UNIVERSITÉ DE GENÈVE

Archive ouverte UNIGE
https://archive-ouverte.unige.ch

Article scientifique | Article | 2011

Published version | Open Access

This is the published version of the publication, made available in accordance with the publisher's policy.

Insights from a simple expression for linear fisher information in a recurrently connected population of spiking neurons

Beck, Jeffrey; Bejjanki, Vikranth R; Pouget, Alexandre

How to cite

BECK, Jeffrey, BEJJANKI, Vikranth R, POUGET, Alexandre. Insights from a simple expression for linear fisher information in a recurrently connected population of spiking neurons. In: Neural computation, 2011, vol. 23, n° 6, p. 1484–1502. doi: 10.1162/NECO_a_00125

This publication URL: https://archive-ouverte.unige.ch//unige:24940

Publication DOI: 10.1162/NECO_a_00125

© This document is protected by copyright. Please refer to copyright holder(s) for terms of use.

LETTER Communicated by Hiroyuki Nakahara

Insights from a Simple Expression for Linear Fisher Information in a Recurrently Connected Population of Spiking Neurons

Jeffrey Beck
jbeck@bcs.rochester.edu
Gatsby Computational Neuroscience Unit, University College London, London, WC1N 3AR, U.K.

Vikranth R. Bejjanki
vrao@bcs.rochester.edu
Alexandre Pouget
alex@bcs.rochester.edu
Department of Brain and Cognitive Science, University of Rochester, Rochester, NY 14627, U.S.A.

A simple expression for a lower bound of Fisher information is derived for a network of recurrently connected spiking neurons that have been driven to a noise-perturbed steady state. We call this lower bound *linear Fisher information*, as it corresponds to the Fisher information that can be recovered by a locally optimal linear estimator. Unlike recent similar calculations, the approach used here includes the effects of nonlinear gain functions and correlated input noise and yields a surprisingly simple and intuitive expression that offers substantial insight into the sources of information degradation across successive layers of a neural network. Here, this expression is used to (1) compute the optimal (i.e., information-maximizing) firing rate of a neuron, (2) demonstrate why sharpening tuning curves by either thresholding or the action of recurrent connectivity is generally a bad idea, (3) show how a single cortical expansion is sufficient to instantiate a redundant population code that can propagate across multiple cortical layers with minimal information loss, and (4) show that optimal recurrent connectivity strongly depends on the covariance structure of the inputs to the network.

1 Introduction

The brain encodes many variables, such as the color of objects and the direction of arm movements, through the concerted activity of populations of noisy spiking neurons, a type of code known as population codes. Understanding these population codes is a key step toward developing neural theories of computation, learning, and information transmission.

 © 2011 Massachusetts Institute of Technology

A natural measure for characterizing the information content of a code when dealing with continuous variables is Fisher information (Abbott & Dayan, 1999). Fisher information is inversely proportional to the smallest change in the encoded stimulus that can be discriminated from the neuronal responses. This measure can be used to explore how to optimize population codes, that is, how to wire neural circuits to maximize Fisher information.

In many population codes, the tuning curve of the neurons, that is, the average response as a function of a real-valued stimulus (denoted s), follows gaussian functions of s. Several studies have investigated how to optimize the parameters of these tuning curves, such as the height and width, when s is a scalar variable (Seung & Sompolinsky, 1993) as well as when $\mathbf{s}$ is a multidimensional vector (Zhang & Sejnowski, 1999). These studies have argued that for scalar s, the brain should use high-amplitude, narrow tuning curves to optimize information transmission and that learning should seek to reduce the width of the tuning curve as a way to improve behavioral performance (Somers, Nelson, & Sur, 1995; Spitzer, Desimone, & Moran, 1988; Murray & Wojciulik, 2004; Schoups, Vogels, Qian, & Orban, 2001; Teich & Qian, 2003).

This conclusion, however, was derived under the assumption that neurons generate independent Poisson spike counts. This is a problem because neurons in vivo are correlated (Zohary, Shadlen, & Newsome, 1994), and correlations can have a significant impact on Fisher information (Abbott & Dayan, 1999; Yoon & Sompolinsky, 1998; Sompolinsky, Yoon, Kang, & Shamir, 2001; Wilke & Eurich, 2002; Wu, Nakahara, & Amari, 2001). These researchers investigated the effects of correlations by considering a variety of physiologically inspired parameterizations of covariance matrices, but they did not consider how a network of spiking neurons might generate these covariance structures. To address this issue, we need an expression for Fisher information in a recurrently connected network of spiking neurons. For scalar variables, such an expression has been recently derived by Toyoizumi, Aihara, and Amari (2006), but only for a single layer of noisy neurons driven by a noiseless or deterministic function of the stimuli of interest. This is a serious limitation for two reasons. First, their approach cannot be applied to a situation in which there is a layered architecture and the quantity of interest is the information content of the final layer. This is because although the first layer of noisy neurons might be driven by a signal that is a deterministic function of the stimulus, the subsequent layers are necessarily driven by a noise-corrupted version of that signal. Second, stimulus-dependent, noiseless inputs convey an infinite amount of Fisher information (assuming invertible transformations), while Fisher information is necessarily finite in the nervous system. For instance, given the image of a contour, it is not possible to know its orientation with infinite precision if only because of noise in the physical world and the noise in the photoreceptors. Indeed, one could argue that the quantity of interest

here is information loss between two layers of cortex. When the input layer is a deterministic function of the stimulus, as it is in the case worked out by Toyoizumi et al. (2006), information loss is infinite. Thus, to model the effects of finite input information and information loss between layers of cortex, we require an expression for Fisher information in a network of noisy neurons that is driven by a noisy input layer.

Here, we expand on previous work that estimated the second-order statistics of spiking neural networks and derive a simple and intuitive expression for a lower bound on Fisher information in a network of spiking neurons (more specifically, linear, nonlinear, Poisson neurons; see below) that fire in response to noisy input spike trains with finite information content. This lower bound corresponds to what we call linear Fisher information, which is the fraction of Fisher information about a stimulus s that can be recovered by a locally optimal linear estimator (i.e., the linear operation on neural activity that can best discriminate between s and $s + \delta s$, where δs is small). In practice, linear Fisher information has been found to provide a tight bound on total Fisher information, in simulations (Seriès, Latham, & Pouget, 2004) and in vivo (Averbeck, Latham, & Pouget, 2006). Consequently, this expression provides a relevant and valuable tool for investigating fundamental questions regarding the computational properties of rate-based population codes.

2 Linear, Nonlinear, Poisson Neurons

The spike response model (SRM) or linear-nonlinear-Poisson model (LNP) of neural activity has become a popular model of spiking neural activity (Gerstner & Kistler, 2002), due in part to its computational simplicity, the ease with which it can be unambiguously fit to neural data (Paninski, 2004), and its ability to approximate more complicated integrate-and-fire neurons (Plesser & Gerstner, 2000). Here we consider an output layer of such LNP neurons with lateral connections, receiving spike trains from an input layer of spiking neurons. Each neuron in the input layer generates a spike train $x_i(t) = \sum_n \delta(t - t_i^n)$, according to a stationary stochastic process with stimulus-dependent mean, $\boldsymbol{\mu}_{\mathbf{x}}(s) = \langle \mathbf{x} \rangle_s$ and covariance $\boldsymbol{\Gamma}_{\mathbf{xx}}(s, t - t') = \langle (\mathbf{x}(t) - \boldsymbol{\mu}_{\mathbf{x}}(s))(\mathbf{x}(t') - \boldsymbol{\mu}_{\mathbf{x}}(s))^{\mathrm{T}} \rangle_s$. Here $\langle \cdot \rangle_s$ indicates an average conditioned on the value of the stimulus of interest, s.

The state of each output neuron, indexed by the letter i, is characterized by a membrane potential proxy $u_i(t)$ that is obtained by taking a weighted linear combination of the spikes from neurons in the input layer, $x_j(t)$, and spikes from the neurons in the output layer, $y_j(t)$:

$$u_i(t) = \sum_j w_{ij} \epsilon * y_j(t) + \sum_j m_{ij} \epsilon * x_j(t), \tag{2.1}$$

where $\epsilon(t)$ gives the time course of the postsynaptic potential associated with each spike, the $*$ indicates a convolution, and the matrices $\mathbf{M}$ and $\mathbf{W}$ specify the feedforward and recurrent connectivity, respectively.

Spikes are then generated from an inhomogeneous Poisson process with the rate given by $\rho_i(t|\hat{t}_i) = \gamma(t - \hat{t}_i)g(u_i(t))$. Here, the gain function, $g(u)$, is a monotonically increasing, nonnegative function; $\hat{t}_i$ is the time of the last spike of neuron i, so that $\gamma(t - \hat{t})$ models the refractoriness of the neuron (in this work, $\gamma(t)$ is either a constant or given by $\gamma = 1 - e^{-kt}$). We use the notation $y_i(t) = \sum_n \delta(t - t_i^n)$ to denote the spike train for output neuron i and $\mathbf{y}(t)$ to refer to the vector of spike trains from all output neurons.

3 Linear Fisher Information

We define linear Fisher information as $\mathbf{I}_\mathbf{y}(s) = \boldsymbol{\mu}'_\mathbf{y}(s) \cdot \boldsymbol{\Gamma}_{\mathbf{yy}}{}^{-1}(s)\boldsymbol{\mu}'_\mathbf{y}(s)$, where $\boldsymbol{\mu}_\mathbf{y}(s)$ and $\boldsymbol{\Gamma}_{\mathbf{yy}}(s)$ are the stimulus-dependent mean and covariance matrix of $\mathbf{y}(t)$, and the notation $(\cdot)$ is meant to indicate a dot or inner product. This corresponds to the part of Fisher information that can be inferred from the variance of the locally optimal linear estimator of the stimulus (Seriès et al., 2004) under the condition that the Cramér-Rao bound is attainable (Wu, Amari, & Nakahara, 2002).

In general, Fisher information contains other terms in addition to the linear term. For instance, when $p(\mathbf{y}|s)$ is a multivariate gaussian distribution, there is a second term, the so-called trace term, that reflects the information content that results from a stimulus-dependent covariance matrix under the gaussian assumption. In theory, this term can contain a large fraction of the information, particularly when the covariance matrix depends on the stimulus (Shamir & Sompolinsky, 2004). Nonetheless, we chose to focus on the linear term because it provides a tight bound on total Fisher information in both simulations (Seriès et al., 2004) and in vivo (Averbeck et al., 2006). Moreover, the trace term is applicable only when the stimulus-conditioned population response is, in fact, gaussian distributed. In general, this assumption of gaussianity may not hold and should be tested. Such a test requires knowledge of the third, fourth, and possibly higher moments. Unfortunately, a theory of correlations of neural networks is agnostic as to moments higher than the second. Thus, such a theory can be used only to estimate linear Fisher information.

This is easily observed by considering a computation of Fisher information for an arbitrary member of the exponential family with sufficient statistics $\mathbf{T}(\mathbf{y})$. In this case, $p(\mathbf{y}|s) = \phi(\mathbf{y}) \exp(\Theta(s) \cdot \mathbf{T}(\mathbf{y}) - \eta(s))$, so that Fisher information is given by

$$I_T(s) = \frac{d}{ds}\langle \mathbf{T} \rangle_s \cdot \boldsymbol{\Gamma}_{TT}^{-1}(s) \frac{d}{ds}\langle \mathbf{T} \rangle_s. \tag{3.1}$$

This equation indicates that explicit computation of Fisher information requires knowledge of the covariance of the sufficient statistic, $\mathbf{T}(\mathbf{y})$. When the sufficient statistic is linear in $\mathbf{y}$ (as is the case for a rate based neural code), this expression yields linear Fisher information, and the first- and second-order statistics (mean and covariance) of $\mathbf{y}$ are all that is needed to compute Fisher information. In contrast, the computation of Fisher information expressed by a neural code that conveys information through the presence or absence of coincident (or time-delayed coincident) spikes requires a sufficient statistic, $\mathbf{T}(\mathbf{y})$, which is influenced by these coincident spikes. An example of such a sufficient statistic would be a $\mathbf{T}(\mathbf{y})$ that spans the space of quadratic functions of $\mathbf{y}$. However, this computation requires an estimate of the covariance of these quadratic elements of $\mathbf{y}$, that is, an estimate of the third and fourth moments of $\mathbf{y}$. As such, a theory of correlations, but not of higher-order statistics, is capable of yielding only an estimate of information associated with a linear-sufficient statistic—$\mathbf{T}(\mathbf{y}) = \mathbf{y}$.

4 Analysis

To compute linear Fisher information, we need an expression for the covariance matrix in the output layer. The main difficulty in doing this comes from the nonlinear activation function $g(u)$. However, we can take advantage of the fact that in an LNP network, the strength of interneuronal connectivity, here characterized by $\mathbf{M}$ and $\mathbf{W}$, is inversely proportional to the number of neurons. Therefore, the variations in the membrane potential proxies, $u_i(t)$, are also inversely proportional to the number of neurons, which implies that they are small in large networks. We can therefore linearize the activation function around its steady state, that is, apply linear response theory (Risken & Frank, 1996) to approximate a linear transfer function between $u_i(t)$ and the associated spike train $y_i(t)$. This is a variation of the approach of Ginzburg and Sompolinsky (1994) and, more recently, Chacron, Longtin, and Maler (2005). Specifically, we seek the function $\chi_i(t)$ such that

$$y_i(t) \approx y_i^0(t) + \chi_i * \delta u_i(t), \tag{4.1}$$

where $\delta u_i(t) = u_i(t) - \bar{u}_i(s)$ and $y_i^0(t)$ is the spike train obtained by driving the neurons only with the stimulus-conditioned mean input: $\langle g(u)|s\rangle$. Here the bracket notation, $\langle g(u)|s\rangle$, is intended to indicate a stimulus-conditioned noise average of $g(u)$. In principle, both of these quantities may depend on time, but for simplicity, we assume stationary statistics. As a simple example, consider an inhomogeneous Poisson process (with no refractory period—$\gamma(t - \hat{t}) = 1$) driven by a stochastically fluctuating membrane potential proxy u with mean $\bar{u}$ and variance σ_u^2 and with a linear gain function g. In this case, the mean drive and, hence, the rate of the unperturbed

process $y^0(t)$ is just the average $\bar{g}(\bar{u}, \sigma_u^2) = \langle g(u) \rangle = \bar{u}$. Similarly, it is easy to show that the output spike train, $y(t)$, has variance $\bar{u} + \sigma_u^2$. From this we can conclude that the linear transfer function χ takes on a constant value of 1. Although this example was limited to the consideration of spike counts, a similar consideration of stationary temporal correlations yields the same result in the linear case.

So far, we have assumed that the fluctuations in u are small, so that $g(u)$ can be approximated by a linear function (Ginzburg & Sompolinsky, 1994). Since large fluctuations in the membrane potential proxy have the effect of smoothing the gain function, we can extend this approach to large fluctuations in u by approximating $g(u)$ with $\bar{g}(\bar{u}, \sigma_u^2) = \langle g(u) \rangle$ and Taylor-expanding this function about the mean drive instead. This has the advantage of allowing neurons that are not driven above threshold by their mean drive but respond due to the stochastic fluctuations to contribute to the network activity and thus also to information content. In this case, an inhomogeneous Poisson spiking neuron would produce an unperturbed spike train, $y^0(t)$, with mean firing rate given by $\bar{g}(\bar{u}, \sigma_u^2)$, and χ would again be constant but with value given by

$$\bar{g}'(\bar{u}, \sigma_u^2) = \frac{d}{dx} \bar{g}'(\bar{u} + x, \sigma_u^2). \tag{4.2}$$

When the refractory term is present, the procedure for estimating χ is essentially the same. Indeed, because we seek a linear transfer function, it is helpful to consider the χ estimation in the Fourier domain, henceforth indicated by a $\tilde{\chi}(\omega)$. In this domain, the estimation of $\tilde{\chi}(\omega)$ is nothing more than the estimation of the frequency response function of a neuron driven by some noise-perturbed membrane potential proxy. This procedure can be performed, numerically if necessary, for any nonlinearity in the gain function or any refractory term (Chacron et al., 2005; Gerstner & Kistler, 2002). The procedure is straightforward: one can simply drive a neuron with some mean input $\bar{u}$ perturbed by white noise with variance σ_u^2 to obtain the statistics of $y^0(t)$. The frequency response function $\tilde{\chi}(\omega)$ can then be obtained by adding to the drive a small frequency-dependent component with frequency ω.

However, in order to make progress analytically, we at first neglect the refractory term and later show numerically that for stationary input statistics, the addition of a nontrivial refractory period seems to have no discernable effect on the estimation of information. Regardless, the procedure we outlined can be used to obtain the linear transfer function of a single neuron (i.e., the χ of equation 4.1) as a function of the mean and variance of the drive of the neuron. This single-neuron linear transfer function can then be used to compute the linear transfer function of an entire network. This is the equivalent of computing matrices $\tilde{\mathbf{A}}(\omega, s)$ and $\tilde{\mathbf{B}}(\omega, s)$ in the

equation

$$\tilde{\mathbf{A}}(\omega, s)(\tilde{\mathbf{y}}(\omega, s) - \boldsymbol{\mu}_{\mathbf{y}}(s)\delta(\omega)) = \tilde{\mathbf{y}}^0(\omega, s) - \boldsymbol{\mu}_{\mathbf{y}}(s)\delta(\omega) + \tilde{\mathbf{B}}(\omega, s)(\tilde{\mathbf{x}}(\omega, s) - \boldsymbol{\mu}_{\mathbf{x}}(s)\delta(\omega)). \tag{4.3}$$

Note that these are perturbation equations expressed in the Fourier domain where terms of the form $\boldsymbol{\mu}_{\mathbf{y}}(s)\delta(\omega)$ and $\boldsymbol{\mu}_{\mathbf{x}}(s)\delta(\omega)$ arise from the removal of the stationary means from $\mathbf{x}(t)$, $\mathbf{y}(t)$, and $\mathbf{y}^0(t)$. Note also that this expression neatly encapsulates the two sources of variability that drive the fluctuations in $\mathbf{y}(t)$, namely, fluctuations that arise from the inputs to the network and fluctuations that arise from the stochastic spiking represented by $\mathbf{y}^0(t)$. In order to compute the second-order statistics, all that remains now is to identify $\tilde{\mathbf{A}}(\omega, s)$ and $\tilde{\mathbf{B}}(\omega, s)$ by comparing the perturbation equation above with the Fourier transform of the linearized form of equation 2.1:

$$\begin{aligned}\tilde{y}_i(\omega) - \boldsymbol{\mu}_{\mathbf{y},i}(s)\delta(\omega) = \tilde{y}_i^0(\omega) - \boldsymbol{\mu}_{\mathbf{y},i}(s)\delta(\omega) \\ + \tilde{\chi}_i(\omega, s)\sum_j W_{ij}\tilde{\epsilon}(\omega)\left(\tilde{y}_j(\omega) - \boldsymbol{\mu}_{\mathbf{y},j}(s)\delta(\omega)\right) \\ + \tilde{\chi}_i(\omega, s)\sum_j M_{ij}\tilde{\epsilon}(\omega)\left(\tilde{x}_j(\omega) - \boldsymbol{\mu}_{\mathbf{x},j}(s)\delta(\omega)\right).\end{aligned} \tag{4.4}$$

This leads to a network transform function of the form

$$\tilde{\mathbf{A}}(\omega, s) = \mathbf{I} - \tilde{\mathbf{D}}(\omega, s)\mathbf{W}, \tag{4.5}$$

$$\tilde{\mathbf{B}}(\omega, s) = \tilde{\mathbf{D}}(\omega, s)\mathbf{M}, \tag{4.6}$$

where diagonal matrix $\tilde{\mathbf{D}}$ is defined by

$$\tilde{D}_{ii}(\omega, s) = \tilde{\chi}_i(\omega, s)\tilde{\epsilon}(\omega). \tag{4.7}$$

We note that in the absence of a refractory term, $\tilde{D}_{ii}$ becomes

$$\tilde{D}_{ii}(\omega, s) = \bar{g}'(\bar{u}_i(s), \sigma^2_{u_i}(s))\tilde{\epsilon}(\omega), \tag{4.8}$$

where the prime indicates a derivative with respect to the first argument of $g(\bar{u}_i(s), \sigma^2_{u_i}(s))$. The covariance of the output spike trains can then be related to the covariance of the input spike trains, $\tilde{\boldsymbol{\Gamma}}_{\mathbf{xx}}(\omega, s)$; the covariance of the unperturbed spikes; and the network transfer function, $\tilde{\mathbf{A}}(\omega, s)$. Finally, since equation 4.3 linearly relates the perturbed output spikes to noisy inputs, $\mathbf{x}$, and the noisy unperturbed spike trains, $\mathbf{y}_0$, it is a simple matter to relate the variability in output spikes to the variability in these two

quantities. Straightforward matrix manipulation of equation 4.3 yields

$$\tilde{\mathbf{\Gamma}}_{\mathbf{yy}}(\omega, s) = \tilde{\mathbf{A}}^{-1}(\omega, s)(\tilde{\mathbf{G}}(\omega, s) + \tilde{\mathbf{D}}(\omega, s)\mathbf{M}\tilde{\mathbf{\Gamma}}_{\mathbf{xx}}(\omega, s)\mathbf{M}^{\mathrm{T}}\tilde{\mathbf{D}}^{\mathrm{T}}(\omega, s))\tilde{\mathbf{A}}^{-\mathrm{T}}(\omega, s), \tag{4.9}$$

where $\tilde{G}_{ii}(\omega, s)$ represents the variability of the Poisson spiking of the recurrently connected neurons arising from the first two terms to the right of the equality in equation 4.3. As such, $\tilde{G}_{ii}(\omega, s)$ is given by the Fourier transform of the autocorrelation function of the unperturbed spike train emanating from output neuron i. In the absence of a refractory term, this is given by

$$\tilde{G}_{ii}(\omega, s) = \bar{g}(\bar{u}_i(s), \sigma^2_{u_i}(s)). \tag{4.10}$$

Similarly, since the input is stationary, the mean activity of the output population is simply $\boldsymbol{\mu}_{\mathbf{y}}(s) = \mathbf{g}(\mathbf{W}\boldsymbol{\mu}_{\mathbf{y}}(s) + \mathbf{M}\boldsymbol{\mu}_{\mathbf{x}}(s), \sigma^2_{\mathbf{u}}(s))$. Since the mean is independent of time, we need to consider only the stationary ($\omega = 0$) mode of the covariance matrix, $\tilde{\mathbf{\Gamma}}_{\mathbf{yy}}(\omega, s)$, to obtain linear Fisher information. Finally, since the derivative of the mean often depends only weakly on the variance of the membrane potential proxies, $\sigma^2_{\mathbf{u}}(s)$, we can neglect this dependence and obtain a very simple expression for the derivative of the mean of the output spikes with respect to the stimulus:

$$\boldsymbol{\mu}'_{\mathbf{y}}(s) = \mathbf{D}\mathbf{W}\boldsymbol{\mu}'_{\mathbf{y}}(s) + \mathbf{D}\mathbf{M}\boldsymbol{\mu}'_{\mathbf{x}}(s). \tag{4.11}$$

This leads to a simple expression for linear Fisher information:

$$\mathbf{I}_{\mathbf{y}}(s) = (\mathbf{M}\boldsymbol{\mu}'_{\mathbf{x}}(s))^{\mathrm{T}}(\mathbf{M}\tilde{\mathbf{\Gamma}}_{\mathbf{xx}}(0, s)\mathbf{M}^{\mathrm{T}} + \tilde{\mathbf{D}}^{-1}(0, s)\tilde{\mathbf{G}}(0, s)\tilde{\mathbf{D}}^{-1}(0, s))^{-1}\mathbf{M}\boldsymbol{\mu}'_{\mathbf{x}}(s). \tag{4.12}$$

If the feedforward connectivity matrix, $\mathbf{M}$, is invertible and we set the term $\tilde{\mathbf{D}}^{-1}(0, s)\tilde{\mathbf{G}}(0, s)\tilde{\mathbf{D}}^{-1}(0, s)$ to zero, equation 4.12 reduces to the linear Fisher information in the input layer $\boldsymbol{\mu}'_{\mathbf{x}}(s)^{\mathrm{T}}\left(\tilde{\mathbf{\Gamma}}_{\mathbf{xx}}(0, s)\right)^{-1}\boldsymbol{\mu}'_{\mathbf{x}}(s)$. Therefore, these two terms, $\mathbf{M}$ and $\tilde{\mathbf{D}}^{-1}(0, s)\tilde{\mathbf{G}}(0, s)\tilde{\mathbf{D}}^{-1}(0, s)$, control the amount of information lost between the input and output layers. The second term can be interpreted as noise with variance, $\tilde{\mathbf{D}}^{-1}(0, s)\tilde{\mathbf{G}}(0, s)\tilde{\mathbf{D}}^{-1}(0, s)$, that is added to the feedforward afferences or inputs of the network: ($\mathbf{Mx}$).

At first sight, it would appear that the recurrent connectivity, $\mathbf{W}$, has no impact on information loss since it does not appear in equation 4.12. However, recurrent connectivity does affect information loss implicitly by modulating the shape of the steady-state tuning curve. This in turn affects noise added to the feedforward afferences by modulating the matrices $\tilde{\mathbf{D}}(0, s)$ and $\tilde{\mathbf{G}}(0, s)$, both of which store quantities evaluated at the steady-state mean activity.

Another important point to emphasize is that equation 4.12 is asymptotically valid for any pattern of feedforward and recurrent connectivity that scales in $O(1/N)$, that is, when the variance of the membrane potential proxy is small. Moreover, this expression can be used regardless of the function that is being computed between the input and output layers. For instance, with a proper choice of connectivity (i.e., a proper choice of $\mathbf{M}$ and $\mathbf{W}$), it is easy to build a network in which the input layer contains neurons with gaussian tuning curves to s and in which the output layer contains optimal tuning curves for some other variable of interest, $z = h(s)$, where $h(s)$ is a nonlinear function of s. The Fisher information about z in the output layer can still be obtained from an equation of the form of equation 4.12 but with the prime now indicating z derivatives.

Although equation 4.12 was derived for a scalar variable s, it is easy to extend this result to the case in which $\mathbf{s}$ is a vector. The ijth entry of the linear Fisher information matrix is now given by

$$\{\mathbf{I_y}\}_{ij}(\mathbf{s}) = \left(\mathbf{M}\frac{d}{ds_i}\boldsymbol{\mu}_\mathbf{x}(\mathbf{s})\right)^{\mathrm{T}} (\mathbf{M}\tilde{\mathbf{\Gamma}}_{\mathbf{xx}}(0,\mathbf{s})\mathbf{M}^{\mathrm{T}} + \tilde{\mathbf{D}}^{-1}(0,\mathbf{s})\tilde{\mathbf{G}}(0,\mathbf{s})\tilde{\mathbf{D}}^{-1}(0,\mathbf{s}))^{-1}\mathbf{M}\frac{d}{ds_j}\boldsymbol{\mu}_\mathbf{x}(\mathbf{s}). \tag{4.13}$$

As a result, this expression can be used to explore how tuning curve parameters like width influence information content regardless of the dimensionality of s. This issue has been studied in the past but only for independent noise (Zhang & Sejnowski, 1999; Brown & Backer, 2006).

We had to make several approximations to obtain the expression in equation 4.12. Specifically, we assumed stationary statistics on input and output populations, linearized about the noise-perturbed gain function, and we ignored refractory effects. To check whether these approximations induce significant errors, we simulated two-layer networks consisting of between 100 and 2000 LNP neurons organized in an orientation hypercolumn (see Seriès et al., 2004, for details about the connectivity). Input patterns of activity, $\mathbf{x}(t)$, were sampled from a Poisson distribution with gaussian-shaped tuning curves of varying amplitude (A^{in}) and width (K^{in}):

$$f_i(s) = A^{\mathrm{in}}\mathrm{e}^{(K^{\mathrm{in}}(cos(s-s_i)-1))}. \tag{4.14}$$

Correlated noise was also added to the rates of these Poisson processes to create correlations in the input spikes. Specifically, we drew an additional random variable z from a zero mean gaussian distribution with covariance matrix given by

$$\mathrm{COV}(z_i, z_j)(s) = C^{\mathrm{in}}\sqrt{f_i(s)f_j(s)}. \tag{4.15}$$

and then sampled input spikes according to $\mathbf{x}(t) = Poi\,(f_i(s) + z_i)$.

For networks with N neurons in the output layer, feedforward connectivity patterns took the form

$$M_{ij} = \frac{A^{\text{ff}}}{N} + \frac{B^{\text{ff}}}{N} \mathrm{e}^{\left(K^{\text{ff}}(\cos(s_i - s_j) - 1)\right)}, \tag{4.16}$$

while recurrent weight patterns were parameterized by

$$W_{ij} = \frac{A^{\text{rec}}}{N} + \frac{B^{\text{rec}}_{exc}}{N} \mathrm{e}^{\left(K_{exc}(\cos(s_i - s_j) - 1)\right)} - \frac{B^{\text{rec}}_{inh}}{N} \mathrm{e}^{\left(K_{inh}(\cos(s_i - s_j) - 1)\right)}. \tag{4.17}$$

Here, s_i indicates the preferred orientation of neuron i. The threshold and slope of a rectified linear gain function were also manipulated, thereby giving us the freedom to implement networks that performed recurrent sharpening, recurrent and feedforward amplification, recurrent sharpening, and sharpening via thresholding. The gain function was parameterized by

$$g(u) = \alpha \log\left(1 + \mathrm{e}^{\frac{u-\theta}{\alpha}}\right), \tag{4.18}$$

allowing us to interpolate between threshold linear and exponential gain functions. Neurons with fast and slow time constants, created by varying the time course of $\epsilon(t)$, were also explored. Further, the effects of strongly coupled neurons were investigated by creating networks in which all connection strengths were identical and order one, but in which the probability of a connection between two neurons was given by functions with the same parametric form as $\mathbf{M}$ and $\mathbf{W}$, as in Seriès et al. (2004).

We then computed the percentage of information preserved in the output layer (obtained from the ratio of linear Fisher information in the output layer to the same quantity in the input layer). This was done by computing the variance of the unbiased locally optimal linear estimator applied to output spikes (Seriès et al., 2004). This empirically observed quantity was then compared to the prediction obtained from equation 4.12. Figure 1 confirms that this expression does indeed provide a very tight bound on linear Fisher information over a wide range of network parameters and activation functions. Curiously, though neglected in the derivation, this expression seems to hold even when refractory effects are included provided $\tilde{D}_{ii}(0, s)$ and $\tilde{G}_{ii}(0, s)$ are simply modulated by the expected value of $\gamma(t - \hat{t})$, that is, $\tilde{D}_{ii}(0, s) \to \bar{\gamma}\tilde{D}_{ii}(0, s)$ and $\tilde{G}_{ii}(0, s) \to \bar{\gamma}\tilde{G}_{ii}(0, s)$. This may be due to the stationary statistics of the input process $\mathbf{x}(t)$, but further investigation is needed to explain this coincidence. We also tested whether there is a significant fraction of information beyond the linear term by estimating information with a nonlinear decoder, namely, a support vector machine with radial basis function kernels (SVM-RBF). We found that these discrete classification algorithms extract less than 3.4% more information

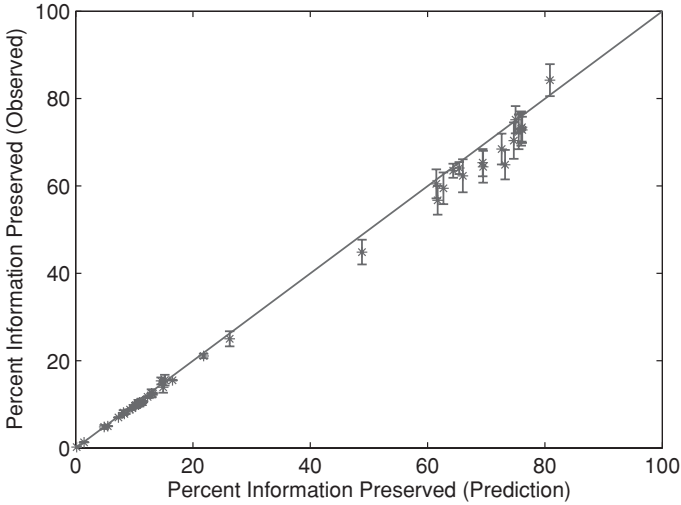


Figure 1: Empirically estimated (Observed) versus predicted percentage of preserved linear Fisher information. The predictions were computed with equation 4.12, while the empirically estimated, Fisher information was computed by estimating the variance of the unbiased locally optimal linear estimator applied to network simulations. Error bars indicate information estimated from training and test data sets using early stopping. Both empirically estimated and predicted Fisher information were normalized by the rate of accumulation of Fisher information in the input population. Thus, values in this figure represent the percentage of the information input into network that is recoverable in the output population. Input populations and feedforward and recurrent connectivity profiles were varied across a wide range of parameter values and network functions, as were input tuning curves and covariance structure. Specifically, for the input population $A^{\text{in}} \in (10\text{ Hz}, 200\text{ Hz})$, $K^{\text{in}} \in (0.5, 4)$, and $C^{\text{in}} \in (0, 0.5)$. For feedforward connection, $A^{\text{ff}} \in (0, 1)$, $B^{\text{ff}} \in (0.5, 5)$, and $K^{\text{ff}} \in (0.5, 5)$. Parameters for the recurrent connectivity were $A^{\text{rec}} \in (-0.6, 0.4)$, $B^{\text{rec}}_{exc} \in (0, 10)$, $B^{\text{rec}}_{inh} \in (0, 10)$, $K^{\text{rec}}_{exc} \in (2, 4)$, and $K^{\text{rec}}_{exc} \in (0, 2)$. Spike counts were obtained from 0.5 second runs, and the time constant of ϵ was between 5 and 10 ms. Each network was driven by a broadly tuned population of independent Poisson spiking neurons. The threshold linear gain function was parameterized by $g(u) = \alpha \log(1 + \mathrm{e}^{\frac{u-\theta}{\alpha}})$, with $\alpha \in (0.1, 10)$ and $\theta \in (-5, 2)$. The refractory function was parameterized by $\gamma(t) = 1 - \exp^{-t/\tau_r}$ where τ_r was between 0 and 10 ms. All parameter values were sampled uniformly from the indicated intervals.

than the local optimal linear estimator does. The comparison between these discrete classification algorithms and Fisher information was performed by mapping the percentage of correct classification onto the equivalent value of d-prime, which is related to the square root of Fisher information (Dayan & Abbott, 2001).

Next, we describe a few applications of the expression for Fisher information derived above.

4.1 Optimal Firing Rate and Recurrent Sharpening. We saw that the information loss in equation 4.12 is controlled by two terms. The second term, $\tilde{G}_{ii}/\tilde{D}_{ii}^2$, is the ratio of the mean firing rate of a given output neuron to the square of the sensitivity of the neuron as described by its linear transfer function. In the absence of a refractory term, this takes the form $\bar{g}(\bar{u}_i, \sigma_i^2)/\bar{g}'(\bar{u}_i, \sigma_i^2)^2$. For an exponential gain function, $\bar{g}'(\bar{u}_i, \sigma_i^2)^2 \propto \bar{g}(\bar{u}_i, \sigma_i^2)^2$, in which case $\tilde{G}_{ii}/\tilde{D}_{ii}^2 \propto 1/\bar{g}(\bar{u}_i, \sigma_i^2)$. From the perspective of a single neuron, this implies that a higher firing rate always preserves more information, with information loss becoming exponentially large as output firing rates go to zero. The opposite result holds for a rectified linear gain function. Indeed, in this case, $\bar{g}'(\bar{u}_i, \sigma_i^2)^2$ is equal to a constant and $\tilde{G}_{ii}/\tilde{D}_{ii}^2 \propto \bar{g}(\bar{u}_i, \sigma_i^2)$. Therefore, somewhat counterintuitively, with a linear activation function, the more a given neuron fires, the less information it transmits.

However, if one uses a linear gain function with LNP neurons, the effective noise-perturbed gain function, $\bar{g}(\bar{u}, \sigma_u^2)$, is exponential for a weak drive and linear for a strong drive (Gerstner & Kistler, 2002; see Figure 2a). In this case, there is a firing rate that minimizes the effective noise added ($\tilde{G}_{ii}/\tilde{D}_{ii}^2$) because the effective noise added to the input reaches a minimum value. A neuron that fires at the rate corresponding to this minimum can be said to be firing at its optimal firing rate. For the particular gain function that was used to create Figure 2b, this optimal firing rate is around 10 Hz. More generally, we can conclude that optimal (information-maximizing) firing rates occur where the gain function has positive curvature and satisfies $\sqrt{\bar{g}(u)}'' = 0$.

This single neuron result can also be used at the network level to show why severe sharpening of input tuning curves may be a particularly bad idea. Consider, for example, the simple case depicted in Figure 2. Here the feedforward afferences, $\mathbf{M}\mathbf{x}(t)$, are assumed to be independent and Poisson with broad tuning curves (the dashed-dot lines in Figures 2c and 2d). Since the afferences are independent, the contribution of each afference to Fisher information in the inputs is given by the ratio of the square of the derivative (with respect to the stimulus) of the mean drive to the mean drive of the afference. These contributions are represented by the (dash-dot in Figures 2e and 2f). Note that as usual, the most informative inputs are those that correspond to the largest slope of the input tuning curves. Now consider two networks: one that sharpens the tuning curves in the output layer (dashed lines in Figure 2c) and one that does not (lines in Figure 2d). The solid line in Figures 2e and 2f shows the effective noise added to each input afference by the Poisson step in the output layer of both networks (corresponding to the term $\bar{g}(\bar{u}_i, \sigma_u^2)/\bar{g}'(\bar{u}_i, \sigma_u^2)^2$). This effective noise determines the fraction

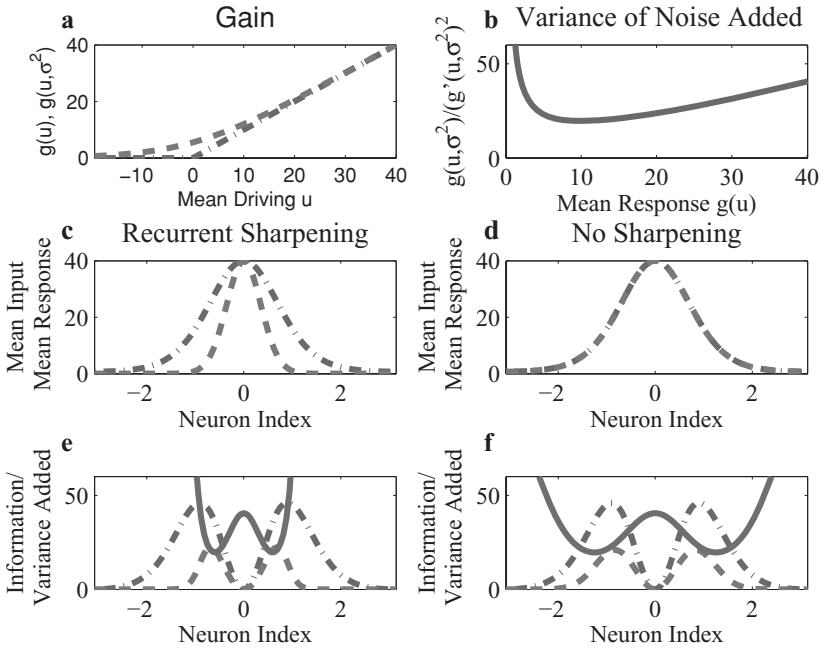


Figure 2: (a) The dashed-dot line shows a threshold linear gain function $g(u)$. When used in a network of LNP neurons, this linear function is smoothed by the noise in the membrane potential proxy, resulting in the effective noise-perturbed gain function shown with the dashed line. (b) Variance of the noise that is effectively added to the input afferences of a neuron as a function of its firing rate, when the noise perturbed gain function in (a dashed line) is used. This gain function is close to an exponential function below 10 Hz. Above 10 Hz, the gain function is approximately linear. (c, d) The dashed-dot lines indicate the mean drive ($\mathbf{M}\boldsymbol{\mu}_{\mathbf{x}}(s)$) of the network when a horizontal ($s = 0$) bar is presented, while the dashed lines indicate the mean response of the output population, $\boldsymbol{\mu}_{\mathbf{y}}(s)$. Output neurons are indexed by their preferred orientation. On the left, c is a network that sharpens its input activity, while on the right, d is a network that does not sharpen. (e, f) The dashed-dot lines indicate the contribution to the Fisher information in the input population that goes into a particular neuron in the output layer, while the dashed line indicates the portion of that information present in the output population. The solid line gives the amount of noise effectively added to inputs to each neuron in the network by the Poisson spiking nonlinearity.

of input information (dashed-dot curve in Figures 2e and 2f) that will be conveyed in the output spike trains (dashed curve in Figures 2e and 2f). The effective noise is minimal for neurons, with an output firing rate close to the optimal value of 10 Hz as shown in Figure 2b.

To optimize information transmission for the entire population of neurons, effective noise added should be the smallest for neurons receiving the most informative inputs. In the no-sharpening network, this is indeed the case. The effective noise is small (solid curve in Figure 2f) when the input information is high (dashed-dot curve in Figure 2f). For the sharpening network, this is no longer the case. A large amount of noise is added to neurons that receive highly informative inputs, resulting in large information loss. This is due to the fact that as firing rates go toward zero, the effective noise scales like one over the firing rate. Indeed, by computing the ratio of the information in the input population to that in the output population, we found that the sharpening network transmits only 27% of the information it receives compared to 49% for the nonsharpening network.

Note also that for a given gain function $g(u)$, this result holds regardless of the specific mechanism by which the sharpening occurs and regardless of the specific spatiotemporal covariance structure induced in the output layer. Note, however, that we are not saying that sharpening is always inefficient. As we will see next, a small amount of sharpening can in fact be helpful; it is severe sharpening that generally destroys information.

4.2 Cortical Expansion, Redundant Codes, and Balanced Excitation and Inhibition. Adding more neurons to a given layer is a well-known way to decrease information loss, but this expression allows us to quantify precisely the impact of the number of neurons on information loss. For example, suppose that each layer is divided into subpopulations of neurons with identical tuning curves and gain functions and that each subpopulation has K neurons in the input layer and N neurons in the output layer. In this case, averaging over the identically tuned neurons results in an effective noise-added term that scales like $\frac{K}{N}$. This indicates that increasing the number of neurons in the output layer, N, while keeping the number of input neurons, K, fixed has the effect of decreasing information loss by an amount proportional to $\frac{K}{N}$. Equation 4.12 also indicates that even when $K = N$, near-perfect information preservation can be achieved as long as the code in the input layer is redundant (i.e., the information is small compared to the number of neurons) and each of the output neurons fires sufficiently close to its optimal firing rate. This is because neurons firing at their optimal rate effectively place a bound on the added-noise term, $\tilde{\mathbf{D}}^{-1}(0, s)\tilde{\mathbf{G}}(0, s)\tilde{\mathbf{D}}^{-1}(0, s)$. When linear Fisher information is small compared to the number of stimulus-tuned neurons in a large network, the eigenvalues of the associated covariance matrix must be large. As a result, the eigenvalues of $\mathbf{M}\tilde{\mathbf{\Gamma}}_{\mathbf{xx}}(0, s)\mathbf{M}^{\mathrm{T}}$ must also be large. When they are sufficiently large compared to the eigenvalues of the matrix that describes the effective noise added, $\tilde{\mathbf{D}}^{-1}(0, s)\tilde{\mathbf{G}}(0, s)\tilde{\mathbf{D}}^{-1}(0, s)$, then this term can be neglected and the linear Fisher information in the output layer will be very close to the linear Fisher information in the input afferences.

This is very convenient as it implies that a single layer of cortical expansion (i.e., a large increase in the number of neurons in the primary sensory cortical areas) is sufficient to instantiate a redundant code, which can then be propagated with small information loss across multiple layers, each of which has the same number of noisy neurons. To see why, consider a single cortical expansion for which the first layer consists of M independent Poisson neurons tuned to s. As a result, the information in the network scales like M. Suppose these neurons project to an output layer that has many more neurons: $N \gg M$. As previously indicated, reasonable constraints on the activity of the output neurons is sufficient to ensure that linear Fisher information is nearly perfectly preserved. But we have actually accomplished more than simple information preservation. We have also instantiated a redundant code in the output layer. This is because the amount of information contained in the network is small (order M) compared to the information capacity of the network, which is order N. This means that the eigenvalues of the covariance matrix of the output layer, $\tilde{\mathbf{\Gamma}}_{\mathbf{yy}}(0, s)$, must be large. Thus, if these output neurons are then used to drive another layer of N, similarly tuned neurons, the covariance of the input afferences to this third layer will satisfy the large eigenvalue condition necessary to ensure that information, once again, is nearly perfectly preserved.

This result indicates that greater information preservation can be accomplished by simply increasing the magnitude of the feedforward connection strengths, $\mathbf{M}$ (and thus increasing the magnitude of the eigenvalues of $\mathbf{M}\tilde{\mathbf{\Gamma}}_{\mathbf{xx}}(0, s)\mathbf{M}^{\mathrm{T}}$), while manipulating recurrent connections to keep the neurons driven by the most informative inputs near the optimal rates. Since connections between cortical areas are excitatory, local recurrent inhibition would be needed to accomplish this. This, then, provides a simple explanation for the information benefit of recurrent networks that balance large excitatory inputs with local recurrent inhibition, a widely observed property of cortical circuits (Marino et al., 2005).

4.3 Optimal Connectivity and Correlations. Finally, the framework described here may be used to demonstrate that optimal connectivity and tuning curve shape depend strongly on the correlations in the input layer. This is illustrated in Figure 3, which shows optimal recurrent connectivity and tuning curve shape in an orientation hypercolumn for two cases: one in which the input population consists of independent neurons and one in which the input neurons are locally positively correlated. In both cases, optimal connectivity was computed by gradient ascent applied to the recurrent weight matrix to maximize Fisher information in response to noisy images of oriented gratings across multiple contrast and image noise levels. Feedforward connectivity was held fixed, as were the parameters of the gain function (once again, partial derivatives with respect to σ_{il}^2 were small and so were ignored). Here, recurrent connections act to ensure that the most informative inputs drive neurons that fire close to the optimal

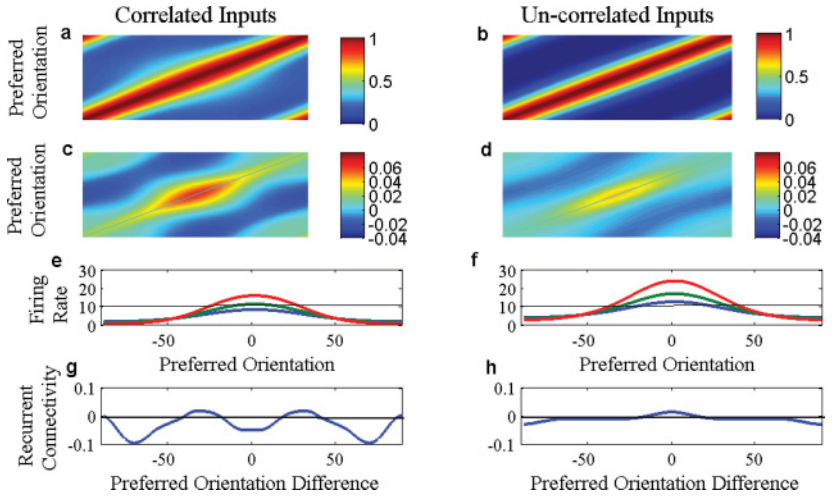


Figure 3: A comparison of the optimal tuning curves and recurrent connectivity in the presence of significant correlations in the input population (left column) and no correlations in the input population (right column). (a, b) The correlation structure of the input afferences $\mathbf{M}\tilde{\mathbf{\Gamma}}_{xx}(0, s)\mathbf{M}^{T}$, where feedforward connection strengths, $\mathbf{M}$, were chosen to have a narrow gaussian profile. Note that we are not plotting $\tilde{\mathbf{\Gamma}}_{xx}(0, s)$, which is why the matrix depicted in b is not diagonal despite the fact that the firing rates in the intput layer are independent. (c, d) The correlation structure of optimal output populations at a moderate value of the contrast with the diagonal removed for plotting purposes. (e, f) Population patterns of activity in the output layer in response to an orientation of 0 degree for low (blue line), medium (green), and high values (red) of contrast. (g, h) Optimal recurrent connectivity profiles. On the right side (uncorrelated inputs), the most informative neurons are the ones close to the inflection point of the population activity. The recurrent connectivity favors local excitation to bring these most informative neurons near the optimal firing rate, which in this network is about 10 Hz. On the left side (correlated inputs), the most informative neurons are now closer to the peak of the population activity. In this case, the recurrent excitation is reduced to keep neurons near the peak around the optimal firing rate of 10 Hz.

rate, that is, the rate that the effective noise added ($\tilde{G}_{ii}/\tilde{D}_{ii}^2$), is minimized for this particular choice of nonlinear gain function. In the independent case, this means that output neurons that lie near the inflection point of the tuning curve fire near the optimal rate of 10 Hz, while the relatively uninformative inputs at the peak of the tuning curve fire at a much higher rate. In the positively correlated case, neurons near the peak of the tuning curve are now relatively more informative than in the independent case, while neurons near the tail are now relatively less informative. This can be

observed by noting the change in the spectra of the covariance matrix of the input afferences. As a result, optimization now penalizes local excitation and has the effect of decreasing the amplitude (and, to a lesser extent, the sharpness) of the tuning curves so that the neurons close to the peak fire near the optimal rate, while the now less informative neurons in the tail have been driven below the optimal rate.

5 Discussion

We have derived a simple expression for linear Fisher information in a network of LNP neurons with arbitrary connectivity. This expression can be used to explore the efficiency of information transmission in networks of spiking neurons computing nonlinear functions, thereby representing an important step toward elucidating the neural basis of processes such as attention and perceptual learning, which allow the nervous system to access more information regarding behaviorally relevant sensory stimuli.

This analysis is limited to linear Fisher information: the fraction of Fisher information that is recoverable by a locally optimal linear estimator. Whether this is a severe limitation remains to be seen. We have found that empirically, it is exceedingly difficult to find any information beyond the linear term in networks of spiking neurons. Moreover, the amount of data required to estimate the nonlinear contributions to Fisher information is typically prohibitively large, because one needs to estimate the third-order and higher-order statistics of spike trains. Similar issues arise with spike timing codes, which convey information through the presence (or absence) of coincident or time-delayed coincident spikes. Such a code is present when the sufficient statistic, $\mathbf{T}(\mathbf{y})$, is influenced by these coincident spikes, as is the case when $\mathbf{T}(\mathbf{y})$ spans the space of quadratic functions of $\mathbf{y}$. Since estimation of Fisher Information requires an estimate of the covariance of the sufficient statistic, the analysis of such a code would require estimates of the third and fourth moments of $\mathbf{y}$.

Finally, we have also assumed that the stimulus is constant over time and that the network has reached a noise-perturbed steady state. While this is sufficient to model a wide variety of behavioral experiments, there is no question that the extension of this work to time-varying stimuli would be of use. However, it is not yet clear that linearizing the Poisson spiking nonlinearity around a nontrivial dynamic state will yield an approximation comparable to that observed in the stationary case.

References

Abbott, L. F., & Dayan, P. (1999). The effect of correlated variability on the accuracy of a population code. *Neural Comput.*, *11*(1), 91–101.

Averbeck, B. B., Latham, P. E., & Pouget, A. (2006). Neural correlations, population coding and computation. *Nature Reviews Neuroscience*, *7*(5), 358–366.

Brown, W. M., & Backer, A. (2006). Optimal neuronal tuning for finite stimulus spaces. *Neural Comput.*, *18*, 1511–1526.

Chacron, M. J., Longtin, A., & Maler, L. (2005). Delayed excitatory and inhibitory feedback shape neural information transmission. *Phys. Rev. E*, *72*(5 pt. 1), 051917.

Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: Computational and mathematical modeling of neural systems*. Cambridge, MA: MIT Press.

Gerstner, W., & Kistler, W. M. (2002). *Spiking neuron models*. Cambridge: Cambridge University Press.

Ginzburg, I., & Sompolinsky, H. (1994). Theory of correlations in stochastic neural networks. *Physical Review E*, *50*, 3171–3191.

Marino, J., Schummers, J., Lyon, D. C., Schwabe, L., Beck, O., Wiesing, P., et al. (2005). Invariant computations in local cortical networks with balanced excitation and inhibition. *Nat. Neurosci.*, *8*, 194–201.

Murray, S. O., & Wojciulik, E. (2004). Attention increases neural selectivity in the human lateral occipital complex. *Nat. Neurosci.*, *7*(1), 70–74.

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

Plesser, H. E., & Gerstner, W. (2000). Noise in integrate-and-fire neurons: From stochastic input to escape rates. *Neural Comput.*, *12*(2), 367–384.

Risken, H., & Frank, T. (1996). *The Fokker-Planck equation: Methods of solutions and applications* (2nd ed.). Berlin: Springer.

Schoups, A., Vogels, R., Qian, N., & Orban, G. (2001). Practising orientation identification improves orientation coding in V1 neurons. *Nature*, *412*(6846), 549–553.

Seriès, P., Latham, P. E., & Pouget, A. (2004). Tuning curve sharpening for orientation selectivity: Coding efficiency and the impact of correlations. *Nature Neuroscience*, *7*(10), 1129–1135.

Seung, H., & Sompolinsky, H. (1993). Simple models for reading neuronal population codes. *Proceedings of the National Academy of Sciences*, *90*, 10749–10753.

Shamir, M., & Sompolinsky, H. (2004). Nonlinear population codes. *Neural Comput.*, *16*(6), 1105–1136.

Somers, D. C., Nelson, S. B., & Sur, M. (1995). An emergent model of orientation selectivity in cat visual cortical simple cells. *J. Neuroscience*, *15*, 5448–5465.

Sompolinsky, H., Yoon, H., Kang, K., & Shamir, M. (2001). Population coding in neuronal systems with correlated noise. *Phys. Rev. E*, *64*(5), 051904.

Spitzer, H., Desimone, R., & Moran, J. (1988). Increased attention enhances both behavioral and neuronal performance. *Science*, *240*(4850), 338–340.

Teich, A. F., & Qian, N. (2003). Learning and adaptation in a recurrent model of V1 orientation selectivity. *J. Neurophysiol.*, *89*(4), 2086–2100.

Toyoizumi, T., Aihara, K., & Amari, S. I. (2006). Fisher information for spike-based population decoding. *Physical Review Letters*, *97*(9), 098102.

Wilke, S. D., & Eurich, C. W. (2002). Representational accuracy of stochastic neural populations. *Neural Comput.*, *14*, 155–189.

Wu, S., Amari, S. I., & Nakahara, H. (2002). Population coding and decoding in a neural field: A computational study. *Neural Comput.*, *14*(5), 999–1026.

Wu, S., Nakahara, H., & Amari, S. I. (2001). Population coding with correlation and an unfaithful model. *Neural Comput.*, *13*(4), 775–797.

Yoon, H., & Sompolinsky, H. (1998). The effect of correlations on the Fisher information of population codes. In M. S. Kearns, S. Solla, & D. Cohn (Eds.), *Advances in neural information processing Systems*, *11* (pp. 167–173). Cambridge, MA: MIT Press.

Zhang, K., & Sejnowski, T. J. (1999). Neuronal tuning: To sharpen or broaden? *Neural Comput.*, *11*, 75–84.

Zohary, E., Shadlen, M. N., & Newsome, W. T. (1994). Correlated neuronal discharge rate and its implications for psychophysical performance. *Nature*, *370*, 140–143.

Received September 29, 2008; accepted October 22, 2010.