Inferring hidden structure in multilayered neural circuits

Niru Maheswaranathan[1], Stephen A. Baccus[2], Surya Ganguli[3*],

1 Neurosciences Graduate Program, Stanford University, Stanford, CA, USA
2 Dept. of Neurobiology, Stanford University, Stanford, CA, USA
3 Dept. of Applied Physics, Stanford University, Stanford, CA, USA

[*]sganguli@stanford.edu

Abstract

A central challenge in sensory neuroscience involves understanding how neural circuits shape computations across multiple nonlinear cell layers. Here we develop a computational framework to reconstruct the response properties of experimentally unobserved neurons in the interior of a multilayered neural circuit. We combine non-smooth regularization techniques with proximal consensus algorithms to overcome traditional difficulties in fitting such models due to the high dimensionality of their parameter space. Our methods are both statistically and computationally efficient, enabling us to not only rapidly learn hierarchical non-linear models, but also efficiently compute widely used descriptive statistics such as the spike triggered average (STA) and covariance (STC) for high dimensional stimuli. We apply our framework to retinal ganglion cell processing, learning STAs and STCs to similar accuracy using just 12% of recorded data, reducing experiment time by an order of magnitude. Furthermore, we learn three layer nonlinear models of retinal circuitry, consisting of thousands of parameters, using only 40 minutes of responses to white noise. Our models demonstrate a 53% improvement in predicting ganglion cell spikes over classical linear-nonlinear (LN) models. The internal structure of these models reveals that hidden nonlinear subunits match the properties of retinal bipolar cells in both receptive field structure and number. Subunits had consistently high thresholds, leading to sparse activity patterns in which only one subunit drives ganglion cell spiking at any time. From the model's parameters, we predict that the removal of visual redundancies through stimulus decorrelation across space–a central tenet of efficient coding theory—originates primarily from bipolar cell synapses. Furthermore, the composite nonlinear computation performed by retinal circuitry corresponds to a boolean OR function applied to bipolar cell feature detectors. Our general computational framework may aid in extracting principles of nonlinear hierarchical sensory processing across diverse modalities from limited data.

Introduction

Motivation

Computational models of neural responses to sensory stimuli have played a central role in addressing fundamental questions about the nervous system, including how sensory stimuli are encoded and represented, the mechanisms that generate such a neural code, and the theoretical principles that govern both the sensory code and underlying mechanisms. Such models often begin with a statistical description of the stimuli that precede a neural response including the spike-triggered average (STA) [1, 2] or covariance (STC) [3–8]. These statistical measures characterize to some extent the set of effective stimuli that drive a response, but do not necessarily reveal how these statistical properties relate to cellular mechanisms or neural pathways. Going beyond these descriptive statistics, an explicit representation of the neural code can be obtained by learning a model to predict neural responses to sensory stimuli. A classic approach involves a single stage of spatiotemporal filtering and a time-independent or static nonlinearity; these models include linear-nonlinear (LN) models with single or multiple pathways [1, 9–11] or generalized linear models (GLMs) [12, 13]. Similar to STA and STC analyses, these models do not directly map onto circuit anatomy and function. As a result, the interpretation of such descriptive statistics and phenomenological models, as well as how they precisely

relate to underlying cellular mechanisms remains unclear. Ideally, one would like to generate more biologically realistic models of sensory circuits, in which the learned parameters are highly interpretable and sub-components of the model map in a one-to-one fashion onto cellular components of neurobiological circuits. For example, model components such as spatiotemporal filtering, thresholding, and summation are readily mapped onto photoreceptor or membrane voltage dynamics, synaptic and spiking thresholds, and dendritic pooling, respectively.

A critical aspect of sensory circuits is that they operate in a hierarchical fashion in which sensory signals propagate through multiple nonlinear cell layers. Fitting models that capture this widespread anatomical structure using measurements of neurons in one layer of a circuit in response to controlled stimuli raises significant statistical as well as computational challenges [14–18], in contrast to LN models [2] and GLMs. A key issue is the high dimensionality of both stimulus and parameter space, as well as the existence of hidden, unobserved neurons in intermediate cell layers that remain unrecorded. The high dimensionality of parameter space can necessitate prohibitively large amounts of data to accurately fit the model (the statistical challenge). Furthermore, this high dimensionality, combined with hidden neurons, can prohibitively extend the computational time required to optimize the model (the computational challenge). One approach to address these difficulties is the use of prior knowledge about the structure and components of the circuits [11, 17, 19–21]. Although prior knowledge of the exact network architecture and sequence of nonlinearities from properties such as synapses and spiking, would greatly constrain the number of possible circuit solutions, such prior knowledge is typically minimal for most neural circuits.

In this work, we present a computational framework that addresses these challenges by incorporating prior knowledge only about the into the estimates of descriptive statistics and the relationship of those statistics to model parameters. We then utilize this framework to formulate and fit parameters of hierarchical nonlinear models to recordings of ganglion cells in the retina. In particular, we focus on models with 3 cell layers connected by two stages of linear-nonlinear processing (LN-LN models). As described below, the cell layers of these models map in one-to-one fashion onto the three principal cell layers of the retina: photoreceptors, bipolar cells, and retinal ganglion cells. We demonstrate that these models are both a more accurate description of the retinal code, as well as more amenable to biophysical interpretation. In particular, we find a match between the properties of subunits in the intermediate, hidden layer of our models and the properties of bipolar cells in the retina. Further analysis of our learned models reveal novel insights into retinal function, namely that, (1) transmission between every subunits and ganglion cell pair is well described by a high threshold expansive nonlinearity, (2) bipolar cells are sparsely active, (3) visual inputs are most de-correlated at the subunit layer, pre-synaptic to ganglion cells, and (4) the composite computation performed by the retinal ganglion cell output corresponds to a boolean OR function of bipolar cell feature detectors. Collectively, these results shed light on the nature of hierarchical nonlinear computation in the retina. Our computational framework is general, however, and we hope it will aid in providing insights into hierarchical nonlinear computations across the nervous system.

Background on retinal physiology and modeling

The retina is a classic system for exploring the relationship between descriptive statistics, quantitative encoding models, and measurements of neurobiological circuit properties [22, 23]. Signals in the retina flow from photoreceptors through populations of horizontal, bipolar, and amacrine cells before reaching the ganglion cell layer. Despite this complex multilayered computation, many studies have characterized retinal ganglion cell responses using simple descriptive statistics such as spike-triggered average or covariance [10, 24, 25]. Responses are often then modeled using a linear-nonlinear (LN) framework. A major reason for the widespread adoption of LN models is their high level of tractability; learning their parameters can be accomplished by solving a simple convex optimization problem [1], or alternatively, estimated using straightforward reverse correlation analyses [2]. While previous studies have found that these simple models can for some neurons capture most of the variance of the responses to spatiotemporal white noise [9, 12, 16], they do not accurately describe responses to stimuli with more structure such as natural scenes [13, 26–28].

Furthermore, the precise relationship between both the descriptive statistics and LN models, and the underlying multilayered neural circuit that generates the ganglion cell response remains unclear. Often, the spatiotemporal linear filter of the LN model is presented as mapping onto the aggregate sequential

mechanisms of phototransduction, signal filtering and transmission through bipolar and amacrine cell pathways, and summation at the ganglion cell, while the nonlinearity is mapped onto the spiking threshold of ganglion cells. However, there can be strong rectification of signals that occurs pre-synaptic to ganglion cells [29–31], breaking the assumption of composite linearity in the pathway from photoreceptors just up to the ganglion cell spiking threshold [32]. Indeed, nonlinear spatial integration within ganglion cell receptive fields was first described in the cat retina by Hochstein and Shapley [33] and Victor and Shapley [34] in Y-type ganglion cells. These authors proposed a hypothetical model for this computation: a cascade of two layers of linear-nonlinear operations (LN-LN). The first major nonlinearity acting at a constant mean luminance is thought to lie at the bipolar-to-ganglion cell synapse, with each of the first layer LN being termed a subunit of the ganglion cell[1]. The second LN layer corresponds to summation or pooling across multiple subunits at the ganglion cell soma, followed by a spiking threshold. The subunit nonlinearities in these models have been shown to underlie many retinal computations including latency encoding [23], object motion sensitivity [35], and sensitivity to fine spatial structure (such as edges) in natural scenes [30]. Figure 1 shows a schematic of the LN-LN cascade and its mapping onto retinal anatomy.

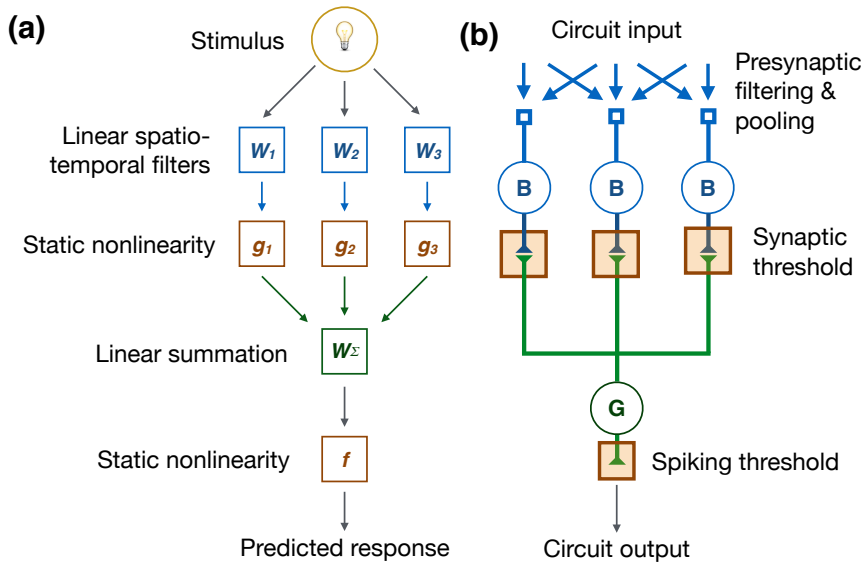

Fig 1. A schematic of the LN-LN cascade and its mapping onto retinal anatomy. (a) Each LN unit consists of a linear spatiotemporal filter followed by a static nonlinearity. The LN-LN cascade is then a bank of LN subunits, whose outputs are pooled at a second linear stage before being passed through a final nonlinearity. (b) The LN-LN model mapped on to a retinal circuit. The first LN stage consists of bipolar cell subunits and the bipolar-to-ganglion cell synaptic threshold. The second LN stage is pooling at the ganglion cell, plus a spiking threshold.

Identifying the parameters of subunit models given limited experimental data is a significant challenge. Typically, simplifying assumptions are made about the form of the subunit nonlinearities [21, 36, 37], the similarity of different subunits [19, 20], or the number of subunits [14]. Another approach is to incorporate prior knowledge about the model parameters, through a process known as regularization in the statistics and machine learning literature [38]. In this paper, we describe computational methods based on proximal consensus algorithms, described below, that allow us to utilize minimal prior knowledge about model parameters in a computationally and statistically efficient manner to perform both spike-triggered analyses and fit hierarchical nonlinear models using much less data than otherwise required.

To achieve our results, we first elucidate the precise relationship between widely used descriptive statistics, like the STA and STC, and the parameters of model neurons. By establishing how these statistics

[1]Note that subunits that combined *linearly* would be indistinguishable from a computational perspective. Due to the roughly linear integration [9] that occurs at bipolar cells, we (computationally) distill mechanisms in photoreceptors, horizontal cells, and their synaptic connections into a single spatiotemporal filter that gives rise to bipolar cell signals.

depend on neuronal filters and nonlinearities, we develop theoretically principled regularization methods for utilizing prior knowledge about neuronal biophysics to rapidly and efficiently estimate descriptive statistics. Moreover, by understanding how the biophysical properties of neural circuits shapes the geometry of the spike-triggered ensemble, we endow these statistics with much greater interpretive power, and enable them to provide a lens through which we can understand the differences between different hierarchical, nonlinear computations. With this computational framework in hand, we then move on to learning hierarchical nonlinear models of retinal responses to spatiotemporal white noise, focusing specifically on the linear-nonlinear cascade (LN-LN) model. We demonstrate that our learned LN-LN models both yield improved predictive performance as well as insights into hierarchical nonlinear computations in the retina. We end by discussing the precise relationship between our work and other work on retinal modeling.

Results

Relationship between descriptive statistics and neural models

We represent a visual stimulus as an N dimensional vector $\mathbf{x}$ of luminance levels. For example, if the stimulus is a segment of a spatiotemporal movie, discretized with N_s spatial bins, and N_t temporal bins, then $N = N_s N_t$, and each component $\mathbf{x}_i$ of the vector reflects the luminance level of the i'th spatiotemporal bin. Moreover, in this section we view a functional neural model as an arbitrary nonlinear function $r = f(\mathbf{x})$, over N dimensional stimulus space, where r determines the probability that the neuron fires in a small time window following a stimulus $\mathbf{x}$: $r(x) = p(\text{spike} \mid \mathbf{x})$. Our goal in this section is to understand at a general level, how and which properties of the function f determine the STA and STC. This analysis will yield a mapping between neural response properties and descriptive statistics that will greatly aid in both the rapid learning and interpretation of descriptive statistics. Readers who are specifically interested in how to fit hierarchical nonlinear models and the lessons we learn from them can safely skip the derivations in this section.

The STA and STC are the mean and covariance, respectively, of the spike-triggered stimulus ensemble, which reflects the collection of stimuli preceding each spike [3]. This distribution over stimuli, conditioned on a spike occurring, can be expressed via Bayes rule,

$$p(\mathbf{x} \mid \text{spike}) = \frac{p(\text{spike} \mid \mathbf{x}) p(\mathbf{x})}{p(\text{spike})}, \tag{1}$$

where $p(\mathbf{x})$ is the prior distribution over stimuli and $p(\text{spike})$ is the average firing probability over all stimuli. Here, we assume a white noise stimulus distribution, in which each component of $\mathbf{x}$ is chosen independently from a Gaussian distribution with zero mean and unit variance. The STA and STC are given by

$$\mathbf{x}_{\text{STA}} = \mathbb{E}_{p(\mathbf{x}|spike)}[\mathbf{x}] \tag{2}$$

$$\mathbf{C}_{\text{STC}} = \mathbb{E}_{p(\mathbf{x}|spike)}[\mathbf{x}\,\mathbf{x}^T] - (\mathbf{x}_{\text{STA}})(\mathbf{x}_{\text{STA}})^T \tag{3}$$

Focusing first on the STA:

$$\begin{aligned} \mathbf{x}_{\text{STA}} &= \int \mathbf{x}\, p(\mathbf{x} \mid \text{spike})\, d\mathbf{x} \\ &= \frac{1}{\mu} \int \mathbf{x}\, r(\mathbf{x}) p(\mathbf{x})\, d\mathbf{x} \\ &= \frac{1}{\mu} \mathbb{E}_{p(\mathbf{x})}[\mathbf{x}\, r(\mathbf{x})] \\ &= \frac{1}{\mu} \mathbb{E}_{p(\mathbf{x})}[\nabla r(\mathbf{x})], \end{aligned} \tag{4}$$

where $\mu = p(\text{spike})$ is the overall probability of spiking. The last step in the derivation uses Stein's lemma, which states that $\mathbb{E}[\mathbf{x} f(\mathbf{x})] = \mathbb{E}[\nabla f(\mathbf{x})]$ if the expectation is taken over a multivariate Gaussian distribution with identity covariance matrix, corresponding to our white noise stimulus assumption. This calculation thus

yields the simple statement that the spike-triggered average is proportional to the gradient (or gain) of the response function, averaged over the input distribution.

Applying Stein's lemma again yields an expression for the STC matrix:

$$\begin{aligned}
\mathbf{C}_{\mathrm{STC}} &= \int \mathbf{x}\mathbf{x}^T p(\mathbf{x} \mid \text{spike}) d\mathbf{x} - (\mathbf{x}_{\mathrm{STA}})(\mathbf{x}_{\mathrm{STA}})^T \\
&= \frac{1}{\mu} \int \mathbf{x}\mathbf{x}^T r(\mathbf{x}) p(\mathbf{x}) d\mathbf{x} - (\mathbf{x}_{\mathrm{STA}})(\mathbf{x}_{\mathrm{STA}})^T \\
&= \frac{1}{\mu} \mathbb{E}_{p(\mathbf{x})}[\mathbf{x}\mathbf{x}^T r(\mathbf{x})] - (\mathbf{x}_{\mathrm{STA}})(\mathbf{x}_{\mathrm{STA}})^T \\
&= \frac{1}{\mu} \mathbb{E}_{p(\mathbf{x})}[\nabla^2 r(\mathbf{x})] - (\mathbf{x}_{\mathrm{STA}})(\mathbf{x}_{\mathrm{STA}})^T \qquad (5)
\end{aligned}$$

Intuitively, these results state that the STA is related to the slope (first derivative) and the STC is related to the Hessian curvature (matrix of second derivatives) of the multi-dimensional nonlinear response function $r(\mathbf{x})$.

For example, consider a linear nonlinear model $r = f(\mathbf{w}^T\mathbf{x})$ which has the following gradient: $\nabla r(\mathbf{x}) = f'(\mathbf{w}^T\mathbf{x})\mathbf{w}$ and Hessian: $\nabla^2 r(\mathbf{x}) = f''(\mathbf{w}^T\mathbf{x})\mathbf{w}\mathbf{w}^T$. Plugging these expressions into equations (4) and (5) reveals that the STA is proportional to $\mathbf{w}$ and the STC is proportional to $\mathbf{w}\mathbf{w}^T$. Therefore, we recover the known result [2] that the STA of the LN model is proportional to the linear filter, and there will be one significant direction in the STC, which is also proportional to the linear filter (with mild assumptions on the nonlinearity, f, to ensure that slope and curvature terms in (4) and (5) are non-zero).

We can extend this to the case of a multilayered circuit with k pathways, each of which first filters the stimulus with a filter $\mathbf{w}_1 \ldots \mathbf{w}_k$. Regardless of how these pathways are then combined, we can write this circuit computation as $r = f(\mathbf{W}^T\mathbf{x})$ where $\mathbf{W}$ is a matrix whose columns are the k pathway filters, and f is a k-dimensional time-independent (static) nonlinear function. We can think of the k dimensional vector $\mathbf{u} = \mathbf{W}^T\mathbf{x}$ as the activity pattern across each of the k pathways before any nonlinearity. Then, the STA and STC can be written explicitly in terms of the pathway filters $\mathbf{w}_i$ and the statistics of the nonlinear neural response function r with respect to the pathway activity pattern $\mathbf{u}$.

The gradient for such a model is $\nabla r(\mathbf{x}) = \mathbf{W}^T \nabla \mathbf{f}(\mathbf{u})$, where $\nabla \mathbf{f}(\mathbf{u})$ is the gradient of the k-dimensional nonlinearity. Using equation (4), the STA is then a linear combination of the pathway filters:

$$\mathbf{x}_{\mathrm{STA}} = \frac{1}{\mu} \sum_{i=1}^{k} \alpha_i \mathbf{w_i},$$

where the weights are given by

$$\alpha_i = \mathbb{E}_{p(\mathbf{x})}[\, \partial_{u_i} r(\mathbf{u}) \,]$$

and they correspond to the average sensitivity, or slope of the neural response r with respect to changes in the activity of the i^{th} filter.

The Hessian for the multilayered model is $\nabla^2 r(\mathbf{x}) = \mathbf{W}\nabla^{\mathbf{2}}\mathbf{f}\mathbf{W}^{\mathbf{T}}$, where $\nabla^{\mathbf{2}}\mathbf{f}$ is the k-by-k matrix of second derivatives of the k-dimensional nonlinearity $f(\mathbf{u})$. From equation (5), the STC is then given by:

$$\mathbf{C}_{\mathrm{STC}} = \frac{1}{\mu^2} \mathbf{W}\mathbf{H}\mathbf{W}^T, \qquad (6)$$

where the k-by-k matrix $\mathbf{H}$ is:

$$\mathbf{H} = \mu \mathbb{E}[\nabla^{\mathbf{2}} f(\mathbf{u})] - \mathbb{E}[\nabla f(\mathbf{u})]\mathbb{E}[\nabla f(\mathbf{u})]^T.$$

This expression implies that nontrivial directions in the column space of $\mathbf{C}_{\mathrm{STC}}$ correspond to (span the same space as) the column space of $\mathbf{W}$. Therefore, the significant eigenvectors of the STC matrix will be linear combinations of the k pathway filters, and the number of significant eigenvectors is at most k.

Regularized spike-triggered analysis

Computing the STA and STC requires estimating N and N^2 parameters, respectively, where $N = N_s N_t$ is the dimensionality of the spatiotemporal stimulus. Increasing the stimulus dimensionality then demands prohibitively long experiments in order to accurately estimate these statistics. To address this difficulty, we formulated a computational framework that allows us to exploit expected structure in either the STA or STC eigenvectors using prior knowledge about these features. This prior knowledge is often previously known to the experimenter, and can otherwise be learned using a small number of experiments and then generalized to many new experiments. This framework allows us to quickly and efficiently estimate the STA and STC using drastically less data than otherwise required. Using prior knowledge to prevent overfitting is known in the mathematics and statistics literature as regularization, so we call our methods regularized spike-triggered analysis.

Although it is not *a priori* obvious how to translate prior knowledge about parallel pathways in multilayered neural circuits into expected structure in the STA/STC, results from the previous section provide a theoretically principled route forward. In particular, if the number of parallel pathways is small relative to the dimensionality of the stimulus then the STA and columns of the STC matrix consist of a linear combination of a small number of individual pathway filters. Therefore, given prior knowledge only related to the number of independent neural pathways, we expect certain types of structure (i.e. smoothness, sparsity, and low spatiotemporal rank) in said pathways to persist after the linear combination.

We thus impose this structure directly on the descriptive statistics through a set of penalty functions, or regularizers. For example, these penalties, as functions on a candidate descriptive statistic, could encourage smoothness in the spatial and temporal domains, low rank spatiotemporal structure, or sparsity (i.e. encouraging many filter coefficients to be zero). The objective we minimize is the sum of these regularizers plus a data fidelity term that preserves similarity (i.e. squared error) between the candidate descriptive statistic and the raw (noisy) spike-triggered statistic.

Regularized STA

Following the above framework, we use the following optimization problem to estimate the regularized STA:

$$\hat{\mathbf{x}}_{\mathrm{STA}} = \min_{\mathbf{x}} \|\mathbf{x} - \mathbf{x}_{\mathrm{STA}}\|_2^2 + \sum_{i=1}^{m} \gamma_i \, \phi_i(\mathbf{x}). \tag{7}$$

Here $\phi_i(x)$ are the regularization penalty functions, and the regularization weight γ_i associated with each penalty function is a hyperparameter that controls how strongly to weight or enforce that penalty. $\mathbf{x}_{\mathrm{STA}}$ is the *raw* STA obtained from simply averaging all stimuli preceding a spike; it is noisy due to sampling spikes from a limited amount of recording time. The output of the optimization $\hat{\mathbf{x}}_{\mathrm{STA}}$ ideally reflects a denoised STA which balances maintaining fidelity to the raw STA (via a least-squares loss term) while conforming to prior structure promoted by the penalty functions. If these penalties $\phi_i(x)$ are convex, then (7) is a convex optimization problem, which we can solve efficiently. To solve it, we employ a *proximal consensus algorithm*. Proximal algorithms are well suited to this problem because they can flexibly incorporate many different penalty terms, including non-smooth terms, and they efficiently scale as the amount of data or stimulus dimensionality increases. For an overview of these algorithms, see Methods. For the convex penalty functions ϕ_i, we use an ℓ_1 penalty that encourages the estimated filter to be sparse (few non-zero coefficients), and a nuclear norm penalty, which is the sum of the singular values of the spatiotemporal filter $\mathbf{x}$ when viewed as a spacetime matrix. This latter penalty encourages many singular values to be zeros, thus causing the filter to be low-rank with respect to this spacetime matrix representation. If all but one are zero, the filter becomes spacetime separable. The nuclear norm penalty is advantageous compared to explicitly forcing the spacetime filter $\mathbf{x}$ to be low-rank, as it is a "soft" penalty which allows for many small singular values, whereas explicitly forcing the filter to be low-rank forces those to be zero. See methods for mathematical details about the regularization penalty functions.

Figure 2a compares a regularized spike-triggered average (rSTA) obtained from (7) with the raw, un-regularized STA. For long recordings, the regularized STA closely matches the raw STA, indicating that regularization does not adversely bias the result when there is sufficient data. For short recordings (less than

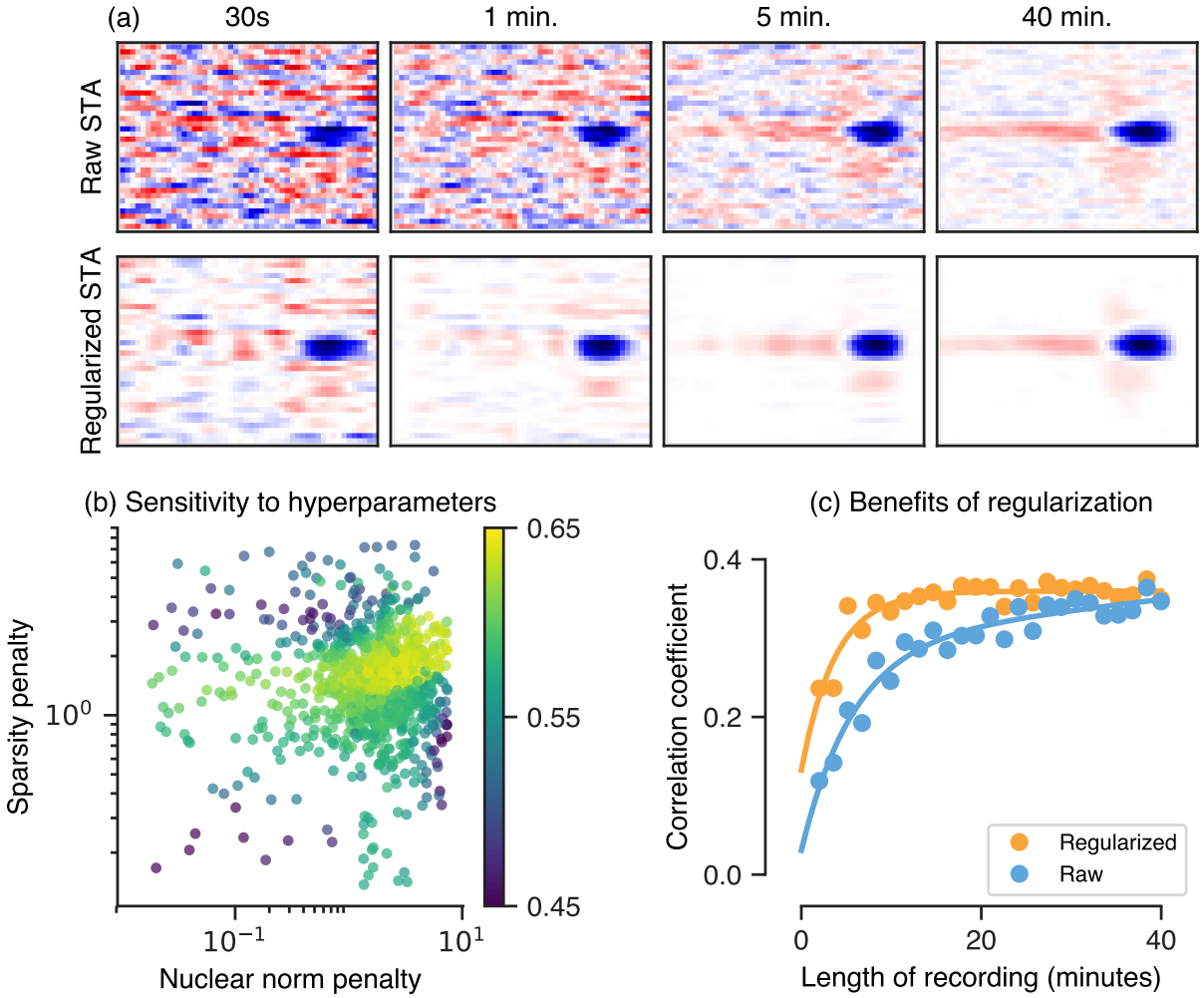

Fig 2. Regularization for estimating receptive fields (via a regularized spike-triggered-average). (a) Top row: the raw spike-triggered average computed using different amounts of data (from left to right, 30s to 40min), bottom row: the regularized spike-triggered average computed using the same amount of data as the corresponding column. (b) Performance (log-likelihood) as a function of two regularization weights, the nuclear norm (x-axis, encourages low-rank structure) and the ℓ_1-norm (y-axis, encourages sparsity). (c) Correlation coefficient of LN models whose filter is fixed to be a regularized or raw (un-regularized) STA, as a function of the amount of training data for model fitting (length of recording).

a few minutes), the regularized STA is much cleaner than the raw STA, and retains much of the structure observed if the STA had been measured using much longer amounts of data. In situations where we wish to quickly estimate a spike-triggered average (e.g. if we are exploring how the STA changes under different contrasts [9], or in response to different adapting stimuli [39]), then utilizing regularization can make new experiments possible that require rapid estimation of the STA.

Figure 2b shows the performance of the regularized STA across different regularization weights (i.e. the γ_i in (7)), scanned over a wide range. It demonstrates that performance is largely insensitive to the strengths of the weights of the ℓ_1 and nuclear norm penalty functions, over a large range. Thus regularization weights need not be fine tuned to achieve superior performance.

We quantified the performance of the regularized STA by employing it as the linear filter of an LN model, with a nonlinearity that was then optimized to fit the data. We measured the ability of the resulting model to predict ganglion cell activity on held-out data, and found that with regularization, about 5 minutes of

recording was sufficient to achieve the performance obtained through 40 minutes of recording without regularization (Figure 2c). Thus regularization not only has the ability to recover a clean STA using little data, but also a useful STA that retains the power to predict the neural response to new stimuli.

Regularized STC Analysis

Beyond the STA, we can also extend our framework to regularize essential information content in the STC matrix. In particular, the eigenvectors of the STC matrix have been interpreted as phenomenological pathway filters [10, 37], and they often constitute an important final target of STC analysis. The STC matrix has N^2 parameters compared to the N parameters of the STA, and small errors in the estimate of each of the many elements of the STC matrix can propagate to generate errors in estimated STC eigenvectors. Thus accurate estimation of STC eigenvectors can require considerably more data than the estimation of the STA itself. Since we expect the eigenvectors of the STC matrix to be linear combinations of biophysically relevant spatiotemporal filters of individual neural pathways, we should be able to utilize prior knowledge about these filters to regularize the eigenvectors directly, thereby estimating them with far less data. Here we introduce a computational framework to achieve this result.

The STC eigenvectors are obtained by an eigendecomposition of the STC matrix $\mathbf{C}$, which is equivalent to solving an optimization problem:

$$\text{maximize} \quad \mathrm{Tr}(\mathbf{U}^T\mathbf{C}\mathbf{U}) \tag{8}$$

$$\text{subject to} \quad \mathbf{U}^T\mathbf{U} = \mathbf{I}, \tag{9}$$

where $\mathbf{U}$ denotes a matrix whose columns are the orthonormal eigenvectors of $\mathbf{C}$. In order to regularize the computation of these eigenvectors, we need to add penalty terms to (8), which precludes a closed form solution to the problem. We circumvent this by reformulating the problem using a convex relaxation, which allows us to solve it efficiently with additional penalty terms (regularization) on the STC eigenvectors. First, we consider the matrix $\mathbf{X} = \mathbf{U}\mathbf{U}^T$, corresponding to the outer product of the eigenvectors. Because of the cyclic property of the trace, namely that $\mathrm{Tr}(\mathbf{U}^T\mathbf{C}\mathbf{U}) = \mathrm{Tr}(\mathbf{U}\mathbf{U}^T\mathbf{C}) = \mathrm{Tr}(\mathbf{X}\mathbf{C})$, the function to be optimized in (8) depends on the eigenvector matrix $\mathbf{U}$ only through the combination $\mathbf{X} = \mathbf{U}\mathbf{U}^T$. Thus we could directly optimize over the variable $\mathbf{X}$. However, the non-convex equality constraint $\mathbf{U}^T\mathbf{U} = \mathbf{I}$ in (9) is not easily expressible in terms of $\mathbf{X}$. $\mathbf{X}$ is however a projection operator, obeying $\mathbf{X}^2 = \mathbf{X}$. Indeed if $\mathbf{U}$ were a truncated N by d eigenvector matrix whose d columns were the top d eigenvectors of $\mathbf{C}$ with the largest eigenvalues, then $\mathbf{X}$ would be a rank d projection operator that projects stimuli onto the d dimensional principal subspace of the STC matrix. Moreover, we have seen that the number of nontrivial eigenvectors, with eigenvalues differing from unity in the STC matrix, corresponds generically to the small number of nonlinear, parallel pathways in the neural circuit. To reflect this prior knowledge in our recovery of clean STC eigenvectors in a computationally tractable manner, we relax the constraint that $\mathbf{X}$ is a rank d projection operator (a non-convex constraint), and replace it with the constraint that $\mathbf{X}$ should be contained within the convex hull of the set of rank-d projection matrices. This space of matrices is a convex body known as the *fantope* [40].

The advantage of this theoretical formulation is that we obtain a convex optimization problem which can be further augmented with additional functions that penalize the columns of $\mathbf{X}$ to impose prior knowledge about the structure of the eigenvectors of $\mathbf{C}$. Each column of $\mathbf{X}$ is a linear combination of the eigenvectors of $\mathbf{C}$, which are themselves in turn linear combinations of the small set of filters that define the relevant low dimensional sensitive subspace of the cell. Therefore, if we expect the true filters $\mathbf{w}_i$ to have certain structure (for example, smooth, low-rank, or sparse), then we also expect to see those properties in both the eigenvectors and in the columns of $\mathbf{X}$.

Putting this logic together, to obtain *regularized* STC eigenvectors, we solve the following convex optimization problem

$$\hat{\mathbf{X}} = \max_{\mathbf{X}} \quad \mathrm{Tr}(\mathbf{X}\mathbf{C}) + \sum_i \gamma_i \, \phi_i(\mathbf{X}) \tag{10}$$

$$\text{subject to} \quad X \in \mathcal{F}^d. \tag{11}$$

Here $\mathbf{C}$ is the *raw* STC matrix, which is noisy due to estimating its elements from limited amounts of data, and $\mathcal{F}^d$ denotes the fantope, or convex hull of all rank d projection matrices. Each ϕ_i is a regularization penalty function applied to each of the columns of $\mathbf{X}$; i.e. $\phi_i(\mathbf{X}) \equiv \sum_{j=1}^{N} \phi_i(\mathbf{x}^j)$, where $\mathbf{x}^j$ denotes the j'th column of $\mathbf{X}$. Again, we can solve this optimization problem efficiently using proximal consensus algorithms (see Methods). The optimization yields a matrix $\hat{\mathbf{X}}$ in the fantope $\mathcal{F}^d$, which may itself have rank higher than d. We perform a final eigendecomposition of this matrix to obtain its top-eigenvectors. These eigenvectors constitute our regularized estimate of the eigenvectors of the STC. Finally, we note that one major computational advantage of this formulation is that we only need to store and work with the N by N raw STC covariance matrix itself. This is in contrast to applying matrix factorization directly to the spike-triggered ensemble, an N by M matrix where M, the number of spikes, is typically much greater than N.

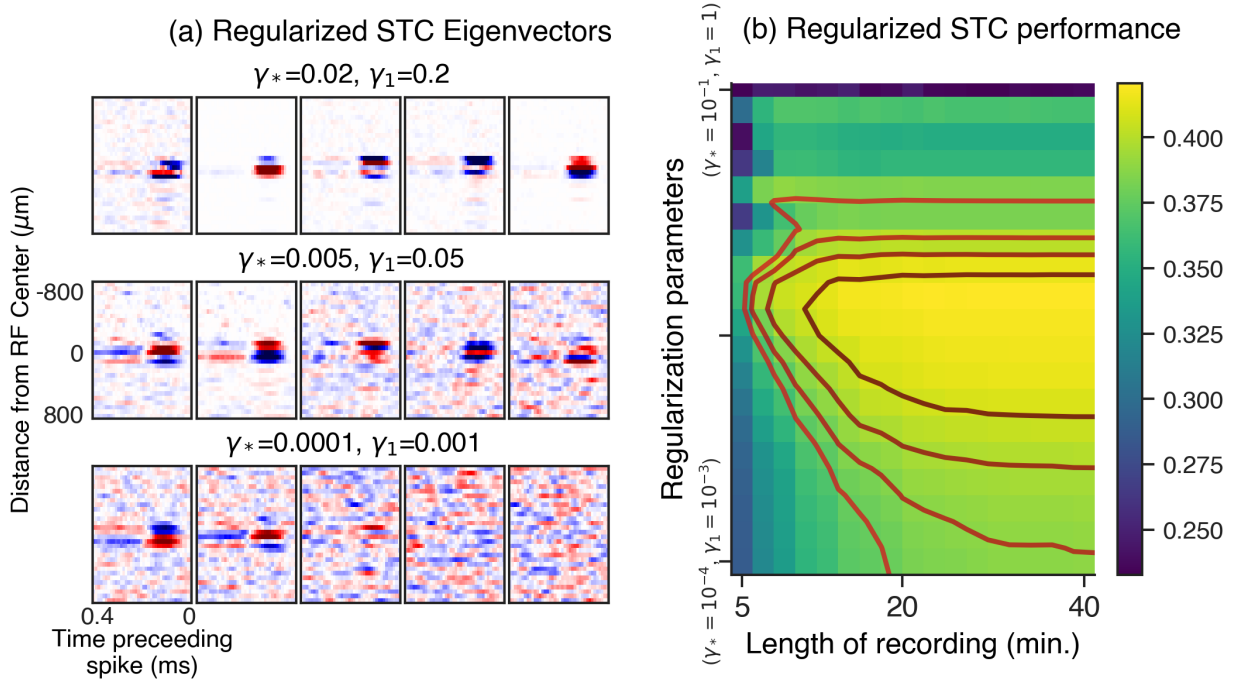

Fig 3. Regularized spike-triggered covariance. (a) Example panels of the output of our regularized spike-triggered covariance algorithm. Each panel contains the five most significant regularized eigenvectors of the STC matrix, reshaped as spatiotemporal filters. The bottom panel shows the result with no regularization added, and the upper panels show the result with increasing weights on the regularization penalties. Here γ_1 is the regularization weight applied to an ℓ_1 penalty encouraging sparsity, and γ_* is a regularization weight applied to a nuclear norm penalty, encouraging approximate spatiotemporal separability of the eigenvectors, when reshaped as spatiotemporal filters. (b) Summary across a population of cells. The heatmap shows the performance of rSTC (measured as the subspace overlap with the best fit LN-LN subspace, see methods for details). The y-axis in (b) represents a line spanning 3 orders of magnitude in two-dimensional regularization parameter space (γ_*, γ_1), ranging from the point $(\gamma_* = 10^{-4}, \gamma_1 = 10^{-3})$ to $(\gamma_* = 10^{-1}, \gamma_1 = 1)$.

Figure 3 demonstrates the improvement in our ability to estimate the relevant subspace spanned by significant STC eigenvectors. Just as in the rSTA, we include only two types of prior knowledge, simply that the eigenvectors should be sparse and low-rank. Sparse and low-rank structure are mild forms of prior knowledge and generalize to a variety of other neural systems (they are not uniquely applicable for retinal data analysis). The former is implemented by applying an ℓ_1 penalty to the columns of $\mathbf{X}$, while the latter is implemented by applying a nuclear norm penalty to the columns of $\mathbf{X}$, with each column reshaped as a spacetime matrix. Our results in Figure 3 reveal that as we increase regularization, we get a better estimate of the STC subspace.

While we obtain a regularized STC subspace spanned by the top d eigenvectors of $\hat{\mathbf{X}}$ in (10), an important issue is what is the appropriate baseline STC subspace for comparison? If we had access to the true model, the correct subspace would be the span of the subunit filters (6). However, in the absence of such ground truth, we can instead score the performance of the STC subspace in terms of how well stimuli, after projection onto the subspace, can be used to predict spikes. We directly fit an LN-LN model with d subunits to predict spikes from stimuli, knowing that the model's d initial subunit filters and its STC matrix span the same subspace. To the extent this model has been optimized, the STC of this model is also optimized to be the most predictive STC subspace for a model of this structure. We compared this model STC subspace to that obtained from our rSTC for different amounts of data and regularization, quantified using subspace overlap (see Methods), a metric that varies between zero if the subspaces are orthogonal and share no common directions, and one if the subspaces are identical. When $d = 1$, this subspace overlap measure reduces to the cosine of the angle between the two single filter directions.

In Figure 3b, we show that with very limited amounts of data, rSTC can approximately recover the best predictive subspace obtained by LN-LN model fitting in the next section. In particular, contour plots of performance reveal that with appropriate regularization, one can recover the best predictive subspace using about 10 minutes of data; without regularization, one requires 40 minutes of data to recover a subspace with comparative predictive accuracy. We note however, that even for the full length of this experiment (40 minutes), regularization still improves our rSTC estimate. This continued improvement indicates that even 40 minutes of data does not suffice to correctly estimate an unregularized STC, in contrast to the STA. These prohibitive data requirements point to the importance of regularization in STC analyses. Indeed our much improved performance of rSTC with limited amounts of data highlights the power of regularized eigenvector recovery from the STC matrix to find stimulus subspaces that are highly predictive of neural spiking, without ever directly fitting a parameterized model of the neuron.

Hierarchical nonlinear models of the retinal response

We now move beyond descriptive statistics for efficiently learning the parameters of hierarchical, nonlinear neural models. We focus on understanding how intermediate nonlinear subunits shape the response of retinal ganglion cells by fitting two layer (LN-LN) models to the activity of ganglion cells in response to white noise flickering bar stimuli. Fitting the parameters of these hierarchical nonlinear models is challenging due to the high-dimensionality of the model parameter space, and because maximum likelihood estimation is a non-convex optimization problem due to hidden subunits intervening between inputs and outputs. Previous efforts to fit this class of models have used particular ganglion cells with a favorable non-overlapping architecture [18] or assumptions of the underlying circuitry such as spatial repetition of the nonlinear subunits [19, 20]. We formalize methods for dealing with both of these challenges by employing the proximal algorithms framework described above.

The LN-LN model architecture is schematized in Figure 1. The stimulus is passed through a set of LN subunits. Each subunit filter is a spatiotemporal stimulus filter, constrained to have unit norm. The subunit nonlinearity is parameterized using a set of basis functions (Gaussian bumps) that tile the input space [12, 16] (see Methods). This parameterization is flexible enough that we could in principle learn, for each individual subunit, any smooth nonlinearity that can be expressed as a linear combination of our basis functions. The second LN layer pools subunits through weighted summation, followed by a spiking nonlinearity that we model using a parameterized soft rectifying function $r(x) = g \log(1 + e^{x-\theta})$. Here g is an overall gain, and θ is a threshold.

Model fitting and performance

We recorded ganglion cell responses to 40 minutes of a flickering Gaussian white noise bars sequence. Using this dataset, we fit both LN and LN-LN models to the same set of cells to see which class of models better predicted cell responses. For both LN and LN-LN models we employed a proximal consensus algorithm (see Methods) to learn the model parameters by optimizing a function similar to that used in learning the regularized STA in (7). However, the data fidelity term in (7) is replaced with the log-likelihood of recorded spikes under a Poisson noise model, with firing rate being the output of the neural model. To combat the

curse of dimensionality due to the large number of parameters, we focused also on regularizing both the LN model filter and the LN-LN model subunit filters through an ℓ_1 regularization term (encouraging sparsity), and a nuclear norm term (encouraging low-rank structure). We chose the weights γ_i of the regularization penalties in (7), both for the LN and LN-LN models, through cross-validation on a small subset of cells, and then held these weights constant across all cells. Our subsequent results indicate that we do not have to fine tune these hyperparameters on a cell by cell basis to achieve good predictive performance. Finally, because different cells may have different numbers of functional subunits, for the LN-LN models, we chose the optimal number of subunits on a cell-by-cell basis by maximizing performance on held-out data through cross-validation. Note that no additional structure is imposed on the subunits such as spatial repetition or overlap, in contrast to previous work [19].

We find that the LN-LN model significantly outperforms the LN model at describing responses of ganglion cells, for all recorded salamander ganglion cells. Figure 4a shows firing rate traces for an example cell, comparing the recorded response (gray) with an LN model (green) and an LN-LN model (red). We quantify the similarity between predicted and recorded firing rate traces using either the Pearson correlation coefficient or the log-likelihood of held-out data under the model. All log-likelihood values are reported as an increase over the log-likelihood of a fixed mean firing rate model, scaled by the firing rate (yielding units of bits/spike). Summarized across $n = 23$ recorded ganglion cells, we find that the LN-LN model yields a consistent improvement over the LN model using either metric (Fig. 4bc). Overall, this demonstrated performance improvement indicates that nonlinear spatial processing is fundamental in driving ganglion cell responses, even to white noise stimuli, and that an LN model is not sufficient to capture the response to spatiotemporal white noise. This salient, intermediate rectification that we computationally identify is consistent with previous measurements of bipolar-to-ganglion cell transmission in the retina [41, 42].

The internal structure of the learned models

Given the improved performance of our hierarchical nonlinear subunit models, we examined the internal structure of the models to assess their potential to reveal insights into retinal structure and computation. Figure 5 shows a visualization of the parameters learned for an example cell. Figure 5ab shows the parameters for the classical LN model, for comparison, while Figure 5cd shows the corresponding subunit filters and subunit nonlinearities in the first stage of the LN-LN model, fit to the same cell. The subunit filters had a similar temporal structure, but smaller spatial profiles compared to that of the LN model and the nonlinearities associated with each subunit were roughly monotonic with high thresholds. These qualitative properties were consistent across all cells. We next examine more quantitatively this learned, and consistent internal structure across cells to extract lessons about the anatomy, physiology, and functional design principles underlying retinal circuitry.

Inferred hidden units quantitatively resemble bipolar cell receptive fields

Mapping the LN-LN model onto retinal anatomy leads us to believe that the first layer filters (some examples of which are shown in Figure 5c) should mimic or capture filtering properties pre-synaptic to bipolar cells in the inner retina. To examine this possibility, we compared the first layer model filters to properties of bipolar cell receptive fields. An example learned model subunit receptive field is shown in Fig. 6a, while a bipolar cell receptive field, obtained from direct intracellular recording of a bipolar cell, is shown in Figure 6b. Qualitatively, we found that the filters in the LN-LN model matched these bipolar cell RFs, as well as previously reported bipolar cell receptive fields [41, 43]: both had center-surround receptive fields with similar spatial extents. To compare these model-derived and ground-truth bipolar cell receptive fields quantitatively, we fit the spatial receptive field with a difference of Gaussians function to estimate the RF center and surround sizes. We find the RF centers for the LN-LN subunit filters are much smaller than the corresponding LN model filter. Furthermore, the size of these LN-LN subunit centers matched the size of the RF center measured from intracellular recordings of bipolar cells (Figure 6c). The recorded bipolar cells, LN model filters (ganglion cells), and LN-LN subunit filters all had similar surround sizes (Figure 6d). We note that this match between LN-LN model subunits and the RF properties of bipolar cells was not an *a priori* constraint placed on our model, but instead arose as an emergent property of predicting spiking responses to

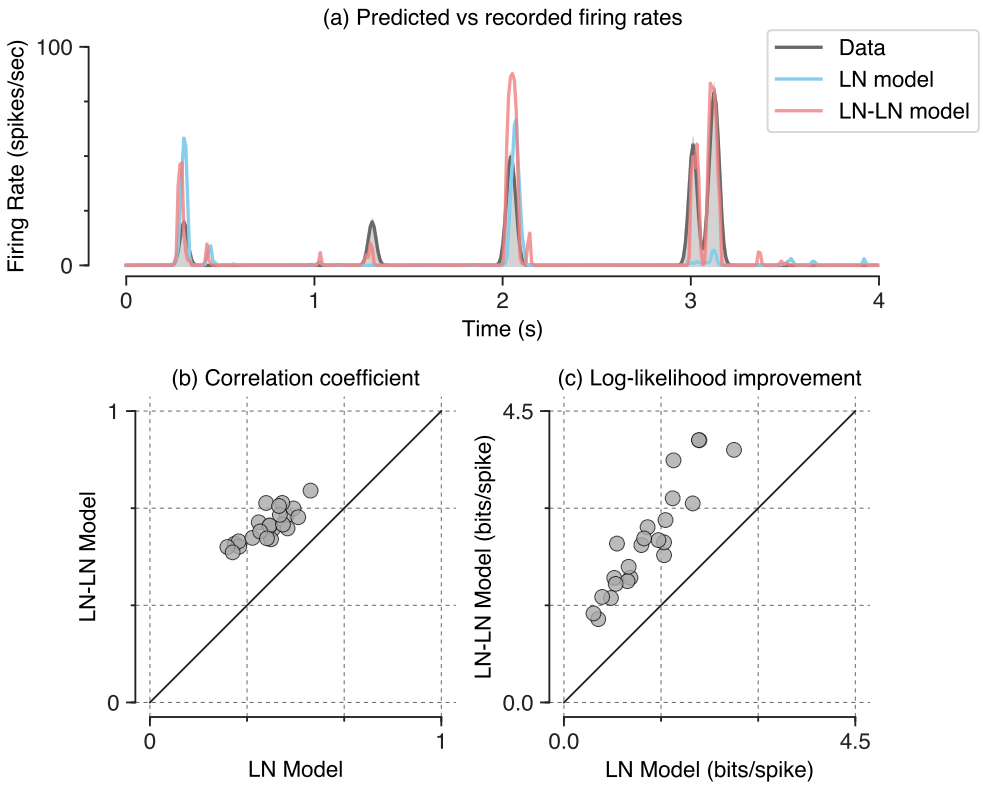

Fig 4. LN-LN models predict held-out response data better than LN models. (a) Firing rates for an example neuron. The recroded firing rate (shaded, gray), is shown along with the LN model prediction (dashed, green) and the LN-LN prediction (solid, red). (b) LN-LN performance on held out data vs. the LN model, measured using correlation coefficient between the model and held out data. Note that all cells are above the diagonal. (c) Same as in (b), but with the performance metric of log-likelihood improvement over the mean rate in bits per spike.

white noise stimuli. These results indicate that our modeling framework not only enables higher performing predictive models of the retinal response, but can also veridically reconstruct important aspects of processing in the unobserved interior of the retina, from measurements of inputs and outputs alone.

Number of inferred subunits

The number of subunits utilized in the LN-LN model for any individual cell was chosen to optimize model predictive performance on a held-out data set via cross-validation. That is, we fit models with different numbers of subunits and selected the one with the best performance on a validation set. We find that for models with more subunits than necessary, extra subunits are ignored (the learned nonlinearity for these subunits is flat, thus they do not modulate the firing rate).

Figure 7a shows the model performance, quantified as the difference between the LN-LN model and the LN model, across a population of cells as a function of the number of subunits included in the LN-LN model. We find that models with four to six subunits maximized model performance on held-out data. Note that the stimuli used here are one-dimensional spatiotemporal bars that have constant luminance across one spatial dimension. Thus each model subunit likely corresponds to the conglomeration of any actual bipolar cell subunits whose receptive fields happen to overlap a particular bar in the stimulus. Previous anatomical studies of bipolar cell density and axonal branching width [44, 45] suggest that a typical ganglion cell in the salamander retina receives input from 10-50 bipolar cells whose receptive fields are tiled across two dimensional space. The number of independently activated groups of such a two dimensional array of bipolar

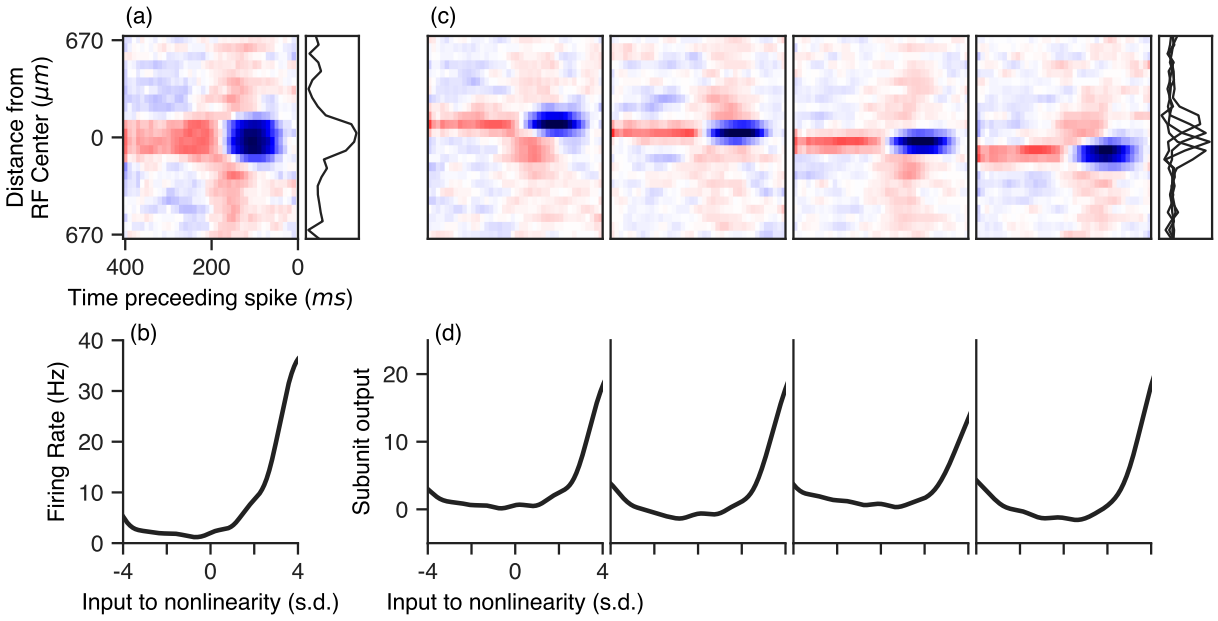

Fig 5. Example LN-LN model parameters fit to a recording of an OFF retinal ganglion cell. (a and b): LN-model parameters, consisting of a single spatial filter (a) and nonlinearity (b). (c and d) LN-LN model parameters. (c) First layer filters (top) and nonlinearities (bottom) of an LN-LN model fit to the same cell. Spatial profiles of filters are shown in gray to the right of the filters. The subunit filters have a much smaller spatial extent compared to the LN filter, but similar temporal profiles.

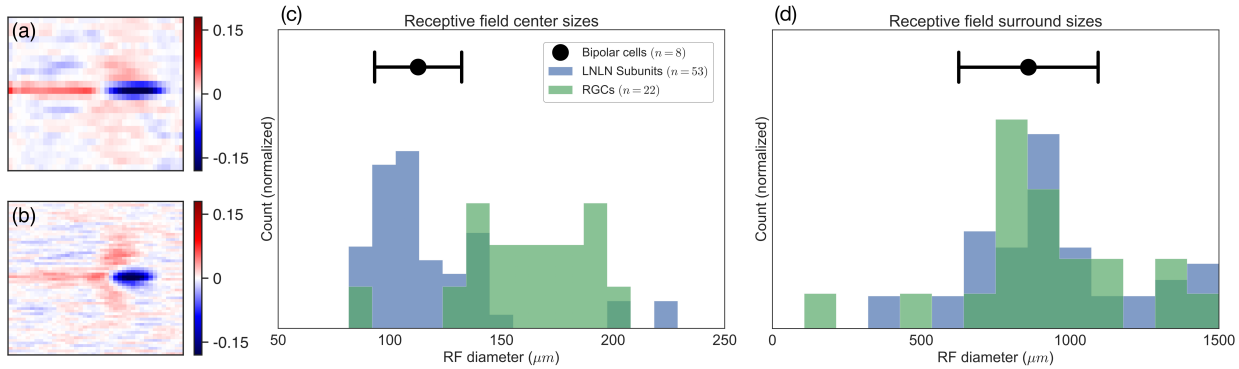

Fig 6. Comparison of subunit filter parameters with intracellular bipolar cell recordings. (a) An example subunit bipolar cell. (b) A recorded bipolar cell receptive field. (c) Receptive field centers sizes for subunit filters (blue), LN model filters (green), and recorded bipolar cells (black point). (c) Same as in (b), but with receptive field surround sizes.

cells, in response to a one dimensional bar stimulus is then expected to be reduced from the total number of bipolar cells, roughly, by a square root factor: i.e. $25 \rightarrow \sqrt{25} = 5$. This estimate is largely consistent with the typical number of subunits in Figure 7a, required to optimize model predictive performance. This estimate also suggests that the large majority of bipolar-to-ganglion cell synapses are rectifying (strongly nonlinear), as linear connections are not uniquely identifiable in an LN-LN cascade. Indeed, in the salamander retina, strong rectification appears to be the norm [41, 42]. Together, these results indicate that our framework not only veridically reconstructs the RF structure of individual bipolar cells, but also the number of bipolar cells presynaptic to a retinal ganglion cell.

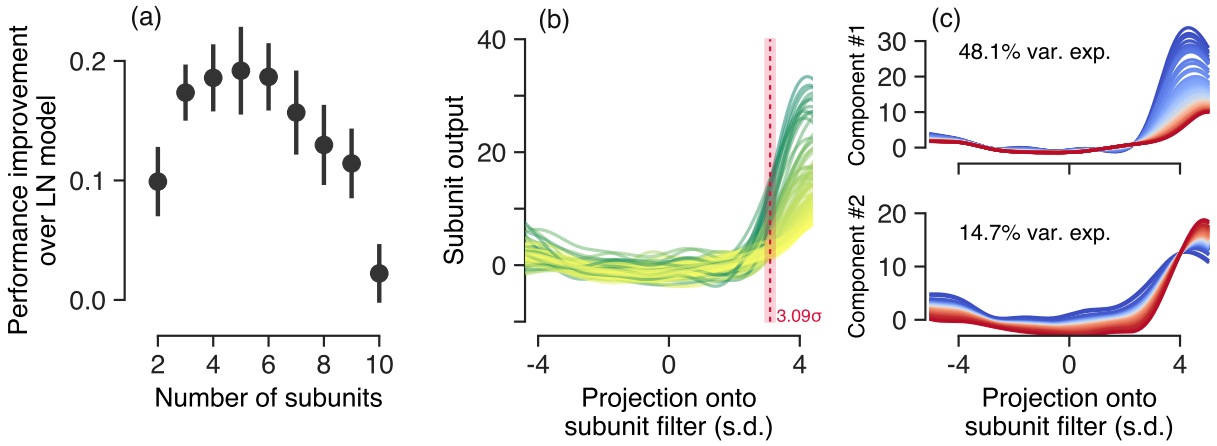

Fig 7. LN-LN model parameter analysis. (a) Performance improvement (increase in correlation coefficient relative to an LN model) as a function of the number of subunits used in the LN-LN model. Error bars indicate the standard error across 23 cells. (b) Subunit nonlinearities learned across all ganglion cells. For reference the white noise input to a subunit nonlinearity has standard deviation 1, which sets the scale of the x-axis. Red line and shaded fill indicate the mean and s.e.m. of nonlinearity thresholds (see text for details). (c) Visualization of the principal axes of variation in subunit nonlinearities by adding or subtracting principal components from the mean nonlinearity. (top) The principal axis of variation in subunit nonlinearities results in a gain change, while (bottom) the second principal axis corresponds to a threshold shift. These two dimensions captured 63% of the nonlinearity variability across cells.

LN-LN models have subunit nonlinearities with consistently high thresholds

Our LN-LN model finds the best fit smooth nonlinearity for each subunit. Each nonlinearity takes as input the projection of the stimulus onto the corresponding subunit spatiotemporal filter. Because the stimulus is a white noise stimulus, and the spatiotemporal filter is constrained to have unit norm, the resulting projection of the stimulus onto the spatiotemporal filter has a standard Normal distribution.

The nonlinearities for all of the measured subunits are overlaid in Figure 7b. Despite the fact that the model could separately learn an arbitrary function over the input for each subunit nonlinearity, we find that the nonlinearities are fairly consistent across the different subunits of many cells. Subunit nonlinearities look roughly like thresholding functions, relatively flat for most inputs but then increasing sharply after a threshold. We quantified the threshold as input for which the nonlinearity reaches 40% of the maximum output, across $n = 92$ model-identified subunits the mean threshold was 3.09 ± 0.14 (s.e.m.) standard deviations. We decomposed the set of nonlinearities using principal components analysis and show the two primary axes of variation in Figure 7c. The primary axis of variation results in a gain change, while the secondary axis induces a threshold shift. Due to the high thresholds of these nonlinearities, subunits only impact ganglion cell firing probability for large input values. We next examine the consequences of these high subunit thresholds for two important aspects of retinal processing: decorrelation across ganglion cells and functional computation within a single ganglion cell.

Stimulus decorrelation at different stages in hierarchical retinal processing

Natural stimuli have highly redundant structure. Early theories of sensory processing, known as efficient coding or redundancy reduction [46, 47], argued that sensory systems ought to remove these redundancies in order to efficiently encode natural stimuli. The simplest such redundancy is that nearby points in space and time contain similar, or correlated, luminance levels [48]. The transmission of such correlated structure would thus be highly inefficient. Efficient coding theories have been used to explain why retinal repsonses are much less correlated than natural scenes, although the mechanistic underpinnings of such decorrelation remain unclear.

Early work [49] suggested a simple mechanism: the linear center-surround receptive field of ganglion cells (and more recently, of bipolar cells [50]) could contribute to redundancy reduction simply by transmitting only differences in stimulus intensity across nearby positions in space. However, it was recently shown [51] using LN models that most of the decorrelation of naturalistic stimuli in the retina could be attributed to ganglion cell nonlinearities, as opposed to linear filtering. Given that we fit an entire layer of subunits pre-synaptic to each ganglion cell layer, we can now ask fundamental questions about the spatial representation of naturalistic images at different stages of hierarchical retinal processing, thereby localizing the mechanistic origin of stimulus decorrelation to a particular retinal layer.

To do so, we generated the response of the entire population of learned nonlinear subunits to a spatial stimulus similar to that used in [51], namely spatially pink noise, low pass filtered in time. We computed the correlation of stimulus intensities as a function of spatial distance, as well as the correlation between pairs of model units as a function of spatial distance. We examined pairs of units across different stages of the LN-LN model: after linear filtering by the subunits, after the subunit nonlinearity, and finally at the ganglion cell firing rates (the final stage). Figure 8 shows that the correlation at these stages drops off with distance between either the subunits or ganglion cells (with distances measured between receptive field centers). Similar to [51], we find that most of the decorrelation is due to nonlinear processing, as opposed to linear filtering. However, this decorrelation is primarily due to the *subunit* nonlinearities, as opposed to ganglion cell spiking nonlinearities. In fact, the correlation between the ganglion cell model firing rates slightly *increases* after pooling across subunits. The most decorrelated representation occurs just after thresholding at the subunit layer. In this manner, our modeling framework provides a more precisely localized mechanistic origin for a central tenet of efficient coding theory. Namely, our results predict that the removal of visual redundancies, through stimulus decorrelation across space, originates primarily from high-threshold nonlinearities associated with bipolar cell synapses.

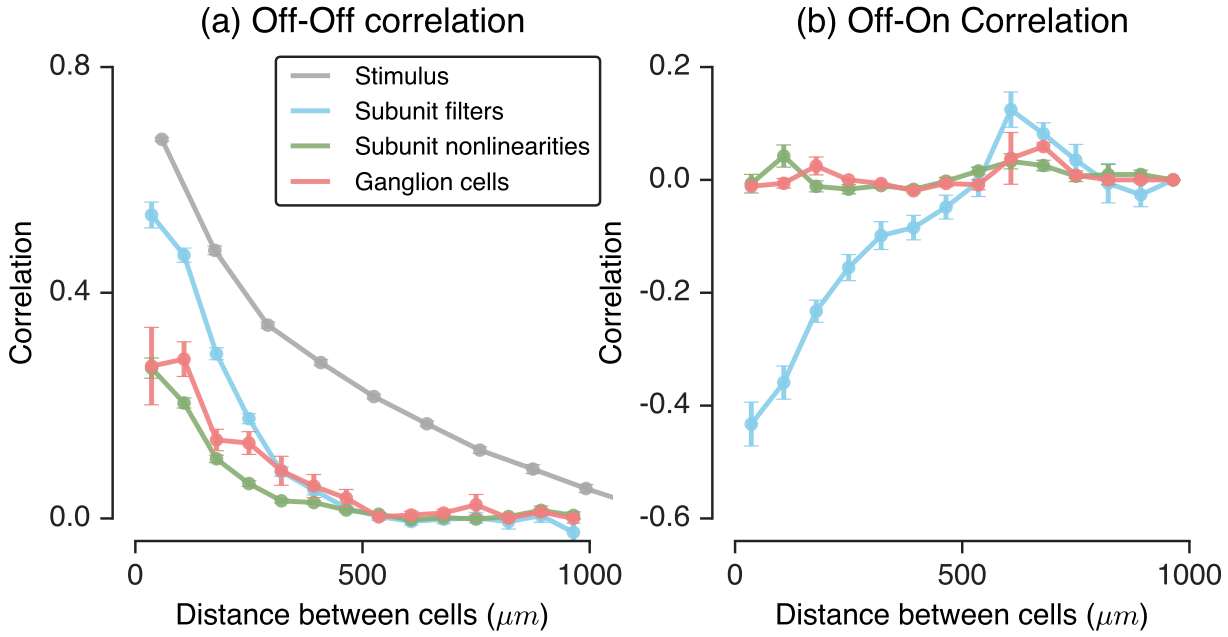

Fig 8. Decorrelation in LN-LN subunit models. A naturalistic (pink noise) stimulus was shown to a population of nonlinear subunits. The correlation in the population after filtering at the subunit layer (blue), after the subunit nonlinearity (green), and after pooling and thresholding at the ganglion cell layer (red), in addition to the stimulus correlations (gray) are shown. Left: the correlation as a function of distance on the retina for Off-Off cell pairs. Right: correlation for Off-On cell pairs. For each plot, distances were binned every $70\mu m$, and error bars are the s.e.m. within each bin.

The nature of nonlinear spatial integration across retinal subunits

Another fundamental aspect of retinal processing is the sparse and precise firing patterns of retinal ganglion cells [52], presumably due to reasons of energy efficiency [51, 53]. LN models can capture such sparse firing in only one way: by using high thresholds relative to the distribution of stimuli projected onto their linear filter. Indeed learned nonlinearities in LN models typically have such high thresholds [9]. LN-LN models, in contrast, can generate sparse responses using two qualitatively distinct operating regimes: either the subunit thresholds (first nonlinearity) could be high and the ganglion cell or spiking threshold (second nonlinearity) could be low, or the subunit thresholds could be low and spiking thresholds could be high. Both of these scenarios give rise to sparse firing at the ganglion cell output. However, they correspond to qualitatively different functional computations.

These various scenarios are diagrammed in Figure 9a-c. Each panel shows the response of a model in a two-dimensional space defined by the projection of the stimulus onto two subunit filters pre-synaptic to the ganglion cell (the two-dimensional space is easier for visualization, but the same picture holds for multiple subunits). We show the response as contours where the firing probability is constant (iso-response contours). Here, the subunit nonlinearities play a key role in shaping the geometry of the response contours, and therefore shape the computation performed by the cell. Note that the ganglion cell nonlinearity would act to rescale the output, but cannot change the shape of the contours. Therefore, it is fundamentally the subunit nonlinearities alone that determine the geometry of the response contours.

Low-threshold subunit nonlinearities give rise to concave contours (Figure 9b), whereas high-threshold subunits give rise to convex contours (Figure 9c). Because final output rate is determined by the subunit and final thresholds, both of these descriptions could yield sparse firing output with the same overall rate (by adjusting the final threshold), but correspond to different computations. Low-threshold subunits can be simultaneously active across many stimuli, and thus yield spiking when subunits are simultaneously active (an AND-like combination of inputs). On the other hand, high-threshold subunits are rarely simultaneously active and thus usually only one subunit is active during a ganglion cell firing event, giving rise to an OR-like combination of inputs. By comparison, a cell that linearly integrates its inputs would have linear contours (Figure 9a).

In our models fit to retinal ganglion cells, we find all cells are much more consistent with the high threshold OR-like model. Subunit nonlinearities tend to have high thresholds, and therefore result in convex contours (shown for different pairs of subunits for two example cells in Figure 9d-e). For each example ganglion cell, we show the corresponding model contours along with the 2 standard deviation contour of the stimulus distribution (gray oval) and the empirical firing histogram (red checkers) in the 2D space defined by the projection of the stimulus onto a given pair of subunit filters identified by the LN-LN cascade model. Note that while the stimulus is uncorrelated (i.e. white, or circular), non-orthogonality of subunit filters themselves yield correlations in the subunit activations obtained by applying each subunit filter to the stimulus. Hence the stimulus distribution in the space of subunit activations (marked by grey ovals) is not circular. In all recorded cells, we find that the composite computation implemented by retinal ganglion cell circuitry corresponds to an OR function associated with high subunit thresholds (as schematized in Figure 9c). Moreover, both the AND computation and the linear model are qualitatively ruled out by the shape of the model response contours as well as the empirical firing histogram over subunit activations, which closely tracks the model response contours (i.e. the boundaries of the red histograms are well captured by the model contours).

These results are consistent with previous studies of nonlinear spatial integration in the retina. For example, Bollinger et. al. [42] discovered convex iso-response contours for a very simple two dimensional spatial stimulus, and Kaardal et. al. [54] performed an explicit hypothesis test between an AND-like and OR-like nonlinear integration over a low dimensional subspace obtained via the un-regularized STC eigenvectors, finding that OR outperformed AND. However, the techniques of [42] are limited to fundamentally low dimensional stimuli, whereas our methods enable the discovery of iso-response contours for high dimensional stimuli by learning LN-LN models with multiple subunits. Moreover, in contrast to the hypothesis testing approach taken in [54], our general methods to learn LN-LN models reveal that an OR-model of nonlinear integration is a good model on an absolute scale of performance amongst all models in the LN-LN family, rather than simply being better than an AND-model.

A simple, qualitative picture of the distinct computational regime in which retinal-ganglion cells operate in response to white noise stimuli can be obtained by considering the geometry of the spike triggered ensemble in N dimensional space. In particular, the distribution of white noise stimuli concentrates on a constant radius sphere in N dimensional stimulus space. More precisely, any high dimensional random stimulus realization $\mathbf{x}$ has approximately the same vector length, because the fluctuations in length across realizations, relative to the mean length, is $O(1/\sqrt{N})$. Thus we can think of all likely white-noise stimuli as occurring on the $N-1$ dimensional surface of a sphere in N dimensional space. Moreover, each subunit filter can be thought of as a vector pointing in a particular direction in N dimensional stimulus space. The corresponding input to the subunit nonlinearity for any stimulus is the inner-product of the stimulus with the subunit filter, when both are viewed as N dimensional vectors. The high threshold of the subunit nonlinearity means that the subunit only responds to a small subset of stimuli on the sphere, corresponding to a small cap centered around the subunit filter. For a single subunit model (i.e. an LN model), the set of stimuli that elicit a spike then corresponds simply to this one cap (Figure 10a). In contrast, the OR like computation implemented by an LN-LN model with high subunit thresholds responds to stimuli in a region consisting of a union of small caps, one for each subunit (Figure 10b).

To verify this conceptual picture, in Figure 10c) we visualize a low, two dimensional projection of the spike-triggered stimulus ensemble for three example ganglion cells, using principal components analysis of the spike-triggered subunit activations. That is, we project the spike-triggered ensemble onto the subunit filters identified in the LN-LN model, and subsequently project those subunit activations onto the two dimensions that capture the most variance in subunit activations. This procedure identifies a subspace that captures the radial spread of subunit filters in high-dimensional stimulus space. We find that the spike-triggered ensemble projected onto this subspace (Figure 10c) curves around the radial shell defined by the stimulus distribution, and matches the conceptual picture shown in Figure 10b. For ease of visualization, we colored elements in the spike-triggered ensemble by which LN-LN model subunit was maximally active during that spike, and we normalize the spike-triggered histogram by the raw stimulus distribution (gray ovals). This picture provides a simple, compelling view for why LN models are insufficient to capture the retinal response to white noise, and further visualizes the aspect of retinal computation LN-LN models capture that the LN model does not: ganglion cells encode the union of different types of large-magnitude stimuli.

Related work

A number of reports have pointed out the limitations of the LN model, the most significant being that it is sensitive to only a single direction in stimulus space. Indeed, the pioneering work of Hochstein and Shapley [33] motivated the use of hierarchical cascade models through careful measurements of the retinal response to stimuli with spatial structures occurring at finer resolution than the width of a ganglion cell receptive field. Early work searching for more than one direction of stimulus sensitivity motivated the use of the significant eigenvectors of the STC matrix as the set of features that drives a cell [10], focusing on low dimensional full field flicker stimuli to reduce data requirements required for correctly estimating these eigenvectors. However, the precise relationship between these eigenvectors and the individual spatiotemporal filtering properties intrinsic to multiple parallel pathways remained unclear. Moreover, the STC eigenvectors obey a biologically implausible orthogonality constraint, if they are to be directly interpreted as filters of individual pathways. Recent work [54] has discussed this issue and proposed a solution by finding linear combinations of STC eigenvectors that align well with a set of pre-specified nonlinearities in order to predict spiking. Our regularization methods for finding an improved STC eigenvector subspace with limited data can greatly aid these methods. But more generally, the approach of directly fitting a hierarchical, nonlinear neural model enables us to jointly learn a set of non-orthogonal, biophysically plausible set of pathway filters, as well as an arbitrary, flexible nonlinearity for each pathway, in contrast to a potentially suboptimal, two-step, process of first finding the STC eigenvectors and then finding filters within this subspace by performing hypothesis tests between different nonlinearities.

Much recent and complementary work has focused on learning hierarchical, nonlinear subunit models [19]. Such models often require simplifying assumptions in order to make model fitting tractable. A common assumption is that subunit stimulus filters are convolutional (that is, there is a single filter that is shifted or

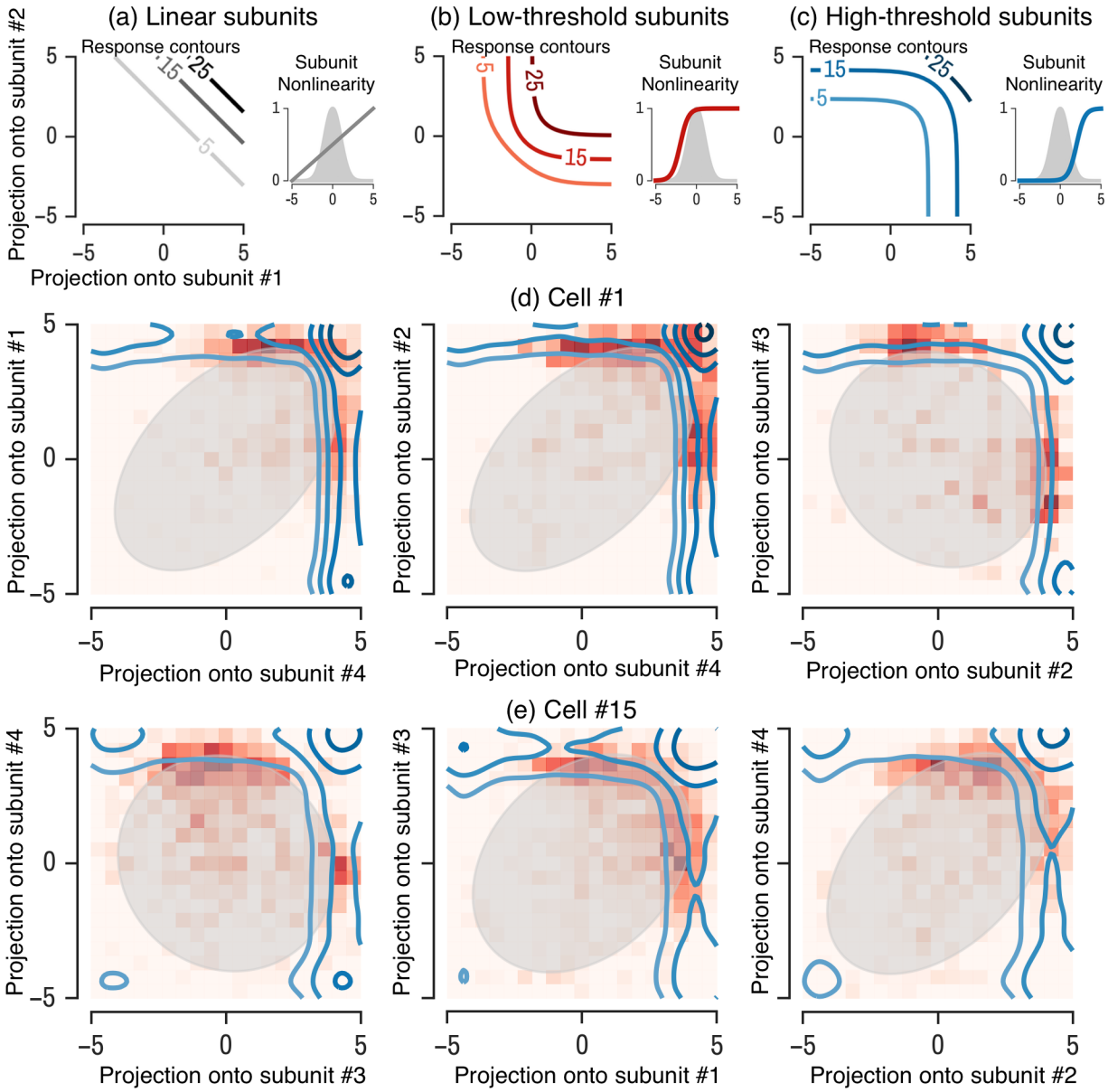

Fig 9. Visualization of subunit contours. Contours of equal firing probability are shown in a 2D space defined by the projection of the visual stimulus along each of two subunits. (a) Example contour plots for a model with low threshold subunit nonlinearities (inset) has concave contours. (b) A model with high threshold subunit nonlinearities has convex contours. (c & d) Contours from a model for two example ganglion cells, for three different pairs of subunits (left to right). In each panel, a histogram of the recorded firing rate is shown (red squares) as well as the stimulus distribution (gray oval).

copied in space) [17, 20] or that the subunit nonlinearities have a particular form (e.g. quadratic) [11, 21]. In contrast, we do not make either assumption. In particular, by not demanding that each subunit type's spatiotemporal filter is shifted across space in a translation invariant manner (the convolutional assumption), our method can detect individual variability in the spatiotemporal filters of subunits of the same type across visual space. Indeed such variability, at least at the level of ganglion cell RFs, has been shown to be functionally important in increasing retinal resolution [55].

Alternatively, Freeman et. al. [18] learned hierarchical subunit models of OFF-type midget ganglion cells

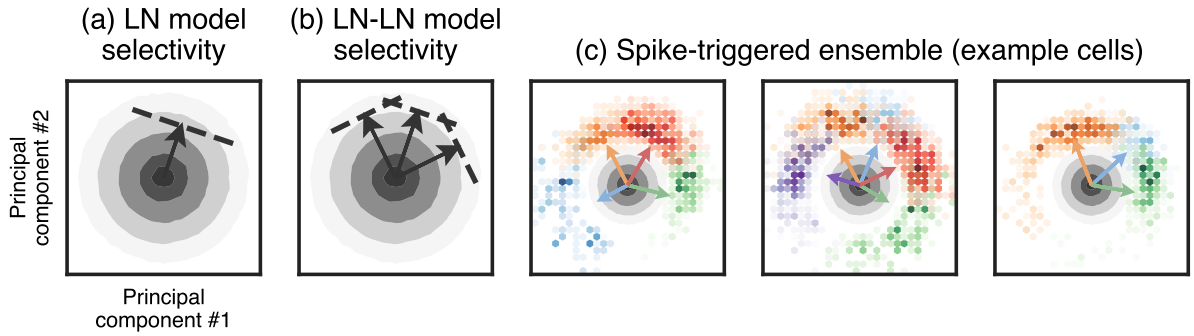

Fig 10. Stimulus selectivity in LN and LN-LN models. Each panel shows the raw stimulus distribution (gray contours) projected onto the top two principal components of the spike-triggered subunit activations (with subunits identified by the LN-LN model). The LN model (a) fires in response to stimuli in a single region, or cap, of stimulus space (indicated by the arrow and dashed threshold), whereas the LN-LN model (b) fires in response to a union of caps, each defined by an individual subunit. (c) Spike-triggered subunit activations for three representative cells are shown as colored histograms (colors indicate which model-identified subunit was maximally active during the spike), with the corresponding subunit filter directions shown as colored arrows (see text for details).

in the primate retina. OFF midget ganglion cells, which mediate high resolution vision in primates, typically receive inputs from a very small number of individual cones. The focus on this particular cell-type enabled several assumptions to ease model fitting in [18]. First, the authors assumed subunits have non-overlapping receptive fields (similar to individual cone RFs). Second, the fact that a very small number of subunits contribute to any OFF midget ganglion cell enabled a greedy algorithm for fitting subunits to succeed. Third, they did not consider the contribution of receptive field surrounds. These three simplifications, while largely valid for this particular cell type, do not hold in general for all cell types. For example, parasol RGCs are not well described by subunits with non-overlapping RFs. In contrast, we do not make any assumptions about subunit RF overlaps.

Finally, earlier work in the retina mapped out arbitrary nonlinear integration over visual space by following iso-contours of the neural response via a closed loop stimulus paradigm [42]. However, because iso-response contours are only one dimension lower than the dimension of stimulus space, this method is fundamentally limited to very low dimensional stimuli. Indeed [42] applied their methods to a 2 dimensional spatial stimulus in which luminance independently varied across two halves of visual space, but was otherwise constant in each half. Within this two-dimensional stimulus space they found one-dimensional nonlinear iso-response contours reflecting nonlinear integration across the two halves of space. In contrast, our methods enable us to learn multiple subunits over many stimulus dimensions, under the assumption of hierarchical nonlinear computation corresponding to an LN-LN cascade.

Other related work has focused on various methods for exploiting prior knowledge to learn STAs or filters in LN models. Bayesian methods are a popular approach for incorporating such knowledge. However, obtaining the minimum mean squared error (MMSE) estimate through exact Bayesian methods involves a high dimensional integral over the space of model parameters, which is generically computationally intractable. Therefore, computationally tractable Bayesian approaches often resort to either approximations, or extremely simple priors that enable analytic integration. Some interesting work applied to the estimation of sensory receptive fields (i.e. STAs and LN models) [56, 57] showed how to use a set of approximation methods known as empirical Bayes to both learn an appropriate prior from data (from a pre-specified set of priors in a parametric family) and to compute the maximum *a-priori* (MAP) estimate of the RF given the learned prior. However, each iteration of the internal loop of the algorithm used to learn the prior naively requires (without any further simplifying assumptions) the inversion of an N by N matrix, where N is the dimension of stimulus space. Such an inversion requires a computational time that is order $O(N^3)$ using the simple Gauss-Jordan elimination algorithm. These methods are thus difficult to scale both to the estimation

of high dimensional RFs, as well as to the estimation of STCs, even for relatively low dimensional stimuli. For example, if these methods were straightforwardly extended to STC estimation using parameterized priors over covariance matrices, they would require the inversion of N^2 by N^2 matrices, thus demanding a step of computational complexity $O(N^6)$ in each iteration of the internal loop required to learn the prior over STC matrices.

In contrast, our methods based on proximal consensus algorithms for regularizing the eigenvectors of STC matrices are much more efficient. The most costly step in terms of computational time for such regularization involves computing the proximal operator for the nuclear norm applied to each of the d columns of the matrix $\mathbf{X}$ in (10), when each column is viewed as an N_s by N_t spatiotemporal matrix. The computational cost of computing any such SVD is $O([\min(N_s, N_t)] \times [\max(N_s, N_t)]^2)$. If the number of spatial bins N_s and number of temporal bins N_t are both of the same order of magnitude (and therefore each is of order $\sqrt{N}$), then the computational cost of a single SVD is $O(N^{3/2})$. Thus the computational cost of STC eigenvector regularization through nuclear norm scales with stimulus dimension N as $N^{3/2}$ in contrast to N^6, as would be the case for empirical Bayes methods applied to STC matrices. Also, the analog of learning the prior from a parametric family of priors in empirical Bayes corresponds to optimizing the hyperparameters, or regularization weights α_i in (10). As we have seen, due to the robustness of our performance results to choices of regularization weights within a large range, we can find these optimal regularization weights by cross-validation through a relatively coarse search over hyperparameter space for a small number of cells, and then hold them fixed for all future recorded cells. One important restriction of the regularization methods that we use is that they cannot handle some of the more flexible priors employed in empirical Bayes methods, especially those involving correlations between different components of STA, STC, or neural model parameters. However, our performance results indicate that we can learn each of these objects using very little data, and therefore demonstrate that even the simple regularizers amenable to our proximal consensus algorithms framework provide both high statistical efficiency (high estimation performance with limited data) as well as high computational efficiency (high estimation performance with limited computational time).

Discussion

In summary, we provide a computational framework to model stimulus driven neural processing in circuits with multiple parallel, hierarchical nonlinear pathways using limited experimental data. This framework elucidates relationships between biophysical circuit properties (spatiotemporal filtering properties of individual pathways, and nonlinear pooling of such pathways across multiple cell layers) and the statistical structure of the spike-triggered stimulus ensemble. We find that the mean of this ensemble (the STA) is simply a weighted combination of pathway filters, weighted by the average gain, or slope of the nonlinearity relating pathway activation to neural spiking probability. Similarly, the second moment of this ensemble (the STC) is a sum of outer products of pathway filters, weighted by the multi-dimensional curvature of the pooling nonlinearity. This latter fact implies that the significant STC eigenvectors, just like the STA, are linear combinations of pathway filters.

These connections enable us to derive theoretically principled algorithms for estimating STAs and STC eigenspaces by translating prior knowledge about the spatiotemporal filtering properties of parallel neural pathways into expected structure in the STA and STC. These theoretically principled algorithms yield highly efficient estimation methods from both a statistical and a computational perspective. Statistically, these methods require little data to achieve high estimation performance; indeed the regularization approach can reduce the amount of data by as much as a factor of 10 to estimate the STA and a factor of 8 to estimate STC eigenvectors, at a level of performance obtained by no regularization. Moreover, computationally, the time complexity of our algorithms scales favorably with respect to stimulus dimensionality N due to the use of efficient proximal operators (see Methods). Indeed, the regularization component of our algorithm for STC eigenspace estimation has time complexity $O(N^{3/2})$, compared to $O(N^6)$ for generic empirical Bayes methods (see Related Work section for more details).

Moving beyond the widely used descriptive statistics of the STA and STC, we also applied our proximal consensus algorithm based framework to directly learn hierarchical nonlinear models of the retinal response to white noise stimuli. These models have the advantage of directly learning biophysically plausible pathway

filters and nonlinearities, rather than simply extracting linear combinations of them with weighting coefficients that depend on the shapes of nonlinearities. We found that models employing two stages of linear and nonlinear computation, namely LN-LN models, demonstrated a robust improvement over the classical standard of LN models at predicting responses to white noise across a population of ganglion cells.

Beyond performance considerations alone, the gross architecture of the LN-LN model maps directly onto the hierarchical, cascaded, anatomy of the retina, thereby enabling the possibility that we can generate quantitative hypotheses about neural signal propagation and computation in the unobserved interior of the retina simply by examining the structure of our model's interior. Since learning our model only requires measurements of the inputs and outputs to the retinal circuit, this approach is tantamount to the computational reconstruction of unobserved hidden layers of a neural circuit. The advantage of applying this method in the retina is that we can experimentally validate aspects of this computational reconstruction procedure.

Indeed, using intracellular recordings of bipolar cells, we found that our learned subunits matched properties of bipolar cells, both in terms of their receptive field center-surround structure, and in terms of the approximate number of bipolar cells connected to a ganglion-cell. However care must be taken not to *directly* identify the learned subunits in our model with bipolar cells in the retina. Instead, they should be thought of as *functional* subunits that reflect the combined contribution of not only bipolar cells, but also horizontal cells and amacrine cells that sculpt the composite response of retinal ganglion cells to stimuli. Nevertheless, the correspondence between subunits and bipolar cell RFs (which are also shaped by horizontal cell processes), suggests learning functional subunits that loosely correspond to the composite effect bipolar cells and associated circuitry have on ganglion-cell synapses is an important aspect of correctly describing the overall ganglion cell response, even to white-noise stimuli.

The interior of our models also reveal several functional principles underlying retinal processing. First, all subunits across all cells had strikingly consistent nonlinearities corresponding to monotonically increasing threshold-like functions with high thresholds. This inferred biophysical property yields several important consequences for neural signal processing in the inner retina. First, it predicts that subunit activation patterns are sparse across the ensemble of stimuli, with typically only one subunit actively contributing to any given ganglion cell spike. Second, it predicts that the dominant source of stimulus decorrelation, a central tenet of efficient coding theory, has its mechanistic origin at the first strongly nonlinear processing stage of the retina, namely in the synapse from bipolar cells to ganglion cells. Third, it implies that the composite function computed by individual retinal ganglion cells corresponds to a Boolean OR function of bipolar cell feature detectors.

Taken together, our framework provides a unified way to estimate hierarchical nonlinear models of sensory processing by combating both the statistical and computational curses of dimensionality associated with learning such models. When applied to the retina, these techniques demonstrably recover known properties in the interior of the retina without requiring direct measurements of these properties. Moreover, by identifying candidate mechanisms for cascaded nonlinear computation in retinal circuitry, our results provide a higher resolution view of retinal processing compared to classic LN models, thereby setting the stage for the next generation of efficient coding theories that may provide a normative explanation for such processing. For example, considerations of efficient coding have been employed to explain aspects of the linear filter [58] and nonlinearity [51] of retinal ganglion cells when viewed through the coarse lens of an LN model. An important direction for future research would be the extension of these basic theories to more sophisticated ones that can explain the higher resolution view of retinal processing uncovered by our learned LN-LN models. Principles that underlie such theories of LN-LN processing might include subthreshold noise rejection [59, 60], sensitivity to higher order statistical structure in natural scenes, and energy efficiency [53]. Indeed the ability to extract these models from data in both a statistically and computationally efficient manner constitutes an important step in the genesis and validation of such a theory.

Another phenomenon robustly observed in the retina is adaptation to the luminance and contrast of the visual scene. Adaptation is thought to be a critical component of the retinal response to natural scenes [26], and a promising direction for extensions of our work would be to include luminance and contrast adaptation in subunit models. Luminance adaptation (adapting to the mean light intensity) is mediated by photoreceptor cells, and could be modeled by prepending a simple photoreceptor model (e.g. [61]) to an LN-LN model. There are two major sites of contrast adaptation, at the bipolar-to-ganglion cell

synapse [62, 63] and at the spiking mechanism of ganglion cells [62, 64]. Extending the simple thresholding nonlinearities in our model with a dynamical model of adaptation (e.g. [15]) is a first step towards understanding the interaction between nonlinear subunits and adaptation.

Beyond the retina, multiple stages of cascaded nonlinear computation constitutes a ubiquitous motif in the structure and function of neural circuits. The tools we have developed to elucidate hierarchical nonlinear processing in the retina are applicable across neural systems more generally. Indeed, the proximal algorithms we use for optimization allow for a single framework in which many possible forms of prior information (regularization) can be utilized. Many types of prior knowledge about model parameters in neural systems involve non-smooth (and non-differentiable) mathematical formulations. For example, stimulus filters, synaptic weight matrices, and population firing patterns, are commonly thought to be some combination of smooth, low-rank, and sparse. Proximal algorithms are a fast and scalable way of utilizing this prior information to fit models and gain insight into neural systems using modest amounts of experimental data. Thus we hope our work provides mathematical and computational tools for efficiently extracting and analyzing both informative descriptive statistics and hierarchical nonlinear models across many different sensory modalities, brain regions, and stimulus ensembles, thereby furthering our understanding of general principles underlying nonlinear neural computation.

Methods

LN-LN models

We optimize the parameters using a maximum likelihood objective assuming a Poisson noise model for spiking. Rather than optimize all of the parameters simultaneously, we alternate between optimizing the filter parameters and optimizing the nonlinearity parameters (and find that this is more stable than optimizing everything jointly).

The nonlinearities are parameterized using a set of tent basis functions (Gaussian bumps) that tile the relevant input space [12, 16]. We typically use 30 evenly spaced Gaussian bumps that tile the range spanned by the projection of the stimulus onto the linear filter. For example, a nonlinearity is parameterized as

$$\begin{aligned} h(u) &= \sum_{j=1}^{p} a_j \phi_j(u) \\ &= \sum_{j=1}^{p} a_j \phi(u - \Delta_j), \end{aligned}$$

where ϕ is the tent basis function, such as $\phi(x) = \exp(-x^2)$, Δ_j indicates the spacing between the tents, and a_j is a parameter. Since the basis functions and spacings are fixed beforehand, the only free parameters are the a_j's.

We choose the number of subunits using a validation set. That is, we fit models with increasing numbers of subunits until held-out performance on a validation set decreases.

Proximal operators and algorithms

The framework of *proximal algorithms* allows us to optimize non-differentiable and non-smooth terms easily. The name *proximal* comes from the fact that these algorithms utilize the *proximal operator* (defined below) as subroutines or steps in the optimization algorithm. For brevity, we skip the derivation of these algorithms, instead referring the reader to the more thorough treatment by Parikh and Boyd [65] or Polson et al. [66]. The proximal operator for a function ϕ given a starting point v is defined as:

$$\mathcal{P}_\phi(v, \rho) = \text{argmin}_x \left[\phi(x) + \frac{\rho}{2} \|x - v\|_2^2 \right]. \tag{12}$$

The proximal operator is a mapping from a starting point v to a new point x that tries to minimize the function $\phi(x)$ (first term above) but stays close to the starting point v (second term). The proximal operator

is a building block that we will use to create more complicated algorithms. We will take advantage of the fact that for many functions ϕ of interest to us, we can analytically compute their proximal operators, thus making these operators a computationally cheap building block.

We used these building blocks to solve optimization problems involving the sum of a number of simpler terms:

$$F(x) = \sum_{i=1}^{k} \phi_i(x) \tag{13}$$

where in our application the ϕ_i's represent either data fitting terms (e.g. a log-likelihood) or different regularization terms or penalty functions on the parameters, x. With respect to learning the parameters of a linear filter in an LN model, the objective consists of a log-likelihood $f(x)$ along with regularization penalties that impose prior beliefs about the filter, x. We focus on two main penalties. Sparsity, which encodes the belief that many filter coefficients are zero, is penalized by the ℓ_1-norm ($\phi_1(x) = \|x\|_1$). Additionally, spatiotemporal filters are often approximately space-time separable (they are well modeled as the outer product of a few spatial and temporal factors). We encoded this penalty by the nuclear norm, ℓ_*, which encourages the parameters x, when reshaped to form a spatiotemporal matrix, to be a low-rank matrix (the nuclear norm ℓ_* of a matrix is simply the sum of its singular values). Another natural penalty would be one that encourages the parameters to be smooth in space and/or time, which could be accomplished by applying an ℓ_1 or ℓ_2 penalty to the spatial or temporal differences in parameters. As shown below, these types of penalties are easy to incorporate into the proximal algorithm framework. Other commonly used regularization penalties, and their corresponding proximal operators, are listed in Table 1.

The proximal consensus algorithm is an iterative algorithm for solving (13) that takes a series of proximal operator steps. It first creates a copy of the variable x for each term ϕ_i in the objective. The algorithm proceeds by alternating between taking proximal steps for each function ϕ_i using that variable copy x_i, and then enforcing all of the different variable copies to agree (reach consensus) by averaging them. The algorithm is:

$$\begin{aligned} x_i^{k+1} &= \mathcal{P}_{\phi_i}(\bar{x}^k - u_i^k) \\ \bar{x}^{k+1} &= \frac{1}{k}\sum_{i=1}^{k} x_i \\ u_i^{k+1} &= u_i^k + x_i^{k+1} - \bar{x}^{k+1}, \end{aligned}$$

where i indexes each of the terms in the objective function, x_i is a copy of the variable, $\bar{x}$ is the average of the variable copies, and u_i is known as a dual variable that can be thought of as keeping a running average of the error between each variable copy and the average. Intuitively, we can think of each variable copy x_i as trying to minimize a single term ϕ_i in the objective, and the average, or consensus $\bar{x}$ forces the different copies to agree on the best value for the global parameters. After convergence, each copy x_i will be close to the mean value $\bar{x}$, which is the set of parameters that minimizes the original composite objective.

This algorithm has a number of desirable properties. First, the updates for each term x_i can be carried out in parallel, therefore allowing for speedups when run on a cluster or multi-core computer. Second, it converges even when terms in the objective are non-differentiable. Due to the repeated application of the proximal operator, this algorithm works best when the terms ϕ_i have proximal operators that are easy to compute.

This is exactly the case for the regularization terms described above: for the ℓ_1 norm, the proximal operator corresponds to soft thresholding of the parameters. For the nuclear norm, the proximal operator corresponds to soft thresholding of the singular values of parameters reshaped as a matrix. Occasionally, the proximal operator may not have a closed form solution. In this case, the proximal step can be carried out through gradient based optimization of (12) directly. This is the case for some log-likelihoods, such as the log-likelihood of a particular firing rate under Poisson spiking. In this case, gradient step based optimization of (12) often dominates the computational cost of the algorithm. As many methods for fitting neural models involve gradient step updates on the log-likelihood, such methods can then be augmented with additional

regularization terms with no appreciable effect on runtime, by using proximal consensus algorithms for optimization. Our code for solving formulating and solving optimization problems using proximal algorithms is provided online at `https://github.com/ganguli-lab/proxalgs`.

Table 1. Common regularization penalties and their proximal operators (in closed form).

Penalty function, $\phi(\mathbf{x})$	**Proximal operator**, $\mathcal{P}_\phi(\mathbf{v}, \rho)$	**Computational complexity**
ℓ_2-norm, $\gamma\|\mathbf{x}\|_2$	$\frac{\mathbf{v}}{1+1/\rho}$	$\mathcal{O}(n)$
ℓ_1-norm, $\gamma\|\mathbf{x}\|_1$	$\begin{cases} v_i - \gamma/\rho & v_i \geq \gamma/\rho \\ v_i + \gamma/\rho & v_i \leq -\gamma/\rho \\ 0 & \text{otherwise} \end{cases}$	$\mathcal{O}(n)$
Nuclear norm, $\|X\|_*$	$\mathbf{X} = \mathbf{USV}^T$, $\mathcal{P}_\phi(\mathbf{v}, \rho) = \mathbf{US}'\mathbf{V}^T$, $\mathbf{S}' = \begin{cases} \sigma_i - \gamma/\rho & \sigma_i \geq \gamma/\rho \\ 0 & \text{otherwise} \end{cases}$	$\mathcal{O}(\min(mn^2, nm^2))$
Non-negativity, $\mathcal{I}(x > 0)$	$\begin{cases} v_i & v_i > 0 \\ 0 & v_i \leq 0 \end{cases}$	$\mathcal{O}(n)$

Experiments

Experimental data was collected from the tiger salamander retina using a multi-electrode array (Multi-channel systems). Isolated ganglion cells were identified using custom spike sorting software. The stimulus used was a 100Hz white noise bars stimulus, where the luminance of each bar was drawn independently from a Gaussian distribution. Spatially, the stimulus spanned approximately 2.8mm on the retina (50 bars at $55.5\mu m$ / bar).

Subspace overlap

We quantify the overlap between two subspaces as the average of the cosine of the principal (or canonical) angles between the subspaces. The principal angles between two subspaces $\mathbf{X} \in \mathcal{R}^{n \times p}$ and $\mathbf{Y} \in \mathcal{R}^{n \times q}$ generalize the idea of angles between vectors. Here we describe a pair of p and q dimensional subspaces in n dimensional space as the span of the columns of the matrices $\mathbf{X}$ and $\mathbf{Y}$. Assuming without loss of generality that $p \leq q$, then we have p principal angles $\theta_1, \ldots, \theta_p$ that are defined recursively for $k = 1, \ldots, p$ as:

$$\cos\theta_k = \max_{\mathbf{x} \in \mathbf{X}} \max_{\mathbf{y} \in \mathbf{Y}} \mathbf{x}^T\mathbf{y} = \mathbf{x_k}^T\mathbf{y_k},$$

subject to the constraints that the vectors are unit vectors ($\mathbf{x}^T\mathbf{x} = \mathbf{y}^T\mathbf{y} = 1$) and they are orthogonal to the previously identified vectors ($\mathbf{x_j}^T\mathbf{x} = 0, \mathbf{y_j}^T\mathbf{y} = 0$ for $j = 1, 2, \ldots, k-1$). That is, the first principal angle is found by finding a unit vector within each subspace such that the correlation, or dot product, between these vectors (these are known as the principal vectors) is maximized. This dot product is the cosine of the principal angle. Then, each subsequent principal angle is found by performing the same maximization but restricting each new pair of vectors to be orthogonal to the previous principal vectors in each subspace. The principal angles can be efficiently computed via the QR decomposition [67]. We define subspace overlap as the average of the cosine of the principal angles, $\frac{1}{p}\sum_{k=1}^{p} \cos\theta_k$. This quantity is at most 1 (for two subspaces that span the same space), and at least 0 (for two orthogonal subspaces that share no common directions).

Acknowledgments

The authors would like to thank David Kastner for bipolar receptive field data, Ben Naecker, Ben Poole, and Lane McIntosh for discussions, and Stéphane Deny, Ben Naecker, and Alex Williams for comments on the manuscript.

References

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

2. Paninski L. Convergence properties of three spike-triggered analysis techniques. Network: Comput Neural Syst. 2003;14:437–464.

3. Schwartz O, Pillow JW, Rust NC, Simoncelli EP. Spike-triggered neural characterization. Journal of Vision. 2006;6(4). doi:10.1167/6.4.13.

4. Aljadeff J, Lansdell BJ, Fairhall AL, Kleinfeld D. Analysis of neuronal spike trains, deconstructed. Neuron. 2016;91(2):221–259.

5. Van Steveninck RDR, Bialek W. 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. 1988;234(1277):379–414.

6. Agüera y Arcas B, Fairhall AL. What causes a neuron to spike? Neural Computation. 2003;15(8):1789–1807.

7. Brenner N, Bialek W, Van Steveninck RdR. Adaptive rescaling maximizes information transmission. Neuron. 2000;26(3):695–702.

8. Schwartz O, Chichilnisky E, Simoncelli EP. Characterizing neural gain control using spike-triggered covariance. Advances in neural information processing systems. 2002;1:269–276.

9. Baccus SA, Meister M. Fast and slow contrast adaptation in retinal circuitry. Neuron. 2002;36(5):909–919.

10. Fairhall AL, Burlingame AC, Narasimhan R, Harris RA, Puchalla JL, Berry MJ. Selectivity for Multiple Stimulus Features in Retinal Ganglion Cells. J Neurophysiol. 2006;96(5):2724–2738. doi:10.1152/jn.00995.2005.

11. Rust NC, Schwartz O, Movshon JA, Simoncelli EP. Spatiotemporal Elements of Macaque {V1} Receptive Fields. Neuron. 2005;46(6):945 – 956. doi:http://dx.doi.org/10.1016/j.neuron.2005.05.021.

12. Pillow J, Shlens J, Paninski L, Sher A, Litke A, Chichilnisky E, et al. Spatio-temporal correlations and visual signalling in a complete neuronal population. Nature. 2008;454(7207):995–9. doi:10.1038/nature07140.

13. Heitman A, Brackbill N, Greschner M, Sher A, Litke AM, Chichilnisky E. Testing pseudo-linear models of responses to natural scenes in primate retina. bioRxiv. 2016; p. 045336.

14. McFarland JM, Cui Y, Butts DA. Inferring nonlinear neuronal computation based on physiologically plausible inputs. PLoS computational biology. 2013;9(7):e1003143. doi:10.1371/journal.pcbi.1003143.

15. Ozuysal Y, Baccus SA. Linking the computational structure of variance adaptation to biophysical mechanisms. Neuron. 2012;73(5):1002–1015.

16. Keat J, Reinagel P, Reid RC, Meister M. Predicting Every Spike: A Model for the Responses of Visual Neurons. Neuron. 2001;30(3):803 – 817. doi:http://dx.doi.org/10.1016/S0896-6273(01)00322-1.

17. Wu A, Park IM, Pillow JW. Convolutional spike-triggered covariance analysis for neural subunit models. In: Advances in Neural Information Processing Systems; 2015. p. 793–801.

18. Freeman J, Field GD, Li PH, Greschner M, Gunning DE, Mathieson K, et al. Mapping nonlinear receptive field structure in primate retina at single cone resolution. eLife. 2015;4:e05241.

19. Real E, Asari H, Gollisch T, Meister M. Neural Circuit Inference from Function to Structure. Current Biology. 2017;.

20. Vintch B, Zaharia AD, Movshon JA, Simoncelli EP, et al. Efficient and direct estimation of a neural subunit model for sensory coding. In: NIPS; 2012. p. 3113–3121.

21. Park IM, Pillow JW. Bayesian spike-triggered covariance analysis. In: Advances in neural information processing systems; 2011. p. 1692–1700.

22. Baccus SA. Timing and Computation in Inner Retinal Circuitry. Annual Review of Physiology. 2007;69(1):271–290. doi:10.1146/annurev.physiol.69.120205.124451.

23. Gollisch T, Meister M. Eye Smarter than Scientists Believed: Neural Computations in Circuits of the Retina. Neuron. 2010;65(2):150 – 164. doi:http://dx.doi.org/10.1016/j.neuron.2009.12.009.

24. Geffen MN, de Vries SEJ, Meister M. Retinal Ganglion Cells Can Rapidly Change Polarity from Off to On. PLoS Biol. 2007;5(3):e65. doi:10.1371/journal.pbio.0050065.

25. Aljadeff J, Segev R, Berry II MJ, Sharpee TO. Spike triggered covariance in strongly correlated Gaussian stimuli. PLoS Comput Biol. 2013;9(9):e1003206.

26. Rieke F, Rudd ME. The challenges natural images pose for visual adaptation. Neuron. 2009;64(5):605–616.

27. Carandini M, Demb JB, Mante V, Tolhurst DJ, Dan Y, Olshausen BA, et al. Do we know what the early visual system does? The Journal of neuroscience. 2005;25(46):10577–10597.

28. McIntosh L, Maheswaranathan N, Nayebi A, Ganguli S, Baccus S. Deep learning models of the retinal response to natural scenes. In: Advances in Neural Information Processing Systems; 2016. p. 1361–1369.

29. Demb JB, Zaghloul K, Haarsma L, Sterling P. Bipolar cells contribute to nonlinear spatial summation in the brisk-transient (Y) ganglion cell in mammalian retina. The Journal of neuroscience. 2001;21(19):7447–7454.

30. Turner MH, Rieke F. Synaptic rectification controls nonlinear spatial integration of natural visual inputs. Neuron. 2016;90(6):1257–1271.

31. Werblin FS. Six different roles for crossover inhibition in the retina: correcting the nonlinearities of synaptic transmission. Visual neuroscience. 2010;27(1-2):1–8.

32. Gollisch T. Features and functions of nonlinear spatial integration by retinal ganglion cells. Journal of Physiology-Paris. 2013;107(5):338–348.

33. Hochstein S, Shapley RM. Linear and nonlinear spatial subunits in Y cat retinal ganglion cells. The Journal of Physiology. 1976;262(2):265–284.

34. Victor JD, Shapley RM. The nonlinear pathway of Y ganglion cells in the cat retina. The Journal of General Physiology. 1979;74(6):671–689. doi:10.1085/jgp.74.6.671.

35. Ölveczky BP, Baccus SA, Meister M. Segregation of object and background motion in the retina. Nature. 2003;423(6938):401–408.

36. Theis L, Chagas A, Arnstein D, Schwarz C, Bethge M. Beyond GLMs: A Generative Mixture Modeling Approach to Neural System Identification. PLoS Computational Biology. 2013;9(11):e1003356. doi:10.1371/journal.pcbi.1003356.

37. Rajan K, Marre O, Tkačik G. Learning quadratic receptive fields from neural responses to natural stimuli. Neural computation. 2013;25(7):1661–1692.

38. Friedman J, Hastie T, Tibshirani R. The elements of statistical learning. vol. 1. Springer series in statistics Springer, Berlin; 2001.

39. Hosoya T, Baccus SA, Meister M. Dynamic predictive coding by the retina. Nature. 2005;436(7047):71–77.

40. Vu VQ, Cho J, Lei J, Rohe K. Fantope Projection and Selection: A near-optimal convex relaxation of sparse PCA. In: Advances in Neural Information Processing Systems 26; 2013. p. 2670–2678.

41. Asari H, Meister M. Divergence of visual channels in the inner retina. Nature neuroscience. 2012;15(11):1581–1589.

42. Bölinger D, Gollisch T. Closed-Loop Measurements of Iso-Response Stimuli Reveal Dynamic Nonlinear Stimulus Integration in the Retina. Neuron. 2012;73(2):333 – 346. doi:http://dx.doi.org/10.1016/j.neuron.2011.10.039.

43. Asari H, Meister M. The projective field of retinal bipolar cells and its modulation by visual context. Neuron. 2014;81(3):641–652.

44. Wu SM, Gao F, Maple BR. Functional architecture of synapses in the inner retina: segregation of visual signals by stratification of bipolar cell axon terminals. The Journal of Neuroscience. 2000;20(12):4462–4470.

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

46. Attneave F. Some informational aspects of visual perception. Psychological review. 1954;61(3):183.

47. Barlow HB. In: Possible principles underlying the transformations of sensory messages. Cambridge, MA: MIT Press; 1961. p. 217–234.

48. Hyvärinen A, Hurri J, Hoyer PO. Natural Image Statistics: A Probabilistic Approach to Early Computational Vision.. vol. 39. Springer Science & Business Media; 2009.

49. Atick JJ, Redlich AN. Towards a theory of early visual processing. Neural Computation. 1990;2(3):308–320.

50. Franke K, Berens P, Schubert T, Bethge M, Euler T, Baden T. Inhibition decorrelates visual feature representations in the inner retina. Nature. 2017;542(7642):439–444.

51. Pitkow X, Meister M. Decorrelation and efficient coding by retinal ganglion cells. Nature neuroscience. 2012;15(4):628–635.

52. Berry MJ, Warland DK, Meister M. The structure and precision of retinal spike trains. Proceedings of the National Academy of Sciences. 1997;94(10):5411–5416.

53. Sterling P, Laughlin S. Principles of neural design. MIT Press; 2015.

54. Kaardal J, Fitzgerald JD, Berry MJ, Sharpee TO. Identifying functional bases for multidimensional neural computations. Neural computation. 2013;25(7):1870–1890.

55. Liu YS, Stevens CF, Sharpee TO. Predictable irregularities in retinal receptive fields. Proceedings of the National Academy of Sciences. 2009;106(38):16499–16504.

56. Sahani M, Linden JF. Evidence optimization techniques for estimating stimulus-response functions. In: Advances in Neural Information Processing Systems 15: Proceedings of the 2002 Conference. vol. 15. MIT Press; 2003. p. 317.

57. Park M, Pillow JW. Receptive field inference with localized priors. PLoS Comput Biol. 2011;7(10):e1002219.

58. Atick JJ, Redlich AN. What does the retina know about natural scenes? Neural computation. 1992;4(2):196–210.

59. Field GD, Rieke F. Nonlinear signal transfer from mouse rods to bipolar cells and implications for visual sensitivity. Neuron. 2002;34(5):773–785.

60. Bialek W. Biophysics: searching for principles. Princeton University Press; 2012.

61. Clark DA, Benichou R, Meister M, da Silveira RA. Dynamical adaptation in photoreceptors. PLOS Comput Biol. 2013;9(11):e1003289.

62. Kim KJ, Rieke F. Temporal contrast adaptation in the input and output signals of salamander retinal ganglion cells. The Journal of Neuroscience. 2001;21(1):287–299.

63. Manookin MB, Demb JB. Presynaptic mechanism for slow contrast adaptation in mammalian retinal ganglion cells. Neuron. 2006;50(3):453–464.

64. Weick M, Demb JB. Delayed-rectifier K channels contribute to contrast adaptation in mammalian retinal ganglion cells. Neuron. 2011;71(1):166–179.

65. Parikh N, Boyd S. Proximal algorithms. Foundations and Trends in optimization. 2013;1(3):123–231.

66. Polson NG, Scott JG, Willard BT, et al. Proximal Algorithms in Statistics and Machine Learning. Statistical Science. 2015;30(4):559–581.

67. Björck Å, Golub GH. Numerical methods for computing angles between linear subspaces. Mathematics of computation. 1973;27(123):579–594.