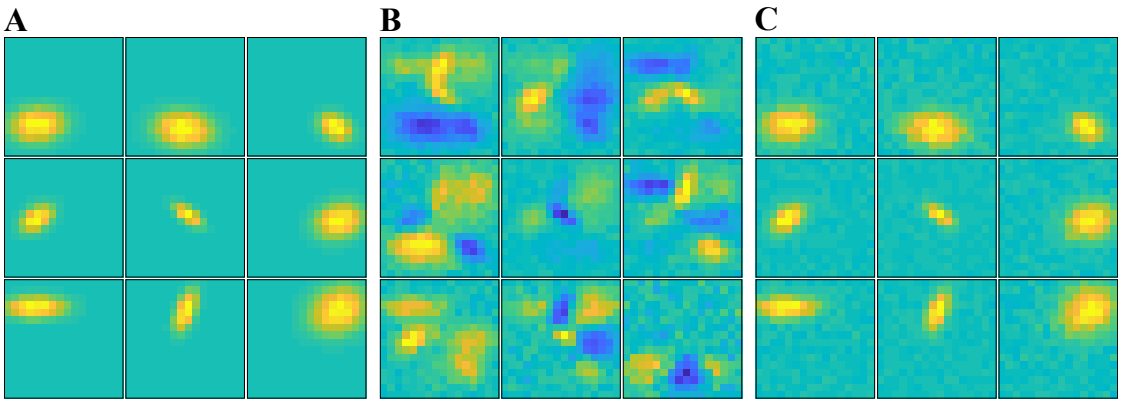

Figure 2.5: **The functional basis method applied to a synthetic retinal ganglion cell (RGC).** (**A**) The functional input components of the synthetic RGC are composed of bivariate Gaussians dispersed across field of vision. (**B**) The receptive field recovered via STC contains sophisticated structures that are difficult to interpret. (**C**) The maximum likelihood estimate of a logical OR model recovers the functional input components.

localized structure and are bimodal in stark contrast to the functional input components. Linear combinations of the STC basis computed with the logical OR model in Fig 2.5**C** bear a strong resemblance to the functional input components. This demonstrates a significant advantage of the functional basis method over convolutional methods [47,48] in that the functional basis method can recover approximate invariances whereas convolutional methods would average out the details of the components.

2.4 Functional neural circuitry of retinal ganglion cells

To study the functional neural circuitry of early vision neurons, the functional basis method was applied to a data set composed of electrophysiological recordings of 53 RGCs from the salamander retina gathered in a prior study by Marre, et al. [49]. The neurons were subjected to visual stimulation in the form of a video where each frame was a 40×40 pixel image of binary white noise. To be more specific, the pixels were spatiotemporally uncorrelated and their values were drawn independently with uniform probability from $s_i \in \{-1, 1\}$ for the ith pixel. The video had a total of 137,145 frames

that were presented at a 60 Hz frame rate ($\sim$ 38 total minutes of stimulation). The total number of spikes each neuron elicited ranged from 118 to 30,908 with a median of 5,453 spikes.

The stimuli were preprocessed by (i) using a filtering technique such that the recovered spatiotemporal receptive field would be transformed into a purely spatial receptive field and (ii) extracting a relevant patch of the images to reduce the dimensionality of the stimulus space. The filtration was performed by computing a spatiotemporal STA (Eq 1.1) where nine additional frames preceding time t are included in each stimulus vector (i.e. $\mathbf{s}_t \leftarrow [\mathbf{s}_{t-9}, \mathbf{s}_{t-8}, \cdots, \mathbf{s}_t]$). The spatiotemporal component recovered from the STA was then reshaped into a 1600×10 matrix where each column corresponds to a frame. The singular value decomposition of this matrix was then computed, matrix $\left(\boldsymbol{\omega}^{(\mathrm{STA})}\right) = \mathbf{K}^{(\mathrm{spat})}\boldsymbol{\Sigma}\mathbf{K}^{(\mathrm{temp})\mathrm{T}}$, where $\mathbf{K}^{(\mathrm{spat})} \in \mathbb{R}^{1600\times 10}$ and $\mathbf{K}^{(\mathrm{temp})} \in \mathbb{R}^{10\times 10}$ are matrices whose columns correspond to bases of spatial and temporal components, respectively, and $\boldsymbol{\Sigma} \in \mathbb{R}^{10\times 10}$ is a diagonal matrix of the singular values in descending order from top-left to bottom-right. Since the largest singular value was much larger than the remainder, the spatial and temporal activity of the receptive field were deemed approximately separable into a spatial and temporal kernel [50] obtained from the first column of $\mathbf{K}^{(\mathrm{spat})}$ and $\mathbf{K}^{(\mathrm{temp})}$. A stimulus vector that would ostensibly produce a purely spatial receptive field was then generated by taking a weighted sum of the ten spatial stimuli that precede time t:

$$\mathbf{s}_t^{(\mathrm{spat})} = \sum_{\tau=0}^{9} \mathbf{s}_{t-\tau} K_{t-\tau,1}^{(\mathrm{temp})}. \tag{2.7}$$

This approach was reasonable in this case because the spatiotemporal receptive field was separable and therefore the dimensionality of the problem could be reduced by removing the temporal dependence of the response without significantly impacting the overall

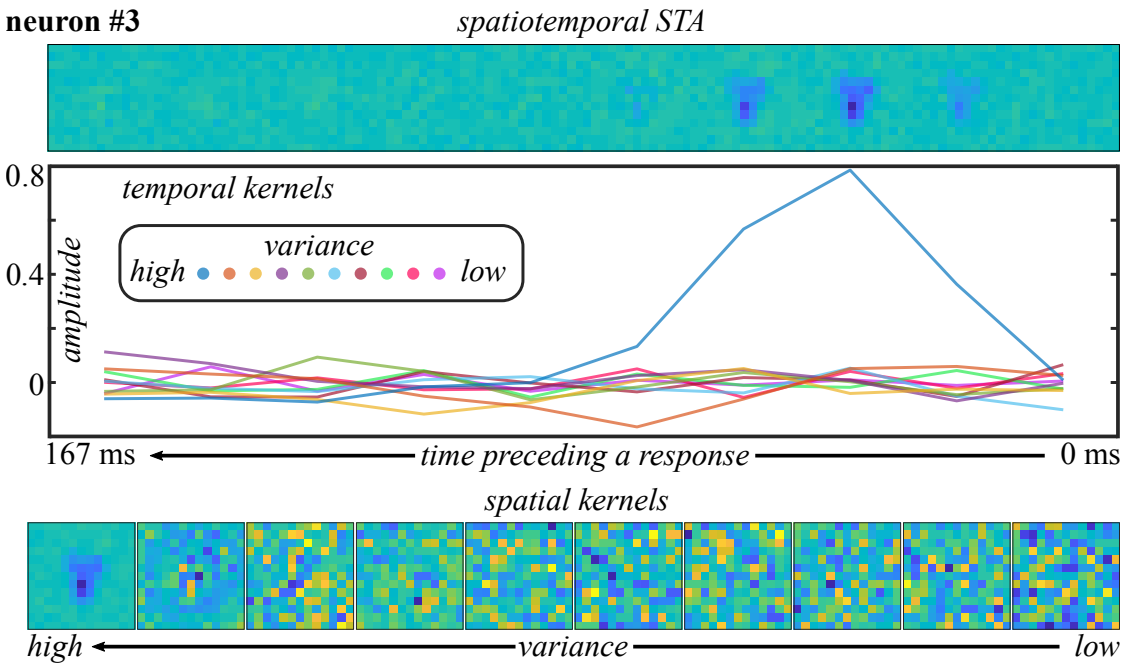

Figure 2.6: **Example showing the spatial and temporal separability of an RGC receptive field.** Here is an example RGC where the spatial and temporal modulation of the STA can be seen to be reasonably separable. At top is the spatiotemporal STA, in the middle are the temporal components weighted by singular values where the largest variance mode stands out, and the bottom shows the spatial kernels where all but the highest variance kernels are dominated by noise (note that the variances are all positive in this case).

shape of the spatial receptive field (see Fig 2.6 for an example). The dimensionality of the stimuli was then reduced further by removing the empty space on the periphery of the STA such that any pixels outside of the relevant $\sim 16 \times 16$ patch of pixels centered on the largest magnitude pixel of the STA were discarded. Because the neurons were sensitive to activity in different sectors of the full, non-patched image, this patching was determined uniquely for each neuron.

The analysis of the RGCs largely mirrored that of the model neurons. The first step was dimensionality reduction. Since the stimulus distribution was radially symmetric, STC remained a reasonable choice for the recovery of the receptive field. Even though the stimuli were not Gaussian distributed, the results of STC analysis did not appear to exhibit noticeable distortion. Due to the presence of noise in the stimulus/response

distribution, the STC method required the introduction of an additional step to resolve which components were significant contributors to the response from those that originated from noise. The eigenvalue spectra from the STC method were compared against those from decorrelated STC where the correlations between the stimuli and responses were broken such that $P(y,\mathbf{s}) \approx P(y)P(\mathbf{s})$. The decorrelation was achieved by first shifting the responses backward in time by a minimal amount to break causation between the stimulus and response at sample t. Then the STC was computed for several different shifts backwards in time and the eigenvalues were pooled. Those positive eigenvalues from the initial correlated STC that exceeded the maximum eigenvalue pooled from the decorrelated STCs and those negative eigenvalues of the initial STC that were more negative than the minimum pooled eigenvalue were considered significant components. It is claimed that this approach is better than randomly shuffling the response because it preserves the overall structure of the spike train [20]. In this specific application, the minimum shift was 100 frames and the eigenvalue spectra of 200 backward shifts were combined.

Of the total 53 neurons, 49 neurons were found to have multicomponent receptive fields and were therefore of particular interest for further analysis (recall that the STC basis should be equal to the functional basis when the receptive field possesses a single component). Note that this is an improvement over the original publication where Kaardal et al. [51] found 30 neurons with multicomponent receptive fields. This improvement was obtained by adjusting the delay between spikes and stimuli using a combination of visual inspection and the magnitude of the largest variance component from the STC method whereas the original publication made this adjustment using the maximum pixel value from the STA. These selected neurons had total spike counts ranging from 192 to 21,894 with a median of 5,426. The receptive fields recovered from two example neurons appear in Fig 2.7. The observed spatial receptive fields are quite similar across the three

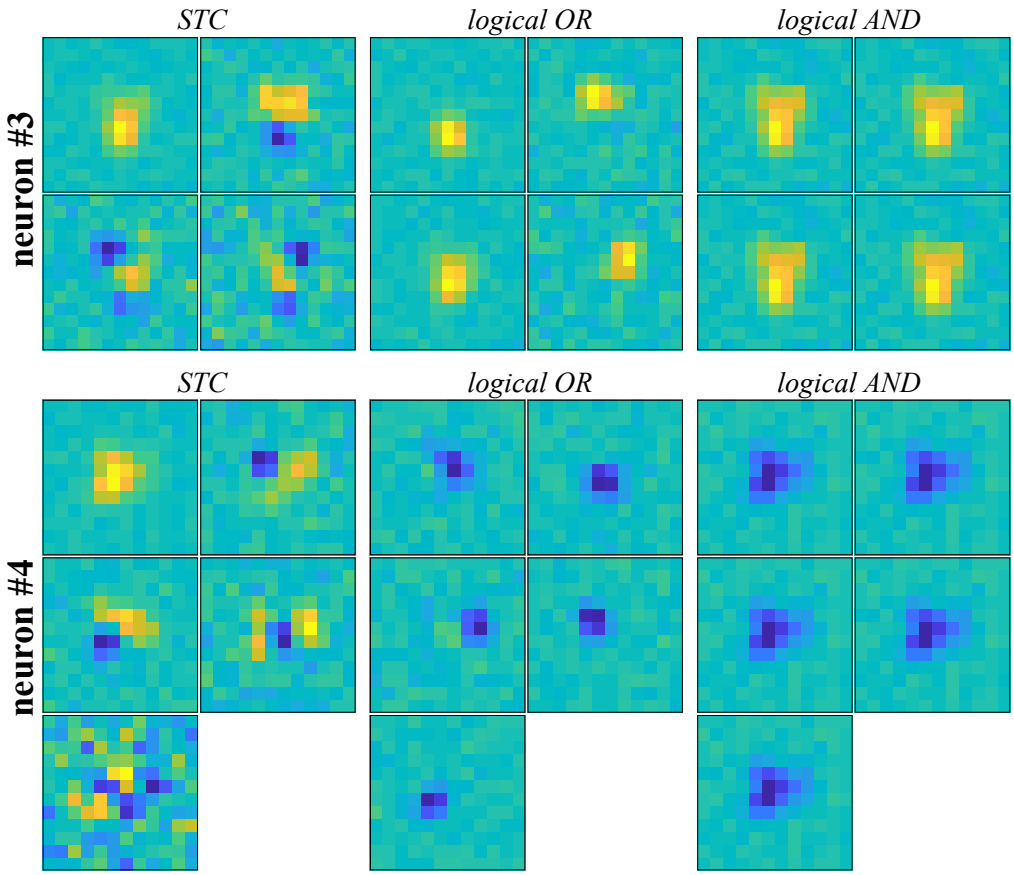

Figure 2.7: **Receptive fields and functional bases from two RGCs.** Estimates of the receptive fields of two example RGC neurons were obtained using STC. Functional bases were computed for each neuron by taking linear combinations of the components spanning the receptive fields as prescribed by maximum likelihood estimates of logical OR and logical AND models.

presented neurons with the most dominant component being monophasic (unmodulated) and the remaining multiphasic with contiguous regions of amplitude and alternating polarity similar to findings in the primary visual cortex [17].

Both logical AND and logical OR models were fit for the 49 selected neurons. The jackknife method [52] was used to form distinct training and cross-validation sets by dividing the stimulus/response pairs into sections containing 25% of samples. The models were then trained on 50% of the samples by choosing two of the sections. The remaining two sections were each assigned to cross-validation and test sets, respectively. Each

model was fit on four unique arrangements of the sections into training, cross-validation, and test sets. To compare the relative quality of fit of two models (such as logical OR versus logical AND), a normalized difference in the log-likelihood was calculated,

$$\Delta L_{\mathrm{A,B}} = \frac{L_{\mathrm{B}} - L_{\mathrm{A}}}{L_{\mathrm{A}} + L_{\mathrm{B}}}, \tag{2.8}$$

where L_{A} and L_{B} are the log-likelihoods of models A and B, respectively, evaluated on the cross-validation set. When the mean difference taken across jackknives, $\Delta L_{\mathrm{A,B}}$, is greater than zero, model A is deemed to be the more predictive model while B is when $\Delta L_{\mathrm{A,B}}$ is less than zero. With regard to the two example neurons, logical OR was found to be the better model for both of them. The neuron labeled **neuron #3** (Fig 2.7) had a relative prediction error of $\Delta L_{\mathrm{OR,AND}} = 0.024 \pm 0.002$ indicating that the logical OR model was a significantly better description than the logical AND model. This was likewise the case for **neuron #4** (Fig 2.7) where $\Delta L_{\mathrm{OR,AND}} = 0.034 \pm 0.003$.

As a baseline assessment of model quality, the functional bases were also compared against predictions made from the STC dimensions. Since STC is nonlinearity-independent, the function relating the linear projections $\mathbf{\Omega}^{\mathrm{T}}\mathbf{s}_t$ to the probabilistic output is ambiguous. To limit bias from choosing a potentially erroneous functional form for the nonlinearity, a primitive choice is to compute an empirical nonlinearity by averaging the responses from the training set into discrete bins as a function of the stimulus projection into the receptive field components. Then predictions can be made by projecting novel stimuli into the receptive field components and looking up the expected response. This approach was used in Kaardal et al. [51] and is partially reproduced here. However, this procedure does not scale well because the nonlinearity falls victim to the curse of dimensionality (Section 1.4.1) and takes exponentially long (as a function of r) to compute. This became an issue in this reproduction because up to eight receptive field

components were found for some neurons while only four were found in Kaardal et al. [51] due to the adjustments made here in the time delays between the alignment of stimuli and responses.

An alternative binning procedure employed here was to instead average over a multivariate Gaussian distribution like so:

$$P(y=1|\mathbf{s}_{t^*}) = \frac{\sum_{t\in T_{\text{train}}} y_t \; e^{-(\Delta\mathbf{x}_{t^*,t})^{\mathrm{T}}\boldsymbol{\mathcal{K}}^{-1}\Delta\mathbf{x}_{t^*,t}}}{\sum_{t\in T_{\text{train}}} e^{-(\Delta\mathbf{x}_{t^*,t})^{\mathrm{T}}\boldsymbol{\mathcal{K}}^{-1}\Delta\mathbf{x}_{t^*,t}}} \tag{2.9}$$

where T_{train} is the set of sample labels that belong to the training set, $\mathbf{s}_{t^*}$ is a novel stimulus ($t^* \notin T_{\text{train}}$), $\Delta\mathbf{x}_{t^*,t} = \mathbf{x}_{t^*} - \mathbf{x}_t$, $\mathbf{x}_t = \boldsymbol{\Omega}^{\mathrm{T}}\mathbf{s}_t$, and $\boldsymbol{\mathcal{K}}$ is a covariance matrix. Intuitively, Eq 2.9 predicts the conditional spiking probability by taking the mean response from the training set weighted by a Gaussian centered at the novel stimulus projected into the receptive field. In this case, the covariance matrix was defined as

$$\boldsymbol{\mathcal{K}}^{-1} = \frac{n_{\text{bins}}}{\sigma^2_{\text{Gauss}}}\mathbf{diag}\left(\mathbf{x}^{(\max)} - \mathbf{x}^{(\min)}\right)^2 \tag{2.10}$$

where

$$x_k^{(\max)} = \max\left(\left\{x_{k,t} = \boldsymbol{\omega}_k \cdot \mathbf{s}_t | \forall t \in T_{\text{train}}\right\}\right), \tag{2.11}$$

$$x_k^{(\min)} = \min\left(\left\{x_{k,t} = \boldsymbol{\omega}_k \cdot \mathbf{s}_t | \forall t \in T_{\text{train}}\right\}\right), \tag{2.12}$$

$n_{\text{bins}} = 10$, and σ_{Gauss} was set such as to minimize the negative log-likelihood of predictions made on the cross-validation sets ($\sigma_{\text{Gauss}} \approx 0.27$). The advantage of this procedure is that the computation scales linearly because there is no need to create a map of the nonlinearity before forming predictions, the extrapolation to unexplored regions of stimulus space is simpler to compute, and the predictions are more stable to adjustments of n_{bins} which may suggest that it more efficiently uses the data than the standard binning

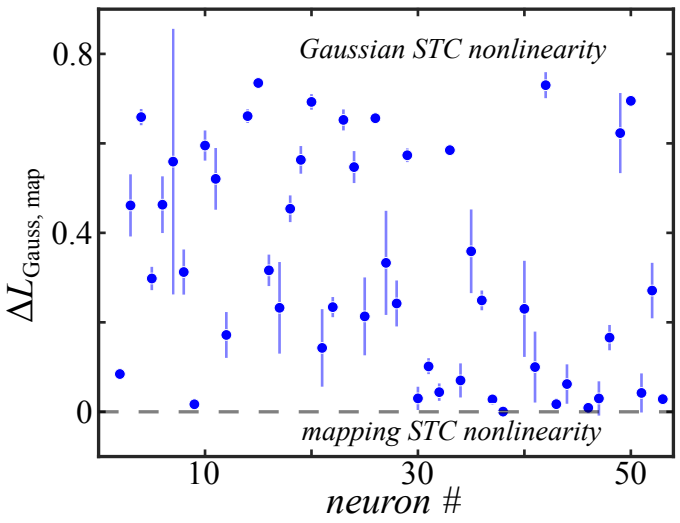

Figure 2.8: **Comparing the prediction error of empirical nonlinearities across the population of RGCs.** The relative prediction error ($\Delta L_{\text{Gauss,map}}$) shows that the empirical Gaussian nonlinearity makes better predictions on the test sets for all neurons in the data set compared to the discrete mapping of the nonlinearity to bins.

procedure. A comparison is made between the new Gaussian and old discrete mapping predictions in Fig 2.8 for neurons where $r \leq 6$ showing that the new procedure forms universally better predictions on the test sets and thus the Gaussian nonlinearity will be used in place of the old approach in the remainder of the chapter.

For both example neurons, it was found that either choice of logical OR or logical AND models performed better than the empirical STC nonlinearity. **Neuron #3**'s logical OR and logical AND models when compared to the STC model performed better with $\Delta L_{\text{OR,STC}} = 0.038 \pm 0.004$ and $\Delta L_{\text{AND,STC}} = 0.014 \pm 0.004$, respectively. Similarly, for **neuron #4**, $\Delta L_{\text{OR,STC}} = 0.16 \pm 0.02$ and $\Delta L_{\text{AND,STC}} = 0.13 \pm 0.02$ for logical OR and logical AND models, respectively.

Across the population of 49 neurons found to have multicomponent receptive fields, the results largely reflected those of the two example neurons. The logical OR models outperformed the logical AND models across the neuron population when the relative prediction error was evaluated on the cross-validation sets (Fig 2.9**A**). The only neurons for which the performance was equal were single-component neurons where the

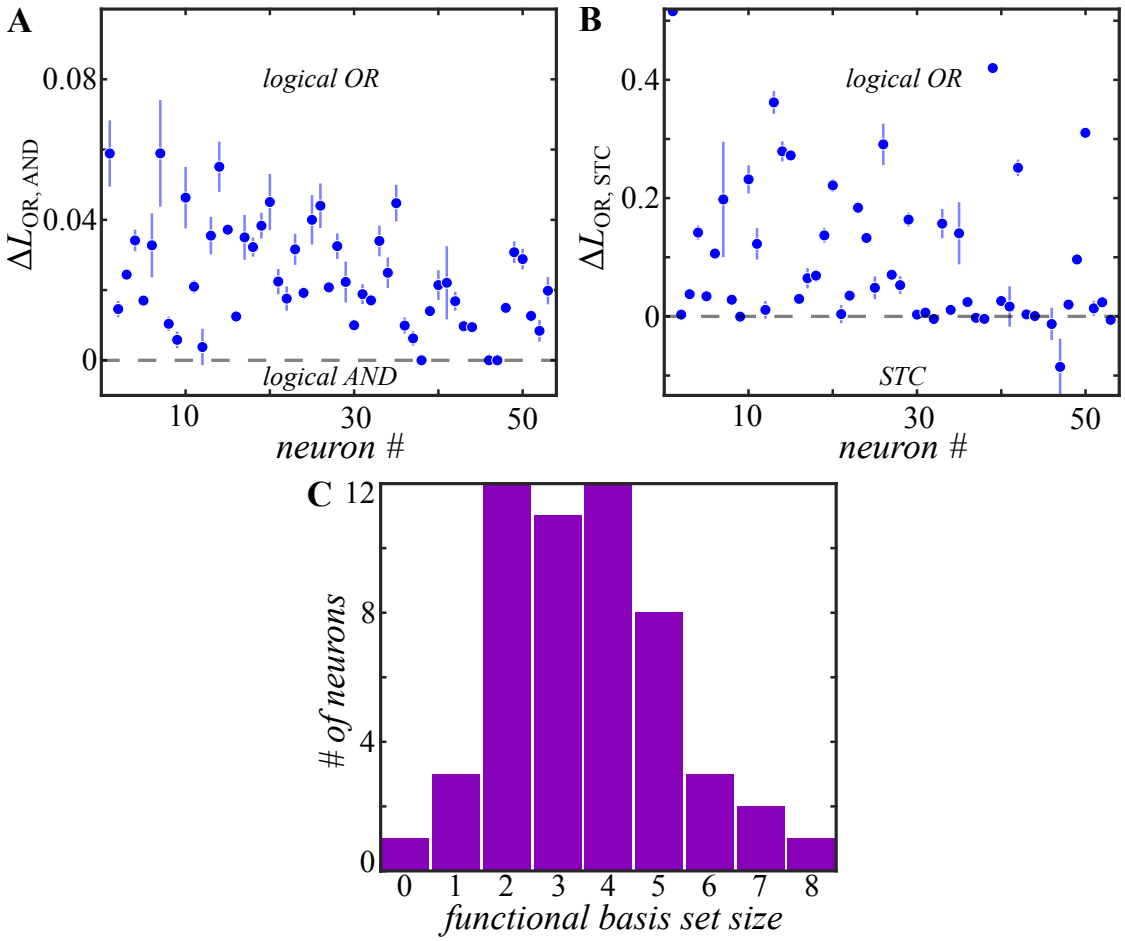

Figure 2.9: **Comparison of the prediction error between Boolean functional bases and empirical nonlinearities and the distribution of functional basis set sizes across RGCs.** (**A**) The relative prediction error (ΔL) between the logical OR and logical AND models evaluated on the cross-validation sets shows that logical OR is the better model across the data set. (**B**) The relative prediction error between the best found functional basis models (which happen to be all logical OR) and the empirical STC nonlinearity evaluated on the test sets shows that the functional basis models are a better fit for the vast majority of neurons and comparable on the rest. (**C**) The distribution of functional basis set sizes across the population for the logical OR models.

logical OR and logical AND models are equivalent. The logical OR models were then shown to have mostly better and sometimes comparable prediction error to the empirical STC nonlinearity (Eq 2.9) evaluated on the test sets (Fig 2.9**B**).

Unlike the STC-recovered components, each component of the logical OR functional basis in Fig 2.7 is spatially localized and monophasic. While each of the compo-

nents explore unique regions of stimulus space, they still overlap near the center of the image and have non-zero projection onto each other. The functional bases of the example neurons bear a strong resemblance to what was hypothesized, closely matching the profile of the literature-inspired RGC in Fig 2.5 where the functional input components were composed of bipolar cell receptive fields. The number of bipolar cell inputs that ought to be expected is known to vary with eccentricity (i.e. the angle of a point on the retina with respect to an axis passing through the fovea and the center of the lens), ganglion cell type, and species. Past physiological studies have concluded that the number of bipolar cell inputs should average somewhere between 2 and 10 [39, 53]. For example, a study by Soodak, Shapley, & Kaplan [53] found that ganglion cells of the domesticated cat averaged between 2 and 5 distinguishable bipolar cell inputs (dependent on the cell type), but noted that the number of bipolar cell inputs could be larger due to indistinguishability of any approximately redundant inputs. Though it is not exactly clear what the expected number of bipolar cell inputs should be for salamander RGCs, the number of components in the estimated functional bases were consistent with general expectations, albeit on the low end of the expected range, where the mean number of functional input components was 4 ± 2 ranging from zero (**neuron #45** had no significant components from the STC analysis) to eight components (see the histogram in Fig 2.9**C** for complete results).

To emphasize the difficulty in interpreting neural computations in the space defined by dimensionality reduction techniques such as STC, empirically binned nonlinearities that map stimulus projections onto pairs of basis vectors to predicted responses are plotted in Fig 2.10 for **neuron #3**. The plotted nonlinearities correspond to the marginal conditional probability distributions $P(y = 1|\mathbf{x}_i, \mathbf{x}_j)$ for $i \neq j$ where $\mathbf{x}_i = \boldsymbol{\omega}_i \cdot \mathbf{s}_t$ for projections onto the STC components and $\mathbf{x}_i = \mathbf{c}_i \cdot \mathbf{s}_t$ for projections onto the functional basis components. It is worth keeping in mind when inspecting Fig 2.10 that the conditional probabilities are not adjusted in any way to account for poorly sampled priors,

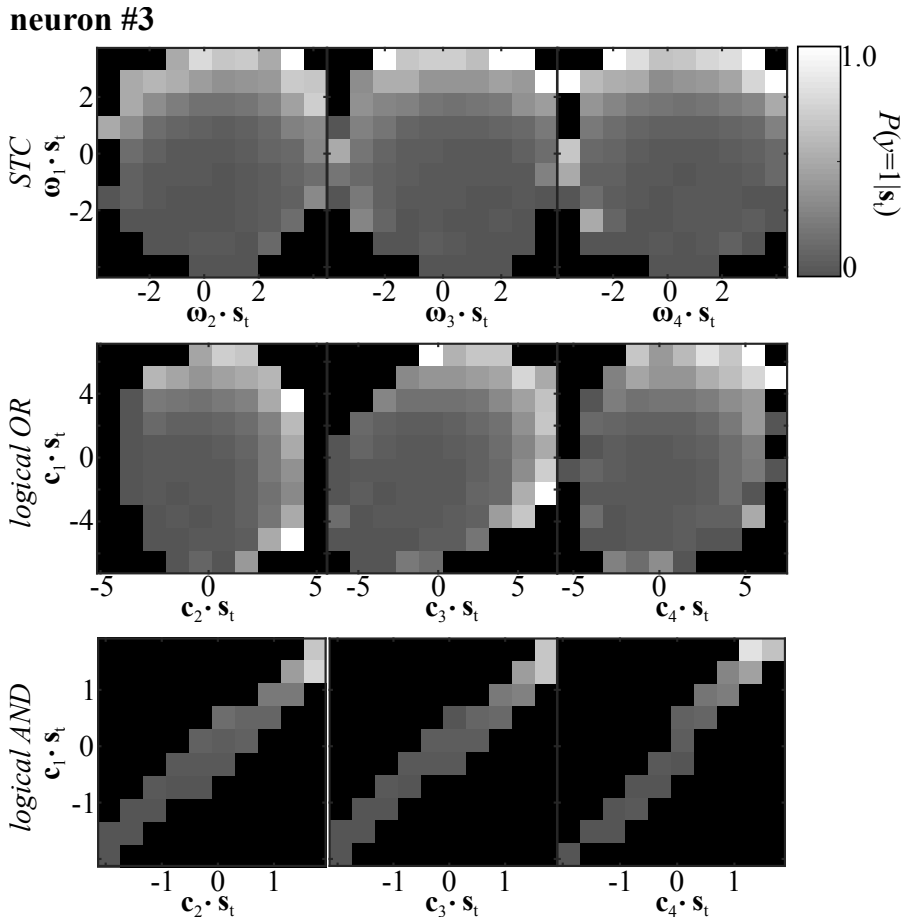

Figure 2.10: **Empirical nonlinearities of an example RGC projected into the receptive field and functional bases.** Empirical nonlinearities are mapped from the training set data along pairs of components from STC, logical OR, and logical AND models. Note that the black bins correspond to bins where no training samples were measured.

$P(\mathbf{x}_i, \mathbf{x}_j)$, on the periphery of the distribution and therefore random noise is more likely to have an influence at the edges (though one would not expect this noise to be correlated between neighboring bins).

Clearly, the computation of this neuron is more interpretable when projected into the logical OR functional basis since the probability of a spike increases approximately monotonically along any $\mathbf{c}_i$ component (Fig 2.10) and it can be readily assumed that this trend will continue for those pairs not pictured. The logical AND nonlinearity is consistent with what would be expected given that **neuron #3**'s logical AND functional

basis (Fig 2.7) is composed of four repeats of the STA. The computation is more complicated in STC space, displaying a crescent-like nonlinearity that has been observed in prior studies [9, 17, 54]. This crescent-like shape makes interpretation of the neural computation difficult because the threshold-crossing from a primarily silent domain to a spiking domain is not a monotonic function along the STC components. While the paired subspaces are computationally meaningful in the functional basis space, the plotted subspaces are arbitrary in the STC basis and the possibility that these subspaces need to be rotated to find a more interpretable view of the nonlinearity is a significant limitation when $r > 2$, unlike the simple examples in Fig 2.1. From Fig 2.1**A** one can also see the resemblance of the logical OR model to a crescent-like nonlinearity in STC space, which may serve as motivation to try logical OR models when crescent-like nonlinearities are observed elsewhere. This demonstrates how useful the functional basis method may be in the effort to better understand the functional neural circuitry.

2.5 Discussion and extensions

While dimensionality reduction techniques have been widely successful in characterizing receptive fields, these methods are ultimately designed to flexibly recover the receptive field independent of the underlying neural computation. In other words, a good dimensionality reduction technique is one that is able to reconstruct the receptive field without significant bias from the specific nonlinearity relating spikes to stimuli. For dimensionality reduction techniques, it does not matter whether the underlying neural computation is, say, a logical OR or logical AND. The receptive field is recoverable by the technique either way. The functional basis method, on the other hand, attempts to uncover the specific neural computation being performed by the neuron by proposing and testing hypotheses about the nonlinearity. Through this, the functional basis method

can discover biologically interpretable components to describe the functional inputs of neurons. By leveraging both dimensionality reduction and the functional basis method in conjunction, it is now possible to gain new insights into the neural circuitry behind the processing of sensory information.

The functional basis also has practical advantages over convolutional dimensionality reduction techniques when identifying invariant inputs. Convolutional techniques succeed when there is a single functional input that is transformed in some sense (e.g. translation or rotation), but will fail when there are any distinct functional inputs that cannot be simply related through invariance constraints. Since these convolutional methods are ultimately optimizing a single component, these techniques may miss any unique structure of the individual functional inputs like those of the bipolar cells in Fig 2.5 & 2.7. The functional basis method, on the other hand, is able to recover invariant or approximately invariant inputs without any explicit imposition of invariance on the model. Of course, whether to use the convolutional methods or the functional basis method may be dependent on knowledge of the problem since either of the methods require some assumption about either an explicit relationship between the input components or the input nonlinearities. It would seem, however, that the functional basis method may be a better choice in general for its flexibility.

Although this chapter has focused on functional bases derived from Boolean logical AND and logical OR nonlinearities, the functional basis method can be extended to include other reasonable hypotheses descriptive of neural computations. For instance, one may expect that some neurons may receive both excitatory and suppressive inputs where the neuron will spike provided any excitatory input when no suppressive inputs are sufficiently activated. If any suppressive inputs are activated, this would be analogous to inhibition of the neuron's spiking activity where the neuron will not spike even when an excitatory input in activated. Such a neuron could be modeled by a product of a logical

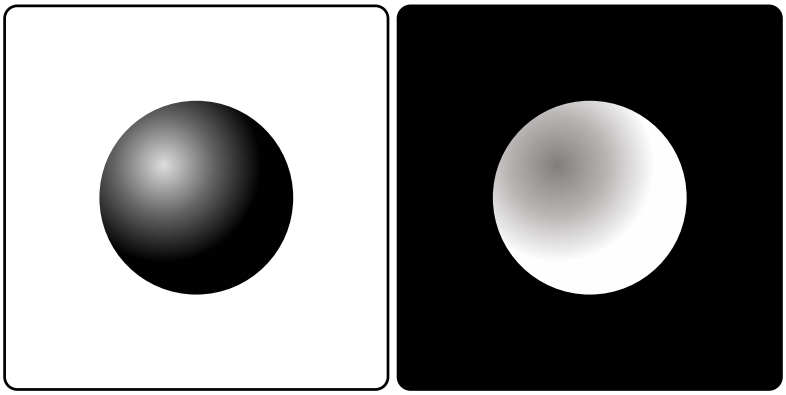

Figure 2.11: **Illustration of quadratic functional inputs.** In which picture can a sphere be identified?

AND and logical OR function,

$$P_{\text{MIX}}(y=1|\mathbf{s}) = P_{\text{OR}}(y=1|\mathbf{s})P_{\text{AND}}(y=0|\mathbf{s}) \tag{2.13}$$

where $P_{\text{AND}}(y=0|\mathbf{s}) = 1 - P_{\text{AND}}(y=1|\mathbf{s})$. Numerous other possibilities can be hypothesized as well, including many more Boolean expressions. The functional input nonlinearities, σ_k, can also be modified to take account of what is known about the sensory system. For instance, a quadratic input nonlinearity,

$$\sigma_k(\mathbf{c}_k^{\text{T}}\mathbf{s}) = \frac{1}{1+e^{-b_k-\zeta_1\mathbf{c}_k^{\text{T}}\mathbf{s}-\zeta_2(\mathbf{c}_k^{\text{T}}\mathbf{s})^2}}, \tag{2.14}$$

where ζ_1 and ζ_2 are weightings of a linear and quadratic term in the activation, could be used to model the functional inputs of a vision neuron that recognizes the presence of an object under different viewing conditions (e.g. Fig 2.11).

It is worth pointing out that there may be some cases where a functional basis may be impossible to define as a type of LN model. For example, radially symmetric nonlinearities where the spiking threshold is circular in some subspace of stimulus space has been proposed as a computational model for complex cells in V1 of the visual

cortex [17, 55]. Yet, imperfect radial symmetry could still allow for a functional basis to be recovered using LN type models. This scenario would appear to be likely given the imperfections observed in the studied RGCs where the bipolar inputs were only approximately invariant. Furthermore, given that the physiological inputs of a neuron are ultimately discrete, such radial symmetry may approximately emerge from functional input vectors starting from some central origin that are distributed at uniform angles across the unit circle with equal thresholds.

Having shown the functional basis approach to be of use in studying low-level vision neurons, the following chapters now expands from the study of early vision neurons to high-level auditory neurons. The study of high-level auditory neurons presents new challenges that cannot be simply addressed with the tools previously discussed in Chapters 1 & 2. Rather, it becomes necessary to return to the problem of dimensionality reduction and develop a new technique which is the subject of the following chapter.

Chapter 2 is based on material that was published in Kaardal, Fitzgerald, Berry, and Sharpee, *Neural Computation* (2013). The dissertation author was the primary investigator and author of this paper.

Chapter 3

Low-rank minimal models for multicomponent computations

Having shown that functional bases can be a valuable tool for identifying biologically interpretable functional inputs to early vision neurons, it is desirable to extend such analyses to gain insight into the neural computations performed in other sensory systems and in higher-level regions of the brain. Analyzing high-level neurons is particularly challenging for dimensionality reduction methods because the computations performed by such neurons are relatively sophisticated compared to those in early sensory regions and are often less responsive to noise stimuli. Intuitively, this unresponsiveness is caused by the unlikelihood that a stimulus will be drawn from a noise distribution with the specific structure necessary for a high-level neuron to recognize an object and therefore elicit a spike.

This alone makes methods such as the STA [2,4] and STC [4,20–25] inappropriate for studying high-level neurons without modifications (see Sections 1.3.1 & 1.3.2) such as stimulus whitening [56] or removing components from the stimulus space [57, 58]. However, both of the latter come with substantial biases. For instance, one may note that

performing STC on a set of whitened stimuli is equivalent to a linear transformation of the estimated receptive field from the stimulus space to a "whitened" space. The ambiguity that remains is whether the receptive field components that appear in this whitened space are meaningful in the stimulus space since a reversal of the linear transformation back into stimulus space negates the whitening procedure. Methods that remove components from the stimulus space have the potential pitfall of removing important structure leading to artifacts in the receptive field estimate (e.g. removing low absolute variance components of the stimulus space can lead to high spatial frequencies missing from the estimate).

Using information-theoretic approaches like MID [26] or MNE [28, 29] (see Sections 1.4 & 1.5) are attractive under such circumstances due to their resistance to bias when presented with correlated stimuli. Since high-level neural computations are expected to be sophisticated, exhibiting invariances such as those in Figs 2.2 & 2.3, one may expect the number of components that span the receptive field to be at least as many as those in early sensory regions like the retina, where the number of components was found to range up to eight (Section 2.4). If true, this would make first-order MID a poor choice due to the curse of dimensionality. Second-order MID [27, 29] may be tractable and has the advantage of being nonlinearity-independent, but the dependence of the results on binning an empirical nonlinearity can be a disadvantageous, especially when the amount of data available is limited. It was decided, therefore, that second-order MNE [28, 29] would be an appropriate choice for studying high-level sensory neurons. Although MNE is nonlinearity-dependent, unlike MID and STC, the nonlinearity is principled and limits bias as discussed in Section 1.5.

Recall that second-order MNE takes the form

$$P(y=1|\mathbf{s}) = \frac{1}{1+e^{-z(\mathbf{s})}}, \quad z(\mathbf{s}) = a + \mathbf{h}^{\mathrm{T}}\mathbf{s} + \mathbf{s}^{\mathrm{T}}\mathbf{J}\mathbf{s} \tag{3.1}$$

where $a \in \mathbb{R}$, $\mathbf{h} \in \mathbb{R}^D$, and $\mathbf{J} \in \mathbb{R}^{D \times D}$ are unknown weights determined by minimizing the negative log-likelihood. Altogether, the number of unique weights necessary to optimize is $1 + D + D(D+1)/2$ (note that the $D(D+1)/2$ term comes from the fact that an arbitrary antisymmetric matrix can be added to $\mathbf{J}$ without changing the output of the nonlinearity) while only $1 + D + rD$ weights are ultimately necessary to specify an r component receptive field. The dimensionality of second-order MNE (and second-order MID, for that matter) is problematic because the number of weights vastly exceeds the number of samples in typical data sets. Systems having more weights than samples have an elevated risk of overfitting because there is at least one predictive variable for each sample encountered. This sampling problem is amplified for correlated and especially natural stimuli where the relationships between stimuli can cause the stimulus distribution to be poorly explored, effectively reducing the sample size of the data set.

In this chapter, the second-order MNE model is extended with applications to high-level sensory neurons in mind by (i) transforming the second-order MNE optimization problem into a structured matrix factorization problem and (ii) applying a specific kind of regularization known as the *nuclear-norm* (or *trace-norm*) regularization [59–61] to de-noise the receptive field estimate. This model extension will be referred to as the *low-rank MNE model* and the optimization procedure the *low-rank MNE method*. Henceforth, the usual *second-order MNE* without the proposed extensions will be referred to as *full-rank MNE* to distinguish it from low-rank MNE. This chapter focuses on the theoretical development of the low-rank MNE method. Later on, Chapter 4 will cover practical elements of the method and applications to high-level auditory neurons where the method is shown to not only lead to a substantial improvement over prior methods, but is the only one of the tested models that recovers multicomponent receptive fields of high-level auditory neurons with statistical significance.

3.1 The low-rank MNE model

Since the purpose of the second-order MNE method is to ultimately recover components spanning the receptive field through factorization of the optimal matrix $\mathbf{J}$, a reasonable strategy to reduce the dimensionality of the weight space is to pre-factorize $\mathbf{J}$ prior to optimization into the bilinear outer-product, $\mathbf{J} = \mathbf{U}\mathbf{V}^{\mathrm{T}}$, of the two matrices, $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{D \times r}$. This bilinear factorization reduces the size of the weight space to $1 + D + 2rD$ and is especially helpful when $r \ll D$. The weights a, $\mathbf{h}$, $\mathbf{U}$, and $\mathbf{V}$ can then be found, once again, by minimizing the negative log-likelihood,

$$L(a, \mathbf{h}, \mathbf{U}, \mathbf{V}) = \frac{1}{N_{\mathrm{samp}}} \sum_{t=1}^{N_{\mathrm{samp}}} \left[y_t \log\left(P_t\right) + (1 - y_t) \log\left(1 - P_t\right) \right], \tag{3.2}$$

which is reproduced here to introduce a shorthand for the nonlinearity at sample t, $P_t = P(y = 1 | \mathbf{s}_t)$. Since real eigenvalues and eigenvectors are desired, the resulting optimal $\mathbf{J}$ can be symmetrized,

$$\mathbf{J}_{\mathrm{sym}} = \frac{1}{2}\left(\mathbf{J} + \mathbf{J}^{\mathrm{T}}\right), \tag{3.3}$$

without loss of generality and then $\mathbf{J}_{\mathrm{sym}}$ can be diagonalized to reveal components of the receptive field.

Following the above procedure, one may be perplexed when the rank of $\mathbf{J}_{\mathrm{sym}}$ is found to be greater than r. This inconsistency of expectations (i.e. $\mathrm{rank}\left(\mathbf{J}_{\mathrm{sym}}\right) = \mathrm{rank}\left(\mathbf{J}\right) \leq r$) versus the emergent reality is due to a subtle problem caused by the bilinear factorization of $\mathbf{J}$. When $\mathbf{J}$ is factorized into $\mathbf{U}$ and $\mathbf{V}$ and optimized, $\mathbf{J}$ and $\mathbf{J}^{\mathrm{T}}$ are not guaranteed to span the same range space unless $\boldsymbol{\mathcal{R}}\left(\mathbf{U}\right) = \boldsymbol{\mathcal{R}}\left(\mathbf{V}\right)$.

Proof. *Case 1: suppose, without loss of generality,* $\boldsymbol{\mathcal{R}}\left(\mathbf{U}\right) \supset \boldsymbol{\mathcal{R}}\left(\mathbf{V}\right)$. *Then the projection of* $\mathbf{U}$ *into the null space of* $\mathbf{V}$ *is non-zero,* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}\left(\mathbf{V}\right)\mathbf{U} \neq \mathbf{0}$, *where* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}\left(\mathbf{V}\right) = \mathbf{I} - \mathbf{V}\mathbf{V}^{\dagger}$

is the null space projection operator and $\mathbf{V}^{\dagger}$ *is the generalized inverse of* $\mathbf{V}$. *Case 2: conversely, if* $\boldsymbol{\mathcal{R}}(\mathbf{U}) = \boldsymbol{\mathcal{R}}(\mathbf{V})$, *then* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\,\mathbf{U} = \mathbf{0}$. *Since in both cases* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\,\mathbf{J}^{\mathrm{T}} = \mathbf{0}$ *and* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{U})\,\mathbf{J} = \mathbf{0}$ *but* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\,\mathbf{J} = \boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{U})\,\mathbf{J}^{\mathrm{T}} = \mathbf{0}$ *is only guaranteed to be true in case 2, then* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{J})\,\mathbf{J}^{\mathrm{T}} = \boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}\left(\mathbf{J}^{\mathrm{T}}\right)\mathbf{J} = \mathbf{0}$ *can only be guaranteed when* $\boldsymbol{\mathcal{R}}(\mathbf{J}) = \boldsymbol{\mathcal{R}}\left(\mathbf{J}^{\mathrm{T}}\right)$.

Consequently, $\boldsymbol{\mathcal{R}}\left(\mathbf{J}_{\mathrm{sym}}\right) \supseteq \boldsymbol{\mathcal{R}}(\mathbf{J})$ and $\mathbf{J}_{\mathrm{sym}}$ may take on $\mathrm{rank}\left(\mathbf{J}_{\mathrm{sym}}\right) \leq 2r$ even while $\mathrm{rank}(\mathbf{J}) \leq r$. This is clearly problematic because if $\mathbf{U}$ and $\mathbf{V}$ are only able to take on a maximum rank of r, there is not generally a sufficient number of weights present to adequately fit a rank $2r$ matrix. A remedy to this problem that will guarantee that $\mathrm{rank}\left(\mathbf{J}_{\mathrm{sym}}\right) = \mathrm{rank}(\mathbf{J})$ is to add the restriction that valid low-rank MNE models must be constrained to satisfy the range space constraint:

$$\boldsymbol{\mathcal{R}}(\mathbf{U}) = \boldsymbol{\mathcal{R}}(\mathbf{V}). \tag{3.4}$$

3.1.1 Constraint formulations

There are several possible ways to constrain $\mathbf{U}$ and $\mathbf{V}$ to satisfy Eq 3.4, a few of which are surveyed here along with a discussion of their respective merits. The first proposal is the set of quadratic constraints expressed by

$$\mathbf{U}\mathbf{U}^{\mathrm{T}} = \mathbf{V}\mathbf{V}^{\mathrm{T}} \tag{3.5}$$

which satisfy Eq 3.4 proven in the following.

Proof. *The quadratic constraints in Eq 3.5 satisfy Eq 3.4 because* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{U})\,\mathbf{U}\mathbf{U}^{\mathrm{T}} = \left[\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{U})\,\mathbf{U}\right]\mathbf{U}^{\mathrm{T}} = \mathbf{0}$ *and* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\,\mathbf{U}\mathbf{U}^{\mathrm{T}} = \boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\,\mathbf{V}\mathbf{V}^{\mathrm{T}} = \mathbf{0}$ *therefore* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\,\mathbf{U} = \mathbf{0}$ *which guarantees the condition in Eq 3.4 is satisfied (and vice versa when* $\mathbf{U}$ *is replaced with* $\mathbf{V}$*).*

The trouble with this formulation is that the there are $D(D+1)/2$ unique constraints

added to the problem that may be intractable to satisfy for large D (and may defeat possible goals like reducing memory usage or increasing computation speed over the full-rank MNE method, for instance) and are independent of r. A similar formulation is the bilinear constraints,

$$\mathbf{U}\mathbf{V}^{\mathrm{T}} = \mathbf{V}\mathbf{U}^{\mathrm{T}}, \tag{3.6}$$

that, when satisfied, ensure that $\mathbf{J}_{\mathrm{sym}} = \mathbf{J} = \mathbf{J}^{\mathrm{T}}$ and trivially satisfy Eq 3.4. This is only a marginal improvement over the quadratic constraints for large D, where there are $D(D-1)/2$ unique constraints.

A third constraint one might propose are the inner-product constraints,

$$\left(\frac{\mathbf{U}_{\bullet,k}\cdot\mathbf{V}_{\bullet,k}}{\left\|\mathbf{U}_{\bullet,k}\right\|_2\left\|\mathbf{V}_{\bullet,k}\right\|_2}\right)^2 = 1 \;\Rightarrow\; \left(\mathbf{U}_{\bullet,k}\cdot\mathbf{V}_{\bullet,k}\right)^2 = \left\|\mathbf{U}_{\bullet,k}\right\|_2^2\left\|\mathbf{V}_{\bullet,k}\right\|_2^2,\ \forall k, \tag{3.7}$$

where $\|\cdot\|_2$ is the ℓ_2-norm (i.e. $\|\mathbf{v}\|_2 = \sqrt{\sum_{i=1}^{D} v_i^2}$). This form of the constraints satisfies Eq 3.4 by forcing $\mathbf{U}_{\bullet,k}$ and $\mathbf{V}_{\bullet,k}$ to be parallel. Unlike the quadratic and bilinear constraints, the inner-product constraints are relatively compact, where the total number of constraints is r, and would work nicely to keep the problem size small. However, numerical experiments performed with an interior-point method found this form of the constraints has difficulty converging to a feasible point, which is likely due to the fact that it is quite nonlinear and nonlinear constraints can potentially lead to convergence at an infeasible stationary point. Of course, the square of the term on the left-hand-side could be replaced with an absolute value, but that would likely not improve the trickiness of the optimization because the absolute value would introduce non-differentiability. In fact, the quadratic (Eq 3.5) and bilinear constraints (Eq 3.6) may not be guaranteed to converge either since the Jacobians of the constraints with respect to the weights can

be rank-deficient which can result in the Karush-Kuhn-Tucker (KKT) conditions (also known as first-order necessary conditions) not being satisfied at a stationary point on the boundary of the feasible region of the weight space [41]. Since the condition in Eq 3.4 implies the necessity of equality constraints, any feasible stationary point must lie on the boundary of the feasible region.

There are several conditions in which the constraints can be guaranteed to satisfy the KKT conditions at a solution known as *constraint qualifications* (or *regularity conditions*) [41]. Unfortunately, all of the constraints proposed thus far do not satisfy any known constraint qualifications. This does not mean that the optimization will not converge with quadratic, bilinear, or inner-product constraints for a given problem, but rather that the optimization cannot be guaranteed to converge. The simplest constraints that guarantee convergence to a feasible stationary point are affine functions (satisfying the linear independence constraint qualification [41]). A set of linear equality constraints can be formulated by splitting $\mathbf{J}$ into the sum of a positive semidefinite (PSD) and negative semidefinite (NSD) matrix, $\mathbf{J} = \mathbf{J}^{(\mathrm{PSD})} + \mathbf{J}^{(\mathrm{NSD})}$. These two matrices each have their own assigned rank, r_{PSD} and r_{NSD}, and the rank of $\mathbf{J}$ is therefore $\mathrm{rank}\,(\mathbf{J}) \leq r_{\mathrm{PSD}} + r_{\mathrm{NSD}}$. Each of these matrices can then be factorized separately into $\mathbf{U}^{(\mathrm{PSD})}, \mathbf{V}^{(\mathrm{PSD})} \in \mathbb{R}^{D \times r_{\mathrm{PSD}}}$ and $\mathbf{U}^{(\mathrm{NSD})}, \mathbf{V}^{(\mathrm{NSD})} \in \mathbb{R}^{D \times r_{\mathrm{NSD}}}$. The linear constraints can be written in terms of the factorization matrices as

$$\mathbf{U}^{(\mathrm{PSD})} = \mathbf{V}^{(\mathrm{PSD})}, \quad \mathbf{U}^{(\mathrm{NSD})} = -\mathbf{V}^{(\mathrm{NSD})}. \tag{3.8}$$

These constraints keep the problem size small (the number of constraints is rD), unlike the quadratic and bilinear constraints, while having better convergence properties than the inner-product constraints and satisfy Eq 3.4. The disadvantage of the linear constraints is having to choose the hyperparameters r_{PSD} and r_{NSD}; but principled strategies are

available that make these hyperparameters less burdensome.

The final constraint proposed here is a relaxation of the linear equality constraints (Eq 3.8) by instead only insisting that $\mathbf{U}$ and $\mathbf{V}$ are equal up to a variable diagonal scaling matrix, $\mathbf{W} \in \mathbb{R}^{r \times r}$. These relaxed linear equality constraints can be written as

$$\mathbf{V} = \mathbf{U}\mathbf{W}. \tag{3.9}$$

This would avoid the necessity of choosing r_{PSD} and r_{NSD}. However, if these constraints are directly inserted into the model such that $z_t = a + \mathbf{h}^{\mathrm{T}}\mathbf{s}_t + \mathbf{s}_t^{\mathrm{T}}\mathbf{U}\mathbf{W}\mathbf{U}^{\mathrm{T}}\mathbf{s}_t$ (Eq 3.1), the optimization would be with respect to a third-order polynomial of the weights (since $\mathbf{W}$ would need to be optimized too) which may be a more difficult optimization. Furthermore, if the constraints are imposed via the method of Lagrange multipliers, the constraints would not only fail to satisfy a known constraint qualification, the objective function has no dependence on $\mathbf{W}$ and therefore the Hessian of the objective function will be perpetually singular.

3.1.2 Nuclear-norm regularization

In its present formulation, the low-rank MNE method attempts to find a low-rank compression of a solution to the full-rank MNE problem (without early stopping or any other form of regularization). Since the full-rank MNE method is likely to overfit, the low-rank compression is therefore prone to recovering components that are dominated by overfitting artifacts. These artifacts can be largely eliminated through a principled application of regularization.

In data-driven applications where the proposed model is intended to find the best solution given the data (as opposed to optimization problems that involve solving an exact function), the ability of a model to generalize to unseen data is often given

great importance. In other words, a good model should be able to predict responses to novel stimuli reasonably well. In such data-driven problems, solutions that lower the generalization error may take on a higher priority than globally optimal solutions on the training data. One technique to reduce generalization error is to penalize the objective function through some form of regularization. The type of regularization that is applied influences the model by biasing it towards some target structure. It is therefore important for one to be careful about what type of regularization is applied since a decrease in generalization error can come at the cost of increasingly biased models [62].

Some well known types of regularization penalties include the Frobenius-norm and LASSO regularization. These types of regularization are applied by adding a penalty function into the objective function; e.g.

$$f(\mathbf{x}) = L(\mathbf{x}) + \varepsilon \ell(\mathbf{x}) \tag{3.10}$$

where $L(\mathbf{x})$ is the negative log-likelihood of some weights, $\mathbf{x}$, $\ell(\mathbf{x}) \geq 0$ is a penalty function, $\varepsilon \geq 0$ is a regularization parameter that adjusts the strength of the penalty, and $f(\mathbf{x})$ is the regularized objective function. Revisiting the full-rank MNE model, one could attenuate the noise in $\mathbf{J}$ by applying Frobenius-norm regularization as such,

$$\ell_{\mathrm{F}}(\mathbf{J}) = \|\mathbf{J}\|_{\mathrm{F}}^2 = \sum_{i=1}^{D} \sum_{j=1}^{D} J_{i,j}^2, \tag{3.11}$$

which has the impact of decreasing the overall magnitude of the matrix elements taken in quadrature and consequently leads to a reduction in the absolute variance of $\mathbf{J}$'s eigenvalue spectrum. If one instead intends to search for a sparse version of $\mathbf{J}$, LASSO

regularization can be generalized from vectors to matrices by the function

$$\ell_1(\mathbf{J}) = \|\mathbf{J}\|_1 = \sum_{i=1}^{D}\sum_{j=1}^{D}|J_{i,j}|, \tag{3.12}$$

where the model is pressed to limit the number of non-zero elements in $\mathbf{J}$. Ideally, penalty functions such as these would be applied where there is justification for the imposed structure on $\mathbf{J}$. Under the present circumstances, there would appear to be little general motivation for employing these two forms of regularization. The only expectation about $\mathbf{J}$ is that its eigendecomposition will uncover at least a partial description of the receptive field. For the sake of interpretability, and because the low-rank MNE method is intended to be a dimensionality reduction technique, one may then hope or even insist that $r \ll D$.

Low-rank structure may be imposed on $\mathbf{J}$ via sparse regularization of its eigenvalue spectrum where the objective function can be penalized by counting the number of non-zero eigenvalues in $\mathbf{J}$'s eigenvalue spectrum. In a way, this is already being done by the bilinear factorization of $\mathbf{J}$ where $r_{\text{PSD}} + r_{\text{NSD}}$ would be the number of non-zero eigenvalues less any additional rank-deficiency of $\mathbf{J}$. It would, however, be desirable to have more flexibility in adjusting the rank of the model since the function that counts the number of non-zero eigenvalues is discontinuous. Furthermore, as will be shown later in Chapter 4, it would also be helpful to have a way to adjust the variance of the receptive field components such that the MNE models have better generalization ability and, importantly, to eliminate insignificant from the receptive field estimate. It will turn out that the addition of regularization is indispensable in practical applications.

This is where the nuclear-norm [59–61] can be of value because the nuclear-norm penalizes $\mathbf{J}$ such that its eigenvalue spectrum is sparse. Similar to how LASSO regularization uses the absolute value to encourage elementwise sparsity, the nuclear-

norm promotes sparsity of the eigenvalue spectrum via the function

$$\ell_*(\mathbf{J}) = \mathrm{Tr}\,|\mathbf{\Lambda}| = \sum_{k=1}^{D} |\Lambda_{k,k}| \tag{3.13}$$

where $\mathrm{Tr}(\cdot)$ is the trace of a matrix. This incarnation of the nuclear-norm may give the reader pause. After all, this function implies the optimization would need to be penalized by a function of the eigendecomposition of $\mathbf{J}$ for which there is not an analytic expression of the form $\mathbf{\Lambda}(\mathbf{J})$. Of course, had $\mathbf{J}$ been guaranteed to be positive semidefinite, then a simple workaround would be to take $\mathrm{Tr}(\mathbf{J})$ (or $-\mathrm{Tr}(\mathbf{J})$ if $\mathbf{J}$ is negative semidefinite). This is unfortunately not generally quite as straight-forward for low-rank MNE because $\mathbf{J}$ cannot be assumed to be a semidefinite matrix of either sort. However, splitting $\mathbf{J}$ into the two semidefinite matrices from above is a resolution to this problem where the nuclear-norm can be simply written as

$$\ell_*(\mathbf{J}) = \mathrm{Tr}\left(\mathbf{J}^{(\mathrm{PSD})} - \mathbf{J}^{(\mathrm{NSD})}\right). \tag{3.14}$$

For the full-rank MNE method, this would require the addition of semidefiniteness constraints, transforming the optimization from a nonlinear program to a nonlinear semidefinite program that may not scale well to large D problems. This regularization is quite tractable for the linearly constrained low-rank MNE problems, however. It follows from Eq 3.14 that

$$\begin{aligned}
\ell_*(\mathbf{U},\mathbf{V}) =& \mathrm{Tr}\left(\mathbf{U}^{(\mathrm{PSD})}\mathbf{V}^{(\mathrm{PSD})^{\mathrm{T}}}\right) - \mathrm{Tr}\left(\mathbf{U}^{(\mathrm{NSD})}\mathbf{V}^{(\mathrm{NSD})^{\mathrm{T}}}\right) \\
=& \frac{1}{2}\mathrm{Tr}\left(\mathbf{U}^{(\mathrm{PSD})}\mathbf{U}^{(\mathrm{PSD})^{\mathrm{T}}} + \mathbf{V}^{(\mathrm{PSD})}\mathbf{V}^{(\mathrm{PSD})^{\mathrm{T}}}\right) \\
&+ \frac{1}{2}\mathrm{Tr}\left(\mathbf{U}^{(\mathrm{NSD})}\mathbf{U}^{(\mathrm{NSD})^{\mathrm{T}}} + \mathbf{V}^{(\mathrm{NSD})}\mathbf{V}^{(\mathrm{NSD})^{\mathrm{T}}}\right) \\
\Rightarrow \ell_*(\mathbf{U},\mathbf{V}) =& \frac{1}{2}\left(\|\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{V}\|_{\mathrm{F}}^2\right)
\end{aligned} \tag{3.15}$$

reached by substitution with the linear equality constraints (Eq 3.8). Apparently, nuclear-norm regularization reduces to Frobenius-norm regularization with respect to $\mathbf{U}$ and $\mathbf{V}$ in this case. Since one may entertain the possibility of using constraints alternative to the linear equality constraints, it can also be shown that the nuclear-norm will follow the same form more generally by invoking a *semidefinite embedding* of $\mathbf{J}$ within a larger positive semidefinite matrix [59–61, 63]. If a matrix $\mathbf{Q}$ is defined as

$$\mathbf{Q}^{\mathrm{T}} = \begin{bmatrix} \mathbf{U}^{\mathrm{T}}, & \mathbf{V}^{\mathrm{T}} \end{bmatrix}, \tag{3.16}$$

then the outer-product is the positive semidefinite matrix

$$\boldsymbol{\mathcal{X}} = \mathbf{Q}\mathbf{Q}^{\mathrm{T}} = \begin{bmatrix} \mathbf{U}\mathbf{U}^{\mathrm{T}}, & \mathbf{J} \\ \mathbf{J}^{\mathrm{T}}, & \mathbf{V}\mathbf{V}^{\mathrm{T}} \end{bmatrix} \tag{3.17}$$

where $\mathbf{J}$ is embedded on the off-diagonal blocks. If one takes the trace of $\boldsymbol{\mathcal{X}}$, the same result from Eq 3.15 is obtained:

$$\frac{1}{2}\mathrm{Tr}\left(\boldsymbol{\mathcal{X}}\right) = \frac{1}{2}\mathrm{Tr}\left(\mathbf{U}\mathbf{U}^{\mathrm{T}}\right) + \frac{1}{2}\mathrm{Tr}\left(\mathbf{V}\mathbf{V}^{\mathrm{T}}\right) = \frac{1}{2}\left(\|\mathbf{U}\|_{\mathrm{F}}^2 + \|\mathbf{V}\|_{\mathrm{F}}^2\right) \tag{3.18}$$

$$\equiv \frac{1}{2}\|\mathbf{Q}\|_{\mathrm{F}}^2. \tag{3.19}$$

Indeed, it can then be proven that regularizing over this semidefinite embedding is an effective proxy for the nuclear-norm even in the absence of the linear equality constraints.

Proof. *Since* $\mathbf{U}$ *spans the same range space as* $\mathbf{U}\mathbf{U}^{\mathrm{T}}$ *and* $\mathbf{V}$ *spans the same range space as* $\mathbf{V}\mathbf{V}^{\mathrm{T}}$*, it can be shown that regularizing over* $\mathrm{Tr}(\boldsymbol{\mathcal{X}})$ *can be used in place of Eq 3.13. Since* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{U})\mathbf{J} = \mathbf{0}$ *and* $\boldsymbol{\mathcal{P}}_{\boldsymbol{\mathcal{N}}}(\mathbf{V})\mathbf{J}^{\mathrm{T}} = \mathbf{0}$*, this means that the left eigenvectors of* $\mathbf{J}$*,* $\boldsymbol{\Omega}_{\mathrm{L}}$ *are a subset of* $\mathbf{U}$ *and the right eigenvectors of* $\mathbf{J}$*,* $\boldsymbol{\Omega}_{\mathrm{R}}$*, are a subset of* $\mathbf{V}$*. This leads to the relation* $\mathrm{rank}(\mathbf{J}) \leq \min(\mathrm{rank}(\mathbf{U}), \mathrm{rank}(\mathbf{V}))$ *and thus it can be concluded that penalizing*

Tr($\boldsymbol{\mathcal{X}}$) *is an effective surrogate for the nuclear-norm penalty.*

An atypical modification to the nuclear-norm regularization procedure is used to gain further improvements on the estimated low-rank MNE models. Instead of using a single regularization parameter to adjust the strength of the nuclear-norm penalty, multiple regularization parameters are defined with respect to each of the paired columns of **U** and **V**. Explicitly, the penalized objective function is (in terms of **Q** from Eq 3.16)

$$f(a,\mathbf{h},\mathbf{Q}) = L(a,\mathbf{h},\mathbf{Q}) + \sum_{k=1}^{r} \varepsilon_k \ell_*(\mathbf{Q}_{\bullet,k}) \tag{3.20}$$

$$= L(a,\mathbf{h},\mathbf{Q}) + \frac{1}{2}\sum_{k=1}^{r} \varepsilon_k \left\|\mathbf{Q}_{\bullet,k}\right\|_{\mathrm{F}}^2 \tag{3.21}$$

where $\varepsilon_k \geq 0$ are the regularization parameters. When using a single regularization parameter, there is a substantial risk that the amount of regularization necessary to eliminate (usually low absolute variance) fictitious components from noisy data may be at the expense of substantial degradation of the higher absolute variance components. Using multiple regularization parameters circumvents this problem but does so by introducing a larger hyperparameter search space.

For a short digression about a Bayesian interpretation of the nuclear-norm, see Appendix A.

3.2 The optimization problem

Putting all the pieces together from the prior section, the low-rank MNE problem is a nonlinear program of the form

$$\begin{aligned} \min_{a,\mathbf{h},\mathbf{Q}} f(a,\mathbf{h},\mathbf{Q}) &= \min_{a,\mathbf{h},\mathbf{Q}} L(a,\mathbf{h},\mathbf{Q}) + \sum_{k=1}^{r} \varepsilon_k \ell_* \left(\mathbf{Q}_{\bullet,k}\right) \\ \text{subject to } \forall k,\ \mathbf{A}_{k,k}\mathbf{Q}_{\bullet,k} &= \mathbf{0} \end{aligned} \tag{3.22}$$

where $\left\{\mathbf{A}_{k,k} \in \mathbb{R}^{D\times 2D} | k \in \{1,\cdots,r\}\right\}$ are a set of matrices defined as

$$\mathbf{A}_{k,k} = \begin{bmatrix} \mathbf{I}, & \pi_k \mathbf{I} \end{bmatrix} \tag{3.23}$$

that impose the linear equality constraints in Eq 3.8 and $\mathbf{Q}$ was defined in Eq 3.16. The parameter $\pi_k \in \{-1,1\}$ determines whether the kth component of $\mathbf{Q}$ is constrained by the positive semidefinite constraints ($\pi_k = -1$) or the negative semidefinite constraints ($\pi_k = 1$). The minimization problem in Eq 3.22 will be called the low-rank MNE problem from here onward. The goal is to find the weights a^*, $\mathbf{h}^*$, and $\mathbf{Q}^*$ that are a feasible local minimizer of the low-rank MNE problem.

There are two approaches that may be used to find a feasible local minimizer. The first way, as touched on before, is to directly substitute $\mathbf{V} = -\pi_k \mathbf{U}$ into f transforming the problem into an unconstrained optimization problem. The second way is to impose the constraints by forming the Lagrangian,

$$\mathcal{L}(a,\mathbf{h},\mathbf{Q};\mathbf{\Psi}) = f(a,\mathbf{h},\mathbf{Q}) - \sum_{k=1}^{r} \mathbf{\Psi}_{\bullet,k}^{\mathrm{T}} \mathbf{A}_{k,k}\mathbf{Q}_{\bullet,k}, \tag{3.24}$$

where $\mathbf{\Psi} \in \mathbb{R}^{D\times r}$ is a matrix of Lagrange multipliers. The results are the same regardless of which approach is used. For the ensuing theoretical discussion, the Lagrangian

approach will be studied since it is easier to transform the analysis to other constraints (if one so chooses to do so).

3.2.1 Optimality conditions

To certify that a set of weights is a feasible local minimizer of the low-rank MNE problem, the first-order necessary conditions (the KKT conditions) and the second-order sufficient conditions must be satisfied. For convex problems like the full-rank MNE problem, it is both necessary and sufficient that the KKT conditions be satisfied since it is a convex problem. The low-rank MNE problem, on the other hand, is shown to be nonconvex where the second-order conditions are needed for sufficiency. These optimality conditions are derived here because they will become useful later. In the following discussion, it will be helpful to define a weight vector,

$$\mathbf{x}^{\mathrm{T}} = \left[\begin{array}{cccc} a, & \mathbf{h}^{\mathrm{T}}, & \mathbf{Q}_{\bullet,1}^{\mathrm{T}}, & \cdots, \quad \mathbf{Q}_{\bullet,r}^{\mathrm{T}} \end{array} \right], \tag{3.25}$$

and a quadratic feature matrix,

$$\mathbf{D}_t = \left[\begin{array}{cc} \mathbf{0}, & \mathbf{s}_t\mathbf{s}_t^{\mathrm{T}} \\ \mathbf{s}_t\mathbf{s}_t^{\mathrm{T}}, & \mathbf{0} \end{array} \right] \tag{3.26}$$

for the tth sample of the stimulus space. Gradients with respect to subsets of the weights are represented by ∇_a, $\boldsymbol{\nabla}_{\mathbf{h}}$, $\boldsymbol{\nabla}_{\mathbf{Q}_{\bullet,k}}$, etc., having the same shape and ordering as the weights represented in the subscript.

The KKT conditions are summarized in Prop 3.1.

Proposition 3.1. *Karush-Kuhn-Tucker (KKT) conditions: the first-order necessary conditions for a feasible local minimum of $f(\mathbf{x})$ are that the gradient of the Lagrangian with*

respect to the weights should be zero,

$$\nabla_a \mathcal{L} = \frac{1}{N_{\text{samp}}} \sum_t (P_t - y_t) = 0 \tag{3.27}$$

$$\boldsymbol{\nabla}_{\mathbf{h}} \mathcal{L} = \frac{1}{N_{\text{samp}}} \sum_t (P_t - y_t)\mathbf{s}_t = \mathbf{0} \tag{3.28}$$

$$\boldsymbol{\nabla}_{\mathbf{Q}_{\bullet,k}} \mathcal{L} = \left[\frac{1}{N_{\text{samp}}} \sum_t (P_t - y_t)\mathbf{D}_t + \varepsilon_k \mathbf{I}\right] \mathbf{Q}_{\bullet,k} - \mathbf{A}_{k,k}^{\mathrm{T}} \boldsymbol{\Psi}_{\bullet,k} = \mathbf{0},\ \forall k, \tag{3.29}$$

and the solution must satisfy the equality constraints,

$$\mathbf{A}_{k,k}\mathbf{Q}_{\bullet,k} = \mathbf{0},\ \forall k. \tag{3.30}$$

As was already noted before, the low-rank MNE problem is guaranteed to satisfy the KKT conditions at a feasible local minimizer because the constraints satisfy the linear independence constraint qualifications [41]. Since the (penalized) objective function, f, of the low-rank MNE problem is bounded from below, there is also guaranteed to be at least one feasible stationary point that satisfies the KKT conditions. However, it is not possible from the KKT conditions alone to determine whether a stationary point is a local minimum, local maximum, or saddle point; hence why the KKT conditions are, in general, necessary but insufficient to certify the type of local optima.

Notably, when the KKT conditions are satisfied $\mathbf{Q}_{\bullet,k}$ must be complementary to Eq 3.29 because $\left(\mathbf{Q}_{\bullet,k}^{\mathrm{T}} \mathbf{A}_{k,k}^{\mathrm{T}}\right) \boldsymbol{\Psi}_{\bullet,k} = \mathbf{0}$ (Eq 3.22) and therefore

$$\begin{aligned} &\mathbf{Q}_{\bullet,k}^{\mathrm{T}} \left[\frac{1}{N_{\text{samp}}} \sum_t (P_t - y_t)\mathbf{D}_t + \varepsilon_k \mathbf{I}\right] \mathbf{Q}_{\bullet,k} = 0 \\ &\Rightarrow \left[\frac{1}{N_{\text{samp}}} \sum_t (P_t - y_t)\mathbf{D}_t + \varepsilon_k \mathbf{I}\right] \mathbf{Q}_{\bullet,k} = \mathbf{0} \end{aligned} \tag{3.31}$$

where the bottom line follows because the bracketed term is a symmetric matrix. This

further implies that $\boldsymbol{\Psi}_{\bullet,k}$ is complementary to the column-space of $\mathbf{A}_{k,k}$. These results will become important in the forthcoming discussion of the locally and globally optimal regularization domains.

To certify that a stationary point is a local minimum requires an assessment of the second-order sufficient conditions in Prop 3.2.

Proposition 3.2. *Second-order sufficient conditions: if the KKT conditions in Prop 3.1 are satisfied at point* $\mathbf{x}^*$ *in weight space and*

$$\boldsymbol{\mathcal{S}} = \boldsymbol{\mathcal{N}}(\mathbf{A})\, \nabla^2_{\mathbf{xx}} f|_{\mathbf{x}^*} \boldsymbol{\mathcal{N}}(\mathbf{A})^{\mathrm{T}} \geq 0 \tag{3.32}$$

where $\mathbf{A} \in \mathbb{R}^{rD\times(1+D+2rD)}$ *is the Jacobian matrix of the constraints and* $\boldsymbol{\mathcal{N}}(\mathbf{A})$ *returns the null space of* $\mathbf{A}$*, then* $\mathbf{x}^*$ *is a feasible local minimum of* f*. Note that the comparison of the matrix* $\boldsymbol{\mathcal{S}}$ *to a scalar indicates the definiteness of the matrix (e.g.* $\boldsymbol{\mathcal{S}} \geq 0$ *means* $\boldsymbol{\mathcal{S}}$ *is positive semidefinite).*

The intuition for Prop 3.2 is that the Hessian of $f(\mathbf{x})$ must be positive semidefinite for displacements of $\mathbf{x}$ along any feasible directions arbitrarily close to a stationary point, $\mathbf{x}^*$, in order for the stationary point to be a local minimum.

In terms of z_t (Eq 3.1), the Hessian is

$$\begin{aligned}\nabla^2_{\mathbf{xx}} f = &\overbrace{\frac{1}{N_{\text{samp}}}\sum_t P_t(1-P_t)\nabla_{\mathbf{x}} z_t \left(\nabla_{\mathbf{x}} z_t\right)^{\mathrm{T}}}^{\text{positive semidefinite, } \mathbf{RR}^{\mathrm{T}}} \\ &+ \underbrace{\frac{1}{N_{\text{samp}}}\sum_t (P_t - y_t)\nabla^2_{\mathbf{xx}} z_t}_{\text{indefinite, } \mathbf{M}} + \underbrace{\sum_{k=1}^{r} \varepsilon_k \nabla^2_{\mathbf{xx}} \ell_*(\mathbf{Q}_{\bullet,k})}_{\text{positive semidefinite, } \mathbf{E}}\end{aligned} \tag{3.33}$$

where the Hessian operator is $\boldsymbol{\nabla}_{\mathbf{xx}} = \boldsymbol{\nabla}_{\mathbf{x}} \boldsymbol{\nabla}_{\mathbf{x}}^{\mathrm{T}}$,

$$\boldsymbol{\nabla}^2_{\mathbf{xx}} z_t = \begin{bmatrix} \mathbf{0}, & \mathbf{0}, & \cdots, & \mathbf{0} \\ \mathbf{0}, & \mathbf{D}_t, & \ddots, & \vdots \\ \vdots, & \ddots, & \ddots, & \mathbf{0} \\ \mathbf{0}, & \cdots, & \mathbf{0}, & \mathbf{D}_t \end{bmatrix}, \quad \text{and} \quad \mathbf{E} = \begin{bmatrix} \mathbf{0}, & \mathbf{0}, & \cdots, & \mathbf{0} \\ \mathbf{0}, & \varepsilon_1 \mathbf{I}, & \ddots, & \vdots \\ \vdots, & \ddots, & \ddots, & \mathbf{0} \\ \mathbf{0}, & \cdots, & \mathbf{0}, & \varepsilon_r \mathbf{I} \end{bmatrix}. \tag{3.34}$$

The matrix $\boldsymbol{\nabla}^2_{\mathbf{xx}} z_t$ is a symmetric indefinite block diagonal matrix and $\mathbf{E}$ is a strictly positive semidefinite matrix ("strictly" is used here to indicate that the matrix is guaranteed to have eigenvalues equal to zero, thus excluding the possibility of positive definiteness). Because of the indefinite matrix $\mathbf{M}$ in Eq 3.33, the objective function f is nonconvex. Application of the linear equality constraints does not change the nonconvexity of the problem and it can be concluded that the low-rank MNE problem is nonconvex.

The nonconvexity of the low-rank MNE problem invites the possibility that the objective function defined on the feasible weight space has saddle points and local minima, many of which may be suboptimal. However, nonconvexity alone does not prove the existence of suboptimal local minima. An empirical approach was used to investigate the existence of suboptimal local minima by generating random optimization problems. These random problems were generated by drawing $N_{\text{samp}} = 100$ stimulus samples of dimensionality $D = 2$ from a normal distribution and drawing random weight vectors, $\mathbf{x}$, from a normal distribution with variance 0.1. The ground truth model defined by $\mathbf{x}$ had rank $r_{\text{opt}} = 2$. Low-rank MNE problems with $r = 1$ models were optimized 10 times with random initializations using an interior-point algorithm (see Section 3.3) with fixed sign for π_1 (Eq 3.22 & 3.23) across trials and $\varepsilon_1 = 0$. If at least two models in these ten trials differed in negative log-likelihood by $1 \cdot 10^{-4}$, the problem was stored for later visual inspection to ensure that this difference was not due to imprecise fitting. This visual inspection was performed by plotting f in $\mathbf{U}$ space and observing the existence

of spatially separated minima (apart from those that are due to symmetry across the origin). Indeed, counterexamples were found that nullify the hypothesis that the low-rank MNE problem does not have suboptimal local minima; one of which is shown in Fig 3.1. However, few counterexamples were witnessed, requiring around $\sim 10^3$ random problems be generated to find a counterexample with the vast majority of random problems having only global minima. It was observed that suboptimal minima appeared to only occur when there was some approximate degeneracy in the length of $\mathbf{Q}_{\bullet,1}$ for each solution and, when suboptimal local minima did exist, there were only up to r_{opt} unique local minima. For random problems where $\mathbf{x}$ was drawn from distributions with larger variance and with $D > 2$ and $r_{\text{opt}} > 2$, counterexamples were similarly difficult to find (in fact, counterexamples were harder to find at higher variances). It appears, then, that suboptimal local minima may be rare at the very least for the most predictive components of the low-rank MNE models.

3.2.2 Locally and globally optimal regularization domains

Since the nuclear-norm penalty functions, ℓ_*, are convex functions with respect to the weights, it is possible to manipulate the regularization parameters, $\{\varepsilon_k\}$, such that the eigenvalues of the positive semidefinite matrix $\mathbf{E}$ overwhelm the contributions of negative variance from the indefinite matrix $\mathbf{M}$ in the Hessian. By doing so, it can be shown that there is a domain for which solutions of the low-rank MNE problem are guaranteed to be globally optimal. First, one may observe that there are some values of $\{\varepsilon_k\}$ such that the Hessian of f becomes positive semidefinite for a given $\mathbf{x}$.

Proposition 3.3. *Given an arbitrary weight vector* $\mathbf{x}$*, there is a threshold value of* ε_k *that satisfies* $\varepsilon_k \leq \lambda_{\max}(\mathbf{M})$ *(where* $\lambda_{\max}(\mathbf{M})$ *is the largest eigenvalue of* $\mathbf{M}$*) such that if all* ε_k *are greater than or equal to this threshold then* $\nabla^2_{\mathbf{xx}} f$ *evaluated at* $\mathbf{x}$ *is guaranteed to be positive semidefinite.*

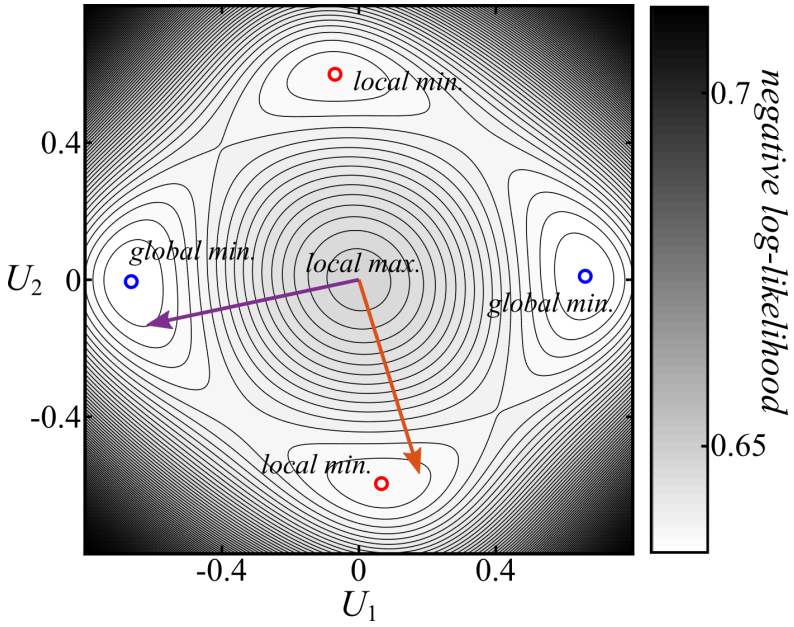

Figure 3.1: **It is possible for suboptimal local minima to exist in the low-rank MNE problem.** The negative log-likelihood, L, is plotted as a function of dimensions U_1 and U_2 (where a and $\mathbf{h}$ are computed for each U_1 and U_2) of a randomly generated problem constructed from ground truth weights: $a = -0.6680$, $\mathbf{h}^{\mathrm{T}} = [0.5251, -0.4768]$, $\mathbf{U}_{\bullet,1}^{\mathrm{T}} = \mathbf{V}_{\bullet,1}^{\mathrm{T}} = [0.1722, -0.5515]$, and $\mathbf{U}_{\bullet,2}^{\mathrm{T}} = \mathbf{V}_{\bullet,2}^{\mathrm{T}} = [-0.6045, -0.1284]$. Note that the components of the ground truth matrix $\mathbf{U}$ are plotted as arrows. Suboptimal local minima (north-south red circles) and global minima (east-west blue circles) are marked where minima symmetric across the origin are equivalent solutions.

Proof. *Assuming that* $\mathbf{x}$ *is fixed, the characteristic polynomial of* $\mathbf{M}+\mathbf{E}$ *is*

$$\det\left(\mathbf{M}+\mathbf{E}-\lambda\mathbf{I}\right) = -\lambda\prod_{k=1}^{r}\det\left(\left[\lambda-\varepsilon_k\right]^2\mathbf{I}-\left[\nabla_{\mathbf{J}}L\right]^2\right) = 0 \tag{3.35}$$

obtained through the block LDU decomposition where λ *is an eigenvalue of the Hessian and* $\nabla_{\mathbf{J}}L \in \mathbb{R}^{D\times D}$ *is a gradient matrix of the negative log-likelihood with respect to* $\mathbf{J}$*. For any* λ *that is a solution to Eq 3.35, there is a corresponding eigenvalue,* λ'*, symmetric across* ε_k *that is also a solution:* $\lambda' = \varepsilon_k \pm |\lambda - \varepsilon_k|$*. Since all* $\varepsilon_k \geq 0$*, the minimum possible eigenvalue of* $\mathbf{M}+\mathbf{E}$ *at* $\mathbf{x}$ *appears when* $\varepsilon_k = 0$ *and is therefore equal to the minimum eigenvalue of* $\mathbf{M}$*,* $\lambda_{\min}(\mathbf{M})$*. Due to symmetry of the eigenvalue spectrum,* $\lambda_{\min}(\mathbf{M}) = -\lambda_{\max}(\mathbf{M})$*. Therefore, if* $\varepsilon_k \geq \lambda_{\max}(\mathbf{M})$ *for all* k *at* $\mathbf{x}$*, the Hessian is then guaranteed*

to be positive semidefinite at $\mathbf{x}$. *Because* $\lambda_{\min}\left(\nabla^2_{\mathbf{xx}} f\right) \geq \lambda_{\min}\left(\mathbf{R}\mathbf{R}^{\mathrm{T}}\right) + \lambda_{\min}\left(\mathbf{M}+\mathbf{E}\right)$, $\varepsilon_k = \lambda_{\max}\left(\mathbf{M}\right)$ *is an upper bound on* ε_k *at and above which the Hessian can be guaranteed to be positive semidefinite.*

Now, suppose a (nonlinear) semidefinite program (SDP) is designed equivalent to the low-rank MNE problem [64]:

$$\begin{aligned} &\min_{a,\mathbf{h},\{\boldsymbol{\mathcal{X}}_k\}} f\left(a,\mathbf{h},\{\boldsymbol{\mathcal{X}}_k\}\right) \\ &\qquad \text{subject to } \forall k, \begin{cases} \mathbf{J}_k - \mathbf{J}_k^{\mathrm{T}} = \mathbf{0}, \\ \operatorname{rank}\left(\boldsymbol{\mathcal{X}}_k\right) \leq 1, \\ \boldsymbol{\mathcal{X}}_k \geq 0 \end{cases} \end{aligned} \tag{3.36}$$

where f is minimized with respect to a, $\mathbf{h}$, and a set of matrices, $\left\{\boldsymbol{\mathcal{X}}_k \in \mathbb{R}^{2D\times 2D} | k \in \{1,\ldots,r\}\right\}$. When the positive semidefiniteness constraint is imposed, each $\boldsymbol{\mathcal{X}}_k$ is equivalent to a semidefinite embedding matrix (Eq 3.17), $\boldsymbol{\mathcal{X}}_k = \mathbf{Q}^{(k)}\mathbf{Q}^{(k)\mathrm{T}}$ where $\mathbf{Q}^{(k)} \in \mathbb{R}^{D\times D}$ are the set of factorization matrices associated with $\boldsymbol{\mathcal{X}}_k$. When the rank constraint is imposed on the SDP, then $\boldsymbol{\mathcal{X}}_k = \mathbf{Q}_{\bullet,k}\mathbf{Q}_{\bullet,k}^{\mathrm{T}}$. The nuclear-norm regularization penalty function is then simply $\ell_*(\boldsymbol{\mathcal{X}}_k) = \frac{1}{2}\operatorname{Tr}\left(\boldsymbol{\mathcal{X}}_k\right)$ for the kth embedding matrix.

In its present form, the SDP is nonconvex due to the rank constraint where any iteration $\boldsymbol{\mathcal{X}}_k^{(n+1)} = \boldsymbol{\mathcal{X}}_k^{(n)} + \alpha\Delta\boldsymbol{\mathcal{X}}_k$ for $\alpha > 0$ from point $\boldsymbol{\mathcal{X}}_k^{(n)}$ towards point $\boldsymbol{\mathcal{X}}_k^{(n+1)}$ is infeasible unless the update $\alpha\Delta\boldsymbol{\mathcal{X}}_k$ is such that (i) $\boldsymbol{\mathcal{R}}\left(\boldsymbol{\mathcal{X}}_k^{(n+1)}\right) = \boldsymbol{\mathcal{R}}\left(\boldsymbol{\mathcal{X}}_k^{(n)}\right)$ and $\operatorname{rank}\left(\boldsymbol{\mathcal{X}}_k^{(n)}\right) = 1$ or (ii) $\boldsymbol{\mathcal{R}}\left(\boldsymbol{\mathcal{X}}_k^{(n+1)}\right) \neq \boldsymbol{\mathcal{R}}\left(\boldsymbol{\mathcal{X}}_k^{(n)}\right)$, $\operatorname{rank}\left(\boldsymbol{\mathcal{X}}_k^{(n)}\right) = 1$, and $\operatorname{rank}\left(\boldsymbol{\mathcal{X}}_k^{(n+1)}\right) = 1$. Updates along the direction of the first case are convex but this case is a relatively boring rescaling of the matrix from iteration n. The second case is more interesting because it fundamentally changes the embedding matrix; but it is nonconvex because it passes through an intermediate infeasible space where $\boldsymbol{\mathcal{X}}_k^{(n)} + \alpha\Delta\boldsymbol{\mathcal{X}}_k$ when $\Delta\boldsymbol{\mathcal{X}}_k$ is non-trivial. If, however, the SDP is relaxed by either dropping the rank constraint or equivalently replacing the

rank constraint with $\text{rank}(\mathbf{X}_k) \leq 2D$, the SDP is well-known to be convex [65, 66]. This convex relaxation of the SDP can be exploited to show that there exists a globally optimal regularization domain.

Proposition 3.4. *(based on proposition 4 from Bach et al. [64] and theorem 2 from Haeffele et al. [67]) If* $\mathbf{x}^*$ *is a feasible local minimizer of the low-rank MNE problem (Eq 3.22 & Eq 3.36) and the condition*

$$\forall k,\ \min(\{\varepsilon_k\}) \geq \lambda_{\max}\left(\frac{1}{N_{\text{samp}}}\sum_t (P_t - y_t)\mathbf{D}_t\right) = 2\lambda_{\max}(\nabla_{\boldsymbol{\mathcal{X}}_k} L) \tag{3.37}$$

is satisfied at $\mathbf{x}^*$ *then* $\mathbf{x}^*$ *is a feasible global minimizer of the low-rank MNE problem (note that* $\nabla_{\boldsymbol{\mathcal{X}}_k} L \in \mathbb{R}^{2D\times 2D}$ *is a gradient matrix of the negative log-likelihood with respect to* $\boldsymbol{\mathcal{X}}_k$*).*

Proof. *According to Prop 3.3, there is some* ε_k *above which* $2\nabla_{\boldsymbol{\mathcal{X}}_k} f$ *is positive definite for a given* $\mathbf{x}$*. It turns out since* $\nabla_{\boldsymbol{\mathcal{X}}_i} L = \nabla_{\boldsymbol{\mathcal{X}}_j} L$ *for any of the ith or jth member of the set* $\{\boldsymbol{\mathcal{X}}_k\}$ *that the value of* ε_k *such that the Hessian is positive semidefinite is equal for all i and j. Thus, if the minimum* ε_k *across all k satisfies Eq 3.37, the Hessian is guaranteed to be positive semidefinite at* $\mathbf{x}$*. Suppose* $\mathbf{x}^*$ *is a feasible local minimizer of the low-rank MNE problem and the conditions in Prop 3.4 are satisfied, then* $\nabla_{\boldsymbol{\mathcal{X}}_k} f$ *are positive semidefinite for all k and* $\nabla_{\boldsymbol{\mathcal{X}}_k} f\, \boldsymbol{\mathcal{X}}_k^* = \mathbf{0}$ *(Eq 3.31) when evaluated at the solution* $\mathbf{x}^*$*. It follows from here that* $\mathbf{x}^*$ *is then a feasible global minimizer of the relaxed SDP and more generally of* f*. This is because the corresponding weights from the SDP for* a^**,* $\mathbf{h}^*$*, and* $\{\boldsymbol{\mathcal{X}}_k^*\}$ *are feasible global minimizers of the relaxed SDP shown through the*

first-order Taylor series expansion of f with respect to $\{\boldsymbol{\mathcal{X}}_k\}$ about the solution:

$$f(a^*,\mathbf{h}^*,\{\boldsymbol{\mathcal{X}}_k\}) \approx f(a^*,\mathbf{h}^*,\{\boldsymbol{\mathcal{X}}_k^*\}) + \sum_{k=1}^{r} \mathrm{Tr}\left(\left[\nabla_{\boldsymbol{\mathcal{X}}_k} f|_{a^*,\mathbf{h}^*,\{\boldsymbol{\mathcal{X}}_k^*\}}\right]^{\mathrm{T}} [\boldsymbol{\mathcal{X}}_k - \boldsymbol{\mathcal{X}}_k^*]\right)$$
$$= f(a^*,\mathbf{h}^*,\{\boldsymbol{\mathcal{X}}_k^*\}) + \sum_{k=1}^{r} \mathrm{Tr}\left(\left[\nabla_{\boldsymbol{\mathcal{X}}_k} f|_{a^*,\mathbf{h}^*,\{\boldsymbol{\mathcal{X}}_k^*\}}\right]^{\mathrm{T}} \boldsymbol{\mathcal{X}}_k\right). \qquad (3.38)$$

No feasible $\boldsymbol{\mathcal{X}}_k$ can locally decrease f because the trace of the product of two positive semidefinite matrices is either positive or zero. This, combined with the fact that the relaxed SDP is convex, leads to the conclusion that $\mathbf{x}^*$ is a feasible global minimizer of f. Therefore, solutions to the low-rank MNE problem are globally optimal when Prop 3.4 is satisfied.

Low-rank MNE models optimized to satisfy Prop 3.4 belong to a regularization domain with globally optimal solutions. It should be made clear, however, that these solutions are globally optimal for a low-rank MNE problem given a specific set of regularization parameters and are not globally optimal solutions to the unregularized problem. In some cases, solutions in the globally optimal domain can be good approximations to the global minimum of the unregularized problem [64, 66, 67]. In past studies of structured matrix factorization problems [64, 66, 67], the use of convex relaxations has been motivated by the goal of obtaining good approximations to the global minimum of some low-rank nonconvex matrix factorization problem. A secondary motivation can also be to find good approximate factorizations of low-rank matrices in extremely large-scale convex programs where solving for $\mathbf{J}$, for instance, is impractical. For problems where $\mathbf{J}$ is not low-rank (either due to the ground truth being of higher-rank or noise corruption) but a low-rank approximation is desirable, the globally optimal domain may poorly approximate the components due to the large amount of regularization required to eliminate undesired components. This makes the locally optimal regularization domain of value because it may be possible to find more generalizable solutions with smaller

regularization parameters. This topic will be discussed further when the low-rank MNE method is applied in Chapter 4. It is worth noting that if $r = D$ in the low-rank MNE model, any local minimum is a global minimum for any feasible $\{\varepsilon_k\}$ because $\boldsymbol{\nabla}_{\boldsymbol{\mathcal{X}}} L = \mathbf{0}$ at a local minimum under the minimal regularization $\varepsilon_k = 0$ for all k.

With regard to the to locally optimal domain, there is a secondary notable consequence of Prop 3.4 that is stated here for the sake of completeness.

Proposition 3.5. *Given a feasible local minimizer* $\mathbf{x}^*$ *of the low-rank MNE problem, the quadratic weights at the solution,* $\mathbf{Q}^*$*, are a unique solution along the unit matrix* $\hat{\mathbf{Q}}^*$ *up to a change in sign of the columns; i.e.* $\pm\hat{\mathbf{Q}}^*_{\bullet,k}$ *for any k are equivalent solutions. If* $\mathbf{Q}^*\mathbf{Q}^{*\mathrm{T}}$ *is degenerate (i.e. has sets of eigenvalues with equal variance), then the solution is unique along directions within the subspace spanned by the degenerate eigenvectors of* $\mathbf{Q}^*\mathbf{Q}^{*\mathrm{T}}$*. Note that a unit matrix is defined here as* $\hat{\mathbf{Q}} = \mathbf{Q}/\left\|\mathbf{Q}\right\|_{\mathrm{F}}$.

Proof. *In contrast to the globally optimal domain,* $\boldsymbol{\nabla}_{\boldsymbol{\mathcal{X}}_k} f$ *is not guaranteed to be positive semidefinite at a feasible local minimizer* $\mathbf{x}^*$*. Yet,* f *remains a convex function of the SDP weights a,* $\mathbf{h}$*, and* $\{\boldsymbol{\mathcal{X}}_k\}$ *and* $\{\boldsymbol{\mathcal{X}}^*_k\}$ *is still a set of matrices complementary to* $\boldsymbol{\nabla}_{\boldsymbol{\mathcal{X}}_k} f$ *due to the satisfaction of Eq 3.31 at* $\mathbf{x}^*$*. Intuitively, this means that the direction of descent of the convex objective function with respect to* $\{\boldsymbol{\mathcal{X}}_k\}$ *is perpendicular to the direction defined by* $\{\boldsymbol{\mathcal{X}}^*_k\}$ *as in Eq 3.38 and* f *is therefore monotonically increasing along the set of unit embedding matrices* $\{\hat{\boldsymbol{\mathcal{X}}}^*_k\}$ *in directions pointing away from the solution. Thus, each* $\boldsymbol{\mathcal{X}}^*_k$ *is a unique solution along unit matrix* $\hat{\boldsymbol{\mathcal{X}}}^*_k$*. In factorized space, these unit embedding matrices can be uniquely decomposed up to a sign as* $\hat{\boldsymbol{\mathcal{X}}}^*_k = \left(\pm\hat{\mathbf{Q}}^*_{\bullet,k}\right)\left(\pm\hat{\mathbf{Q}}^{*\mathrm{T}}_{\bullet,k}\right) = \hat{\mathbf{Q}}^*_{\bullet,k}\hat{\mathbf{Q}}^{*\mathrm{T}}_{\bullet,k}$ *unless some subset of* $\{\boldsymbol{\mathcal{X}}^*_k\}$ *are degenerate in which case the decomposition of the degenerate embedding matrices is unique up to a rotation of the collection of their eigenvectors.*

The intuitive meaning of Prop 3.5 is that a solution with $\mathbf{Q}^*$ is globally optimal along the unit matrix $\hat{\mathbf{Q}}$. This conclusion could be potentially of interest for finding

globally optimal solutions in the locally optimal domain and is derived in preparation for possible future advancements in mathematical optimization methods. At the moment, applying Prop 3.5 would theoretically decrease the number of iterations required to reach a solution in a branch and bound algorithm. This would, however, be impractical for the problem sizes in this paper (at least at the time of writing this) since the modified branch and bound algorithm would still be at worst an exponential time algorithm.

3.3 Optimizing the model weights

The focus of this section is on solving the low-rank MNE problem in Eq 3.22 for a given set of nuclear-norm regularization parameters, $\{\varepsilon_k\}$. The topic of finding appropriate settings for the regularization parameters is treated separately in Section 4.1. More specifically, the focus here is to propose algorithms that guarantee convergence to a feasible local minimizer of the objective function, f, with respect to the weights at some finite feasible point in weight space, a^*, $\mathbf{h}^*$, $\mathbf{Q}^*$. Any gradient-based solver can be used with some degree of success, though second-order methods such as Newton's method and quasi-Newton methods are recommended, depending on the problem size, since these methods can explicitly avoid convergence to saddle points.

To leave open the option of using the method of Lagrange multipliers, a brief summary of an interior-point method based on Chapter 19 of Nocedal & Wright [41] is provided. The interior-point method searches iteratively for a feasible local minimizer, $\mathbf{x}^*$, of f by recursively solving the linear system

$$\underbrace{\begin{bmatrix} \nabla^2_{\mathbf{xx}}\mathcal{L}, & \mathbf{A}^{\mathrm{T}} \\ \mathbf{A}, & \mathbf{0} \end{bmatrix}}_{\text{constrained Hessian, } \mathcal{H}} \begin{bmatrix} \mathbf{p}_{\mathbf{x}} \\ -\mathbf{p}_{\Psi} \end{bmatrix} = - \underbrace{\begin{bmatrix} \nabla_{\mathbf{x}}\mathcal{L} \\ \nabla_{\Psi}\mathcal{L} \end{bmatrix}}_{\text{KKT conditions}} \tag{3.39}$$

where $\mathbf{A}$ is the full Jacobian of the constraints, $\mathbf{p_x}$ is the update direction of the weight vector, $\mathbf{x}$, $\mathbf{p_\Psi}$ is the update direction of the Lagrange multipliers unrolled into a vector, and $\nabla_\Psi \mathcal{L}$ is the gradient of the Lagrangian with respect to the Lagrange multipliers unrolled into a vector. Note that for the linear equality constraints, $\nabla^2_{\mathbf{xx}} \mathcal{L} = \nabla^2_{\mathbf{xx}} f$ and

$$(\nabla_\Psi \mathcal{L})^{\mathrm{T}} = \left[\begin{array}{cccc} \mathbf{Q}^{\mathrm{T}}_{\bullet,1} \mathbf{A}^{\mathrm{T}}_{1,1}, & \mathbf{Q}^{\mathrm{T}}_{\bullet,2} \mathbf{A}^{\mathrm{T}}_{2,2}, & \cdots, & \mathbf{Q}^{\mathrm{T}}_{\bullet,r} \mathbf{A}^{\mathrm{T}}_{r,r} \end{array} \right]. \tag{3.40}$$

Apparently, the optimization completes when the vector on the right-hand-side of Eq 3.39 is zero since the vector is equal to the KKT conditions (Prop 3.1).

The remaining problem is making sure that the interior-point method converges to a feasible local minimizer rather than a local maximum or saddle point. This is done with a backtracking line search where a candidate point is accepted only when there is sufficient decrease of f and infeasibility and by adding a diagonal shift matrix to the Hessian to maintain appropriate matrix inertia [41]. The diagonal shift matrix, $\delta \mathbf{I}$ where $\delta \geq 0$ is added to the Hessian of the Lagrangian, $\nabla^2_{\mathbf{xx}} \mathcal{L} + \delta \mathbf{I}$ such that the number of positive eigenvalues, m, the number of negative eigenvalues, n, and the of eigenvalues equal to zero, l, are equal to the to total number of weights, the total number of constraints, and zero, respectively. If the constrained Hessian does not meet these conditions, δ is adjusted to enforce the matrix inertia [41]. For an unconstrained problem, this procedure is equivalent to maintaining positive definiteness of the Hessian.

For large-scale problems, where computing and inverting the constrained Hessian is impractical, the Hessian of the Lagrangian may be approximated using L-BFGS [41,68]. Since the L-BFGS procedure is lengthy to describe, it will not be described here in more detail except to note that the algorithm gains speed over using the exact Hessian by reducing the size of the matrices that must be inverted (or eliminates the inversion entirely for unconstrained problems). Furthermore, the approximation to the Hessian by

L-BFGS is positive definite by definition and maintains the proper matrix inertia making it less likely to converge to a saddle point compared to first-order methods like gradient descent [41]. The main downside of L-BFGS is that it takes more iterations to converge than the exact Hessian which, for low dimensional problems, may actually cause L-BFGS to be slower. Although for low dimensional problems where this would occur, the timing difference is not likely to be significant making L-BFGS often a good choice regardless.

Block coordinate descent is another algorithm that may be used to reduce the size of large scale problems. Block coordinate descent can be particularly useful when the problem size is so large that even computing and storing the gradient of the full problem is impractical. This may be useful for extremely high-resolution imaging, for example. Block coordinate descent can also be used in place of L-BFGS where the problem is broken up such that the exact Hessians of the weights in each block are more practical to compute. Of course, block coordinate descent requires solving a subproblem with respect to the block weights which can also be solved with L-BFGS (or gradient descent, etc.). A block coordinate descent algorithm was developed for the low-rank MNE problem where block weights $\mathbf{x}_k^{\mathrm{T}} = \left[a, \mathbf{h}^{\mathrm{T}}, \mathbf{Q}_{\bullet,k}^{\mathrm{T}}\right]$ are defined for all $k \in \{1, \cdots, r\}$. The algorithm then cyclically solves the r block subproblems:

$$\text{block } k \text{ subproblem: } \begin{cases} \min\limits_{a,\mathbf{h},\mathbf{Q}_{\bullet,k}} f(a,\mathbf{h},\mathbf{Q}) \\ \qquad \text{subject to } \mathbf{A}_{k,k}\mathbf{Q}_{\bullet,k} = \mathbf{0} \end{cases} \tag{3.41}$$

until the KKT conditions (Prop 3.1) and second-order sufficient conditions (Prop 3.2) are satisfied. This algorithm provably converges under very mild conditions to a local minimum of the low-rank MNE problem as shown in Appendix B.

3.4 Hyperparameter optimization problems

The hyperparameters of the optimization are the structural parameters of the model and the objective function that may be adjusted but are not optimized to minimize the objective function, f, evaluated on the training set. The hyperparameters of the low-rank MNE problem are $\{\varepsilon_k\}$, r_{PSD}, and r_{NSD}. It is easy to see why $\{\varepsilon_k\}$ in particular cannot be optimized to minimize f on the training set because the solution would be trivially $\varepsilon_k = 0$ for all k. Similarly, setting r_{PSD} and r_{NSD} to minimize f on the training set bears the significant risk of overfitting since more weights will produce a model that fits the training set equal to or better than models with fewer weights. This section discusses principled ways to choose the hyperparameter settings of the low-rank MNE problem.

3.4.1 Regularization parameters

Finding reasonable settings for the nuclear-norm regularization parameters is goal-dependent. For instance, does one seek a consistent solution that approximates the global minimum of the unregularized problem or a solution that will generalize well to novel data? In principle, one could achieve both in tandem, but these two scenarios will be treated separately and then combined later as one optimization problem. The two cases that will be looked at first are solutions in the globally optimal domain that are close approximations to the global minimum of L evaluated on the training set and solutions in either domain that best fit the cross-validation set data.

Starting with the former, globally optimal approximations to the unregularized problem would mean finding those $\{\varepsilon_k\}$ that satisfy Prop 3.4. However, since one would like the optimization problem to be as close as possible to the unregularized problem while minimizing bias from the nuclear-norm penalty, one would want to apply the minimal amount of regularization necessary to reach the globally optimal regularization

domain (Prop 3.4). Explicitly, one searches for a solution to the nonlinear program

$$\min_{a,\mathbf{h},\mathbf{Q},\{\varepsilon_k\}} f(a,\mathbf{h},\mathbf{Q};\{\varepsilon_k\}) \quad \text{subject to } \forall k, \begin{cases} \mathbf{A}_{k,k}\mathbf{Q}_{\bullet,k} = \mathbf{0} \\ \varepsilon_k - 2\lambda_{\max}\left(\boldsymbol{\nabla}_{\boldsymbol{\mathcal{X}}_k} L\right) = 0. \end{cases} \tag{3.42}$$

Note that inequality in Eq 3.37 has been replaced with an equality. Furthermore, it is already known from Props 3.35 & 3.4 that $\varepsilon_i = \varepsilon_j$ for any i and j and therefore this optimization only has one independent regularization parameter, $\varepsilon_k = \varepsilon$ for all k. Solutions to Eq 3.42 are called the *globally optimal approximation*.

Regularization domain-agnostic solutions that instead search for $\{\varepsilon_k\}$ that best generalizes to novel data instead takes on the form

$$\begin{aligned} \{\varepsilon_k^*\} &= \arg\min_{\{\varepsilon_k\}} \; L(\mathbf{x}^*)|_{T_{\text{CV}}} \\ &\text{subject to} \quad \varepsilon_k \le \varepsilon_{k+1}, \\ &\qquad\qquad \forall k \in K_{\text{PSD}} \bigcup K_{\text{NSD}} \end{aligned} \quad \left\{ \begin{aligned} \mathbf{x}^* &= \arg\min_{\mathbf{x}} \; f(\mathbf{x};\{\varepsilon_k\})|_{T_{\text{train}}} \\ &\text{subject to} \quad \mathbf{A}_{k,k}\mathbf{Q}_{\bullet,k} = \mathbf{0}, \\ &\qquad\qquad \forall k \in \{1,\cdots r\} \end{aligned} \right. \tag{3.43}$$

where the weight vector $\mathbf{x}$ is used for conciseness, T_{CV} and T_{train} are non-intersecting sets of sample indices (t) that belong to the cross-validation and training sets, respectively. The sets K_{PSD} and K_{NSD} are column indices for $\mathbf{Q}$ where $K_{\text{PSD}} = \left\{1, 2, \cdots, r_{\text{PSD}} - 1\right\}$ and $K_{\text{NSD}} = \left\{r_{\text{PSD}} + 1, r_{\text{PSD}} + 2, \cdots, r - 1\right\}$ and the components of $\mathbf{Q}_{\bullet,k}$ are assumed to be arranged such that $\pi_k = -1$ (Eq 3.23) for any $k \in \left\{1, 2, \cdots, r_{\text{PSD}}\right\}$ and $\pi_k = 1$ for any $k \in \left\{r_{\text{PSD}} + 1, r_{\text{PSD}} + 2, \cdots, r\right\}$. The minimization problem to the right of the bracket in Eq 3.43 constitutes a subproblem that must be minimized and whose solution acts as fixed input to the problem on the left of the bracket. Solving the problem in Eq 3.43 will be called *empirical model selection*.

One may notice that the problem in Eq 3.43 now includes a new set of constraints

on the regularization parameters. These constraints are introduced because the nonlinearity $P(y=1|\mathbf{s}_t)$ is invariant to the rearrangement of $\mathbf{Q}$'s columns and therefore any problems that are the same up to a permutation of the pairs ε_k and π_k are equivalent optimization problems. Therefore, the volume of the regularization parameter search space can be substantially decreased without loss of generality.

Finally, one may insist on finding a solution that will generalize well to novel data while at the same time belonging in the globally optimal domain. This can be achieved simply by adding the constraint from Eq 3.37 without modification (i.e. keep the inequality this time) to the problem in Eq 3.43.

3.4.2 Rank parameters

The strategy for setting the maximum rank hyperparameters, $r = r_{\text{PSD}} + r_{\text{NSD}}$, is goal and problem dependent. For instance, the strategies may differ if a globally optimal approximation is sought versus generalizable approximations. The strategy may be further modified if one's interest is in finding a receptive field of optimal rank versus a receptive field of a given rank r.

The case that will become of most interest in this volume is finding generalizable approximations to receptive fields with unknown optimal rank. Compared to the regularization parameters, setting the maximum rank $r = r_{\text{PSD}} + r_{\text{NSD}}$ of the low-rank MNE model is relatively easy in this case. Since nuclear-norm regularization may be used to lower the overall rank of $\mathbf{J}$, it therefore makes sense to set r_{PSD} and r_{NSD} relative to an upper bound on the expected rank of the model. For example, if one knows (or suspects) that the receptive field of a neuron is spanned by four components but does not know what the sign of those components might be, one can set $r_{\text{PSD}} = r_{\text{NSD}} = 5$ and let the regularization procedure "prune" irrelevant components. The reason why r_{PSD} and r_{NSD} are set to five instead of four is because excess dimensions provide infor-

mation about whether the number of prescribed components is sufficient. If more than enough positive and negative components are provided to capture the receptive field, then $\text{rank}(\mathbf{J}^{(\text{PSD})}) < r_{\text{PSD}}$ and $\text{rank}(\mathbf{J}^{(\text{NSD})}) < r_{\text{NSD}}$ and raising r_{PSD} or r_{NSD} will not provide any new non-zero components to the basis. Procedurally, one should thus increment r_{PSD} and r_{NSD} until this condition is met.

If instead one wants to recover a compressed reconstruction of the receptive field of a particular maximum rank r using either the globally optimal approximation or empirical model selection, the above procedure is not applicable. In this case, one would need to instead enumerate the different combinations of r_{PSD} and r_{NSD} such that $r_{\text{PSD}} + r_{\text{NSD}} = r$. This is not difficult to perform, but it does mean an increase in the number of models that must be fit relative to the upper bounding procedure. Depending on one's preferences, either the training set or the cross-validation set can be used for model selection (i.e. determining the most appropriate model for the application). One can also use other principled methods for setting r_{PSD} and r_{NSD}; for example, using the signs from the largest absolute variance components of $\mathbf{J}$ obtained from the full-rank MNE model.

The globally optimal approximation was left out from the aforementioned pruning procedure because the globally optimal approximation does not necessarily guarantee that an upper bound r greater than the optimal rank will ultimately yield an equivalent model. This is because the ε that satisfies the relevant equality constraint in Eq 3.42 may change for different r.

This concludes the theoretical background of the low-rank MNE method. In the next chapter, the low-rank MNE method will be discussed in practical terms, including the introduction of algorithms to determine the regularization parameter settings and simulated analyses to demonstrate the application of the low-rank MNE method. The practical discussion will culminate in an application to recovering spectrotemporal

receptive fields of high-level auditory neurons.

Chapter 3 contains work that was published in Kaardal, Theunissen, and Sharpee, *Frontiers in Computational Neuroscience* (2017). The dissertation author was the primary investigator and author of the paper.

Chapter 4

Low-rank minimal models in practice: applications to the auditory system

In the prior chapter, the theoretical foundation of the low-rank MNE method was established and various properties of the optimization problem were analyzed including the derivation of globally optimal and locally optimal regularization domains. In the present chapter, the focus shifts to practical applications of the low-rank MNE method towards reconstructing the spectrotemporal receptive fields of auditory neurons recorded from the zebra finch auditory forebrain. Before delving into the analysis of the auditory system, algorithms are proposed for solving the optimization problems in Eqs 3.42 & 3.43 and the analysis is simulated on synthetic neurons for validation.

4.1 Optimization algorithms

Finding a globally optimal approximations (Eq 3.42) may be achieved in a couple ways. One simple way is to initialize $\varepsilon_k = \varepsilon = 0$ for all k and then solve Eq 3.22 using standard methods for optimizing continuously differentiable functions (such as

Algorithm 4.1 Globally optimal approximation of a low-rank MNE model.

inputs: r_{PSD}, r_{NSD}, $\{(y_t, \mathbf{s}_t)\}$; initial guess for a, $\mathbf{h}$, $\mathbf{U}$, and $\mathbf{V}$
initialization: $\varepsilon \leftarrow 0$

while $\varepsilon \neq 2\lambda_{\max}(\nabla_{\boldsymbol{\mathcal{X}}} L)$ **do**
    Reinitialize $\mathbf{U}_{\bullet,k}$ and $\mathbf{V}_{\bullet,k}$ for any k if $\mathbf{U}_{\bullet,k} = \mathbf{0}$ or $\mathbf{V}_{\bullet,k} = \mathbf{0}$
    a, $\mathbf{h}$, $\mathbf{U}$, $\mathbf{V} \leftarrow$ solve Eq 3.22
    compute $\lambda_L \leftarrow 2\lambda_{\max}(\nabla_{\boldsymbol{\mathcal{X}}} L)$
    if $\varepsilon > \lambda_L$ **then**
        choose new ε from $[\lambda_L,\ \varepsilon)$
    else if $\varepsilon < \lambda_L$ **then**
        choose new ε from $(\varepsilon, \lambda_L]$
$\mathbf{J} \leftarrow \mathbf{U}\mathbf{V}^{\mathrm{T}}$

returns: a, $\mathbf{h}$, $\mathbf{J}$

the interior-point/Newton's method from Section 3.3). Then, one can test if ε is feasible by checking whether $\varepsilon - 2\lambda_{\max}(\nabla_{\boldsymbol{\mathcal{X}}} L) = 0$ holds (defining $\boldsymbol{\mathcal{X}} = \sum_{k=1}^{r} \boldsymbol{\mathcal{X}}_k$) where the affirmative would indicate ε is feasible and the negative that ε is infeasible. If ε is feasible and the rest of the KKT conditions are satisfied (Prop 3.1), then a globally optimal approximation has been found. If ε is infeasible but the rest of the KKT conditions are satisfied (Prop 3.1), then ε is adjusted to be closer to the present value of $2\lambda_{\max}\left(\nabla_{\boldsymbol{\mathcal{X}}_k} L\right)$. Pseudocode of this algorithm appears in Alg 4.1. It is possible a more sophisticated algorithm may be derived by forming the Lagrangian of Eq 3.42,

$$\mathcal{L}_{\text{global}}(a, \mathbf{h}, \mathbf{Q}; \varepsilon) = f(a, \mathbf{h}, \mathbf{Q}; \varepsilon) - \sum_{k=1}^{r} \boldsymbol{\Psi}_{\bullet,k}^{\mathrm{T}} \mathbf{A}_{k,k} \mathbf{Q}_{\bullet,k} - \Psi_{\varepsilon}\left[\varepsilon - 2\lambda_{\max}(\nabla_{\boldsymbol{\mathcal{X}}} L)\right] \tag{4.1}$$

where Ψ_{ε} is a Lagrange multiplier. The considerable downside of this constraint is that it does not have an analytic gradient. It is unclear whether there is a quick way to solve this problem, so this is left behind in favor of the simple algorithm from Alg 4.1.

Despite the global optimality of the solutions, the nonconvexity of the problem means that one must still take care to avoid poor initializations. For instance, it is trivially

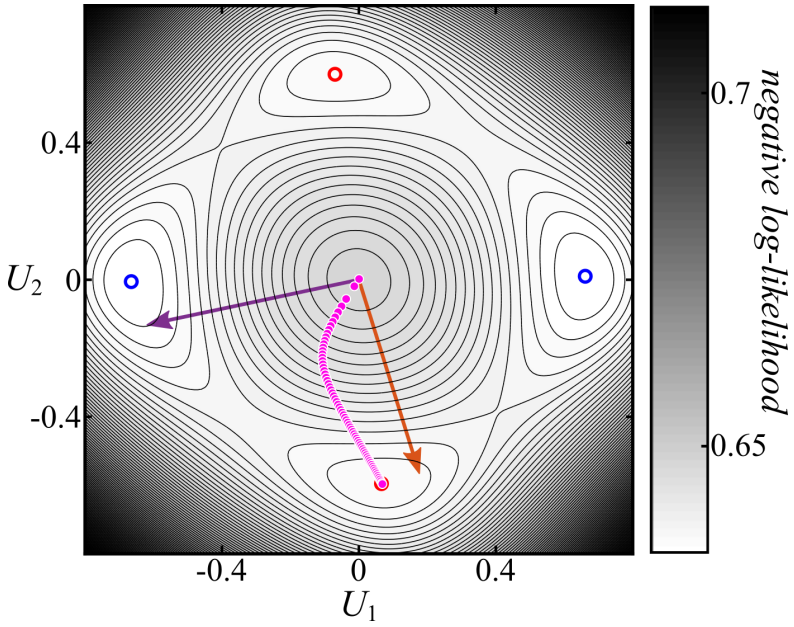

Figure 4.1: **Relaxation from a globally optimal approximation to a locally optimal solution does not guarantee global optimality.** This is the same problem as appears in Fig 3.1. The magenta dots show progress of a slow relaxation (annealing) of the regularization parameter ε from the globally optimal domain where $\varepsilon > 0$ to $\varepsilon = 0$ via the update $\varepsilon \leftarrow \varepsilon - \eta$ where here $\eta = 1 \cdot 10^{-3}$. Each magenta dot corresponds to a solution to the low-rank MNE problem given a value of ε using an interior-point method. The relaxation of the regularization parameter results in convergence to the southern local minimum where $L = 0.63\mathbf{6464}$ while the global minima to the east and west have function value $L = 0.63\mathbf{4719}$. Reducing η to $1 \cdot 10^{-4}$ did not change the result. The arrows correspond to the ground truth $\mathbf{U}$ components as defined in Fig 3.1.

the case that $\nabla_{\mathbf{Q}_{\bullet,k}} f = \mathbf{0}$ when $\mathbf{Q}_{\bullet,k} = \mathbf{0}$ (Prop 3.1) but $\mathbf{Q}_{\bullet,k} = \mathbf{0}$ is not necessarily a local minimum and can in fact be a local maximum. It is worth noting that relaxation of the nuclear-norm regularization parameter (where the the regularization parameter is annealed from a globally optimal ε to zero) from a solution in the globally optimal domain into a solution of the unregularized problem (where $\varepsilon = 0$) does not guarantee convergence to the global minimum of L. Examples were found among randomly generated problems (Section 3.2.1) that relaxed into suboptimal local minima instead; for instance, the problem in Fig 4.1 where the solution relaxes into a suboptimal local minimum.

The empirical model selection (Eq 3.43) method must be solved with a completely different approach since there are no constraints to prevent $\{\varepsilon_k\}$ from taking on the trivial solution. Measuring the optimality of a model in terms of the negative log-likelihood evaluated on the cross-validation set requires a search algorithm that is different from the usual gradient-based algorithms because L (evaluated on the cross-validation set) does not have an analytic form in terms of $\{\varepsilon_k\}$. Three choices that may be used, in order of sophistication, are a grid search [69], random search [69], and Bayesian optimization [69–71]. Since only the grid search and Bayesian optimization approaches were implemented, these are described in more detail below.

4.1.1 Grid search

In a grid search, a grid is formed over the domain of $\{\varepsilon_k\}$ where a set of discrete points on the domain of ε_k is denoted by g_k. The discrete, finite set that enumerates all possible combinations of elements from every g_k is defined by the Cartesian product, $G = g_1 \times g_2 \times \cdots \times g_r$. To perform the grid search, then one would iterate over all elements of G, $\{\varepsilon_k\} \in G$, solving Eq 3.22 for each set of coordinates, and return the model that leads to the minimal L evaluated on the cross-validation set. This is considered the most generalizable estimate of the receptive field. This makes a grid search an easy and intuitive approach to hyperparameter optimization that works well provided the cardinality of G is reasonably small. Unfortunately, with increasing r, one may be forced to reduce the cardinality of each g_k for the search to remain tractable. Supposing that the domain is discretized by the same number of samples, n, along each ε_k, the number of feasible grid coordinates (accounting for the inequality constraints in Eq 3.43) is a

product of two figurate numbers:

$$\mathbf{card}\,(G) = \sum_{k_1=1}^{k_2} \cdots \sum_{k_{r_{\mathrm{PSD}}}=1}^{n} \sum_{k_{r_{\mathrm{PSD}}+1}}^{k_{r_{\mathrm{PSD}}+2}} \cdots \sum_{k_r}^{n} 1 = \frac{\prod_{j=0}^{r_{\mathrm{PSD}}-1}(n+j)}{r_{\mathrm{PSD}}!} \frac{\prod_{j=0}^{r_{\mathrm{NSD}}-1}(n+j)}{r_{\mathrm{NSD}}!} \tag{4.2}$$

(note that when $r_{\cdot\mathrm{SD}} = 0$ then the product $\prod_{j=0}^{r_{\cdot\mathrm{SD}}-1}(n+j)$ is defined to be one). Consequently, grid search is susceptible to an explosion of dimensionality as the rank of the model increases since $\mathbf{card}\,(G)$ is a monotonically increasing function of r. Furthermore, it can be shown that $\mathbf{card}\,(G)$ is divergent for large r when $n > 1$.

Proof. *Suppose n is some positive integer where* $n > 1$ *and r is an arbitrary positive integer where it can be assumed that either* $r_{\mathrm{PSD}} = 0$ *or* $r_{\mathrm{NSD}} = 0$ *without loss of generality. It is then possible to show rigorously that the cardinality of the set G diverges as r asymptotically increases:*

$$\lim_{r\to\infty} \mathbf{card}\,(G) = \lim_{r\to\infty} \frac{(n+r-1)!}{(n-1)!\,r!} = \frac{1}{(n-1)!} \lim_{r\to\infty} \prod_{j=1}^{n-1}(r+j) = \infty. \tag{4.3}$$

Therefore, $\mathbf{card}\,(G)$ *is a monotonically increasing function of r and diverges when* $n > 1$.

This explosion of dimensionality sets a practical limit on the precision to which a grid search can be used to solve Eq 3.43 for multiple nuclear-norm regularization parameters.

In response to this problem, a heuristic algorithm was proposed [72] using block coordinate descent (Eq 3.41) that exploits the blockwise dependence on a single regularization parameter. For the kth block, a grid search is performed over g_k while holding the remaining dimensions fixed, keeping the block solution with minimal prediction error on the cross-validation set before moving on to the next block. This procedure repeats cyclically until several consecutive cycles through all r blocks fail to lead to a reduction in the prediction error. This procedure bears some similarity to the empirical form of early stopping since the optimization pulls the model towards a local minimum

Algorithm 4.2 Empirical model selection using a block coordinate descent heuristic for low-rank MNE model estimation.

inputs: r_{PSD}, r_{NSD}, $\{(y_t, \mathbf{s}_t)\}$; initial guess for a, $\mathbf{h}$, $\mathbf{U}$, $\mathbf{V}$; define sets sampling the domains of the regularization parameters $\{g_k\}$; training and cross-validation sets T_{train} and T_{CV}, respectively, where $T_{\text{train}} \cap T_{\text{CV}} = \emptyset$; maximum failures to find a better solution, M_{fail}
Initialization: $a' \leftarrow a$, $\mathbf{h}' \leftarrow \mathbf{h}$, $\mathbf{U}' \leftarrow \mathbf{U}$, $\mathbf{V}' \leftarrow \mathbf{V}$, $L_{\text{best}} \leftarrow L(a, \mathbf{h}, \mathbf{U}, \mathbf{V})|_{T_{\text{CV}}}$, $m_{\text{fail}} \leftarrow 0$

while $m_{\text{fail}} < M_{\text{fail}}$ **do**
    initialize failure switch $\xi \leftarrow 1$
    for $k \in \{1, \cdots, r\}$ **do**
        for $\varepsilon_k \in g_k$ **do**
            a', $\mathbf{h}'$, $\mathbf{U}'_{\bullet,k}$, $\mathbf{V}'_{\bullet,k} \leftarrow$ solve Eq 3.41 with respect to the training set, T_{train}
            compute $L_{\text{new}} \leftarrow L(a', \mathbf{h}', \mathbf{U}', \mathbf{V}')$
            if L_{new} is sufficiently less that L_{best} **then**
                / better solution found */*
                $L_{\text{best}} \leftarrow L_{\text{new}}$
                $a \leftarrow a'$, $\mathbf{h} \leftarrow \mathbf{h}'$, $\mathbf{U}_{\bullet,k} \leftarrow \mathbf{U}'_{\bullet,k}$, $\mathbf{V}_{\bullet,k} \leftarrow \mathbf{V}'_{\bullet,k}$
                $m_{\text{fail}} \leftarrow 0$
                $\xi \leftarrow 0$
    $m_{\text{fail}} \leftarrow m_{\text{fail}} + \xi$
$\mathbf{J} \leftarrow \mathbf{U}\mathbf{V}^{\mathrm{T}}$

returns: a, $\mathbf{h}$, $\mathbf{J}$

on the training set but does not necessarily return a converged solution. A pseudocode implementation of this algorithm may be found in Alg 4.2. It is recommended, when passing through the kth set g_k to optimize the models in order from the lowest to highest values of the regularization parameter. By doing so, one may use the solution obtained with the previous ε_k as input to the problem with the next largest ε_k on the grid for a swifter optimization. Passing through the grid from highest to lowest is not recommended without protecting against poor initialization. This heuristic algorithm was found to work reasonably well in Kaardal et al. [72]. The main drawback of this algorithm is that this it requires substantial computation time (though usually not nearly as much as the standard grid search would require).

4.1.2 Bayesian optimization

Bayesian optimization may be used as an alternative to grid search for optimizing the hyperparameters of a problem [70, 71]. Indeed, after the publication of Kaardal et al. [72], it was soon found that Bayesian optimization is not only competitive, but can often be a much faster approach to finding appropriate regularization parameter settings for the low-rank MNE problem in Eq 3.43.

Generally speaking, the goal of Bayesian optimization is to find the global optimum of a function on some finite domain of the independent variables in (hopefully) sub-exponential time. What makes Bayesian optimization different from other deterministic methods for global optimization is that it can also be used to efficiently find the global optimum of an unknown or hidden function. In the case of the low-rank MNE problem, the negative log-likelihood evaluated on the cross-validation set as a function of the nuclear-norm regularization parameters, $L(\boldsymbol{\epsilon})$ where $\boldsymbol{\epsilon} = [\varepsilon_1, \cdots, \varepsilon_r]$, would be an example of such an unknown function because it does not have an analytic form. Provided $L(\boldsymbol{\epsilon})$ can be assumed to be continuous, the gradient of $L(\boldsymbol{\epsilon})$ is bounded on some finite domain in $\boldsymbol{\epsilon}$. Thus, given a point $\boldsymbol{\epsilon}$ and the maximum gradient defined on the domain, a linear extrapolation outwards from $\boldsymbol{\epsilon}$ sets a lower bound on the objective function at any other point on the domain (doing this in branch and bound form is known as Lipschitzian optimization). However, the maximum slope of $L(\boldsymbol{\epsilon})$ is not known; and as such, a probabilistic approach may be taken where instead $L(\boldsymbol{\epsilon})$ is treated as a random variable drawn from some distribution, typically a Gaussian process.

Given $\boldsymbol{\epsilon}_\rho$ and L_ρ where $\rho \in \{1, \cdots, N_\rho\}$ is the sample label of $L(\boldsymbol{\epsilon})$ measured at point $\boldsymbol{\epsilon}_\rho$, a Gaussian process is defined where each L_ρ is a Gaussian distributed random variable parameterized by ε_ρ and any subset of $\{L_\rho\}$ of cardinality greater than one is drawn from a multivariate Gaussian distribution [73]. The Gaussian process is completely defined by the mean, $\mu(\boldsymbol{\epsilon})$, and a covariance function, $\kappa(\boldsymbol{\epsilon}, \boldsymbol{\epsilon}')$, where $\boldsymbol{\epsilon}'$ is

a reference point on the domain of ϵ. Without loss of generality, $\mu(\epsilon)$ can be set to zero [70, 73]. In Bayesian optimization, a Gaussian process is used as a prior distribution on the objective function: $L(\epsilon) \sim \mathcal{GP}\big(0, \kappa(\epsilon,\ \epsilon')\big)$. Bayesian inference is then used to recursively update the prior distribution, $P\left(L; S_{N_\rho}\right)$, as new samples are added to the set of pairs $S_{N_\rho} = \big\{(L_\rho, \epsilon_\rho)\big\}$. The probability distribution of the Gaussian random number $L_{N_\rho+1}$ at some new point $\epsilon_{N_\rho+1}$ is estimated through the posterior, $P(L_{N_\rho+1}|\epsilon_{N_\rho+1}; S_{N_\rho})$. Explicitly, a new point $\epsilon_{N_\rho+1}$ is incorporated into the covariance matrix like so

$$\mathcal{K}_{1:N_\rho+1,1:N_\rho+1} = \begin{bmatrix} \kappa(\epsilon_1,\epsilon_1), & \kappa(\epsilon_1,\epsilon_2), & \cdots, & \kappa(\epsilon_1,\epsilon_{N_\rho+1}) \\ \kappa(\epsilon_2,\epsilon_1), & \kappa(\epsilon_2,\epsilon_2), & \ddots, & \vdots \\ \vdots, & \ddots, & \ddots, & \kappa(\epsilon_{N_\rho},\epsilon_{N_\rho+1}) \\ \kappa(\epsilon_{N_\rho+1},\epsilon_1), & \cdots, & \kappa(\epsilon_{N_\rho+1},\epsilon_{N_\rho}), & \kappa(\epsilon_{N_\rho+1},\epsilon_{N_\rho+1}) \end{bmatrix} \tag{4.4}$$

and $L_{N_\rho+1}$ is drawn from a univariate normal distribution, $L_{N_\rho+1} \sim \mathcal{N}_{\text{norm}}\left(\mu_{N_\rho+1}, \sigma^2_{N_\rho+1}\right)$, at $\epsilon_{N_\rho+1}$ where the mean and variance are derived from $\mathcal{K}_{1:N_\rho+1}$:

$$\mu_{N_\rho+1} = \mathcal{K}_{N_\rho+1,1:N_\rho}\mathcal{K}^{-1}_{1:N_\rho+1,1:N_\rho+1}\mathbf{L}_{1:N_\rho} \tag{4.5}$$

$$\sigma^2_{N_\rho+1} = \kappa(\epsilon_{N_\rho+1},\epsilon_{N_\rho+1}) - \mathcal{K}_{N_\rho+1,1:N_\rho}\mathcal{K}^{-1}_{1:N_\rho+1,1:N_\rho+1}\mathcal{K}_{1:N_\rho,N_\rho+1} \tag{4.6}$$

(for the derivation, see Brochu et al. [70]). Here, $\mathbf{L}^{\mathrm{T}}_{1:N_\rho} = \big[L_1, \cdots, L_{N_\rho}\big]$. This vector of objective function values is then augmented to include the measured value of $L_{N_\rho+1}$ by concatenation to the end of $\mathbf{L}_{1:N_\rho}$ giving $\mathbf{L}_{1:N_\rho+1}$. The new prior distribution is then $L(\epsilon) \sim \mathcal{N}\left(0, \mathcal{K}_{1:N_\rho+1,1:N_\rho+1}\right)$ and then the procedure is repeated.

The choice of covariance kernel is chosen to be based on a Euclidean distance metric (i.e. $|\epsilon - \epsilon'|$) since $L(\epsilon)$ has been found empirically to be at least approximately continuous. In Snoek et al. [71], the authors recommend using the Matérn 5/2 kernel [74] while the python software that was used for the low-rank MNE problem, *GPyOpt* [75],

used the Matérn 3/2 kernel [74] as a default. There was not found to be significant practical differences between the two kernels so the default Matérn 3/2 kernel was used. Explicitly, the Matérn 3/2 kernel is

$$\kappa(\epsilon,\epsilon') = \nu^2\left(1+\sqrt{3}\frac{\|\epsilon-\epsilon'\|_2}{\eta}\right)e^{-\sqrt{3}\frac{\|\epsilon-\epsilon'\|_2}{\eta}} \tag{4.7}$$

where ν is the variance and η is the length scale which can be optimized to best fit the observed covariance of the measurements.

One of the strongest points of Bayesian optimization is how it chooses the next trial point, $\epsilon_{N_\rho+1}$. Since the goal is to find a global optimum of $L(\epsilon)$, a reasonable goal for making the choice would be to develop an *acquisition function* that balances between exploitation (or convergence) and exploration. Several of these are available, including *probability of improvement*, *expected improvement*, and *lower confidence bound* [70, 71]. Without going into the details of each of the acquisition functions, it was found that the software's [75] lower confidence bound acquisition function worked better than expected improvement. Probability of improvement was not evaluated because it is a legacy method that seems to have largely fallen into disfavor [70, 71]. The lower confidence bound also includes a parameter that can be adjusted to either force the optimization to spend more time exploring or to quench into a minimum of $L(\epsilon)$. The lower confidence bound acquisition function is

$$f_{\mathrm{LCB}}(\epsilon_{N_\rho+1}) = \mu_{N_\rho+1} - \tau\sigma_{N_\rho+1} \tag{4.8}$$

where $\tau \geq 0$ is the exploration weighting. The new trial is the feasible $\epsilon_{N_\rho+1}$ that minimizes f_{LCB}. Intuitively, one can think of the lower confidence bound as being some fractional standard deviation estimate of the lower bound on the function L. When τ is close to zero, the optimization will prefer exploitation and choose points with minimal

estimated mean. Large τ promotes exploration where $L(\boldsymbol{\epsilon}_{N_\mathrm{p}+1})$ with large standard deviation can be more favorable.

In the acquisition function minimization routine, it became necessary to make a small modification to the Bayesian optimization software [75]. A Monte Carlo search was used to initialize $\boldsymbol{\epsilon}$ when minimizing f_{LCB} where an initial point was chosen with uniform probability from within the domain of $\boldsymbol{\epsilon}$ and any infeasible point according to the constraints in Eq 3.43 was rejected. There are two problems related to this procedure: (i) if an infinite number of Monte Carlo trials are tested until the desired number of feasible initializations are found, it may take a long time to find a feasible trial point and (ii) if a finite number of Monte Carlo trials are tested, it is possible that no feasible trial points will be recovered. This is particularly problematic for the low-rank MNE problem where the introduction of the simplifying constraints on $\{\varepsilon_k\}$ in Eq 3.43 inadvertently leads to increasing difficulty in randomly selecting a feasible point as r increases as shown in the following.

Proof. *Suppose the domain of all members of* $\{\varepsilon_k\}$ *is* $[0, w]$ *where* $w > 0$*. The volume of the feasible region is*

$$\mathbf{vol}\,(G) = \lim_{n\to\infty}\left[\mathbf{card}\,(G)\,\frac{w^r}{n^r}\right] = \int_0^w \cdots \int_0^{\varepsilon_{r_{\mathrm{PSD}}+2}} \int_0^w \cdots \int_0^{\varepsilon_2} \mathrm{d}\boldsymbol{\epsilon} = \frac{w^{r_{\mathrm{PSD}}}}{r_{\mathrm{PSD}}!}\frac{w^{r_{\mathrm{NSD}}}}{r_{\mathrm{NSD}}!} \tag{4.9}$$

where G and n are the same as those in Eq 4.3 and $\mathrm{d}\boldsymbol{\epsilon} = \prod_{k=1}^{r} \mathrm{d}\varepsilon_k$*. Since the volume of the domain is* w^r*, the probability of choosing a feasible random point uniformly from the domain is* $p = (r_{\mathrm{PSD}}!)^{-1}(r_{\mathrm{NSD}}!)^{-1}$*. Because* p *is a monotonically decreasing function in* $r_{\cdot\mathrm{SD}}$ *and* $\lim_{r_{\cdot\mathrm{SD}}\to\infty} p = 0$*, it is increasingly unlikely that a random point in the domain of* $\boldsymbol{\epsilon}$ *will be feasible as* r_{PSD} *and* r_{NSD} *grow larger and the volume is asymptotically zero.*

This problem did not become noticeable until $r_{\mathrm{PSD}} \geq 5$ and $r_{\mathrm{NSD}} \geq 5$ low-rank MNE models were tested where it was found, more often than not, that a feasible point could

not be found among 5,000 trial points. This was fixed by introducing a customized Monte Carlo where the random samples $\{\varepsilon_k\}$ were rearranged such that any trial point was feasible (Eq 3.43).

If one so desires, Bayesian optimization can also be used to find settings for discrete hyperparameters such as r_{PSD} and r_{NSD} which can add an extra degree of flexibility to the optimization procedure. However, this was not implemented.

4.2 Validation on synthetic neurons

A simulated analysis of three synthetic neurons was performed to demonstrate the low-rank MNE method's ability to recover receptive fields. The results were validated against the ground truth receptive field and compared to performance of other dimensionality reduction techniques; specifically, STC, first-order MNE, and full-rank (second-order) MNE methods. The dimensionality reduction techniques were compared and contrasted based on their abilities to recover the correct subspace of stimulus space that spans the receptive field and to predict neural responses to novel stimuli. Both globally optimal approximations (Eq 3.42) and empirical model selection (Eq 3.43) will be shown to have practical merit for recovering receptive fields.

4.2.1 "Auditory" neuron

A synthetic "auditory" neuron was constructed by generating six bivariate Gaussians in (16×16) spectrotemporal space such that the sum of the six Gaussians (Fig 4.2) bears a receptive field resembling a bivariate "Mexican hat" distribution similar to what has been observed in single component auditory receptive fields [76, 77]. These Gaussian components were then combined to form the ground truth matrix $\mathbf{J}_{\text{GT}} \in \mathbb{R}^{D \times D}$ via the weighted outer product $\mathbf{J}_{\text{GT}} = \mathbf{F}\mathbf{W}\mathbf{F}^{\text{T}}$ where $\mathbf{F} \in \mathbb{R}^{D \times r_{\text{GT}}}$ is a matrix with $r_{\text{GT}} = 6$

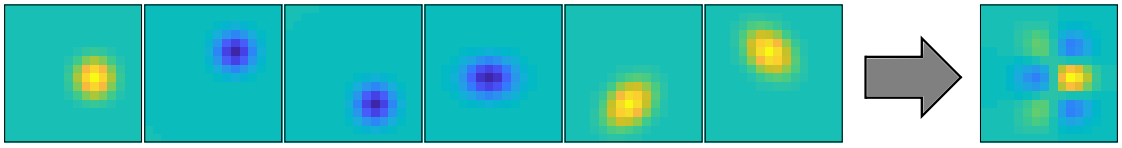

Figure 4.2: **Constructing the receptive field of a synthetic auditory neuron.** A receptive field reminiscent of single component reconstructions of spectrotemporal auditory receptive fields is constructed from bivariate Gaussians plotted left to right in order of high to low variance. The weighted average of these components appears at right following the arrow.

columns corresponding to the six $D = 256$ components in Fig 4.2. $\mathbf{W} \in \mathbb{R}^{r_{\mathrm{GT}} \times r_{\mathrm{GT}}}$ is a diagonal matrix that weighs the contribution of each component to the receptive field. Three of the components are excitatory (positive eigenvalues) while the remaining three are suppressive (negative eigenvalues). The ground truth receptive field is then obtained by diagonalizing $\mathbf{J}_{\mathrm{GT}}$ and the resulting receptive field components are plotted in Fig 4.3.

The synthetic neuron was stimulated by spectrograms (in logarithmic amplitude and linear time/frequency scales) drawn from a mildly-correlated Gaussian noise distribution where each stimulus was generated from a multivariate Gaussian distribution with covariance kernel

$$\mathcal{K} = \mathbf{G}^{(\mathrm{kern})^{\mathrm{T}}} \mathbf{G}^{(\mathrm{kern})} \tag{4.10}$$

where

$$G^{(\mathrm{kern})}_{x_i, y_j} \propto \sum_{k=1}^{D} \sum_{l=1}^{D} \exp\left(\begin{bmatrix} x_i - x_k, & y_j - y_l \end{bmatrix} \mathbf{C}_{\mathrm{cov}}^{-1} \begin{bmatrix} x_i - x_k \\ y_j - y_l \end{bmatrix} \right), \tag{4.11}$$

x_i and y_i are the temporal and spectral coordinates of the spectrogram, respectively, and $\mathbf{C}_{\mathrm{cov}} \in \mathbb{R}^{2\times 2}$ is the covariance matrix $\mathbf{C}_{\mathrm{cov}} = \mathbf{diag}([3,\ 3])$. Intuitively, the elements that make up the spectrogram are most strongly correlated with nearby elements and the correlations are isotropic. The tth stimulus vector is then generated by drawing a vector

$\hat{\mathbf{s}}_t \in \mathbb{R}^{D\times 1}$ from an uncorrelated multivariate normal distribution and multiplying by the matrix square root of the covariance kernel: $\mathbf{s}_t = \boldsymbol{\mathcal{K}}^{\frac{1}{2}}\hat{\mathbf{s}}_t$. A total of $N_{\text{samp}} = 100,000$ total stimulus samples were generated.

The response of the synthetic neuron was fabricated by calculating the nonlinearity, $P(y=1|\mathbf{s}_t)$, according to Eq 3.1 for each of the 100,000 stimulus samples. For each sample t, a uniform random number $\xi_t \in [0,1]$ was drawn. If $\xi_t < P(y=1|\mathbf{s}_t)$, then the response, y_t, was set to one; otherwise, y_t was zero. The synthetic neuron had a mean spiking probability of $\sim 25\%$ across all samples.

The data was first analyzed using STC (see Chapter 1.3.2) using the full data set. The highest variance components of the reconstructed receptive field are shown in Fig 4.3**A**. Looking at this reconstruction, there is clear distortion of the receptive field where the regions of non-zero amplitude are much broader and exhibit additional regions of activity compared to the ground truth in Fig 4.3**A**. This observation is not, however, the result of a benign rotation of the receptive field because the overlap (Eq 2.6) between the reconstructed and ground truth receptive fields is 0.003. Despite appearances, this indicates that the receptive fields largely are not spanning the same subspace. The observed distortion is consistent with what was alluded to in Section 1.3.3 where STC is known to be biased when presented with correlated stimulus distributions.

Full-rank MNE was fit using the empirical early stopping procedure (Chapter 1.5.1) as a mild form of regularization that attempts to limit overfitting. The data was divided into 70% training, 20% cross-validation (used for early stopping), and 10% testing sections where each sample belonged to one section only. Jackknife analysis was performed by circularly shifting the indices of the samples in each of the sections by 25% of the total samples as shown in Fig 4.4. Following optimization, $\mathbf{J}$ was averaged across the four jackknives and was then diagonalized, revealing the receptive field captured by the three largest positive and three most negative variance components in Figs 4.3**A**

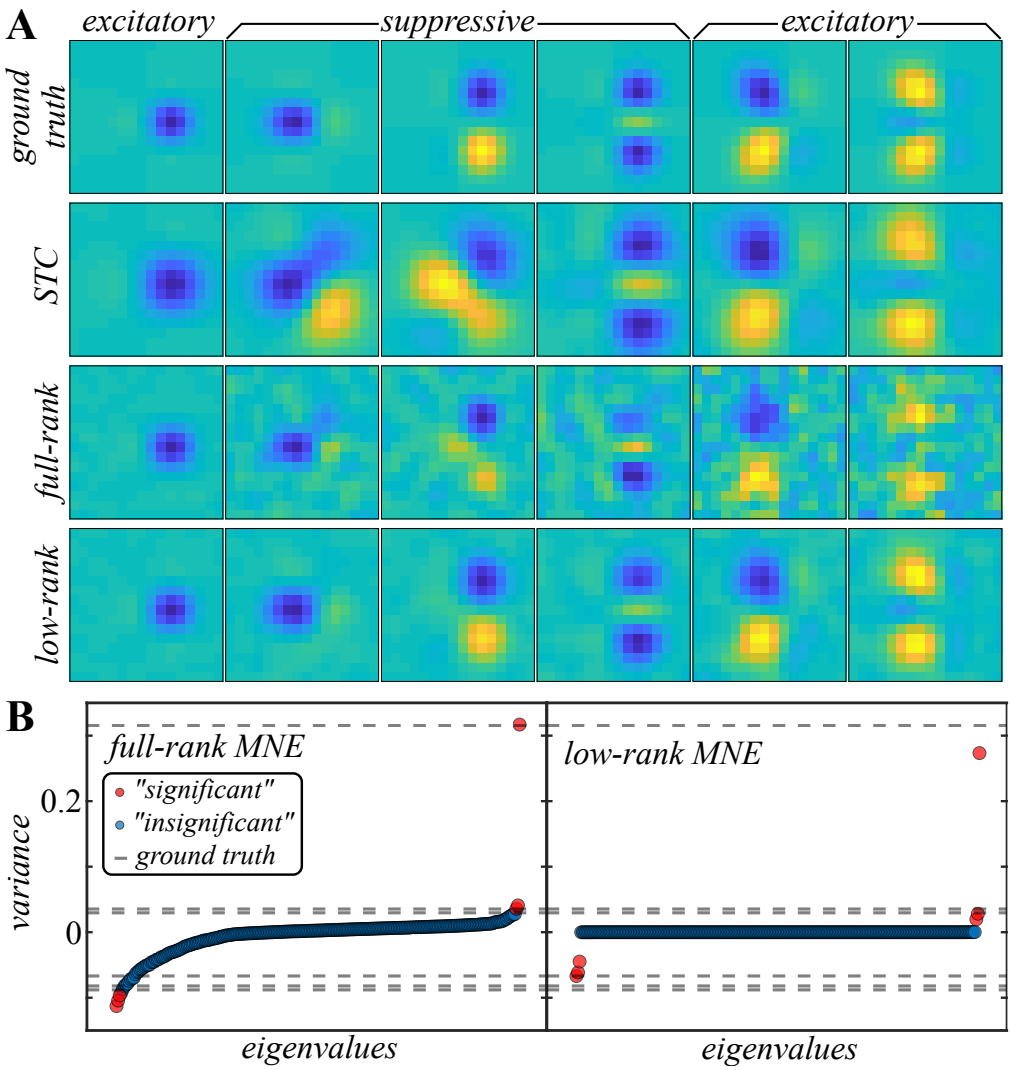

Figure 4.3: **The receptive fields of the synthetic auditory neuron recovered using STC, full-rank MNE, and low-rank MNE methods.** (**A**) The ground truth receptive field of the synthetic auditory neuron is plotted beside the reconstructions estimated by the six most significant components from each dimensionality reduction method. The ground truth is plotted from left to right from highest to lowest variance. (**B**) The eigenvalues of the full-rank and low-rank MNE models are plotted for comparison to the ground truth (dashed lines).

& **B**. Full-rank MNE appears to have done a reasonable job of capturing the first three or four components, but the fifth and sixth components are heavily corrupted by noise. Notably, two of the largest positive variance components have lower magnitude than several fictitious negative components and five of the six significant components are

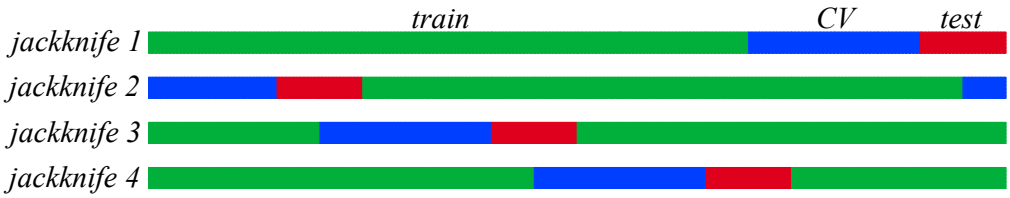

Figure 4.4: **Diagram showing how the data is divided into training, cross-validation, and test sets.** Analysis is performed on four jackknives where data is split between training (green), cross-validation (blue), and test (red) sets.

nearly engulfed by fictitious components. One may be tempted to say that STC has outperformed full-rank MNE in this task given the lack of noise corruption in the STC reconstruction. However, even with its flaws, full-rank MNE still produces a better subspace with an overlap of 0.7 ± 0.1.

Low-rank MNE models with $r_{\text{PSD}} = 3$ and $r_{\text{NSD}} = 3$ were optimized to find a globally optimal approximation via Alg 4.1 with the equality constraints (Eq 3.8) directly inserted into the nonlinearity (Eq 3.1) and L-BFGS was used to minimize the subproblems. The analysis was performed with the data divided into jackknives in exactly the same way as the full-rank MNE analysis (Fig 4.4). The amount of regularization necessary to transition to the globally optimal regularization domain (Prop 3.4) across all jackknives was ~ 0.045 which was low enough that distortion was not a substantial issue. The six largest variance components obtained from diagonalizing the mean $\mathbf{J}$ across jackknives are plotted in Fig 4.3**A**. Unlike the full-rank MNE models, the low-rank MNE models capture all six of the ground truth components (Fig 4.3**A**) with remarkable clarity and without referencing the cross-validation set. The low-rank MNE model's receptive field captures the ground truth components almost exactly with an overlap of 0.9861 ± 0.0009 on average across jackknives.

One potential drawback of the low-rank MNE method is that nuclear-norm regularization may ultimately attenuate the components' variance relative to the ground truth which may lead to either distortion of the receptive field (when the regularization

parameters are large) and perhaps worse predictions on the test sets. Of course, the latter is less important if the ultimate goal is not to make predictions but instead to employ the recovered receptive field in a separate linear-nonlinear model or to compute a functional basis. The attenuation of the components can be seen in the eigenvalue spectrum of the mean **J** that appears in Fig 4.3**B** where all six of the outstanding eigenvalues exhibit a noticeable damping of their absolute variance. This reduction in magnitude did not end up being detrimental to the predictive power of the low-rank MNE models where the mean negative log-likelihood evaluated on the test sets was $L = 0.210 \pm 0.003$. This is a significant improvement over the full-rank MNE model where $L = 0.229 \pm 0.003$ and the first-order MNE model where $L = 0.564 \pm 0.002$.

The low-rank MNE method was used to recover a globally optimal low-rank compression of the receptive field where $r_{\text{PSD}} = 2$ and $r_{\text{NSD}} = 2$ and a globally optimal expanded basis where $r_{\text{PSD}} = 4$ and $r_{\text{NSD}} = 4$. The mean compressed and expanded basis receptive fields may be found in Fig 4.5. The compressed receptive field still has good overlap (0.9752 ± 0.0007) indicating that the compressed components largely lie within a smaller subspace of the ground truth receptive field, as ought to be the case for a good compression. In terms of prediction error, the compressed models perform better than the first-order MNE models with $L = 0.278 \pm 0.002$. The compressed models perform worse by this measure compared to the full-rank MNE models, but that is because predictions from the full-rank MNE models are made using all components of **J** sans dimensionality reduction. The expanded basis (Fig 4.5) performs equivalently to the $r = 6$ low-rank MNE models from above with a prediction error of $L = 0.210 \pm 0.003$. The difference in the negative log-likelihood estimate is not statistically significant. These two models perform equivalently because the two extra dimensions that appear in the expanded basis have variance $\sim 10^{-4}$ and do not contribute significantly to the predicted response. The subspace is also of high quality since the ground truth receptive field

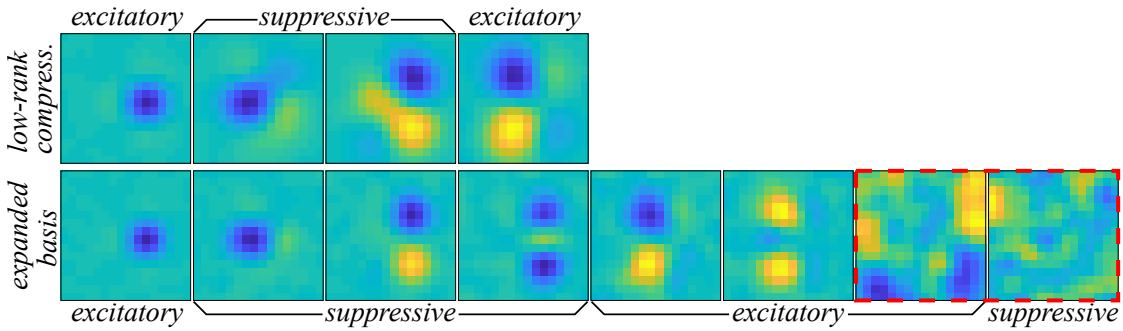

Figure 4.5: **A globally optimal compression and expansion of the synthetic auditory neuron's receptive field using the low-rank MNE method.** The extra dimensions in the expanded basis boxed in the red dashed-line line have variance $\sim 10^{-4}$.

components lie almost perfectly within the recovered receptive field yielding an overlap of 0.9894 ± 0.0009.

4.2.2 "Vision" neurons

In some cases, reaching a globally optimal approximation (Eq 3.42) may require an excessively large nuclear-norm regularization parameter such that the components spanning the receptive field become heavily distorted. This problem may also occur when performing empirical model selection (Eq 3.43) with a single nuclear-norm regularization parameter when the components that span the receptive field have dissimilar absolute variances. In such cases, empirical model selection with multiple regularization parameters may produce better models. Fitting low-rank MNE models using empirical model selection will be demonstrated on two vision-inspired synthetic neurons that share the same vaguely center-surround receptive field one may expect to encounter in the early visual system [78, 79]. As with the synthetic auditory neuron, the receptive field is constructed from $r_{\text{GT}} = 4$ bivariate Gaussians (Fig 4.6) in a 20×20 pixel image space ($D = 400$). From here, the data is generated with an identical procedure to that of the synthetic auditory neuron with one exception: instead of generating correlated Gaussian stimuli from the covariance kernel in Eqs 4.10 & 4.11, the covariance kernel is

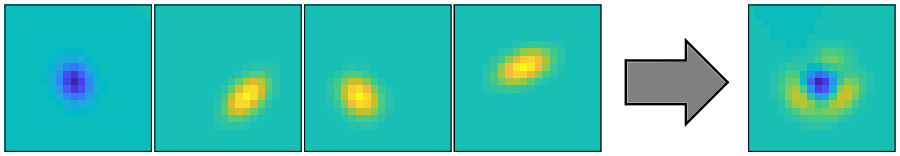

Figure 4.6: **Constructing the synthetic vision neurons' receptive field.** The high and low SNR synthetic vision neurons have a center-surround type receptive field constructed from bivariate Gaussians.

instead derived from a covariance matrix of images from the Van Hateren natural image database [80]. The total number of stimuli and response samples is $N_{\text{samp}} = 48,510$.

The two neurons are distinguished by the gain of the nonlinearity where one neuron has a high signal-to-noise ratio (SNR) while the other has a low SNR. To be clear, what is meant by SNR here is the relative decisiveness of the neuron where the high SNR neuron is more likely to have spiking probability closer to either zero or one relative to the low SNR neuron which is more likely to take on intermediate probabilities. The total number of spikes elicited by the high SNR neuron was 11,031 and the low SNR neuron was 10,434.

As before, STC and full-rank MNE methods are used as a performance standard. However, for these neurons, the STC method is performed on each of the jackknives (in the same 70%/20%/10% sections as in Fig 4.4) to compare the predictive power of STC to the other dimensionality reduction methods. STC performs as expected as seen in Fig 4.7**B** where the non-zero amplitude of the four largest variance components is distorted even more so than was the case for the synthetic auditory neuron in Fig 4.3 caused by the stronger correlations in the stimulus distribution. This distortion is manifested quantitatively by a mean overlap of 0.32 ± 0.05 across jackknives for both neurons. Dimensionality reduction was performed in the typical way for STC (see Sections 1.3.2 & 2.4) and the prediction error on the test sets was determined by fitting a full-rank MNE model with the stimuli projected into the STC receptive field (in order to place the competing receptive field estimates on equal footing) and computing the mean negative

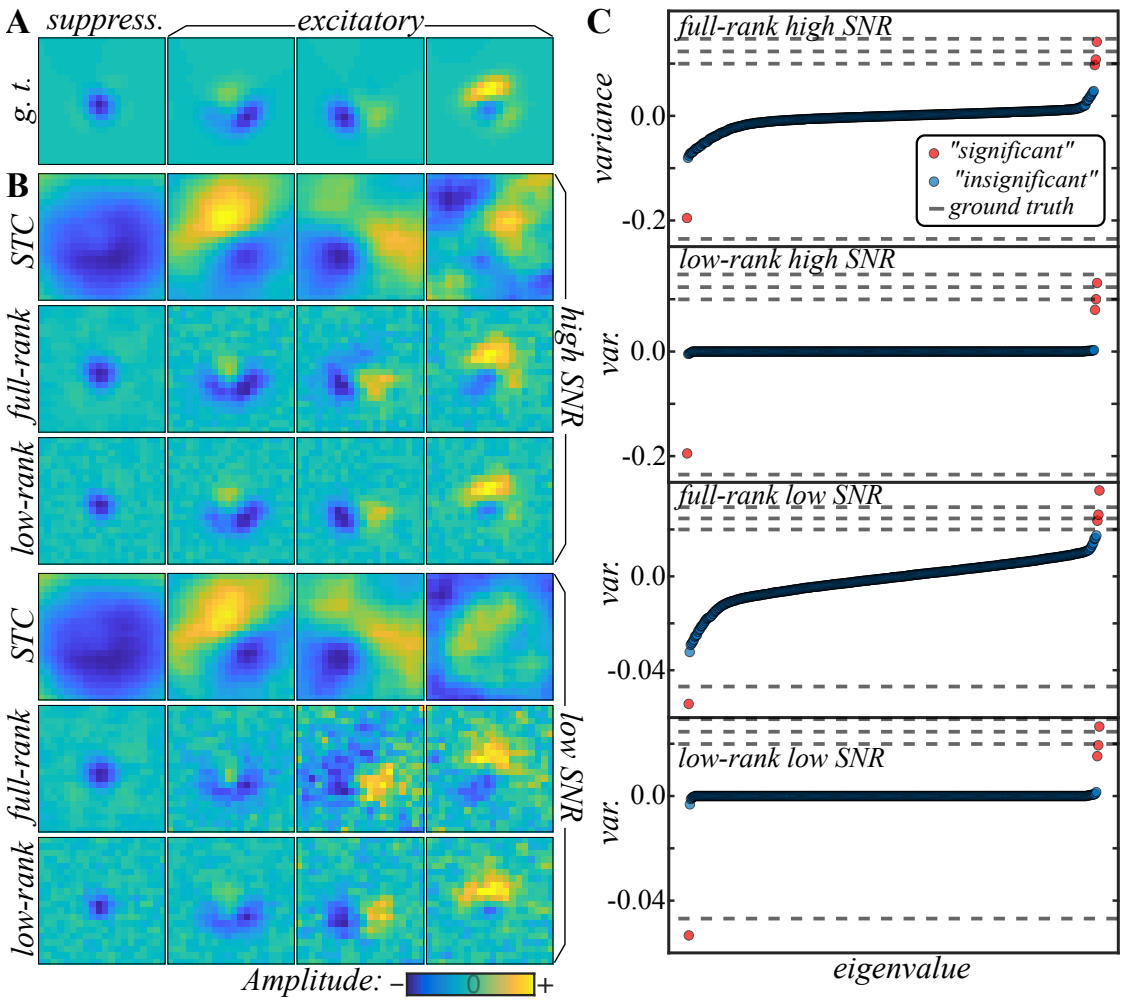

Figure 4.7: **Receptive fields recovered using STC, full-rank MNE, and the empirical model selection approach to the low-rank MNE method for two synthetic vision neurons.** (**A**) The ground truth receptive field of the high and low SNR neurons is plotted as four orthonormal vectors. (**B**) The four "significant" components from the STC, full-rank MNE, and low-rank MNE methods demonstrate how well each method recovers the receptive field for each neuron. (**C**) The mean eigenvalue spectrum from each method is plotted for comparison against the ground truth spectrum (dashed lines) of each neuron.

log-likelihood across the test sets. The prediction error of STC with a logistic nonlinearity was $L = 0.40 \pm 0.01$ and $L = 0.47 \pm 0.01$ for the high and low SNR neuron, respectively.

Compared to STC, the receptive field recovered by the full-rank MNE models leads to an improvement on the high SNR neuron (Fig 4.7**B**) where the four largest variance components have an overlap of 0.909 ± 0.008 and the models have a mean

prediction error of $L = 0.32 \pm 0.02$. However, on the low SNR neuron the full-rank MNE models performed worse with an overlap of 0.17 ± 0.05 and $L = 0.50 \pm 0.01$. The overlap is so poor in this case because the four largest variance components do not correspond to the four most significant or relevant components in this case (see Fig 4.7**C**). Two of the positive variance components have a lower absolute variance than the that of some of the fictitious components. Unlike in Kaardal et al. [72] where the four largest variance components are analyzed, the rigidity of using variance to determine the most relevant components is relaxed for the moment to reveal the two right-most components in Fig 4.7**B**. Using these significant components instead, the overlap increases to 0.74 ± 0.02 and, despite the dominance of noise in the two lowest absolute variance components, this choice better represents the underlying receptive field than STC. This problem of choosing significant components will be returned to later on in the discussion section.

Low-rank MNE models were optimized using the block coordinate descent heuristic in Alg 4.2 where the interior-point method [41] from Section 3.3 was used to solve the block subproblems. Models were fit for each $r \in \{1, \cdots, 8\}$ using the signs of the r largest variance eigenvalues of $\mathbf{J}$ averaged across jackknives from the full-rank MNE models of the high SNR neuron to assign r_{PSD} and r_{NSD}. The grid search iterated over the domain $\varepsilon_k \in [0, 1]$ for all k with a resolution of 0.001 and the block coordinate descent heuristic proceeded until $M_{\text{fail}} = 3$, the maximum number of consecutive cycles through the blocks that failed to uncover a solution that reduced the generalization error. The upper bound on the regularization parameters was chosen because $\mathbf{U}_{\bullet,k} \approx \mathbf{0}$ when $\varepsilon_k = 1$. Since $\min\left(\{\varepsilon_k^*\}\right)$ (i.e. the regularization parameter settings that minimize the negative log-likelihood evaluated on the cross-validation set) did not satisfy the condition required for certifiable global optimality (Eq 3.37) for all optimized models on either neuron, these low-rank MNE models lie in the locally optimal regularization domain.

In Section 3.4.2, it was proposed that low-rank MNE models may be optimized

by setting an expected upper bound on r_{PSD} and r_{NSD}. This proposal was tested and the results are shown in Fig 4.8**A** where one expects the negative log-likelihood evaluated on the cross-validation set to saturate for $r \geq 4$ much like what was observed for the functional basis set size (Section 2.3, in particular Fig 2.2**F**). Indeed, the predicted saturation occurs at $r = 4$ where all $r > 4$ models are equivalent to the $r = 4$ model. This is possible because the extra components in the $r > 4$ models are eliminated by the regularization parameters and have variance approximately equal to zero. By the $r = 6$ models, both $\mathbf{J}^{(\text{PSD})}$ and $\mathbf{J}^{(\text{NSD})}$ are rank deficient with rank below r_{PSD} and r_{NSD}, respectively, satisfying the convergence condition from the upper bounding procedure in Section 3.4.2. The compressed models also behave as expected, having higher generalization error than the $r \geq 4$ models while each additional component up to $r = 4$ leads to improvement. This analysis provides evidence that setting upper bounds on the rank can be used to fit models of optimal rank.

The recovered receptive fields of the $r = 4$ models appear in Fig 4.7**B** for both the high and low SNR neurons. Qualitatively, the recovered receptive fields do particularly well at capturing the details of the lower variance components relative to the receptive fields captured by the full-rank MNE method. The overlap of the recovered receptive fields of the $r = 4$ low-rank MNE models was 0.933 ± 0.007 and 0.83 ± 0.02 for the high and low SNR neurons, respectively. The prediction error was $L = 0.233 \pm 0.009$ for the high SNR neuron and $L = 0.45 \pm 0.01$ for the low SNR neuron. Each of these measures indicate that the low-rank MNE models much better capture the receptive field and predict responses than the STC and full-rank MNE models. The overlaps and prediction errors of the models are summarized in Fig 4.8**B**.

Returning to the statement at the beginning of this section, there are improvements that may be gained by finding solutions in the locally optimal regularization domain on this problem. Using Alg 4.1 to find globally optimal approximations yields the

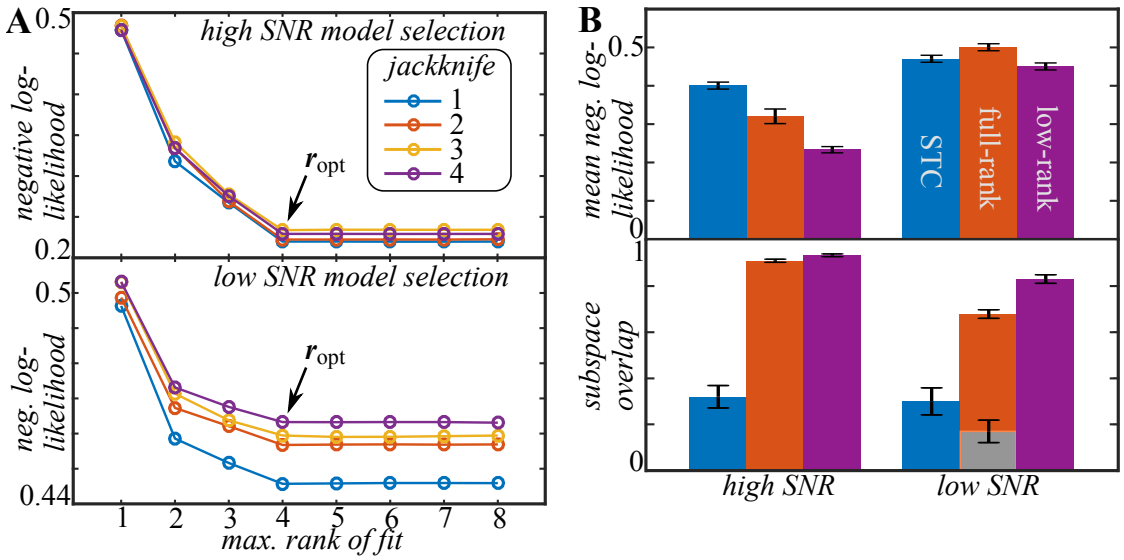

Figure 4.8: **Selecting the best model among STC, full-rank MNE, and the empirically selected low-rank MNE models for two synthetic vision neurons.** (**A**) Plots of the prediction error of low-rank MNE models evaluated on four cross-validation sets as a function of rank saturate as the rank increases above the optimal rank of $r = 4$ ($r_{\text{PSD}} = 3$ and $r_{\text{NSD}} = 1$). (**B**) A summary of performance measures from different dimensionality reduction methods show that the low-rank MNE models perform best. The gray bar on the bottom plot refers to the mean overlap of the four largest variance components of the full-rank MNE models while the orange bar is the mean overlap of the full-rank components plotted in Fig 4.7**B**.

receptive field estimates in Fig 4.9. The estimate for the high SNR neuron has a similar appearance to the full-rank MNE reconstruction in Fig 4.7**B** though with some apparent smoothing of the noise. Compared to the full-rank MNE model, the globally optimal approximation is still an improvement with an overlap of 0.954 ± 0.003 and prediction error $L = 0.252 \pm 0.006$. While the subspace overlap is higher than the receptive fields obtained through empirical model selection, the empirical model selection produces receptive fields that better predict the neural responses by a large margin. With respect to the low SNR neuron, the globally optimal approximation exhibits substantial distortion reducing the overlap to 0.85 ± 0.02 which is comparable to the locally optimal models. However, the locally optimal models still have the edge in predicting responses since the globally optimal approximation has a prediction error of $L = 0.46 \pm 0.01$ on the low

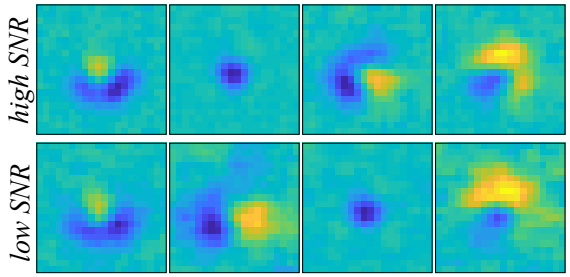

Figure 4.9: **Globally optimal approximations to the receptive field of two synthetic vision neurons.** The recovered receptive fields of the high and low SNR neurons are shown in order of highest (left) to lowest (right) absolute variance.

SNR neuron.

The fact that the empirical model selection procedure leads to better predictions is likely a results of targeting solutions towards generalizability. The cross-validation procedure has the additional advantage of allowing optimization from an upper bound on r. By contrast, the globally optimal approximation cannot be relied upon to saturate above $r_{\text{PSD}} = 3$ and $r_{\text{NSD}} = 1$ since the necessary regularization is inversely proportional to these quantities (a consequence of the bracketed term in Eq 3.29). This is shown in Fig 4.10 where, compared to the saturation observed in Fig 4.8**A**, the globally optimal approximations do not reach a definitive saturation for $r \geq 4$, especially for the low SNR neuron where the optimal rank is dubious. Therefore, the upper bounding from Section 3.4.2 suggested for use in computing generalizable models of optimal rank is unreliable for globally optimal approximations.

Another area in which the solutions in the locally optimal domain outperformed the globally optimal approximation was in more closely capturing the variance of each component. The components in Fig 4.9 are presented deliberately in order of high to low variance to emphasize that the components have not maintained the same order as a function of absolute variance from the ground truth in Fig 4.7**A**. If one compares the eigenvalue spectra of the globally optimal approximations in Fig 4.11 to those of the locally optimal models in Fig 4.7**C**, the attenuation of the eigenvalue spectra in the

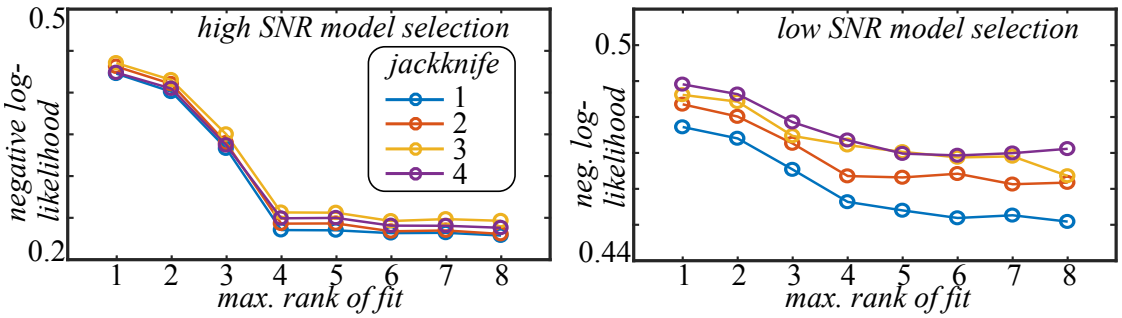

Figure 4.10: **Model selection between globally optimal approximations of different rank is unreliable.** The prediction error of the globally optimal approximations is measured on the cross-validation sets for the high and low SNR synthetic neurons. Unlike empirical model selection, the globally optimal approximations do not as decisively saturate for $r \geq 4$ low-rank MNE models.

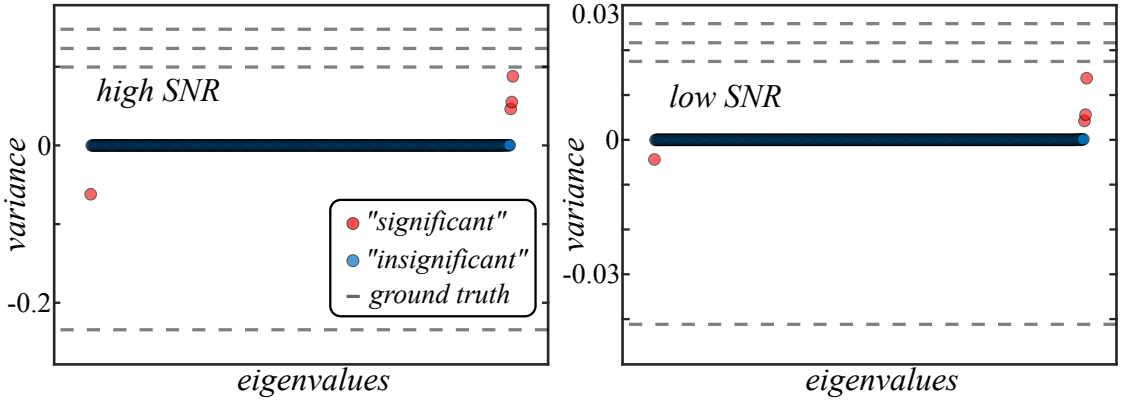

Figure 4.11: **Eigenvalue spectra of the globally optimal approximations to the receptive fields of the two synthetic vision neurons.** Eigenvalue spectra of mean **J** from the globally optimal approximation are plotted for the high and low SNR neurons.

globally optimal models is much more severe, being much closer to the baseline than to the ground truth eigenvalues.

Finally, the experiments on the synthetic neurons conclude with a test of Bayesian optimization to determine regularization parameter settings that minimize the generalization error. Low-rank models with ranks $r_{\text{PSD}} = 10$ and $r_{\text{NSD}} = 10$ were fit to recover the receptive fields of the high and low SNR neurons. The domain of the regularization parameters was $\varepsilon_k \in [0, 1]$ for all k. In the Bayesian optimization software [75], the primary bottleneck was found to be the update of the hyperparameters ν and η in the

Mat'ern 3/2 kernel (Eq 4.7) since the problem size grew superlinearly as a function of the number of iterations. To mitigate this, the hyperparameters were fixed to $\nu^2 = 1$ and $\eta = 0.1$ which appeared to work well in practice. The Bayesian optimization began with up to 300 iterations allocated to first solving Eq 3.43 with all ε_k constrained to be equal to ε, a universal regularization parameter. Following this phase, the regularization parameters were allowed to vary independently, only constrained by the inequality constraints in Eq 3.43 and the box constraints defined by the domain. The exploration weighting, τ in Eq 4.8, was annealed from 2 to 0 over the remaining 600 iterations to encourage exploration at the beginning and exploitation as the optimization neared its imposed end. L-BFGS was used to optimize the subproblem to the right of the bracket in Eq 3.43 with the equality constraints on $\mathbf{U}$ and $\mathbf{V}$ directly inserted into the nonlinearity.

The results of the Bayesian optimization are featured in Fig 4.12. The eight largest variance components are pictured in Fig 4.12 to show what becomes of the insignificant components in the low-rank MNE method. The insignificant components (those boxed in by red dashed-lines) are almost entirely composed of random noise and have variance that is approximately zero. The significant components match closely to what was found using Alg 4.2. The overlap of the four largest variance components and the ground truth components (Fig 4.7**A**) is 0.93 ± 0.01 and 0.85 ± 0.01 for the high and low SNR neuron, respectively. The prediction error is $L = 0.234 \pm 0.009$ for the high SNR neuron and $L = 0.45 \pm 0.01$ for the low SNR neuron. The results are equivalent to those found using Alg 4.2 and either algorithm may be used to find solutions for the low-rank MNE problem in Eq 3.43.

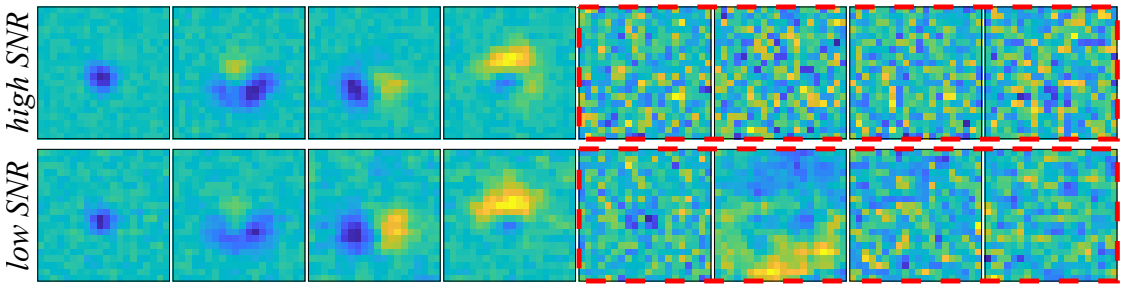

Figure 4.12: **Receptive field components of two synthetic vision neurons recovered using Bayesian optimization.** The reconstructed receptive fields of the high and low SNR neurons where empirical model selection is performed using the Bayesian optimization approach appear with four extra insignificant components boxed in the red dashed lines.

4.3 Neural coding in the auditory system

Here it is sought to make progress on three contemporary problems in the field of computational neuroscience. The first is the more general goal of showing that the low-rank MNE method is a novel dimensionality reduction technique that may be used to recover multiple receptive field components from both low and high-levels of sensory processing whereas this goal has been out of reach with previous methods. High-level, in this case, refers to neurons at a depth in the neural circuitry where the neurons become selective for sophisticated representations of objects in and properties of the environment that depart from the simplicity of, for example, (for vision neurons) center-surround and (for auditory neurons) monotones and "clicks" (a fleeting burst of sound that spans a wide range of perceptible frequency). There is, of course, no strict threshold above which a neuron can be defined to be "sufficiently sophisticated" as to be high-level. However, a working definition of high-level sensory neurons are those that are unlikely to be stimulated by external sensory stimuli drawn from an uncorrelated noise distribution over the course of a typical experiment. The second problem where progress is made is on characterizing multicomponent receptive fields of high-level auditory neurons which has not, as far as the author is aware, been achieved with confidence in prior studies. The

final problem that is addressed is a modest description of the functional neural circuitry in high-level auditory regions, especially insofar as it suggests a divergence in how neurons from some auditory and visual regions process sensory input.

4.3.1 The dataset and preprocessing procedure

Data from the CRCNS database used with permission from the Theunissen laboratory at the University of California, Berkeley was composed of *in vivo* electrophysiological recordings of anesthetized adult male zebra finches subject to auditory stimulation [81, 82]. From this dataset, analysis focused on neurons from the high-level auditory regions field L and the caudal mesopallium (CM) from the auditory forebrain. Single neuron action potentials were recorded from a total of 189 neurons from field L and 37 neurons from CM. The responses were sampled at a 1 ms resolution. These responses were elicited due to stimulation by two types of auditory stimuli: (i) 2 second samples of conspecific birdsong recorded from 20 male zebra finches and (ii) 10 synthesized spectrotemporal ripple stimuli. These stimuli were bandpass filtered from 250 Hz to 8 kHz (within the perceptible frequency range of the zebra finch [83, 84]). Each stimulus was presented up to ten times to the zebra finches via speakers in a sound attenuation chamber.

The stimuli were preprocessed by first transforming from amplitude-time space to spectrotemporal space with the spectrogram function in MATLAB. Using a Hamming window where neighboring windows had a 50% overlap, it was found that a 250 Hz frequency resolution coupled with a 2 ms temporal resolution provided a reasonable balance between the spectral and temporal resolution of the spectrograms as observed through visual inspection of the linear weights ($\mathbf{h}$) of first-order MNE models. With these settings, the linear weights resolved clear spectrotemporal structure (without significant bias along either the spectral or temporal axes) while providing plenty of stimulus/response pairs.

The response was provided as a list of spike times that were accumulated in equally spaced temporal bins with 2 ms width to match the resolution of the spectrotemporal stimuli. Any empty bins were set to zero (no spikes). For repeated trials of stimuli, the corresponding response bins were averaged across trials providing an estimate of the mean firing rate given a stimulus. Since the MNE methods require that $y_t \in [0, 1]$, each bin is divided by the maximum firing rate, $y_{\max} = \max\left(\{y_1, y_2, \cdots, y_{N_{\text{samp}}}\}\right)$ which is equivalent to upsampling the temporal resolution by the factor $y_{\max}$.

Stimulus samples were extracted from the spectrograms by taking 40-60 ms windows preceding the response y_t and unrolling the spectrogram into the stimulus vector $\mathbf{s}_t$. To reduce the dimensionality of the stimulus vectors, first-order MNE was used to refine the frequency range of stimuli where frequency bins well above and below the observed single component receptive fields were excluded. The stimulus/response pairs were then randomly shuffled to break correlations between neighboring stimuli and to ensure a wide sampling of the stimulus/response distribution in the training, cross-validation, and test sets which were divided into 70%/20%/10% sections as was previously done with respect to the synthetic neurons (Fig 4.4).

Of the 189 field L and 37 CM neurons, 41 field L and 9 CM neurons were selected for further analysis. This selection was based on whether a spectrotemporal window could be identified exhibiting observable structure in the single component receptive field estimate. The alignment between the stimulus and response samples (due to a time delay between presentation of stimuli and recording of spikes) and spectrotemporal windowing of the stimuli were adjusted manually until any observed amplitude of the receptive field was confined to the spectrotemporal window. Initially, the STA method was used due to its simplicity. However, in the best cases, it was prone to overestimating the spectrotemporal extent of the receptive fields (e.g. the bias observed in Fig 4.9) leading to more windows that were at times much larger than necessary. In the worse

(and indeed most) cases, the STA was misleading, often exhibiting amplitude at any alignment; even those greater than 100 ms from the spike-onset time. The first-order MNE method was not prone to these well-known biases of the STA and was determined to be an adequate substitute for the STA. The 50 neurons chosen were those that exhibited spectrotemporal structure in the linear weights of the first-order MNE models. Second-order methods such as STC and second-order MNE were not applied to this windowing procedure for the sake of expediency and because the first-order MNE method is usually a linear combination of the second-order components making it an adequate proxy for determining the spatiotemporal extent of a multicomponent receptive field. While this procedure may bias the subsequent analysis towards first-order MNE models, the later results show that this bias, if it exists, can still be overcome by the low-rank MNE models.

Among this subset of neurons, the number of samples range from 9,800 to 58,169 with a median of 42,474. The total number of spikes (prior to preprocessing) ranged from 276 to 29,121 with a median total of 6,120. STC, first-order MNE, full-rank MNE, low-rank MNE, and functional basis methods were optimized across four jackknives for each of the neurons.

4.3.2 Multicomponent receptive fields of high-level auditory neurons

In this section, it is shown that the low-rank MNE method can recover multicomponent receptive fields of high-level auditory neurons from regions field L and CM of the songbird brain. Since the number of components that span the receptive fields of the auditory neurons is not known *a priori*, the upper bounding procedure from Section 3.4.2 is used. Tests using the globally optimal approximation method (Eq 3.42 and Alg 4.1) found that the amount of regularization necessary to reach the globally optimal regularization domain (Prop 3.4) was comparable to the model neurons in the prior section.

However, as touched on in the prior section, the upper bounding method can and did fail on the tested neurons to find appropriately sparse receptive fields. While globally optimal approximation and empirical model selection produced models that had similar prediction error on the tested neurons, the empirical model selection did so with fewer non-zero components and with greater clarity. Under the circumstances it was therefore decided to search for generalizable models via empirical model selection rather than globally optimal approximations to the receptive fields. Since the ground truth was not known, the overlap metric could not be applied here. Instead, the relative quality of the models was judged first by prediction error (i.e. the model that minimizes the negative log-likelihood across test sets) and second, when the two models had similar prediction error, by the model that is able to attain that prediction error while optimizing the least number of weights.

Solutions to the low-rank MNE problems were found using the block coordinate descent heuristic (Alg 4.2) with the interior-point method [41] from Section 3.3 used to solve the block subproblems (Eq 3.41). This choice was made over Bayesian optimization simply because the data analysis with Alg 4.2 was already completed before Bayesian optimization was considered as an alternative. Optimized models with upper bounds $r_{\text{PSD}} = 10$ and $r_{\text{NSD}} = 10$ were found to satisfy the conditions $\text{rank}\,(\mathbf{J}_{\text{PSD}}) < 10$ and $\text{rank}\,(\mathbf{J}_{\text{NSD}}) < 10$ to $\sim 10^{-4}$ precision on at least two jackknives (and usually all) for each neuron in the population.

One remaining data analysis challenge was resolving which components were significant contributions to the receptive field after averaging $\mathbf{J}$ across jackknives since $\text{rank}\,(\mathbf{J})$ may be slightly different for each jackknife. An approach to significance testing based on random matrix theory was used to make this determination and is described in Appendix C. The rank of $\mathbf{J}$ from each jackknife was usually similar, being only different by a few components, so the disparity is not overly troublesome.

The multicomponent receptive fields of two example neurons, named **blabla0713_3_B** and **oo2015_8_A**, recovered using the low-rank MNE, full-rank MNE, and STC methods appear in Fig 4.13. The components shown are the largest variance components of the receptive fields where the number of components plotted was determined through significance testing of the eigenvalues of mean **J** from the low-rank MNE models. The low-rank MNE method produced much sharper components that were more localized in both frequency and time compared to either of the full-rank MNE and STC models. The contrast between the STC components and the low-rank MNE components is particularly stark, though the failure of STC is expected given the statistics of the stimuli. The largest absolute variance components of the low-rank and full-rank MNE models bear a strong resemblance, but the structures that appear in the components diverge for the lower absolute variance components at the right-hand-side of Fig 4.13. For example, the four lowest absolute variance components in Fig 4.13 in the full-rank model present receptive field components that feature frequency gratings with long temporal extent that one may surmise to be fictitious given their absence in the first-order and low-rank MNE receptive fields, given that the prediction errors of the first-order and low-rank MNE models are significantly reduced compared to the full-rank MNE model. The low-rank MNE models also outperformed both the STC models on these two example neurons. For neuron **blabla0713_3_B**, the prediction error was 0.133 ± 0.002, 0.158 ± 0.004, and 0.16 ± 0.02 for the low-rank MNE, full-rank MNE, and STC models, respectively (where the prediction error of the STC models was computed using a logistic nonlinearity as described in Section 4.2.2). For neuron **oo2015_8_A**, the prediction error was 0.166 ± 0.007, 0.188 ± 0.007, and 0.19 ± 0.03 for the low-rank MNE, full-rank MNE, and STC models, respectively.

The improvement of the low-rank MNE models over the full-rank MNE models for the two example neurons is sustained over the entire neuron population. In Fig 4.14**A**,

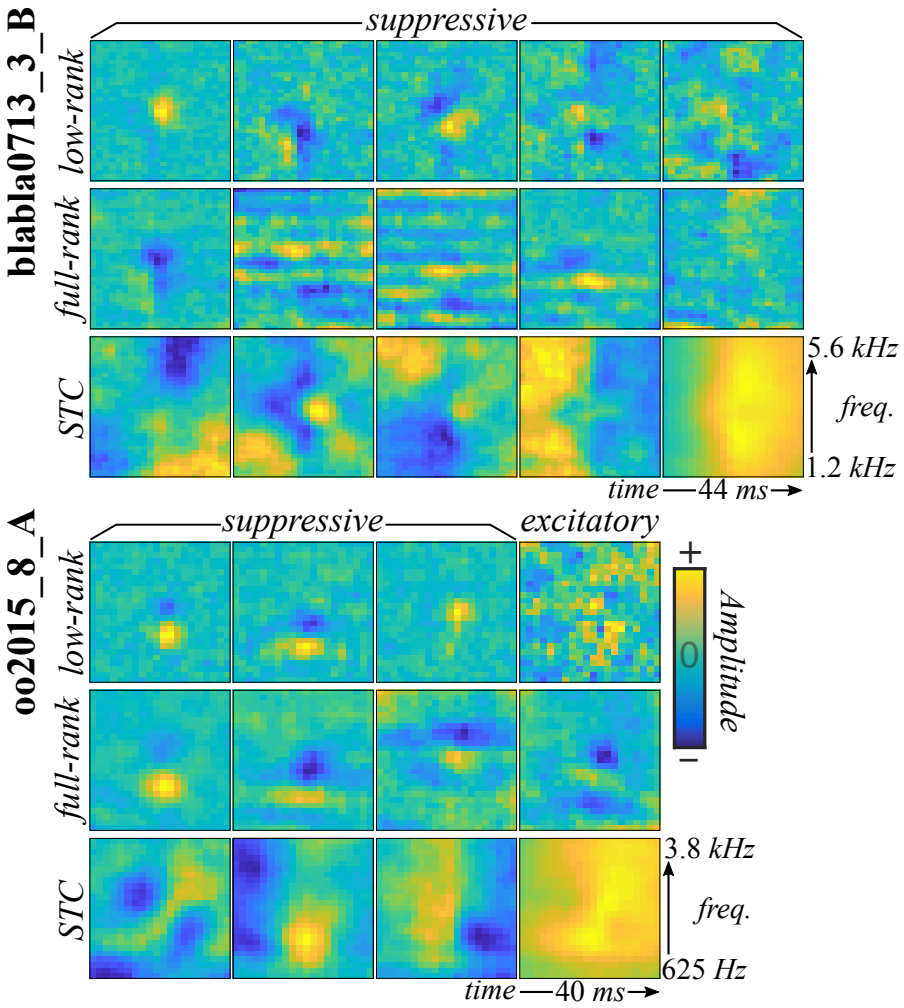

Figure 4.13: **Multicomponent receptive fields recovered from two field L neurons.** Multicomponent receptive fields of two field L neurons, **blabla0713_3_B** and **oo2015_8_A**, are recovered via the low-rank MNE, full-rank MNE, and STC dimensionality reduction methods. The MNE components are ordered from highest (left) to lowest (right) variance.

two graphs show how the prediction error of the low-rank MNE models compares to the full-rank MNE and STC models using the first-order MNE models as a baseline for comparison. The left-most plot is particularly enlightening not just because the low-rank MNE models universally outperform the full-rank MNE models, but the full-rank MNE models are also outperformed by the first-order MNE models for all neurons in the population even though the first-order MNE models recover only a single component.

This may provide evidence for why second-order methods have failed to characterize high-level auditory neurons in the past: none of the second-order components are significant. Furthermore, this gives reason to be skeptical of the observed structure in the full-rank MNE components. While the full-rank MNE receptive fields are insignificant, most of the low-rank MNE models by contrast perform better than the first-order MNE models suggesting that the low-rank MNE models may be more reliable estimates of the spectrotemporal receptive fields (Fig 4.14**B**). To be more specific, the low-rank MNE method leads to improved predictions over the first-order MNE models in 37 of the 50 neurons, including the two example neurons in Fig 4.13. Unsurprisingly, the low-rank MNE models are also better at predicting responses on the test sets than the STC models as shown in the central graph of Fig 4.14**A**. To try to improve the STC results, STC was performed on zero-phase whitened [56] stimuli but the receptive fields in the whitened space lead to substantially worse predictions over unwhitened STC [72].

The novel components characterized by the low-rank MNE method have provided new insights into the receptive fields of high-level sensory neurons. Additionally, these results demonstrate an advancement in techniques for reconstructing receptive fields of high-level sensory neurons. Based on the provided evidence, one might surmise that the low-rank MNE method may be key to making at least partial progress on solving the standing problems in computational neuroscience that were introduced at the beginning of this section.

4.3.3 Functional neural circuitry of high-level auditory neurons

Receptive field reconstructions are alone limited in their ability to provide insight into the conceptual aspects of neural computation. As such, more may be learned about the auditory system by studying more informative aspects of sensory systems such as the functional neural circuitry. Here the functional basis method [51] (Chapter 2)

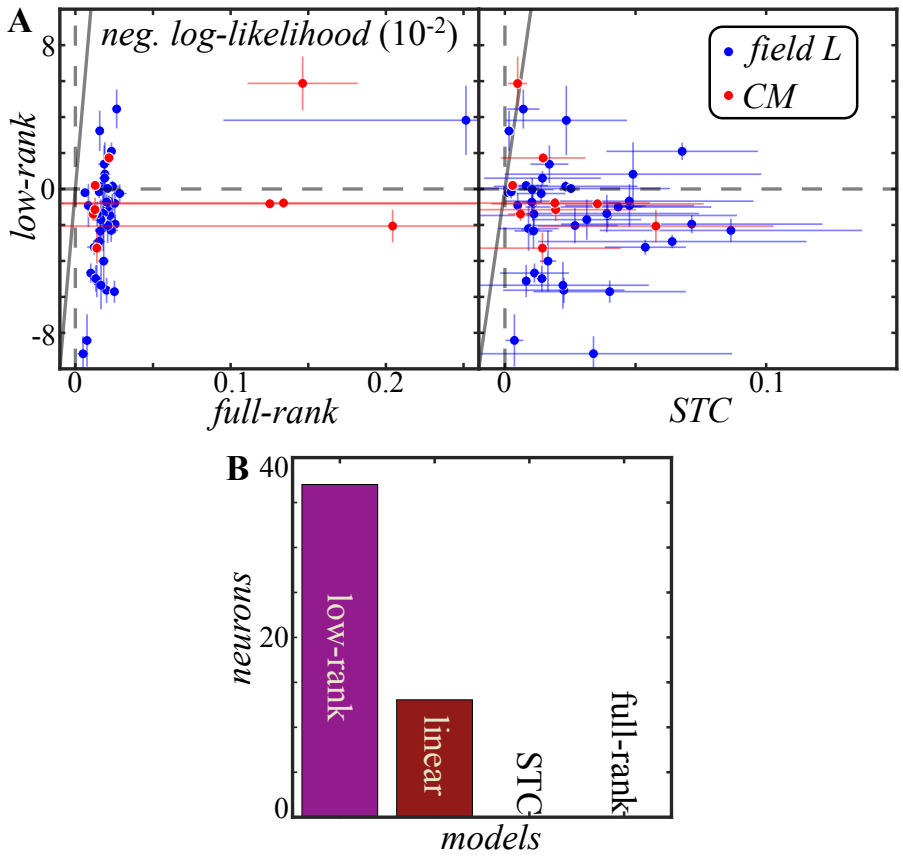

Figure 4.14: **Selecting the best model among STC and first-order, full-rank, and low-rank MNE methods across the population of auditory neurons.** (**A**) The difference in the negative log-likelihood evaluated on the test sets between the labeled models and the first-order MNE models is computed to compare the relative prediction error of first-order MNE, low-rank MNE, full-rank MNE, and STC across the auditory neurons. Above the horizontal dashed line, the first-order MNE model performs better than the low-rank MNE model. To the right of the vertical dashed line, the first-order MNE model performs better than the model labeled on the *x*-axis. The solid gray line compares the relative prediction error of the low-rank MNE model vs. the model on the *x*-axis where points below the line indicate that the low-rank MNE model performs better. (**B**) A bar plot shows the number of auditory neurons best fit by each of the models as determined from **A** where here "linear" refers to the first-order MNE models.

takes a reprising role where the reconstructed receptive fields from the low-rank MNE method will be utilized to reach new conclusions about the computations performed by populations of high-level auditory neurons.

The analysis focuses particularly on functional bases derived from the simple Boolean operations logical OR and logical AND due to their ease of interpretation

as being primarily integrative or discriminatory, respectively, towards collections of functional inputs. More intuitively, a logical OR operation is integrative because it expresses a form of invariance where any relevant input has the ability to cause a neuron to elicit a response. Logical AND is discriminative because the neuron responds only when a coincidence of all relevant sensory inputs are present which can be viewed as negation of the logical OR operation where logical AND is invariant to the absence of a relevant sensory input leading to a silent response.

Logical OR and logical AND functional bases were computed for each neuron across the population within the mean relevant subspace from each of the multicomponent dimensionality reduction methods (low-rank MNE, full-rank MNE, and STC) using quadratic input nonlinearities for added flexibility (Eq 2.14). As usual, four jackknives were computed with the data set divided into 70% training/20% cross-validation/10% test sets consistent with (Fig 4.4). The procedure otherwise follows along to the same steps used to optimize the functional bases in Chapter 2 where L-BFGS is employed to minimize the negative log-likelihood evaluated on the training set and the optimal number of functional basis components is determined via saturation of the negative log-likelihood evaluated on the cross-validation sets. In this case, the global optimization procedure completed following 50 consecutive failures of the algorithm to find a better training set solution with random weight initializations.

The two example neurons that appeared in Fig 4.13 return in Fig 4.15 where the logical AND functional basis is pictured for linear combinations of the receptive field estimates from each method in Fig 4.13. Logical AND was chosen for these neurons because the logical AND models had a lower prediction error than the logical OR models when measured on the cross-validation sets. For example, the normalized relative prediction errors (Eq 2.8) of the functional bases derived from the low-rank MNE estimates of the receptive fields were $\Delta L_{\text{OR,AND}} = -0.0235 \pm 0.001$ and $\Delta L_{\text{OR,AND}} =$

$(-3 \pm 1) \cdot 10^{-3}$ for neurons **blabla0713_3_B** and **oo2015_8_A**, respectively, indicating logical AND was a significantly better hypothesis than logical OR.

Clearly, STC provides inadequate estimates of the receptive fields for this application where the repeated dimensions in Fig 4.15 reveal an undercomplete basis which ought to be unlikely unless the number of components spanning the receptive field is overestimated. If one had reasonably speculated that the broad temporal gratings observed in the full-rank MNE receptive field reconstruction of neuron **blabla0713_3_B** from Fig 4.13 were benign and would perhaps cancel out when computing the functional basis, the result in Fig 4.15 dispels that notion (at least for logical OR and logical AND models) given that the gratings still occur prominently in four of the five functional basis components. For this neuron, the functional basis remains largely divergent from those that are reconstructed from the low-rank MNE models with arguably the exception of two of the five components. The **oo2015_8_A** neuron fares a little better since two of the four components are similar between the full-rank and low-rank MNE models but the remaining two functional basis components derived from the full-rank MNE models are nearly empty spectrograms.

The population of auditory neurons was by far better modeled by logical AND operations with respect to the low-rank MNE receptive field estimates. This is shown graphically in Fig 4.16 where the the vast majority of the neurons, including 40 of the 41 field L neurons and 8 of the 9 CM neurons, are positioned below the horizontal dashed line indicating that logical AND best predicts responses across the test sets. This plot also shows how the difference in performance between the logical AND and logical OR models is correlated with the overall balance of the positive and negative components of the eigenvalue spectra represented by $\mathrm{Tr}(\mathbf{J})$ on the x-axis of Fig 4.16. Receptive fields dominated by negative variance components were better described by logical AND models instead of logical OR models with a t-test p-value of 0.1%. Overall, the

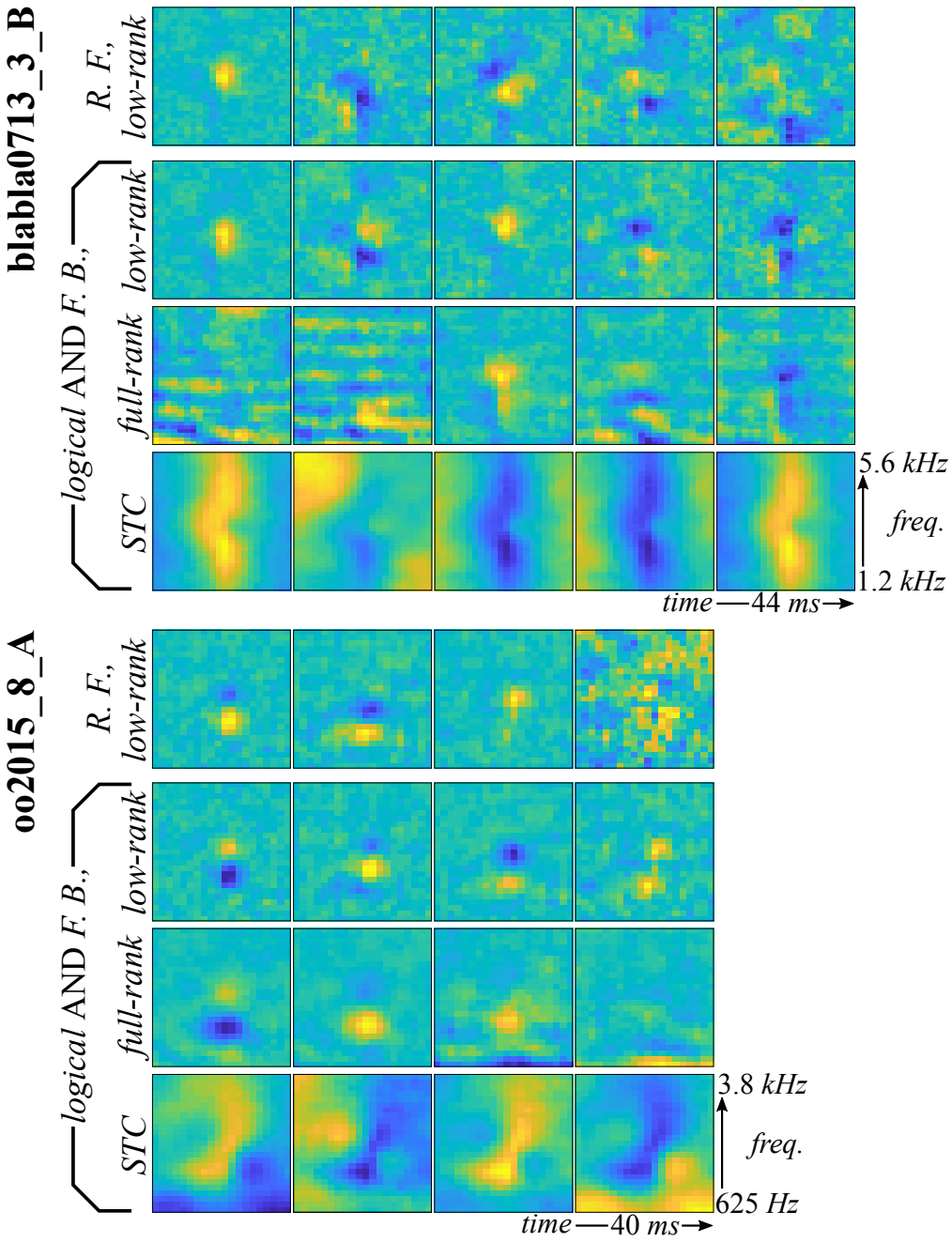

Figure 4.15: **Functional bases of two field L neurons.** Functional bases ("F. B.") derived from a logical AND model are plotted for the two field L neurons confined to the recovered receptive fields from the low-rank MNE, full-rank MNE, or STC models. The receptive fields ("R. F.") reconstructed by the low-rank MNE models are reproduced here for the reader's convenience.

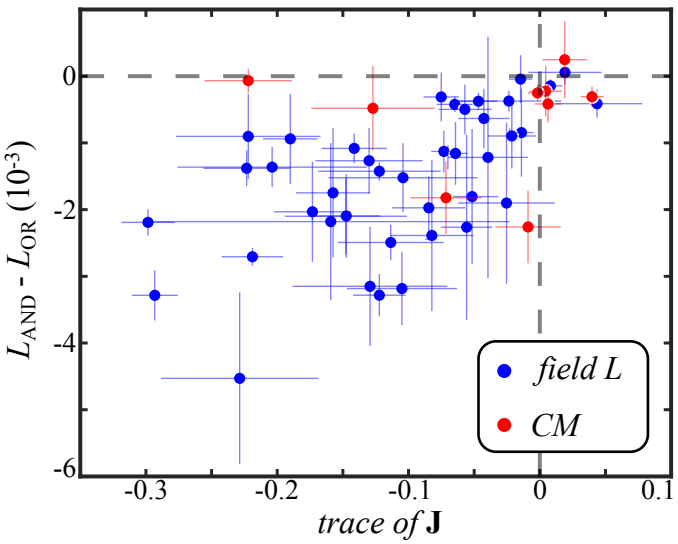

Figure 4.16: **The logical AND functional basis is the dominant description of the auditory neurons and is correlated with an imbalance of excitation and suppression.** Differences in the negative log-likelihood of the logical AND and logical OR models evaluated on the test sets are plotted as a function of $\mathrm{Tr}(\mathbf{J})$ from the low-rank MNE models for all of the auditory neurons. Neurons above the horizontal gray dashed line are better fit by logical OR models while those below are better fit by logical AND models. Neurons to the left of the vertical dashed line have receptive fields composed of components dominated by negative-variance while those on the right are dominated by positive-variance.

dominance of logical AND computations in the population suggests that auditory neurons in field L and CM are discriminatory. This could be interpreted as meaning that these neurons are selective for very specific qualities of auditory stimuli.

4.4 Discussion

By developing a low-rank MNE method designed to resist overfitting and bias from arbitrary stimulus distributions, it was possible to estimate multicomponent receptive fields of high-level auditory neurons with greater precision and significance than was previously possible. Dramatic improvements were observed not only in the quality of the receptive fields (certifiably in the case of the synthetic neurons) but also in the ability of the low-rank MNE models to predict responses of the neurons to novel

stimuli. Furthermore, the reconstructions of the receptive fields even at this high-level of processing were found to be spanned by few components ($r \leq 20$) which is a small enough space to allow for interpretable models to be constructed that describe the underlying neural computations.

Despite the relatively few components recovered from each auditory neuron, the low-rank MNE models were shown to outperform the full-rank MNE models. At the other extreme, the low-rank MNE models were shown to also outperform the first-order MNE models on the majority of neurons in the population. There are several reasons why these improvements were encountered. As was repeated on several occasions throughout this volume (and will be restated once more), MNE models often yield better reconstructions than spike-triggered methods like STC because MNE models were principally designed to have low susceptibility to bias when faced with non-Gaussian stimulus distributions. Examples of STC exhibiting sometimes substantial biases were presented throughout the chapter where the low-rank and first-order MNE methods were ubiquitous improvements over the STC models.

Compared to the full-rank MNE method, the low-rank MNE method has three primary advantages: (i) the explicit rank constraint reduces the number of weights to optimize, (ii) nuclear-norm regularization can limit or entirely eliminate the influence of fictitious components, and (iii) the matrix $\mathbf{J}$ is decomposed via a nonlinear matrix factorization. The first case simply renders the low-rank MNE model less prone to overfitting by reducing the number of weights per data sample. In the second case, the nuclear-norm imposes a well-defined low-rank structure on the matrix $\mathbf{J}$; a property of which is fundamental to dimensionality reduction. The full-rank MNE method's usage of early stopping [28] may behave unpredictably as was observed throughout the chapter and it certainly cannot be expected to push the optimization towards rank-deficient solutions. Yet, dropping the early stopping procedure would make matters worse since

the full-rank MNE model would proceed to overfit, at least on the data set sizes in this chapter. Lastly, the third point is important because one cannot necessarily assume there is a strict relationship between the variance of components in $\mathbf{J}$ and the contribution a component makes towards the predictive power of the model. One cannot generally assume that the components of $\mathbf{J}$ will behave like a linear matrix factorization where the eigenvalues/eigenvectors are local minima and larger absolute variance components are proportional to the depth of the minimum. In fact, in the process of generating the random problems in Section 3.2.1, counterexamples were found where the component located at a suboptimal local minimum could exist with higher variance than the the globally optimal component. The low-rank MNE problem circumvents this issue by fitting a low-rank bilinear factorization of $\mathbf{J}$ rather than the two-stage procedure (optimization of $\mathbf{J}$ followed by factorization) employed in the full-rank method.

The results highlight an interesting possible difference in how information is processed between visual and auditory systems. Recent investigations [51, 85], including those in Chapter 2, have shown that logical OR operations provide a better description of the computations in early vision. This is in contrast to the findings in this chapter that the computations performed by high-level auditory neurons from the avian auditory forebrain are overwhelmingly better described by logical AND operations. It is possible that these results indicate that auditory and vision neurons are sensitive to the presence of different types of stimulus transformations. The logical OR models can be interpreted as a max pooling of responses along excitatory dimensions of the inputs while logical AND models perform a max pooling operation along suppressive input dimensions and the two operations are related through negation. Another possibility is that early sensory regions are better described by logical OR while higher-level sensory regions are better described by logical AND. Determining which of these hypotheses, or others, are true would require the study of additional sensory regions and would be an interesting direction for future

research.

Chapter 4 contains work that was published in Kaardal, Theunissen, and Sharpee, *Frontiers in Computational Neuroscience* (2017). The dissertation author was the primary investigator and author of the paper.

Appendix A

Bayesian interpretation of the nuclear-norm

The nuclear-norm penalty function can be viewed from the Bayesian perspective as a prior distribution on the weights of the form

$$P(a,\mathbf{h},\mathbf{Q}) = \prod_{k=1}^{r} P(\mathbf{Q}_{\bullet,k}) = \prod_{k=1}^{r} \left(\frac{\varepsilon_k}{2\pi}\right)^{\frac{D}{2}} e^{-\frac{\varepsilon_k}{2}\mathbf{Q}_{\bullet,k}^{\mathrm{T}}\mathbf{Q}_{\bullet,k}} \tag{A.1}$$

which is a product of multivariate normal distributions with covariance $\varepsilon_k^{-1}\mathbf{I}$. If the weights are assumed to be drawn from this prior distribution, the objective function can be reformulated in terms of the posterior distribution, $P(a,\mathbf{h},\mathbf{Q}|y,\mathbf{s}_t) \propto P(y|\mathbf{s}_t;a,\mathbf{h},\mathbf{Q})P(a,\mathbf{h},\mathbf{Q})$, as (where, to be concise, $P(y)$ has been dropped since it is a constant):

$$\begin{aligned} f(a,\mathbf{h},\mathbf{Q}) = -\frac{1}{N_{\text{samp}}} \sum_t \Big[& y_t \log\left(P(y=1|\mathbf{s}_t;a,\mathbf{h},\mathbf{Q})P(a,\mathbf{h},\mathbf{Q})\right) \\ & + (1-y_t)\log\left(P(y=0|\mathbf{s}_t;a,\mathbf{h},\mathbf{Q})P(a,\mathbf{h},\mathbf{Q})\right)\Big]. \end{aligned} \tag{A.2}$$

This can be shown to reduce to the form in Eq 3.21 as follows:

$$\begin{aligned}\Rightarrow & -\frac{1}{N_{\text{samp}}}\sum_{t}\left[y_t \log P(y=1|\mathbf{s}_t)+(1-y_t)\log P(y=0|\mathbf{s}_t)\right]-\frac{1}{N_{\text{samp}}}\sum_{t}\log P(a,\mathbf{h},\mathbf{Q}) \\ & = L(a,\mathbf{h},\mathbf{Q})+\frac{1}{2}\sum_{k=1}^{r}\varepsilon_k\left\|\mathbf{Q}_{\bullet,k}\right\|_{\text{F}}^2+\frac{D}{2N_{\text{samp}}}\sum_{k=1}^{r}\log\frac{2\pi}{\varepsilon_k}\end{aligned}$$

where $P(y|\mathbf{s}_t) \equiv P(y|\mathbf{s}_t;a,\mathbf{h},\mathbf{Q})$ and $\mathbf{Q}_{\bullet,k}^{\text{T}}\mathbf{Q}_{\bullet,k} \equiv \left\|\mathbf{Q}_{\bullet,k}\right\|_{\text{F}}^2$. The right-most sum can be safely ignored because it does not explicitly depend on any of the weights and therefore has no impact on the solutions to the minimization problem. Thus, the nuclear-norm can be interpreted as a product of independent prior distributions composed of multivariate normal distributions on the components of $\mathbf{Q}$ and uniform distributions on a and $\mathbf{h}$. This prior distribution assumes that the variances of the components that make up $\mathbf{Q}$ are more likely to lie close to zero.

Appendix B

Convergence of the block coordinate descent algorithm

A block coordinate descent algorithm can be used to find a feasible local minimizer of the low-rank MNE problem (Eq 3.22). The block coordinate descent is taken with respect blocks of weights unrolled into block vectors

$$\mathbf{x}_k^{\mathrm{T}} = \left[\begin{array}{ccc} a, & \mathbf{h}^{\mathrm{T}}, & \mathbf{Q}_{\bullet,k}^{\mathrm{T}} \end{array} \right] \tag{B.1}$$

and the block k subproblems (Eq 3.41) can be solved cyclically. To prove that the block coordinate descent algorithm converges, one must show that the KKT conditions (Prop 3.1) and second-order sufficient conditions (Prop 3.2) of the low-rank MNE problem (Eq 3.22) are satisfied when the block KKT and block second-order sufficient conditions are satisfied across all blocks.

The block k subproblem (Eq 3.41) can be minimized by recursively solving the

linear system

$$\begin{bmatrix} \nabla^2_{\mathbf{x}_k\mathbf{x}_k}\mathcal{L}, & \mathbf{A}^{(k)^{\mathrm{T}}} \\ \mathbf{A}^{(k)}, & \mathbf{0} \end{bmatrix} \begin{bmatrix} \mathbf{p}_{\mathbf{x}_k} \\ -\mathbf{p}_{\Psi_k} \end{bmatrix} = - \begin{bmatrix} \nabla_{\mathbf{x}_k}\mathcal{L} \\ \nabla_{\Psi_k}\mathcal{L} \end{bmatrix} \tag{B.2}$$

with the remaining $\mathbf{Q}_{\bullet,j}$ $(j \neq k)$ held fixed using the interior-point method discussed in Section 3.3. The new matrices, $\mathbf{A}^{(k)} \in \mathbb{R}^{D\times 1+D+2rD}$, are the block constraint Jacobians. The cyclic block minimization leads to a monotonically decreasing series $f(\mathbf{x}_1^{(j)}) \geq f(\mathbf{x}_2^{(j)}) \geq \cdots \geq f(\mathbf{x}_r^{(j)})$ for the jth cycle through the blocks. Since the objective function is bounded from below, $f(\mathbf{x}) \geq 0$, the objective function cannot decrease indefinitely and will eventually saturate as $j \to \infty$ to a stationary point. At this stationary point, the block KKT and block second-order sufficient conditions are satisfied.

Proposition B.1. *Block KKT conditions: the first-order necessary conditions for* $\mathbf{x}_k^*$ *to be a feasible local minimizer of the block subproblem (Eq 3.41) are*

$$\nabla_{\mathbf{x}_k}\mathcal{L} = \mathbf{0},\ \nabla_{\Psi_{\bullet,k}}\mathcal{L} = \mathbf{0} \tag{B.3}$$

where the latter equation corresponds to satisfaction of the linear equality constraints and both equations are evaluated at $\mathbf{x}_k^*$.

As was the case for the full problem in Eq 3.22, the block subproblems can be guaranteed to satisfy the KKT conditions at a stationary point because the constraints are linear.

Proposition B.2. *Block second-order sufficient conditions: for the block weights* $\mathbf{x}_k^*$ *to be a feasible local minimizer of the block subproblem (Eq 3.41), the second-order sufficient conditions are*

$$\mathcal{S}_k = \mathcal{N}\left(\mathbf{A}^{(k)}\right) \nabla_{\mathbf{x}_k\mathbf{x}_k} f|_{\mathbf{x}_k^*} \mathcal{N}\left(\mathbf{A}^{(k)}\right)^{\mathrm{T}} \geq 0. \tag{B.4}$$

Since the block subproblem is nonconvex, it is both necessary and sufficient that a solution $\mathbf{x}_k^*$ satisfy Props B.1 & B.2.

Before beginning with the proof, the following submatrices are defined to make the notation less cumbersome:

$$\mathbf{A}_{i:j,i:j}^{*\mathrm{T}} = \begin{bmatrix} \mathbf{A}_{i,i}^{*\mathrm{T}}, & \mathbf{0}, & \cdots, & \mathbf{0} \\ \mathbf{0}, & \mathbf{A}_{i+1,i+1}^{*\mathrm{T}}, & \ddots, & \vdots \\ \vdots, & \ddots, & \ddots, & \mathbf{0} \\ \mathbf{0}, & \ldots, & \mathbf{0}, & \mathbf{A}_{j,j}^{*\mathrm{T}} \end{bmatrix}, \tag{B.5}$$

$$\mathbf{B} = \frac{1}{N}\sum_t P_t(1-P_t) \begin{bmatrix} 1 \\ \mathbf{s}_t \end{bmatrix} \begin{bmatrix} 1, & \mathbf{s}_t \end{bmatrix}, \tag{B.6}$$

$$\mathbf{R}_{i:j,\bullet}\mathbf{R}_{i':j',\bullet}^{\mathrm{T}} = \frac{1}{N}\sum_t P_t(1-P_t) \begin{bmatrix} \mathbf{D}_t\mathbf{Q}_{\bullet,i} \\ \mathbf{D}_t\mathbf{Q}_{\bullet,i+1} \\ \vdots \\ \mathbf{D}_t\mathbf{Q}_{\bullet,j} \end{bmatrix} \begin{bmatrix} \mathbf{D}_t\mathbf{Q}_{\bullet,i'} \\ \mathbf{D}_t\mathbf{Q}_{\bullet,i'+1} \\ \vdots \\ \mathbf{D}_t\mathbf{Q}_{\bullet,j'} \end{bmatrix}^{\mathrm{T}}, \tag{B.7}$$

$$\mathbf{Y}_{i:j,\bullet} = \frac{1}{N}\sum_t P_t(1-P_t) \begin{bmatrix} \mathbf{D}_t\mathbf{Q}_{\bullet,i} \\ \mathbf{D}_t\mathbf{Q}_{\bullet,i+1} \\ \vdots \\ \mathbf{D}_t\mathbf{Q}_{\bullet,j} \end{bmatrix} \begin{bmatrix} 1, & \mathbf{s}_t^{\mathrm{T}} \end{bmatrix}, \tag{B.8}$$

$$\mathbf{Z}_{i:j,i:j} = \frac{1}{N}\sum_t (P_t - y_t) \begin{bmatrix} \mathbf{D}_t, & \mathbf{0}, & \cdots, & \mathbf{0} \\ \mathbf{0}, & \mathbf{D}_t, & \ddots, & \vdots \\ \vdots, & \ddots, & \ddots, & \mathbf{0} \\ \mathbf{0}, & \cdots, & \mathbf{0}, & \mathbf{D}_t \end{bmatrix} + \begin{bmatrix} \varepsilon_i \mathbf{I}, & \mathbf{0}, & \cdots, & \mathbf{0} \\ \mathbf{0}, & \varepsilon_{i+1}\mathbf{I}, & \ddots, & \vdots \\ \vdots, & \ddots, & \ddots, & \mathbf{0} \\ \mathbf{0}, & \cdots, & \mathbf{0}, & \varepsilon_j \mathbf{I} \end{bmatrix} \tag{B.9}$$

where i and j are indices that extract a submatrix from a larger matrix and $0 < i \leq j \leq r$ (the same goes for i' and j').

The proof of convergence is made under two mild assumptions.

Assumption B.1. *The first-order feature space satisfies the condition*

$$\operatorname{rank}\left(\frac{1}{N_{\text{samp}}}\sum_t \begin{bmatrix} 1 \\ \mathbf{s}_t \end{bmatrix} \begin{bmatrix} 1, & \mathbf{s}_t^{\mathrm{T}} \end{bmatrix}\right) = D + 1 \tag{B.10}$$

or has been transformed such that this is true.

This first assumption means that the covariance of the first-order feature space (the stimulus vector augmented by the element one) should be full-rank. Even if this assumption is not immediately satisfied, the feature space can be transformed without loss of generality by projecting the first-order feature space into the non-zero principal components of the feature space.

Assumption B.2. *Any overlap in the subspace that spans the positive variance components of* $\mathbf{R}_{k,\bullet}\mathbf{R}_{k,\bullet}^{\mathrm{T}}$ *and the subspace that spans the negative variance components of* $\mathbf{Z}_{k,k}$ *have non-degenerate magnitude.*

Intuitively, the second assumption means that if eigendecomposition of $\mathbf{R}_{k,\bullet}\mathbf{R}_{k,\bullet}^{\mathrm{T}}$ and $\mathbf{Z}_{k,k}$ produce overlapping subspaces, the variance of these subspaces is assumed to be such that $\mathcal{R}\left(\mathbf{R}_{k,\bullet}\mathbf{R}_{k,\bullet}^{\mathrm{T}} + \mathbf{Z}_{k,k}\right) \supseteq \mathcal{R}\left(\mathbf{R}_{k,\bullet}\mathbf{R}_{k,\bullet}^{\mathrm{T}}\right)$. Given the dissimilarity of these two matrices, it is unlikely that this assumption would be broken. Furthermore, this assumption

can theoretically be satisfied through an infinitesimal adjustment of the nuclear-norm regularization parameters. The reasoning behind the second assumption is not obvious and will be made clear later on. These assumptions are laid out here not because the cyclic block coordinate descent will not converge otherwise, but instead to emphasize that the following proof only guarantees convergence under these assumptions.

Now, it will be shown that under Assumptions B.1 & B.2 that satisfaction of Props B.1 & B.2 simultaneously leads to the satisfaction of Props 3.1 & 3.2.

Proposition B.3. *If, for all k, each* $\mathbf{x}_k^*$ *is a feasible local minimizer of their respective block subproblem in Eq 3.41 satisfying Prop B.1, then* $\mathbf{x}^*$ *(the full weight vector) is a KKT point of the low-rank MNE problem Eq 3.22 satisfying Prop 3.1.*

Proof. *The KKT conditions in Prop 3.1 are trivially satisfied when Prop B.1 is simultaneously satisfied by all r blocks because the gradient of the Lagrangian of the full-rank MNE problem must be zero and* $\mathbf{x}^*$ *must be feasible in order to satisfy the block KKT conditions in Prop B.1.*

Proving that the second-order sufficient conditions are satisfied is quite a bit more involved and makes up the remainder of this section.

Proposition B.4. *Under Assumptions B.1 & B.2 and supposing that the KKT conditions are satisfied according to Prop B.3 for all* $\mathbf{x}_k^*$ *that are feasible local minima of the block subproblem (Eq 3.41, satisfaction of the block second-order sufficient conditions in Prop B.2 is sufficient to guarantee that the full weight vector* $\mathbf{x}^*$ *simultaneously satisfies the second-order sufficient conditions of the full low-rank MNE problem are satisfied (Prop 3.2).*

Proof. *Under Assumption B.1, the matrix* $\mathbf{R}_{i:j,\bullet}\mathbf{R}_{i:j,\bullet}^{\mathrm{T}}$ *is positive definite provided* $\mathbf{Q}_{\bullet,k} \neq \mathbf{0}$ *for all* $k \in \{i, \cdots, j\}$ *and strictly positive semidefinite if any* $\mathbf{Q}_{\bullet,k} = \mathbf{0}$*. When all of the*

nuclear-norm regularization parameters, $\{\varepsilon_k\}$, satisfy Prop 3.4, the matrix $\mathbf{Z}_{i:j,i:j}$ is positive semidefinite; otherwise $\mathbf{Z}_{i:j,i:j}$ is indefinite. From Eq 3.31, it is known that the rank-deficiency of $\mathbf{Z}_{i:j,i:j}$ is $\operatorname{rank}\left(\boldsymbol{\mathcal{N}}\left(\mathbf{Z}_{i:j,i:j}\right)\right) \geq \operatorname{rank}\left(\left[\mathbf{Q}_{\bullet,i}, \cdots, \mathbf{Q}_{\bullet,j}\right]\right)$. *The matrix $\mathbf{B}$ is positive definite.*

In terms of the abbreviations in Eqs B.5, B.6, B.7, B.8, & B.9, $\boldsymbol{\mathcal{S}}_k$ (Prop B.2) may written as

$$\begin{aligned}\begin{bmatrix} \mathbf{I}, & \mathbf{0} \\ \mathbf{0}, & \mathbf{A}^{*}_{k,k} \end{bmatrix} \begin{bmatrix} \mathbf{B}, & \mathbf{Y}^{\mathrm{T}}_{k,\bullet} \\ \mathbf{Y}_{k,\bullet}, & \mathbf{R}_{k,\bullet}\mathbf{R}^{\mathrm{T}}_{k,\bullet} + \mathbf{Z}_{k,k} \end{bmatrix} \begin{bmatrix} \mathbf{I}, & \mathbf{0} \\ \mathbf{0}, & \mathbf{A}^{*\mathrm{T}}_{k,k} \end{bmatrix} \\ = \begin{bmatrix} \mathbf{B}, & \mathbf{Y}^{\mathrm{T}}_{k,\bullet}\mathbf{A}^{*\mathrm{T}}_{k,k} \\ \mathbf{A}^{*}_{k,k}\mathbf{Y}_{k,\bullet}, & \mathbf{A}^{*}_{k,k}\left(\mathbf{R}_{k,\bullet}\mathbf{R}^{\mathrm{T}}_{k,\bullet} + \mathbf{Z}_{k,k}\right)\mathbf{A}^{*\mathrm{T}}_{k,k} \end{bmatrix}.\end{aligned} \tag{B.11}$$

The Schur complement of $\boldsymbol{\mathcal{S}}_k$ taken over $\mathbf{B}$ is

$$\left(\boldsymbol{\mathcal{S}}_k / \mathbf{B}\right) = \mathbf{A}^{*}_{k,k}\left(\mathbf{R}_{k,\bullet}\mathbf{R}^{\mathrm{T}}_{k,\bullet} + \mathbf{Z}_{k,k}\right)\mathbf{A}^{*\mathrm{T}}_{k,k} - \mathbf{A}^{*}_{k,k}\mathbf{Y}_{k,\bullet}\mathbf{B}^{-1}\mathbf{Y}^{\mathrm{T}}_{k,\bullet}\mathbf{A}^{*\mathrm{T}}_{k,k} \tag{B.12}$$

which is positive definite because $\boldsymbol{\mathcal{S}}_k$ is positive definite at a local minimizer of each block. Since $\mathbf{S}_k$ is positive semidefinite and $\mathbf{B}$ is positive definite, it follows from the Schur complement condition for positive semidefiniteness that

$$\boldsymbol{\Theta}_{k,k} = \mathbf{A}^{*}_{k,k}\left(\mathbf{R}_{k,\bullet}\mathbf{R}^{\mathrm{T}}_{k,\bullet} + \mathbf{Z}_{k,k}\right)\mathbf{A}^{*\mathrm{T}}_{k,k} \tag{B.13}$$

is also positive semidefinite. Note that Assumptions B.1 & B.2 guarantee that $\boldsymbol{\Theta}_{k,k}$ is only strictly positive semidefinite when $\mathbf{Q}_{\bullet,k} = \mathbf{0}$ and is otherwise positive definite.

Using Eq B.13 and the conclusions about the definiteness of $\boldsymbol{\Theta}_{k,k}$, convergence of cyclic block coordinate descent to a feasible local minimizer of the low-rank MNE problem (Eq 3.22) can be shown through recursive application of the Schur complement.

First, the second-order sufficient conditions of the low-rank MNE problem, $\boldsymbol{\mathcal{S}}$ (Prop 3.2), can be rewritten as

$$\begin{aligned}\begin{bmatrix} \mathbf{I}, & \mathbf{0} \\ \mathbf{0}, & \mathbf{A}^{*}_{1:r,1:r} \end{bmatrix} \begin{bmatrix} \mathbf{B}, & \mathbf{Y}^{\mathrm{T}}_{1:r,\bullet} \\ \mathbf{Y}_{1:r,\bullet}, & \mathbf{R}_{1:r,\bullet}\mathbf{R}^{\mathrm{T}}_{1:r,\bullet} + \mathbf{Z}_{1:r,1:r} \end{bmatrix} \begin{bmatrix} \mathbf{I}, & \mathbf{0} \\ \mathbf{0}, & \mathbf{A}^{*\mathrm{T}}_{1:r,1:r} \end{bmatrix} \\ = \begin{bmatrix} \mathbf{B}, & \mathbf{Y}^{\mathrm{T}}_{1:r,\bullet}\mathbf{A}^{*\mathrm{T}}_{1:r,1:r} \\ \mathbf{A}^{*}_{1:r,1:r}\mathbf{Y}_{1:r,\bullet}, & \mathbf{A}^{*}_{1:r,1:r}\left(\mathbf{R}_{1:r,\bullet}\mathbf{R}^{\mathrm{T}}_{1:r,\bullet} + \mathbf{Z}_{1:r,1:r}\right)\mathbf{A}^{*\mathrm{T}}_{1:r,1:r} \end{bmatrix}.\end{aligned} \tag{B.14}$$

Then, the Schur complement of $\boldsymbol{\mathcal{S}}$ is taken with respect to $\mathbf{B}$:

$$(\boldsymbol{\mathcal{S}}/\mathbf{B}) = \boldsymbol{\Theta}_{1:r,1:r} - \mathbf{A}^{*}_{1:r,1:r}\mathbf{Y}_{1:r,\bullet}\mathbf{B}^{-1}\mathbf{Y}^{\mathrm{T}}_{1:r,\bullet}\mathbf{A}^{*\mathrm{T}}_{1:r,1:r}, \tag{B.15}$$

where

$$\boldsymbol{\Theta}_{i:j,i':j'} = \mathbf{A}^{*}_{i:j,i:j}\left(\mathbf{R}_{i:j,\bullet}\mathbf{R}^{\mathrm{T}}_{i':j',\bullet} + \mathbf{Z}_{i:j,i':j'}\right)\mathbf{A}^{*\mathrm{T}}_{i':j',i':j'}. \tag{B.16}$$

From here, the Schur complement of $\boldsymbol{\Theta}_{k:r,k:r}$ over $\boldsymbol{\Theta}_{k,k}$ forms the sequence

$$\begin{aligned}(\boldsymbol{\Theta}_{1:r,1:r}/\boldsymbol{\Theta}_{1,1}) &= \boldsymbol{\Theta}_{2:r,2:r} - \boldsymbol{\Theta}_{2:r,1}\boldsymbol{\Theta}^{-1}_{1,1}\boldsymbol{\Theta}_{1,2:r} \\ (\boldsymbol{\Theta}_{2:r,2:r}/\boldsymbol{\Theta}_{2,2}) &= \boldsymbol{\Theta}_{3:r,3:r} - \boldsymbol{\Theta}_{3:r,2}\boldsymbol{\Theta}^{-1}_{2,2}\boldsymbol{\Theta}_{2,3:r} \\ (\boldsymbol{\Theta}_{3:r,3:r}/\boldsymbol{\Theta}_{3,3}) &= \boldsymbol{\Theta}_{4:r,4:r} - \boldsymbol{\Theta}_{4:r,3}\boldsymbol{\Theta}^{-1}_{3,3}\boldsymbol{\Theta}_{3,4:r} \\ &\vdots \\ (\boldsymbol{\Theta}_{r-1:r,r-1:r}/\boldsymbol{\Theta}_{r-1,r-1}) &= \boldsymbol{\Theta}_{r,r} - \boldsymbol{\Theta}_{r-1:r,r-1}\boldsymbol{\Theta}^{-1}_{r-1,r-1}\boldsymbol{\Theta}_{r-1,r-1:r}.\end{aligned} \tag{B.17}$$

When all $\mathbf{Q}_{\bullet,k} \neq \mathbf{0}$, then according to Eq B.13 all $\boldsymbol{\Theta}_{k,k}$ are positive definite. Therefore, the last Schur complement of sequence indicates that $\boldsymbol{\Theta}_{r-1:r,r-1:r}$ must be positive definite because $\boldsymbol{\Theta}_{r,r}$ and $\boldsymbol{\Theta}_{r-1,r-1}$ are positive definite. This result can than be used to show that $\boldsymbol{\Theta}_{r-2:r,r-2:r}$ is also positive definite. In the same fashion, by inserting

the results below into the equation above and backtracking through the sequence from bottom to top, one would find that all $\boldsymbol{\theta}_{k:r,k:r}$ *are positive definite when all* $\mathbf{Q}_{\bullet,k} \neq \mathbf{0}$ *and therefore* $\boldsymbol{\mathcal{S}}$ *is positive definite and* $\mathbf{x}^*$ *is a feasible local minimizer of the low-rank MNE problem.*

If, on the other hand, $\mathbf{Q}_{\bullet,k} = \mathbf{0}$ *for some k, then* $\boldsymbol{\Theta}_{k,k}$ *is strictly positive semidefinite for that k. Let* $\mathbf{Q}$ *have* r_{opt} *non-zero columns where* $r_{\text{opt}} < r$*. Without loss of generality, the columns of* $\mathbf{Q}$ *can be rearranged such that the first* $r - r_{\text{opt}}$ *columns are zero (*$\mathbf{Q}_{\bullet,k} = \mathbf{0}$ *for* $k \leq r - r_{\text{opt}}$*). Then the elements of the submatrices* $\mathbf{R}_{1:r-r_{\text{opt}}}\mathbf{R}^{\mathrm{T}}_{1:r,\bullet}$ *and* $\mathbf{R}_{1:r,\bullet}\mathbf{R}^{\mathrm{T}}_{1:r-r_{\text{opt}},\bullet}$ *are all zero and therefore* $\boldsymbol{\Theta}_{1:r-r_{\text{opt}},1:r-r_{\text{opt}}} = \mathbf{Z}_{1:r-r_{\text{opt}},1:r-r_{\text{opt}}}$ *is a block diagonal matrix,* $\boldsymbol{\Theta}_{r-r_{\text{opt}}+1:r,r-r_{\text{opt}}+1:r}$ *is positive definite, and the remaining submatrices of* $\mathbf{R}_{1:r,\bullet}\mathbf{R}^{\mathrm{T}}_{1:r,\bullet}$ *are all zeros. Because* $\boldsymbol{\Theta}_{k,k}$ *for* $k < r - r_{\text{opt}}$ *is rank-deficient,* $\boldsymbol{\Theta}_{k:r,k:r}$ *can be proven to be positive semidefinite via the generalized Schur complement. To do so, the matrix inverses,* $\boldsymbol{\Theta}^{-1}_{k,k}$*, of the top* $r - r_{\text{opt}}$ *lines of the recursive Schur complement (Eq B.17) must be substituted by a generalized inverse,* $\boldsymbol{\Theta}^{\dagger}_{k,k}$*, and satisfy the additional requirement that* $\boldsymbol{\mathcal{N}}\left(\boldsymbol{\Theta}_{k,k+1:r}\right) \supseteq \boldsymbol{\mathcal{N}}\left(\boldsymbol{\Theta}_{k,k}\right)$ *for all k. Of course, this condition is trivially satisfied because* $\mathbf{R}_{k,\bullet} = \mathbf{0}$ *when* $\mathbf{Q}_{\bullet,k} = \mathbf{0}$ *and therefore* $\boldsymbol{\Theta}_{k,k+1:r} = \mathbf{A}^{*}_{k,k}\mathbf{R}_{k,\bullet}\mathbf{R}^{\mathrm{T}}_{k+1:r}\mathbf{A}^{*\mathrm{T}}_{k+1:r,k+1:r} = \mathbf{0}$ *and* $\boldsymbol{\mathcal{N}}\left(\boldsymbol{\Theta}_{k,k+1:r}\right)$ *is full-rank. Backtracking once again from the bottom equation in the sequence to the top of the sequence in Eq B.17 shows that* $\boldsymbol{\mathcal{S}}$ *is positive semidefinite when some* $\mathbf{Q}_{\bullet,k} = \mathbf{0}$*. Therefore, cyclic block coordinate descent converges to a feasible local minimizer of the low-rank MNE problem (Eq 3.22) under Assumptions B.1 & B.2.*

Appendix B contains work that was published in Kaardal, Theunissen, and Sharpee (2017). The dissertation author was the primary investigator and author of the paper.

Appendix C

Resolving the optimal rank of second-order MNE models

In second-order MNE models where the multicomponent receptive field is recovered by diagonalizing (symmetrized) **J**, solutions for **J** originating from multiple fits of the models to different data sets may produce conflicting answers with regard to the number of significant components that make up **J**. One quick way to resolve this issue is to instead determine the number of significant components from **J** averaged across the different fits. However, this still leaves open the question of a strategy for determining which components of the averaged **J** ought to be considered significant. The approach taken here determines the number of significant components based on ideas from random matrix theory.

Suppose that the mean **J** matrix was drawn from a distribution of distribution of random numbers, $\mathcal{J}$, with zero mean and positive variance $\hat{\sigma}^2$. If **J** is a large matrix with $D \gg 1$, then the distribution of eigenvalues of random matrices whose elements are

drawn from $\mathcal{J}$ are distributed according to the Wigner semi-circle law:

$$P(\beta) = \begin{cases} \frac{1}{2\pi\hat{\sigma}^2}\sqrt{4\hat{\sigma}^2 - \beta^2} & \text{if } -2|\hat{\sigma}| \leq \beta \leq 2|\hat{\sigma}| \\ 0 & \text{otherwise} \end{cases} \tag{C.1}$$

where β is the variance of an eigenvalue. Unfortunately, it is not known from which probability distribution **J** has been drawn; but a conservative estimate can be made on the bounds of the null distribution in Eq C.1 under the assumption that the distribution $\mathcal{J}$ has similar statistics to matrix **J** itself. To limit any further assumptions being set on $\mathcal{J}$, no explicit functional form is imposed on $\mathcal{J}$. Rather, a large number of random matrices is generated by symmetrically shuffling the elements of **J** with the sign of each element decided with uniform probability to ensure that the mean of the distribution remains at zero. Each of these random symmetric matrices are diagonalized and the largest magnitude positive and negative eigenvalues are aggregated from each matrix. The number of significant components is then determined to the desired probability, p_{thres}, that the eigenvalue is within the bounds of the null distribution (Eq C.1) defined by the aggregated eigenvalues from the random matrices. The significant components are outliers from the null distribution with probability $1 - p_{\text{thres}}$. A pseudocode outline of this procedure appears in Alg C.1.

Appendix C contains work that was published in Kaardal, Theunissen, and Sharpee (2017). The dissertation author was the primary investigator and author of the paper.

Algorithm C.1 Resolving the rank of the matrix J.

inputs: $\mathbf{J}$, p_{thres}, the number of random matrices to generate M
initialization: compute a vector of eigenvalues $\beta \leftarrow \mathbf{eig}(\mathbf{J})$, initialize the significant number of components $r_{\text{opt}} \leftarrow 0$, initialize an empty set to store the eigenvalue bounds $\zeta \leftarrow \emptyset$

for $m \in \{1, \cdots, M\}$ **do**
    $\hat{\mathbf{Y}} \leftarrow$ randomly sample $D(D+1)/2$ elements from $\pm\mathbf{J}$ with uniform probability into a $D \times D$ symmetric matrix
    $\zeta \leftarrow \zeta \bigcup \left\{ |\min\left(\hat{\mathbf{Y}}\right)|, \max\left(\hat{\mathbf{Y}}\right) \right\}$
for $k \in \{1, \cdots, D\}$ **do**
    $p \leftarrow \dfrac{1}{2M} \sum_{m=1}^{2M} H\left(\zeta_m - |\beta_k|\right)$ */* where $H(\cdot)$ is the Heaviside step function */*
    if $p \geq p_{\text{thres}}$ **then**
        break
    else
        $r_{\text{opt}} \leftarrow k$
returns: r_{opt}

Bibliography

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

[2] Egbert De Boer and Paul Kuyper. Triggered correlation. *IEEE Transactions on Biomedical Engineering*, 15(3):169–179, 1968.

[3] Markus Meister and Michael J Berry. The neural code of the retina. *Neuron*, 22(3):435–450, 1999.

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

[5] J Victor and R Shapley. A method of nonlinear analysis in the frequency domain. *Biophysical Journal*, 29(3):459–483, 1980.

[6] Craig A Atencio, Tatyana O Sharpee, and Christoph E Schreiner. Cooperative nonlinearities in auditory cortical neurons. *Neuron*, 58(6):956–966, 2008.

[7] Donald R Cantrell, Jianhua Cang, John B Troy, and Xiaorong Liu. Non-centered spike-triggered covariance analysis reveals neurotrophin-3 as a developmental regulator of receptive field properties of on-off retinal ganglion cells. *PLoS computational biology*, 6(10):e1000967, 2010.

[8] Xiaodong Chen, Feng Han, Mu-ming Poo, and Yang Dan. Excitatory and suppressive receptive field subunits in awake monkey primary visual cortex (v1). *Proceedings of the National Academy of Sciences*, 104(48):19120–19125, 2007.

[9] Adrienne L Fairhall, C Andrew Burlingame, Ramesh Narasimhan, Robert A Harris, Jason L Puchalla, and Michael J Berry. Selectivity for multiple stimulus features in retinal ganglion cells. *Journal of neurophysiology*, 96(5):2724–2738, 2006.

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

[11] Jessica L Fox, Adrienne L Fairhall, and Thomas L Daniel. Encoding properties of haltere neurons enable motion feature detection in a biological gyroscope. *Proceedings of the National Academy of Sciences*, 107(8):3840–3845, 2010.

[12] Sungho Hong, Blaise Agüera y Arcas, and Adrienne L Fairhall. Single neuron computation: from dynamical system to feature detector. *Neural computation*, 19(12):3133–3172, 2007.

[13] Gregory D Horwitz, EJ Chichilnisky, and Thomas D Albright. Blue-yellow signals are enhanced by spatiotemporal luminance contrast in macaque v1. *Journal of Neurophysiology*, 93(4):2263–2278, 2005.

[14] Gregory D Horwitz, EJ Chichilnisky, and Thomas D Albright. Cone inputs to simple and complex cells in v1 of awake macaque. *Journal of Neurophysiology*, 97(4):3070–3081, 2007.

[15] Anmo J Kim, Aurel A Lazar, and Yevgeniy B Slutskiy. System identification of drosophila olfactory sensory neurons. *Journal of computational neuroscience*, 30(1):143–161, 2011.

[16] Miguel Maravall, Rasmus S Petersen, Adrienne L Fairhall, Ehsan Arabzadeh, and Mathew E Diamond. Shifts in coding properties and maintenance of information transmission during adaptation in barrel cortex. *PLoS biology*, 5(2):e19, 2007.

[17] Nicole C Rust, Odelia Schwartz, J Anthony Movshon, and Eero P Simoncelli. Spatiotemporal elements of macaque v1 receptive fields. *Neuron*, 46(6):945–956, 2005.

[18] Lawrence C Sincich, Jonathan C Horton, and Tatyana O Sharpee. Preserving information in neural transmission. *Journal of Neuroscience*, 29(19):6207–6216, 2009.

[19] Jon Touryan, Brian Lau, and Yang Dan. Isolation of relevant visual features from random stimuli for cortical complex cells. *Journal of Neuroscience*, 22(24):10811–10818, 2002.

[20] William Bialek and Rob R van Steveninck. Features and dimensions: Motion estimation in fly vision. *arXiv preprint q-bio/0505003*, 2005.

[21] R De Ruyter Van Steveninck and W Bialek. Real-time performance of a movement-sensitive neuron in the blowfly visual system: coding and information transfer in short spike sequences. *Proceedings of the Royal Society of London B: Biological Sciences*, 234(1277):379–414, 1988.

[22] Liam Paninski. Convergence properties of some spike-triggered analysis techniques. In *Advances in neural information processing systems*, pages 189–196, 2003.

[23] Il Memming Park and Jonathan W Pillow. Bayesian spike-triggered covariance analysis. In *Advances in neural information processing systems*, pages 1692–1700, 2011.

[24] Jonathan W Pillow and Eero P Simoncelli. Dimensionality reduction in neural models: an information-theoretic generalization of spike-triggered average and covariance analysis. *Journal of vision*, 6(4):9–9, 2006.

[25] Odelia Schwartz, EJ Chichilnisky, and Eero P Simoncelli. Characterizing neural gain control using spike-triggered covariance. In *Advances in neural information processing systems*, pages 269–276, 2002.

[26] Tatyana Sharpee, Nicole C Rust, and William Bialek. Analyzing neural responses to natural signals: maximally informative dimensions. *Neural computation*, 16(2):223–250, 2004.

[27] Kanaka Rajan and William Bialek. Maximally informative âĂIJstimulus energiesâĂİ in the analysis of neural responses to natural signals. *PloS one*, 8(11):e71959, 2013.

[28] Jeffrey D Fitzgerald, Ryan J Rowekamp, Lawrence C Sincich, and Tatyana O Sharpee. Second order dimensionality reduction using minimum and maximum mutual information models. *PLoS computational biology*, 7(10):e1002249, 2011.

[29] Jeffrey D Fitzgerald, Lawrence C Sincich, and Tatyana O Sharpee. Minimal models of multidimensional computations. *PLoS computational biology*, 7(3):e1001111, 2011.

[30] Claude Elwood Shannon. Communication in the presence of noise. *Proceedings of the IRE*, 37(1):10–21, 1949.

[31] Edwin Thompson Jaynes. *Probability Theory: The Logic of Science*. Cambridge University Press, 2003.

[32] David A Drachman. Do we have brain to spare? *Neurology*, 64(12):2004–2005, 2005.

[33] Maximilian Riesenhuber and Tomaso Poggio. Hierarchical models of object recognition in cortex. *Nature neuroscience*, 2(11):1019–1025, 1999.

[34] Thomas Serre, Lior Wolf, Stanley Bileschi, Maximilian Riesenhuber, and Tomaso Poggio. Robust object recognition with cortex-like mechanisms. *IEEE transactions on pattern analysis and machine intelligence*, 29(3):411–426, 2007.

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

[36] HB Barlow and William R Levick. The mechanism of directionally selective units in rabbit's retina. *The Journal of physiology*, 178(3):477–504, 1965.

[37] Catherine E Carr and Masakazu Konishi. Axonal delay lines for time measurement in the owl's brainstem. *Proceedings of the National Academy of Sciences*, 85(21):8311–8315, 1988.

[38] Hiroki Asari and Markus Meister. Divergence of visual channels in the inner retina. *Nature neuroscience*, 15(11):1581–1589, 2012.

[39] Ethan Cohen and Peter Sterling. Microcircuitry related to the receptive field center of the on-beta ganglion cell. *Journal of Neurophysiology*, 65(2):352–359, 1991.

[40] William H Press, Saul A Teukolsky, William T Vetterling, and Brian P Flannery. *Numerical recipes in C*, volume 2. Cambridge university press Cambridge, 1996.

[41] Jorge Nocedal and Stephen Wright. *Numerical optimization*. Springer Science & Business Media, 2006.

[42] Léon Bottou. Large-scale machine learning with stochastic gradient descent. In *Proceedings of COMPSTAT'2010*, pages 177–186. Springer, 2010.

[43] Claire S Adjiman, Stefan Dallwig, Christodoulos A Floudas, and Arnold Neumaier. A global optimization method, αbb, for general twice-differentiable constrained nlpsâĂŤi. theoretical advances. *Computers & Chemical Engineering*, 22(9):1137–1158, 1998.

[44] Ryan J Rowekamp and Tatyana O Sharpee. Analyzing multicomponent receptive fields from neural responses to natural stimuli. *Network: Computation in Neural Systems*, 22(1-4):45–73, 2011.

[45] Ryan J Rowekamp. *Characterizing neural responses to natural stimuli*. PhD thesis, UC San Diego: Physics (Biophysics), 2014. b8207231.

[46] David J Ketchen Jr and Christopher L Shook. The application of cluster analysis in strategic management research: an analysis and critique. *Strategic management journal*, pages 441–458, 1996.

[47] Michael Eickenberg, Ryan J Rowekamp, Minjoon Kouh, and Tatyana O Sharpee. Characterizing responses of translation-invariant neurons to natural stimuli: Maximally informative invariant dimensions. *Neural computation*, 24(9):2384–2421, 2012.

[48] Brett Vintch, Andrew Zaharia, J Movshon, and Eero P Simoncelli. Efficient and direct estimation of a neural subunit model for sensory coding. In *Advances in neural information processing systems*, pages 3104–3112, 2012.

[49] Olivier Marre, Dario Amodei, Nikhil Deshmukh, Kolia Sadeghi, Frederick Soo, Timothy E Holy, and Michael J Berry. Mapping a complete neural population in the retina. *Journal of Neuroscience*, 32(43):14859–14873, 2012.

[50] Michael P Eckert, Gershon Buchsbaum, and Andrew B Watson. Separability of spatiotemporal spectra of image sequences. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(12):1210–1213, 1992.

[51] Joel Kaardal, Jeffrey D Fitzgerald, Michael J Berry, and Tatyana O Sharpee. Identifying functional bases for multidimensional neural computations. *Neural computation*, 25(7):1870–1890, 2013.

[52] Bradley Efron and Robert J Tibshirani. *An introduction to the bootstrap*. CRC press, 1994.

[53] RE Soodak, RM Shapley, and E Kaplan. Fine structure of receptive-field centers of x and y cells of the cat. *Visual neuroscience*, 6(6):621–628, 1991.

[54] Tatyana O Sharpee, Katherine I Nagel, and Allison J Doupe. Two-dimensional adaptation in the auditory forebrain. *Journal of neurophysiology*, 106(4):1841–1861, 2011.

[55] Edward H Adelson and James R Bergen. Spatiotemporal energy models for the perception of motion. *JOSA A*, 2(2):284–299, 1985.

[56] Anthony J Bell and Terrence J Sejnowski. The âĂIJindependent componentsâĂİ of natural scenes are edge filters. *Vision research*, 37(23):3327–3338, 1997.

[57] Tatyana O Sharpee, Kenneth D Miller, and Michael P Stryker. On the importance of static nonlinearity in estimating spatiotemporal neural filters with natural stimuli. *Journal of neurophysiology*, 99(5):2496–2509, 2008.

[58] Johnatan Aljadeff, Ronen Segev, Michael J Berry II, and Tatyana O Sharpee. Spike triggered covariance in strongly correlated gaussian stimuli. *PLoS computational biology*, 9(9):e1003206, 2013.

[59] Maryam Fazel. *Matrix rank minimization with applications*. PhD thesis, PhD thesis, Stanford University, 2002.

[60] Maryam Fazel, Haitham Hindi, and Stephen P Boyd. Log-det heuristic for matrix rank minimization with applications to hankel and euclidean distance matrices. In *American Control Conference, 2003. Proceedings of the 2003*, volume 3, pages 2156–2162. IEEE, 2003.

[61] Benjamin Recht, Maryam Fazel, and Pablo A Parrilo. Guaranteed minimum-rank solutions of linear matrix equations via nuclear norm minimization. *SIAM review*, 52(3):471–501, 2010.

[62] Stuart Geman, Elie Bienenstock, and René Doursat. Neural networks and the bias/variance dilemma. *Neural computation*, 4(1):1–58, 1992.

[63] Ricardo Cabral, Fernando De la Torre, João P Costeira, and Alexandre Bernardino. Unifying nuclear norm and bilinear factorization approaches for low-rank matrix decomposition. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2488–2495, 2013.

[64] Francis Bach, Julien Mairal, and Jean Ponce. Convex sparse matrix factorizations. *arXiv preprint arXiv:0812.1869*, 2008.

[65] Lieven Vandenberghe and Stephen Boyd. Semidefinite programming. *SIAM review*, 38(1):49–95, 1996.

[66] Samuel Burer and Renato DC Monteiro. A nonlinear programming algorithm for solving semidefinite programs via low-rank factorization. *Mathematical Programming*, 95(2):329–357, 2003.

[67] Benjamin Haeffele, Eric Young, and Rene Vidal. Structured low-rank matrix factorization: Optimality, algorithm, and applications to image processing. In *International Conference on Machine Learning*, pages 2007–2015, 2014.

[68] Richard H Byrd, Jorge Nocedal, and Robert B Schnabel. Representations of quasi-newton matrices and their use in limited memory methods. *Mathematical Programming*, 63(1):129–156, 1994.

[69] James Bergstra and Yoshua Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.

[70] Eric Brochu, Vlad M Cora, and Nando De Freitas. A tutorial on bayesian optimization of expensive cost functions, with application to active user modeling and hierarchical reinforcement learning. *arXiv preprint arXiv:1012.2599*, 2010.

[71] Jasper Snoek, Hugo Larochelle, and Ryan P Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, pages 2951–2959, 2012.

[72] Joel T. Kaardal, Frédéric E. Theunissen, and Tatyana O. Sharpee. A low-rank method for characterizing high-level neural computations. *Frontiers in Computational Neuroscience*, 11:68, 2017.

[73] Christopher KI Williams and Carl Edward Rasmussen. Gaussian processes for regression. In *Advances in neural information processing systems*, pages 514–520, 1996.

[74] Bertil Matérn. *Spatial variation*, volume 36. Springer Science & Business Media, 2013.

[75] The GPyOpt authors. GPyOpt: A bayesian optimization framework in python. http://github.com/SheffieldML/GPyOpt, 2016.

[76] Katherine I Nagel and Allison J Doupe. Organizing principles of spectro-temporal encoding in the avian primary auditory area field l. *Neuron*, 58(6):938–955, 2008.

[77] Dario L Ringach, Michael J Hawken, and Robert Shapley. Dynamics of orientation tuning in macaque primary visual cortex. *Nature*, 387(6630):281, 1997.

[78] Stephen W Kuffler. Discharge patterns and functional organization of mammalian retina. *Journal of neurophysiology*, 16(1):37–68, 1953.

[79] Gerald Westheimer. Center-surround antagonism in spatial vision: Retinal or cortical locus? *Vision research*, 44(21):2457–2465, 2004.

[80] J Hans van Hateren and Arjen van der Schaaf. Independent component filters of natural images compared with simple cells in primary visual cortex. *Proceedings of the Royal Society of London B: Biological Sciences*, 265(1394):359–366, 1998.

[81] Noopur Amin, Patrick Gill, and Frédéric E Theunissen. Role of the zebra finch auditory thalamus in generating complex representations for natural sounds. *Journal of neurophysiology*, 104(2):784–798, 2010.

[82] Patrick Gill, Junli Zhang, Sarah MN Woolley, Thane Fremouw, and Frédéric E Theunissen. Sound representation methods for spectro-temporal receptive field estimation. *Journal of computational neuroscience*, 21(1):5, 2006.

[83] Kazuo Okanoya and Robert J Dooling. Hearing in passerine and psittacine birds: a comparative study of absolute and masked auditory thresholds. *Journal of Comparative Psychology*, 101(1):7–15, 1987.

[84] Robert J Dooling. Auditory perception in birds. *Acoustic communication in birds*, 1:95–130, 1982.

[85] Thomas Serre, Aude Oliva, and Tomaso Poggio. A feedforward architecture accounts for rapid categorization. *Proceedings of the national academy of sciences*, 104(15):6424–6429, 2007.