Fast inference in generalized linear models via expected log-likelihoods

Alexandro D. Ramirez [1,*], Liam Paninski [2]

[1] Weill Cornell Medical College, NY. NY, U.S.A
[*] alr2038@med.cornell.edu
[2] Columbia University Department of Statistics,
Center for Theoretical Neuroscience, Grossman Center for the Statistics of Mind,
Kavli Institute for Brain Science, NY. NY, U.S.A

Abstract

Generalized linear models play an essential role in a wide variety of statistical applications. This paper discusses an approximation of the likelihood in these models that can greatly facilitate computation. The basic idea is to replace a sum that appears in the exact log-likelihood by an expectation over the model covariates; the resulting "expected log-likelihood" can in many cases be computed significantly faster than the exact log-likelihood. In many neuroscience experiments the distribution over model covariates is controlled by the experimenter and the expected log-likelihood approximation becomes particularly useful; for example, estimators based on maximizing this expected log-likelihood (or a penalized version thereof) can often be obtained with orders of magnitude computational savings compared to the exact maximum likelihood estimators. A risk analysis establishes that these maximum EL estimators often come with little cost in accuracy (and in some cases even improved accuracy) compared to standard maximum likelihood estimates. Finally, we find that these methods can significantly decrease the computation time of marginal likelihood calculations for model selection and of Markov chain Monte Carlo methods for sampling from the posterior parameter distribution. We illustrate our results by applying these methods to a computationally-challenging dataset of neural spike trains obtained via large-scale multi-electrode recordings in the primate retina.

1 Introduction

Systems neuroscience has experienced impressive technological development over the last decade. For example, ongoing improvements in multi-electrode recording (Brown et al., 2004; Field et al., 2010; Stevenson and Kording, 2011) and imaging techniques (Cossart et al., 2003; Ohki et al., 2005; Lütcke et al., 2010) have made it possible to observe the activity of hundreds or even thousands of neurons simultaneously. To fully realize the potential of these new high-throughput recording techniques, it will be necessary to develop analytical methods that scale well to large neural population sizes. The need for efficient computational methods is especially pressing in the context of on-line, closed-loop experiments (Donoghue, 2002; Santhanam et al., 2006; Lewi et al., 2009).

Our goal in this paper is to develop scalable methods for neural spike train analysis based on a generalized linear model (GLM) framework (McCullagh and Nelder, 1989). This model class has proven useful for quantifying the relationship between neural responses and external stimuli or behavior (Brillinger, 1988; Paninski, 2004; Truccolo et al., 2005; Pillow et al., 2008; Truccolo et al., 2010), and has been applied successfully in a wide variety of brain areas; see, e.g., (Vidne et al., 2011) for a recent review (Of course, GLMs are well-established as a fundamental tool in applied statistics more generally.) GLMs offer a convenient likelihood-based approach for predicting responses to novel stimuli and for neuronal decoding (Paninski et al., 2007), but computations involving the likelihood can become challenging if the stimulus is very high-dimensional or if many neurons are observed.

The key idea presented here is that the GLM log-likelihood can be approximated cheaply in many cases by exploiting the law of large numbers: we replace an expensive sum that appears in the exact log-likelihood (involving functions of the parameters and observed covariates) by its expectation to obtain an approximate "expected log-likelihood" (EL) (a phrase coined by Park and Pillow in (Park and Pillow, 2011)). Computing this expectation requires knowledge, or at least an approximation, of the covariate distribution. In many neuroscience experiments the covariates correspond to stimuli which are under the control of the experimenter, and therefore the stimulus distribution (or least some moments of this distribution) may be considered known a priori, making the required expectations analytically or numerically tractable. The resulting EL approximation can often be computed significantly more quickly than the exact log-likelihood. This approximation has been exploited previously in some special cases (e.g., Gaussian process regression (Sollich and Williams, 2005; Rasmussen and Williams, 2005) and maximum likelihood estimation of a Poisson regression model (Paninski, 2004; Field et al., 2010; Park and Pillow, 2011; Sadeghi et al., 2013)). We generalize the basic idea behind the EL from the specific models where it has been applied previously to all GLMs in canonical form and discuss the associated computational savings. We then examine a number of novel applications of the EL towards parameter estimation, marginal likelihood calculations, and Monte Carlo sampling from the posterior parameter distribution.

2 Results

2.1 Generalized linear models

Consider a vector of observed responses, $r = (r_1, ..., r_N)$, resulting from N presentations of a p dimensional stimulus vector, x_i (for $i = 1, ..., N$). Under a GLM, with model parameters θ, the likelihood for r is chosen from an exponential family of distributions (Lehmann and Casella, 1998). If we model the observations as conditionally independent given x (an assumption we will later relax), we can write the log-likelihood for r as

$$L(\theta) \equiv \log p(r|\theta, \{x_i\}) = \sum_{n=1}^{N} \frac{1}{c(\phi)} \Big(a(x_n^T\theta) r_n - G(x_n^T\theta) \Big) + \text{const}(\theta), \tag{1}$$

for some functions $a(), G()$, and $c()$, with ϕ an auxiliary parameter (McCullagh and Nelder, 1989), and where we have written terms that are constant with respect to θ as const(θ). For the rest of the paper we will consider the scale factor $c(\phi)$ to be known and for convenience we will set it to one. In addition, we will specialize to the "canonical" case that $a(x_n^T\theta) = x_n^T\theta$, i.e., $a(.)$ is the identity function. With these choices, we see that the GLM log-likelihood is the sum of a linear and non-linear function of θ,

$$L(\theta) \quad = \quad \sum_{n=1}^{N} (x_n^T\theta) r_n - G(x_n^T\theta) + \text{const}(\theta). \tag{2}$$

This expression will be the jumping-off point for the EL approximation. However, first it is useful to review a few familiar examples of this GLM form.

First consider the standard linear regression model, in which the observations r are normally distributed with mean given by the inner product of the parameter vector θ and stimulus vector x. The log-likelihood for r is then

$$L(\theta) \quad = \quad \sum_{n=1}^{N} -\frac{(r_n - x_n^T\theta)^2}{2\sigma^2} + \text{const}(\theta) \tag{3}$$

$$\propto \quad \sum_{n=1}^{N} (x_n^T\theta) r_n - \frac{1}{2}(x_n^T\theta)^2 + \text{const}(\theta), \tag{4}$$

where for clarity in the second line we have suppressed the scale factor set by the noise variance σ^2. The non-linear function $G(.)$ in this case is seen to be proportional to $\frac{1}{2}(x_n^T\theta)^2$.

As another example, in the standard Poisson regression model, responses are distributed by an inhomogeneous Poisson process whose rate is given by the exponential of the inner product of θ and x. (In the neuroscience literature this model is often referred to as a linear-nonlinear-Poisson (LNP) model (Simoncelli et al., 2004).) If we discretize time so that r_n denotes the number of events (e.g., in the neuroscience setting the number of emitted spikes) in time bin n, the log-likelihood is

$$L(\theta) = \sum_{n=1}^{N} \log \left(\exp\Big(-\exp(x_n^T\theta)\Big) \frac{\Big(\exp(x_n^T\theta)\Big)^{r_n}}{r_n!} \right) \tag{5}$$

$$= \sum_{n=1}^{N} (x_n^T\theta) r_n - \exp(x_n^T\theta) + \text{const}(\theta). \tag{6}$$

In this case we see that $G(.) = \exp(.)$.

As a final example, consider the case where responses are distributed according to a binary logistic regression model, so that r_n only takes two values, say 0 or 1, with $p_n \equiv p(r_n = 1|x_n^T, \theta)$ defined according to the canonical "logit" link function

$$\log\left(\frac{p_n}{1-p_n}\right) = x_n^T\theta. \tag{7}$$

Here the log-likelihood is

$$L(\theta) = \sum_{n=1}^{N} \log\Big(p_n^{r_n}(1-p_n)^{1-r_n}\Big) \tag{8}$$

$$= \sum_{n=1}^{N} (x_n^T\theta) r_n + \log(1-p_n) \tag{9}$$

$$= \sum_{n=1}^{N} (x_n^T\theta) r_n - \log\Big(1+\exp\Big(x_n^T\theta\Big)\Big), \tag{10}$$

so $G(.) = \log\Big(1+\exp(.)\Big)$.

2.2 The computational advantage of using expected log-likelihoods over log-likelihoods in a GLM

Now let's examine eq. (2) more closely. For large values of N and p there is a significant difference in the computational cost between the linear and non-linear terms in this expression. Because we can trivially rearrange the linear term as $\sum_{n=1}^{N}(x_n^T\theta)r_n = (\sum_{n=1}^{N} x_n^T r_n)\theta$, its computation only requires a single evaluation of the weighted sum over vectors x $\sum_{n=1}^{N}(x_n^T r_n)$, no matter how many times the log-likelihood is evaluated. (Remember that the simple linear structure of the first term is a special feature of the canonical link function; our results below depend on this canonical assumption.) More precisely, if we evaluate the log-likelihood K times, the number of operations to compute the linear term is $O(Np + Kp)$; computing the non-linear sum, in contrast, requires $O(KNp)$ operations in general. Therefore, the main burden in evaluating the log-likelihood is in the computation of the non-linear term. The EL, denoted by $\tilde{L}(\theta)$, is an approximation to the log-likelihood that can alleviate the computational cost of the non-linear term. We invoke the law of large numbers to approximate the sum over the non-linearity in equation 2 by

its expectation (Paninski, 2004; Field et al., 2010; Park and Pillow, 2011; Sadeghi et al., 2013):

$$L(\theta) = \sum_{n=1}^{N} \Big((x_n^T\theta) r_n - G(x_n^T\theta) \Big) + \text{const}(\theta) \tag{11}$$

$$\approx \left(\sum_{n=1}^{N} x_n^T r_n \right) \theta - N\mathbf{E}\Big[G(x^T\theta)\Big] \equiv \tilde{L}(\theta), \tag{12}$$

where the expectation is with respect to the distribution of x. The EL trades in the $O(KNp)$ cost of computing the nonlinear sum for the cost of computing $\mathbf{E}\Big[G(x^T\theta)\Big]$ at K different values of θ, resulting in order $O(Kz)$ cost, where z denotes the cost of computing the expectation $\mathbf{E}\Big[G(x^T\theta)\Big]$. Thus the nonlinear term of the EL can be be computed about $\frac{Np}{z}$ times faster than the dominant term in the exact GLM log-likelihood. Similar gains are available in computing the gradient and Hessian of these terms with respect to θ.

How hard is the integral $\mathbf{E}\Big[G(x^T\theta)\Big]$ in practice? I.e., how large is z? First, note that because G only depends on the projection of x onto θ, calculating this expectation only requires the computation of a one-dimensional integral:

$$\mathbf{E}\Big[G(x^T\theta)\Big] = \int G(x^T\theta)p(x)dx = \int G(q)\zeta_\theta(q)dq, \tag{13}$$

where ζ_θ is the (θ-dependent) distribution of the one-dimensional variable $q = x^T\theta$. If ζ_θ is available analytically, then we can simply apply standard unidimensional numerical integration methods to evaluate the expectation.

In certain cases this integral can be performed analytically. Assume (wlog) that $\mathbf{E}\Big[x\Big] = 0$, for simplicity. Consider the standard regression case: recall that in this example

$$G(x^T\theta) \propto \frac{\theta^T x x^T \theta}{2}, \tag{14}$$

implying that

$$\mathbf{E}\Big[G(x^T\theta)\Big] = \frac{\theta^T C\theta}{2}, \tag{15}$$

where we have abbreviated $\mathbf{E}\Big[xx^T\Big] = C$. It should be noted that for this Gaussian example one only needs to compute the non-linear sum in the exact likelihood once, since $\sum_n G(x_n^T\theta) = \theta^T(\sum_n x_n x_n^T)\theta$ and $\sum_n x_n x_n^T$ can be precomputed. However, as discussed in section 2.3, if C is chosen to have some special structure, e.g., banded, circulant, Toeplitz, etc., estimates of θ can still be computed orders of magnitude faster using the EL instead of the exact likelihood.

The LNP model provides another example. If $p(x)$ is Gaussian with mean zero and covariance C, then

$$\mathbf{E}\Big[G(x^T\theta)\Big] = \int \exp(x^T\theta) \frac{1}{(2\pi)^{\frac{p}{2}}|C|^{\frac{1}{2}}} \exp\Big(-x^T C^{-1}x/2\Big)dx \tag{16}$$

$$= \exp\Big(\frac{\theta^T C\theta}{2}\Big), \tag{17}$$

where we have recognized the moment-generating function of the multivariate Gaussian distribution.

Note that in each of the above cases, $\mathbf{E}\Big[G(x^T\theta)\Big]$ depends only on $\theta^T C\theta$. This will always be the case (for any nonlinearity $G(.)$) if $p(x)$ is elliptically symmetric, i.e.,

$$p(x) = h(x^T C^{-1}x), \tag{18}$$

for some nonnegative function $h(.)$[1]. In this case we have

$$\mathbf{E}\Big[G(x^T\theta)\Big] = \int G(x^T\theta)h(x^TC^{-1}x)dx \tag{19}$$

$$= \int G(y^T\theta')h(||y||_2^2)|C|^{1/2}dy, \tag{20}$$

where we have made the change of variables $y = C^{-1/2}x, \theta' = C^{1/2}\theta$. Note that the last integral depends on θ' only through its norm; the integral is invariant with respect to transformations of the form $\theta' \to O\theta'$, for any orthogonal matrix O (as can be seen by the change of variables $z = O^Ty$). Thus we only need to compute this integral once for all values of $||\theta'||_2^2 = \theta^TC\theta$, up to some desired accuracy. This can be precomputed off-line and stored in a one-d lookup table before any EL computations are required, making the amortized cost z very small.

What if $p(x)$ is non-elliptical and we cannot compute ζ_θ easily? We can still compute $\mathbf{E}\Big[G(x^T\theta)\Big]$ approximately in most cases with an appeal to the central limit theorem (Sadeghi et al., 2013): we approximate $q = x^T\theta$ in equation 13 as Gaussian, with mean $\mathbf{E}\Big[\theta^Tx\Big] = \theta^T\mathbf{E}\Big[x\Big] = 0$ and variance $\text{var}(\theta^Tx) = \theta^TC\theta$. This approximation can be justified by the classic results of (Diaconis and Freedman, 1984), which imply that under certain conditions, if d is sufficiently large, then ζ_θ is approximately Gaussian for most projections θ. (Of course in practice this approximation is most accurate when the vector x consists of many weakly-dependent, light-tailed random variables and θ has large support, so that q is a weighted sum of many weakly-dependent, light-tailed random variables.) Thus, again, we can precompute a lookup function for $\mathbf{E}\Big[G(x^T\theta)\Big]$, this time over the two-d table of all desired values of the mean and variance of q. Numerically, we find that this approximation often works quite well; Figure 1 illustrates the approximation for simulated stimuli drawn from two non-elliptic distributions (binary white noise stimuli in A and Weibull-distributed stimuli in B).

2.3 Computational efficiency of maximum expected log-likelihood estimation for the LNP and Gaussian model

As a first application of the EL approximation, let us examine estimators that maximize the likelihood or penalized likelihood. We begin with the standard maximum likelihood estimator,

$$\hat{\theta}_{MLE} = \arg\max_\theta \; L(\theta). \tag{21}$$

Given the discussion above, it is natural to maximize the expected likelihood instead:

$$\hat{\theta}_{MELE} = \arg\max_\theta \; \tilde{L}(\theta), \tag{22}$$

where "MELE" abbreviates "maximum EL estimator." We expect that the MELE should be computationally cheaper than the MLE by a factor of approximately Np/z, since computing the EL is approximately a factor of Np/z faster than computing the exact likelihood. In fact, in many cases the MELE can be computed analytically while the MLE requires a numerical optimization, making the MELE even faster.

Let's start by looking at the standard regression (Gaussian noise) case. The EL here is proportional to

$$\tilde{L}(\theta) \propto \theta^TX^Tr - N\frac{\theta^TC\theta}{2}, \tag{23}$$

[1]Examples of such distributions include the multivariate normal and Student's-t, and exponential power families (Fang et al., 1990). Elliptically symmetric distributions are important in the theory of GLMs because they guarantee the consistency of the maximum likelihood estimator for θ even under certain cases of model misspecification; see (Paninski, 2004) for further discussion.

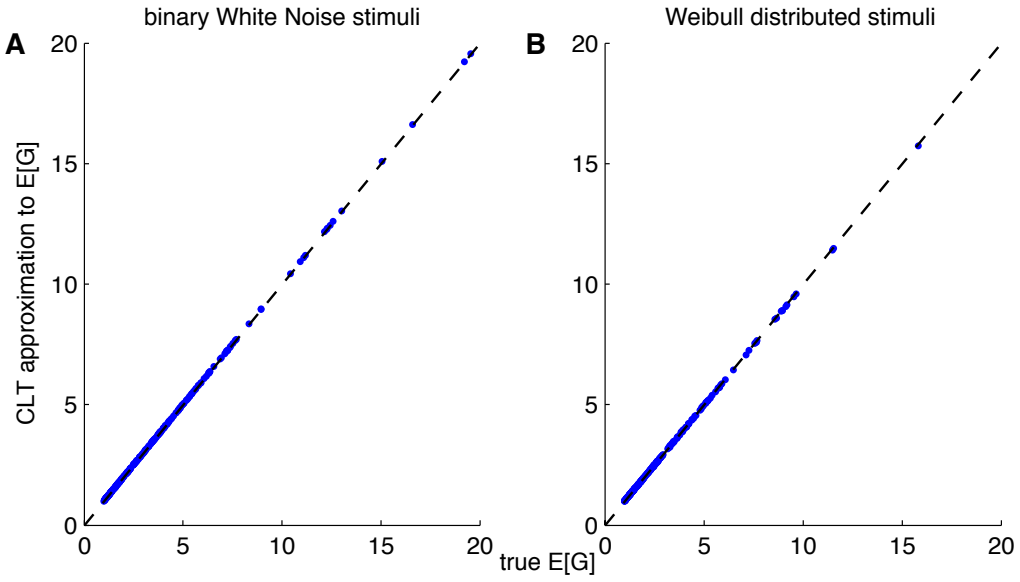

Figure 1: The normal approximation for ζ_θ is often quite accurate for the computation of $\mathbf{E}\Big[G\Big]$ (equation 13 in the text). Vertical axis corresponds to central limit theorem approximation of $\mathbf{E}\Big[G\Big]$, and horizontal axis corresponds to the true $\mathbf{E}\Big[G\Big]$, computed numerically via brute-force Monte Carlo. We used a standard Poisson regression model here, corresponding to an exponential G function. The stimulus vector x is composed of 600 i.i.d. binary (A) or Weibull (B) variables; x has mean 0.36 in panel A, and the Weibull distribution in panel B has scale and shape parameters 0.15 and 0.5, respectively. Each dot corresponds to a different value of the stimulus filter θ. These were zero-mean, Gaussian functions with randomly-chosen norm (uniformly distributed on the interval 0 to 0.5) and scale (found by taking the absolute value of a normally distributed variable with variance equal to 2).

where we have used equation 15 and defined $X = (x_1, ..., x_N)^T$. This is a quadratic function of θ; optimizing directly, we find that the MELE is given by

$$\hat{\theta}_{\mathrm{MELE}} = (NC)^{-1} X^T r. \tag{24}$$

Meanwhile, the MLE here is of the standard least-squares form

$$\hat{\theta}_{MLE} = (X^T X)^{-1} X^T r, \tag{25}$$

assuming $X^T X$ is invertible (the solution is non-unique otherwise).

The computational cost of determining both estimators is determined by the cost of solving a p-dimensional linear system of equations; in general, this requires $O(p^3)$ time. However, if C has some special structure, e.g., banded, circulant, Toeplitz, etc. (as is often the case in real neuroscience experiments), this cost can be reduced to $O(p)$ (in the banded case) or $O(p \log(p))$ (in the circulant case) (Golub and van Van Loan, 1996). The MLE will typically not enjoy this decrease in computational cost, since in general $X^T X$ will be unstructured even when C is highly structured. (Though counterexamples do exist; for example, if X is highly sparse, then $X^T X$ may be sparse even if C is not, for sufficiently small N.)

As another example, consider the LNP model. Somewhat surprisingly, the MELE can be computed analytically for this model (Park and Pillow, 2011) if $p(x)$ is Gaussian and we modify the model slightly to include an offset term so that the Poisson rate in the n-th time bin is given by

$$\lambda_n = \exp(\theta_0 + x_n^T \theta), \tag{26}$$

with the likelihood and EL modified appropriately. The details are provided in (Park and Pillow, 2011) and also, for completeness, the methods section of this paper; the key result is that if one first optimizes the EL (equation 12) with respect to the offset θ_0 and then substitutes the optimal θ_0 back into the EL, the resulting "profile" expected log-likelihood $\max_{\theta_0} \tilde{L}(\theta, \theta_0)$ is a quadratic function of θ, which can be optimized easily to obtain the MELE:

$$\hat{\theta}_{\mathrm{MELE}} = \arg\max_{\theta} \max_{\theta_0} \tilde{L}(\theta, \theta_0) \tag{27}$$

$$= \arg\max_{\theta} \theta^T X^T r - \sum_{n=1}^{N} r_n \frac{\theta^T C \theta}{2} \tag{28}$$

$$= \left((\sum_n r_n) C \right)^{-1} X^T r. \tag{29}$$

Note that this is essentially the same quadratic problem as in the Gaussian case (equation 23) with the total number of spikes $\sum_n r_n$ replacing the number of samples N in equation 23. In the neuroscience literature, the function $\frac{X^T r}{\sum_n r_n}$ is referred to as the spike-triggered average, since if time is discretized finely enough so that the entries of r are either 0 or 1, the product $X^T \frac{r}{\sum_n r_n}$ is simply an average of the stimulus conditioned on the occurrence of a 'spike' ($r = 1$). The computational cost for computing $\hat{\theta}_{\mathrm{MELE}}$ here is clearly identical to that in the Gaussian model (only a simple linear equation solve is required), while to compute the MLE we need to resort to numerical optimization methods, costing $O(KNp)$, with K typically depending superlinearly on p. The MELE can therefore be orders of magnitude faster than the MLE here if Np is large, particularly if C has some structure that can be exploited. See (Park and Pillow, 2011; Sadeghi et al., 2013) for further discussion.

What about estimators that maximize a penalized likelihood? Define the maximum penalized expected log-likelihood estimator (MPELE)

$$\hat{\theta}_{\mathrm{MPELE}} = \arg\max_{\theta} \tilde{L}(\theta) + \log(f(\theta)), \tag{30}$$

where $\log(f(\theta))$ represents a penalty on θ; in many cases $f(\theta)$ has a natural interpretation as the prior distribution of θ. We can exploit special structure in C when solving for the MPELE as well. For example, if we use a mean-zero (potentially improper) Gaussian prior, so that $\log(f(\theta)) = -\frac{1}{2}\theta^T R \theta$ for some positive semidefinite matrix R, the MPELE for the LNP model is again a regularized spike-triggered average (see (Park and Pillow, 2011) and methods)

$$\hat{\theta}_{\mathrm{MPELE}} = \left(C + \frac{R}{\sum_n r_n} \right)^{-1} \frac{X^T r}{\sum_n r_n}. \tag{31}$$

For general matrices R and C, the dominant cost of computing $\hat{\theta}_{\mathrm{MPELE}}$ will be $O(Np + p^3)$. (The exact maximum a posteriori (MAP) estimator has cost comparable to the MLE here, $O(KNp)$.) Again, when C and R share some special structure, e.g. C and R are both circulant or banded, the cost of $\hat{\theta}_{\mathrm{MPELE}}$ drops further.

If we use a sparsening L_1 penalty instead (David et al., 2007; Calabrese et al., 2011), i.e., $\log(f(\theta)) = -\lambda \|\theta\|_1$, with λ a scalar, $\hat{\theta}_{\mathrm{MPELE}}$ under a Gaussian model is defined as

$$\hat{\theta}_{\mathrm{MPELE}} = \arg\max_{\theta} \theta^T X^T r - N \frac{\theta^T C \theta}{2} - \lambda \|\theta\|_1; \tag{32}$$

the MPELE under an LNP model is of nearly identical form. If C is a diagonal matrix, classic results from subdifferential calculus (Nesterov, 2004) show that $\hat{\theta}_{\mathrm{MPELE}}$ is a solution to equation 32 if and only if $\hat{\theta}_{\mathrm{MPELE}}$ satisfies the subgradient optimality conditions

$$-NC_{jj}(\hat{\theta}_{\mathrm{MPELE}})_j + (X^T r)_j = \lambda \ \mathrm{sign}(\hat{\theta}_{\mathrm{MPELE}})_j \ \text{ if } (\hat{\theta}_{\mathrm{MPELE}})_j \neq 0 \tag{33}$$

$$\left| -NC_{jj}(\hat{\theta}_{\mathrm{MPELE}})_j + (X^T r)_j \right| \leq \lambda \ \text{ otherwise}, \tag{34}$$

for $j = 1, ..., p$. The above equations imply that $\hat{\theta}_{\text{MPELE}}$ is a soft-thresholded function of $X^T r$: $(\hat{\theta}_{\text{MPELE}})_j = 0$ if $|(X^T r)_j| \leq \lambda$, and otherwise

$$(\hat{\theta}_{\text{MPELE}})_j = \frac{1}{NC_{jj}}\Big((X^T r)_j - \lambda \text{ sign}(\hat{\theta}_{\text{MPELE}})_j\Big), \tag{35}$$

for $j = 1, ..., p$. Note that equation 35 implies that we can independently solve for each element of $\hat{\theta}_{\text{MPELE}}$ along all values of λ (the so-called regularization path). Since only a single matrix-vector multiply $(X^T r)$ is required, the total complexity in this case is just $O(Np)$. Once again, because $X^T X$ is typically unstructured, computation of the exact MAP is generally much more expensive.

When C is not diagonal we can typically no longer solve equation 32 analytically. However, we can still solve this equation numerically, e.g., using interior-point methods (Boyd and Vandenberghe, 2004). Briefly, these methods solve a sequence of auxiliary, convex problems whose solutions converge to the desired vector. Unlike problems with an L_1 penalty, these auxiliary problems are constructed to be smooth, and can therefore be solved in a small number of iterations using standard methods (e.g., Newton-Raphson (NR) or conjugate gradient (CG)). Computing the Newton direction requires a linear matrix solve of the form $(C + D)\theta = b$, where D is a diagonal matrix and b is a vector. Again, structure in C can often be exploited here; for example, if C is banded, or diagonal plus low-rank, each Newton step requires $O(p)$ time, leading to significant computational gains over the exact MAP.

To summarize, because the population covariance C is typically more structured than the sample covariance $X^T X$, the MELE and MPELE can often be computed much more quickly than the MLE or the MAP estimator. We have examined penalized estimators based on L_2 and L_1 penalization as illustrative examples here; similar conclusions hold for more exotic examples, including group penalties, rank-penalizing penalties, and various combinations of L_2 and L_1 penalties.

2.4 Analytic comparison of the accuracy of EL estimators with the accuracy of maximum-likelihood estimators

We have seen that EL-based estimators can be fast. How accurate are they, compared to the corresponding MLE or MAP estimators? First, note that the MELE inherits the classical consistency properties of the MLE; i.e., if the model parameters are held fixed and the amount of data N tends to infinity, then both of these estimators will recover the correct parameter θ, under suitable conditions. This result follows from the classical proof of the consistency of the MLE (van der Vaart, 1998) if we note that both $(1/N)L(\theta)$ and $(1/N)\tilde{L}(\theta)$ converge to the same limiting function of θ.

To obtain a more detailed view, it is useful to take a close look at the linear regression model, where we can analytically calculate the mean-squared error (MSE) of these estimators. Recall that we assume

$$r|x \sim \mathcal{N}(x^T\theta, I), \tag{36}$$
$$x \sim \mathcal{N}(0, I). \tag{37}$$

(For convenience we have set $\sigma^2 = 1$.) We derive the following MSE formulas in the methods:

$$\mathbf{E}\Big[\|\hat{\theta}_{\text{MELE}} - \theta\|_2^2\Big] = \frac{\theta^T\theta + p(\theta^T\theta + 1)}{N}, \tag{38}$$
$$\mathbf{E}\Big[\|\hat{\theta}_{MLE} - \theta\|_2^2\Big] = \frac{p}{N - p - 1}; \tag{39}$$

see (Shaffer, 1991) for some related results. In the classical limit, for which p is fixed and $N \to \infty$ (and the MSE of both estimators approaches zero), we see that, unless $\theta = 0$, the MLE outperforms the MELE:

$$\lim_{N\to\infty} N\mathbf{E}\Big[\|\hat{\theta}_{\text{MELE}} - \theta\|_2^2\Big] > \lim_{N\to\infty} N\mathbf{E}\Big[\|\hat{\theta}_{MLE} - \theta\|_2^2\Big] = p. \tag{40}$$

However, for many applications (particularly the neuroscience applications we will focus on below), it is more appropriate to consider the limit where the number of samples and parameters are large, both

$N \to \infty$ and $p \to \infty$, but their ratio $\frac{p}{N} = \rho$ is bounded away from zero and infinity. In this limit we see that

$$\mathbf{E}\Big[\|\hat{\theta}_{\text{MELE}} - \theta\|_2^2\Big] \quad \to \quad \rho(\theta^T\theta + 1) \tag{41}$$

$$\mathbf{E}\Big[\|\hat{\theta}_{MLE} - \theta\|_2^2\Big] \quad \to \quad \frac{\rho}{1-\rho}. \tag{42}$$

See figure 7 for an illustration of the accuracy of this approximation for finite N and p.

Figure 2A (left panel) plots these limiting MSE curves as a function of ρ. Note that we do not plot the MSE for values of $\rho > 1$ because the MLE is non-unique when p is greater than N; also note that eq. (42) diverges as $\rho \nearrow 1$, though the MELE MSE remains finite in this regime. We examine these curves for a few different values of $\theta^T\theta$; note that since $\sigma^2 = 1$, $\theta^T\theta$ can be interpreted as the signal variance divided by the noise variance, i.e., the signal-to-noise ratio (SNR):

$$SNR \quad = \quad \frac{\mathbf{E}\Big[\theta^T x^T x \theta\Big]}{\sigma^2} \tag{43}$$

$$= \quad \frac{\theta^T\theta}{1}. \tag{44}$$

The second line follows from the fact that we choose stimuli with identity covariance. The key conclusion is that the MELE outperforms the MLE for all $\rho > \frac{SNR}{1+SNR}$. (This may seem surprising, since for a given X, a classic result in linear regression is that the MLE has the lowest MSE amongst all unbiased estimators of θ (Bickel and Doksum, 2007). However, the MELE is biased given X, and can therefore have a lower MSE than the MLE by having a smaller variance.)

What if we examine the penalized versions of these estimators? In the methods we calculate the MSE of the MAP and MPELE given a simple ridge penalty of the form $\log(f(\theta)) = -\frac{R}{2}\|\theta\|_2^2$, for scalar R. Figure 2B (top panel) plots the MSE for both estimators (see equations 85, 101 in the methods for the equations being plotted) as a function of R and ρ for an SNR value of 1. Note that we now plot MSE values for $\rho > 1$ since regularization makes the MAP solution unique. We see that the two estimators have similar accuracy over a large region of parameter space. For each value of ρ we also compute each estimator's optimal MSE — i.e., the MSE corresponding to the the value of R that minimizes each estimator's MSE. This is plotted in Figure 2A (right panel). Again, the two estimators perform similarly.

In conclusion, in the limit of large but comparable N and p, the MELE outperforms the MLE in low-SNR or high-(p/N) regimes. The ridge-regularized estimates (the MPELE and MAP) perform similarly across a broad range of (p/N) and regularization values (Figure 2B). These analytic results motivate the applications (using non-Gaussian GLMs) on real data treated in the next section.

2.5 Fast methods for refining maximum expected log-likelihood estimators to obtain MAP accuracy

In settings where the MAP provides a more accurate estimator than the MPELE, a reasonable approach is to use $\hat{\theta}_{\text{MPELE}}$ as a quickly-computable initializer for optimization algorithms used to compute $\hat{\theta}_{MAP}$. An even faster approach would be to initialize our search at $\hat{\theta}_{\text{MPELE}}$, then take just enough steps towards $\hat{\theta}_{MAP}$ to achieve an estimation accuracy which is indistinguishable from that of the exact MAP estimator. (Related ideas based on stochastic gradient ascent methods have seen increasing attention in the machine learning literature (Bottou, 1998; Boyles et al., 2011).) We tested this idea on real data, by fitting an LNP model to a population of ON and OFF parasol ganglion cells (RGCs) recorded *in vitro*, using methods similar to those described in (Shlens et al., 2006; Pillow et al., 2008); see these earlier papers for full experimental details. The observed cells responded to either binary white-noise stimuli or naturalistic stimuli (spatiotemporally correlated Gaussian noise with spatial correlations having a 1/F power spectrum and temporal correlations defined by a first-order autoregressive process; see methods for details). As

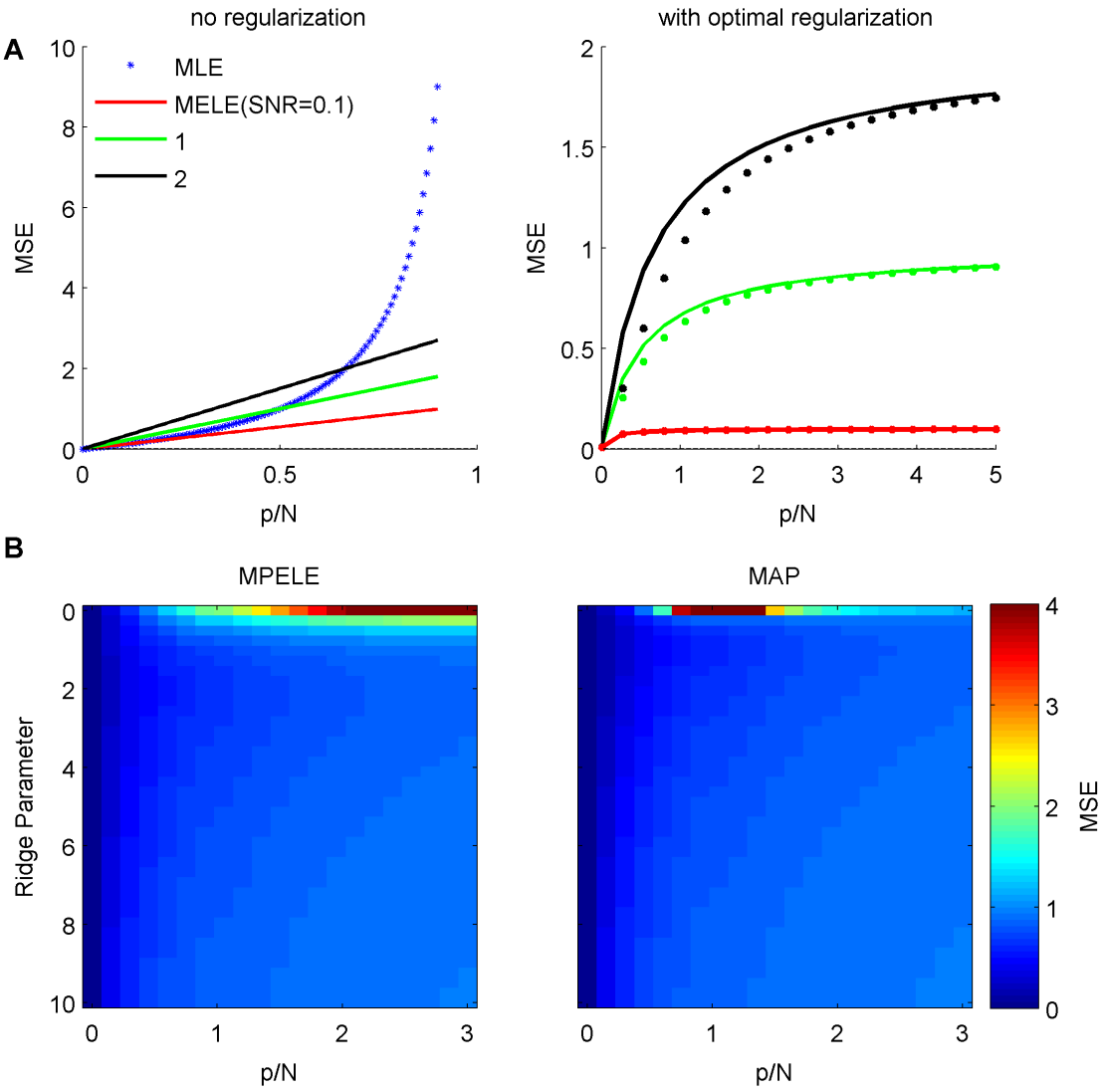

Figure 2: Comparing the accuracy of the MAP and MPELE in the standard linear regression model with Gaussian noise. A.) (left) The mean squared error (MSE) for the MELE (solid lines) and the MLE (dotted line) is shown as a function of p/N the ratio of the number of parameters to number of samples. We plot results for the asymptotic case where the number of samples and dimensions goes to infinity but the ratio p/N remains finite. Different colors denote different values for the true filter norm; recall that the MSE of the MLE is independent of the true value of θ, since the MLE is an unbiased estimator of θ in this model. The MLE mean squared error is larger than that of the MELE when p/N is large. B.) MSE for both estimators when L2 regularization is added. The MSE is similar for both estimators for a large range of ridge parameters and values of p/N. A.) (right) For each value of p/N, separate ridge parameters are chosen for the MPELE and MAP estimators to minimize their respective mean squared errors. Solid curves correspond to MPELE (as in left panel); dotted curves to MAP estimates. The difference in performance between the two optimally-regularized estimators remains small for a wide range of values of SNR and p/N. Similar results are observed numerically in the Poisson regression (LNP) case (data not shown).

described in the methods, each receptive field was specified by 810 parameters, with $\frac{p}{N} = 0.021$. For the MAP, we use a simple ridge penalty of the form $\log(f(\theta)) = -\frac{R}{2}\|\theta\|_2^2$.

Many iterative algorithms are available for ascending the posterior to approximate the MAP. Preconditioned conjugate gradient ascent (PCG) (Shewchuk, 1994) is particularly attractive here, for two reasons. First, each iteration is fairly fast: the gradient requires $O(Np)$ time, and multiplication by the preconditioner turns out to be fast, as discussed below. Second, only a few iterations are necessary, because the MPELE is typically fairly close to the exact MAP in this setting (recall that $\hat{\theta}_{\mathrm{MPELE}} \to \hat{\theta}_{MAP}$ as $N/p \to \infty$), and we have access to a good preconditioner, ensuring that the PCG iterates converge quickly. We chose the inverse Hessian of the EL evaluated at the MELE or MPELE as a pre-conditioner. In this case, using the same notation as in equations 27 and 31, the preconditioner is simply given by $(C\sum_n r_n)^{-1}$ or $(C\sum_n r_n + RI)^{-1}$. Since the EL Hessian provides a good approximation for the log-likelihood Hessian, the preconditioner is quite accurate; since the stimulus covariance C is either proportional to the identity (in the white-noise case) or of block-Toeplitz form (in the spatiotemporally-correlated case), computation with the preconditioner is fast ($O(p)$ or $O(p \log p)$, respectively).

For binary white-noise stimuli we find that the MELE (given by equation 27 with $C = \mathbf{I}$) and MLE yield similar filters and accuracy, with the MLE slightly outperforming the MELE (see figure 3A). Note that in this case, $\hat{\theta}_{\mathrm{MELE}}$ can be computed quickly, $O(pN)$, since we only need to compute the matrix-vector multiplication $X^T r$. On average across a population of 126 cells, we find that terminating the PCG algorithm, initialized at the MELE, after just two iterations yielded an estimator with the same accuracy as the MAP. To measure accuracy we use the cross-validated log-likelihood (see methods). It took about 15× longer to compute the MLE to default precision than the PCG-based approximate MLE (88±2 vs 6±0.1 seconds on an Intel Core 2.8 GHz processor running Matlab; all timings quoted in this paper use the same computer). In the case of spatiotemporally correlated stimuli (with $\hat{\theta}_{\mathrm{MPELE}}$ given by equation 31), we find that 9 PCG iterations are required to reach MAP accuracy (see figure 3B); the MAP estimator was still slower to compute by a significant factor (107±8 vs 33±1 seconds).

2.5.1 Scalable modeling of interneuronal dependencies

So far we have only discussed models which assume conditional independence of responses given an external stimulus. However, it is known the predictive performance of the GLM can be improved in a variety of brain areas by allowing some conditional dependence between neurons (e.g., (Truccolo et al., 2005; Pillow et al., 2008); see (Vidne et al., 2011) for a recent review of related approaches). One convenient way to incorporate these dependencies is to include additional covariates in the GLM. Specifically, assume we have recordings from M neurons and let r_i be the vector of responses across time of the ith neuron. Each neuron is modeled with a GLM where the weighted covariates, $x_n^T\theta_i$, can be broken into an offset term, an external stimulus component x^s, and a spike history-dependent component

$$x_n^T\theta_i \quad = \quad \theta_{i0} + (x^s)^T\theta_i^s + \sum_{j=1}^{M}\sum_{k=1}^{\tau} r_{j,n-k}\theta_{ijk}^H, \tag{45}$$

for $n = 1, ..., N$, where τ denotes the maximal number of lags used to predict the firing rate of neuron i given the activity of the the observed neurons indexed by j. Note that this is the same model as before when $\theta_{ijk}^H = 0$ for $j = 1, 2, ..., M$; in this special case, we recover an inhomogeneous Poisson model, but in general, the outputs of the coupled GLM will be non-Poisson. A key challenge for this model is developing estimation methods that scale well with the number of observed neurons; this is critical since, as discussed above, experimental methods continue to improve, resulting in rapid growth of the number of simultaneously observable neurons during a given experiment (Stevenson and Kording, 2011).

One such scalable approach uses the MELE of an LNP model to fit a fully-coupled GLM with history terms. The idea is that estimates of θ^s fit with $\theta_{ijk}^H = 0$ will often be similar to estimates of θ^s without this hard constraint. Thus we can again use the MELE (or MPELE) as an initializer for θ^s, and then use fast iterative methods to refine our estimate, if necessary. More precisely, to infer θ_i for $i = 1, 2, ..., M$, we

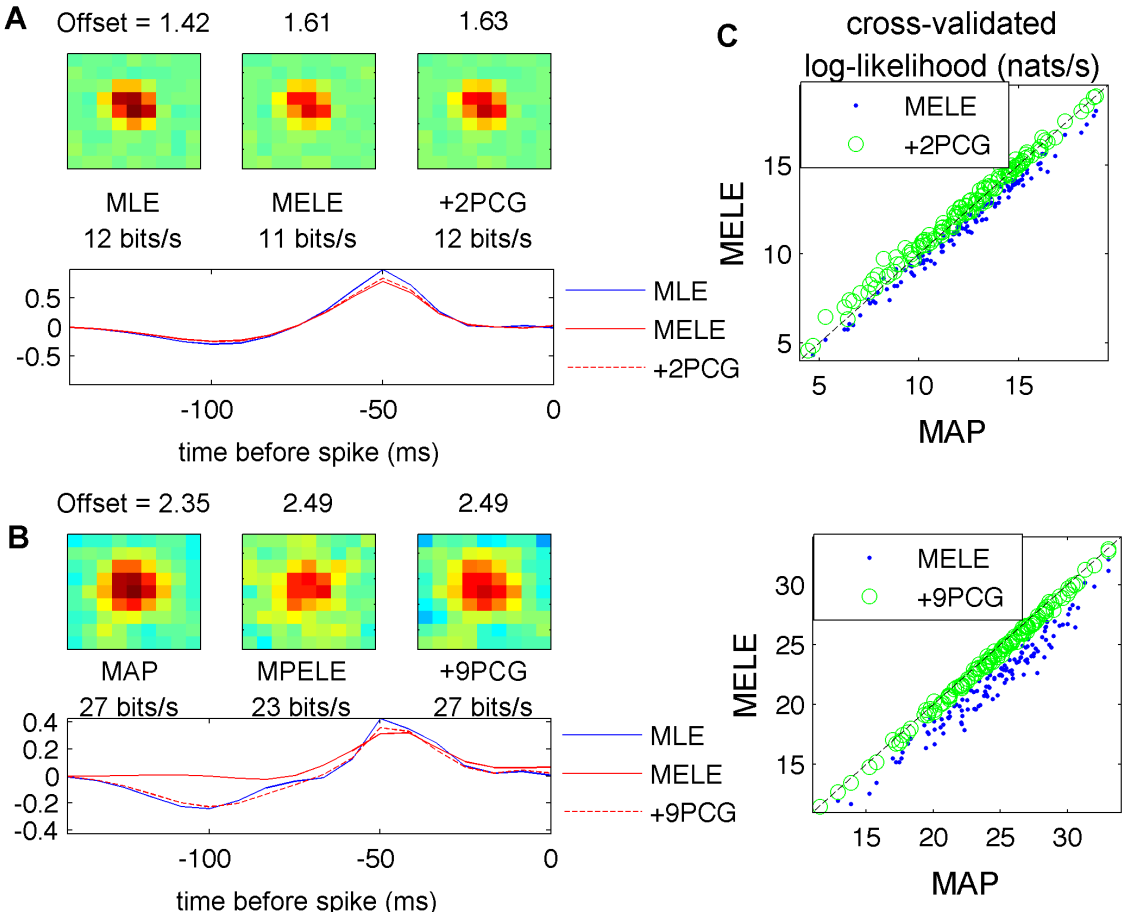

Figure 3: The MPELE provides a good initializer for finding approximate MAP estimators using fast gradient-based optimization methods in the LNP model. A.) The spatiotemporal receptive field of a typical retinal ganglion cell (RGC) in our database responding to binary white-noise stimuli and fit with a linear-nonlinear Poisson (LNP) model with exponential non-linearity, via the MLE. The receptive field of the same cell fit using the MELE is also plotted. The goodness-of-fit of the MLE (measured in terms of cross-validated log-likelihood; see methods) is slightly higher than that of the MELE (12 versus 11 bits/s). However, this difference disappears after a couple pre-conditioned conjugate gradient (PCG) iterations using the true likelihood initialized at the MELE (see label +2PCG). Note that we are only showing representative spatial and temporal slices of the 9x9x10 dimensional receptive field, for clarity. B.) Similar results hold when the same cell responds to 1/f correlated Gaussian noise stimuli (for correlated Gaussian responses, the MAP and MPELE are both fit with a ridge penalty). In this case 9 PCG iterations sufficed to compute an estimator with a goodness-of-fit equal to that of the MAP. C.) These results are consistent over the observed population. (top) Scatterplot of cross-validated log-likelihood across a population of 91 cells, responding to binary white-noise stimuli, each fit independently using the MLE, MELE or a couple PCG iterations using the true likelihood and initialized at the MELE. (bottom) Log-likelihood for the same population, responding to 1/f Gaussian noise, fit independently using an L2 regularized MAP, MPELE or 9 PCG iterations using the true regularized likelihood initialized at the MPELE.

first estimate θ_i^s, assuming no coupling or self-history terms ($\theta_{ijk}^H = 0$ for $j = 1, 2, ..., M$) using the MELE (equation 27 or a regularized version). We then update the history components and offset by optimizing the GLM likelihood with the stimulus filter held fixed up to a gain factor, α_i, and the interneuronal terms

$\theta^H_{ijk} = 0$ for $i \neq j$:

$$(\theta_{i0}, \alpha_i, \theta^H_i) = \arg \max_{(\theta_0, \alpha, \theta^H_{ijk}=0 \ \forall i \neq j)} L\Big((\theta_0, \alpha\hat{\theta}_{\mathrm{MPELE}}, \theta^H)\Big) + \log\Big(f(\theta_0, \alpha\hat{\theta}_{\mathrm{MPELE}}, \theta^H)\Big). \quad (46)$$

Holding the shape of the stimulus filter fixed greatly reduces the number of free parameters, and hence the computational cost, of finding the history components. Note that all steps so far scale linearly in the number of observed neurons M. Finally, perform a few iterative ascent steps on the full posterior, incorporating a sparse prior on the interneuronal terms (exploiting the fact that neural populations can often be modeled as weakly conditionally dependent given the stimulus; see (Pillow et al., 2008) and (Mishchenko and Paninski, 2011) for further discussion). If such a sparse solution is available, this step once again often requires just $O(M)$ time, if we exploit efficient methods for optimizing posteriors based on sparsity-promoting priors (Friedman et al., 2010; Zhang, 2011).

We investigated this approach by fitting the GLM specified by equation 45 to a population of 101 RGC cells responding to binary white-noise using 250 stimulus parameters and 105 parameters related to neuronal history (see methods for full details; $\frac{p}{N} = 0.01$). We regularize the coupling history components using a sparsity-promoting L_1 penalty of the form $\lambda \sum_j |\theta^H_{ij}|$; for simplicity, we parametrize each coupling filter θ^H_{ij} with a single basis function, so that it is not necessary to sum over many k indices. (However, note that group-sparsity-promoting approaches are also straightforward here (Pillow et al., 2008).) We compute our estimates over a large range of the sparsity parameter λ using the "glmnet" coordinate ascent algorithm discussed by (Friedman et al., 2010).

Figure 4A compares filter estimates for two example cells using the fast approximate method and the MAP. The filter estimates are similar (though not identical); both methods find the same coupled, nearest neighbor cells. Both methods achieve the same cross-validated prediction accuracy (Fig. 4B). The fast approximate method took an average of 1 minute to find the entire regularization path; computing the full MAP path took 16 minutes on average.

2.6 Marginal likelihood calculations

In the previous examples we have focused on applications that require us to maximize the (penalized) EL. In many applications we are more interested in integrating over the likelihood, or sampling from a posterior, rather than simply optimization. The EL approximation can play an important role in these applications as well. For example, in Bayesian applications one is often interested in computing the marginal likelihood: the probability of the data, $p(r)$, with the dependence on the parameter θ integrated out (Gelman et al., 2003; Kass and Raftery, 1995). This is done by assigning a prior distribution $f(\theta|R)$, with its own "hyper-parameters" R, over θ and calculating

$$F(R) \equiv p(r|x_1, ..., x_N, R) = \int p(r, \theta|x_1, ..., x_N, R)d\theta = \int p(r|\theta, x_1, ..., x_N)f(\theta|R)d\theta. \quad (47)$$

Hierarchical models in which R sets the prior over θ_i for many neurons i simultaneously are also useful (Behseta et al., 2005); the methods discussed below extend easily to this setting.

We first consider a case for which this integral is analytically tractable. We let the prior on θ be Gaussian, and use the standard regression model for $r|\theta$:

$$p(r, \theta|X, R) = p(r|X, \theta)f(\theta|R) \quad (48)$$
$$= \mathcal{N}(X\theta, \sigma^2\mathbf{I})\mathcal{N}(0, R^{-1}), \quad (49)$$

where $\mathbf{I}$ is the identity matrix. Computing the resulting Gaussian integral, we have

$$\log F(R) = \frac{1}{2}\log\Big(\det(\Sigma R)\Big) + \frac{r^T X \Sigma X^T r}{2\sigma^4} + \mathrm{const}(R), \quad (50)$$
$$\Sigma = (X^T X\sigma^2 + R)^{-1}. \quad (51)$$

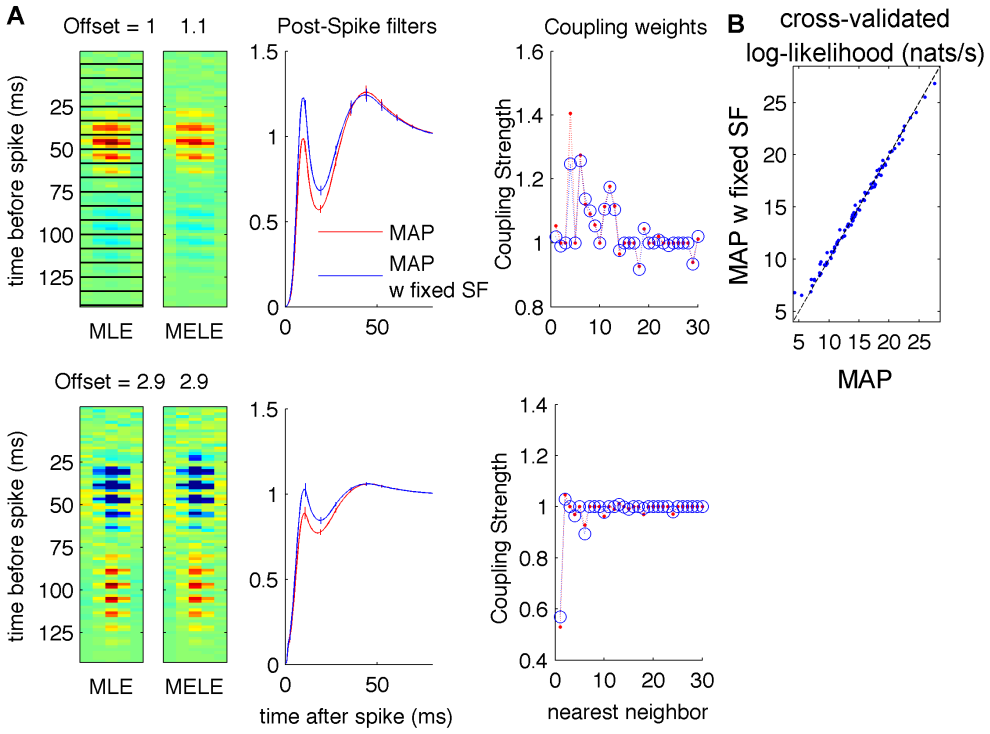

Figure 4: Initialization with the MPELE using an LNP model, then coordinate descent using a sparsity-promoting prior, efficiently estimates a full, coupled, non-Poisson neural population GLM. A.) Example stimulus, self-history, and coupling filters for two different RGC cells (top and bottom rows). The stimulus filters are laid out in a rasterized fashion to map the three-dimensional filters (two spatial dimensions and one temporal dimension) onto the two-dimensional representation shown here. Each filter is shown as a stack (corresponding to the temporal dimension) of two dimensional spatial filters, which we outline in black in the top left to aid the visualization. MAP parameters are found using coordinate descent to optimize the exact GLM log-likelihood with L_1-penalized coupling filters (labeled MAP). Fast estimates of the self-history (see methods for details of errorbar computations) and coupling filters are found by running the same coordinate descent algorithm with the stimulus filter (SF) fixed, up to a gain, to the MELE of an LNP model (labeled 'w fixed SF'; see text for details). Note that estimates obtained using these two approaches are qualitatively similar. In particular, note that both estimates find coupling terms that are largest for neighboring cells, as in (Pillow et al., 2008; Vidne et al., 2011). We do not plot the coupling weights for cells 31-100 since most of these are zero or small. B.) Scatterplot comparing the cross-validated log-likelihood over 101 different RGC cells show that the two approaches lead to comparable predictive accuracy.

If R and $X^T X$ do not share any special structure, each evaluation of $F(R)$ requires $O(p^3)$ time.

On the other hand, if we approximate $p(r|X,\theta)$ by the the expected likelihood, the integral is still tractable and has the same form as equation 50, but with $\Sigma = (NC\sigma^2 + R)^{-1}$ assuming $\mathrm{Cov}[x] = C$. In many cases the resulting $F(R)$ calculations are faster: for example, if C and R can be diagonalized (or made banded) in a convenient shared basis, then $F(R)$ can often be computed in just $O(p)$ time.

When the likelihood or prior is not Gaussian we can approximate the marginal likelihood using the

Laplace approximation (Kass and Raftery, 1995)

$$\begin{aligned}\log F(R) &\approx \log p(\theta_{MAP}|R) - \frac{1}{2}\log(\det(-H(\theta_{MAP}))) \\ &= \theta_{MAP}^T X^T r - \sum_n G(x_n^T\theta_{MAP}) + \log\Big(f(\theta_{MAP}|R)\Big) - \frac{1}{2}\log(\det(-H(\theta_{MAP}))),\end{aligned} \tag{52}$$

again neglecting factors that are constant with respect to R. $H(\theta_{MAP})$ is the posterior Hessian

$$H_{ij} = \frac{\partial^2}{\partial\theta_i\partial\theta_j}\Big(-\sum_n G(x_n^T\theta) + \log\Big(f(\theta|R)\Big)\Big), \tag{53}$$

evaluated at $\theta = \theta_{MAP}$. We note that there are other methods for approximating the integral in equation 47, such as Evidence Propagation (Minka, 2001; Bishop, 2006); we leave an exploration of EL-based approximations of these alternative methods for future work.

We can clearly apply the EL approximation in eqs. (52-53). For example, consider the LNP model, where we can approximate $\mathbf{E}\Big[G(\theta_0 + x^T\theta)\Big]$ as in equation 16; if we use a normal prior of the form $f(\theta^s|R) = \mathcal{N}(0, R^{-1})$, then the resulting EL approximation to the marginal likelihood looks very much like the Gaussian case, with the attending gains in efficiency if C and R share an exploitable structure, as in our derivation of the MPELE in this model (recall section 2.3).

This EL-approximate marginal likelihood has several potential applications. For example, marginal likelihoods are often used in the context of model selection: if we need to choose between many possible models indexed by the hyperparameter R, then a common approach is to maximize the marginal likelihood as a function of R (Gelman et al., 2003). Denote $\hat{R} = \arg\max_R F(R)$. Computing $\hat{R}$ directly is often expensive, but computing the maximizer of the EL-approximate marginal likelihood instead is often much cheaper. In our LNP example, for instance, the EL-approximate $\hat{R}$ can be computed analytically if we can choose a basis such that $C = \mathbf{I}$ and $R \propto \mathbf{I}$:

$$\hat{R} = \begin{cases} \frac{p}{\frac{q}{N_s^2} - \frac{p}{N_s}}\mathbf{I} & \text{if } p < \frac{q}{N_s} \\ \infty & \text{if } p \geq \frac{q}{N_s}, \end{cases} \tag{54}$$

with $q \equiv \|X^T r\|_2^2$ and $N_s = \sum_{n=1}^N r_n$; see methods for the derivation. Since $\hat{\theta}_{\text{MPELE}} = \Big(C + \frac{R}{\sum_n r_n}\Big)^{-1}\frac{X^T r}{\sum_n r_n}$ (equation 31), an infinite value of R corresponds to $\hat{\theta}_{\text{MPELE}}$ equal to zero: infinite penalization. The intuitive interpretation of equation 54 is that the MPELE should be shrunk entirely to zero when there isn't enough data, quantified by $\frac{q}{N_s}$, compared to the dimensionality of the problem p. Similar results hold when there are correlations in the stimulus, $C \neq I$, as discussed in more detail in the methods.

Fig. 5 presents a numerical illustration of the resulting penalized estimators. We simulated Poisson responses to white-noise Gaussian stimuli using stimulus filters with $p = 250$ parameters. We use a standard iterative method for computing the exact $\hat{R}$: it is known that the optimal $\hat{R}$ obeys the equation

$$\hat{R} = \frac{p - \hat{R}\sum_i H^{-1}(\hat{\theta}_{MAP})_{ii}}{\hat{\theta}_{MAP}(\hat{R})^T\hat{\theta}_{MAP}(\hat{R})}, \tag{55}$$

under the Laplace approximation (Bishop, 2006). Note that this equation only implicitly solves for $\hat{R}$, since $\hat{R}$ is present on both sides of the equation. However, this leads to the following common fixed-point iteration:

$$\hat{R}_{i+1} = \frac{p - \hat{R}_i\sum_i H^{-1}(\hat{\theta}_{MAP})_{ii}}{\hat{\theta}_{MAP}(\hat{R}_i)^T\hat{\theta}_{MAP}(\hat{R}_i)}, \tag{56}$$

with $\hat{R}$ estimated as $\lim_{i\to\infty}\hat{R}_i$, when this limit exists. We find that the distance between the exact and approximate $\hat{R}$ values increases with $\frac{p}{N}$, with the EL systematically choosing lower values of $\hat{R}$ (Figure 5A

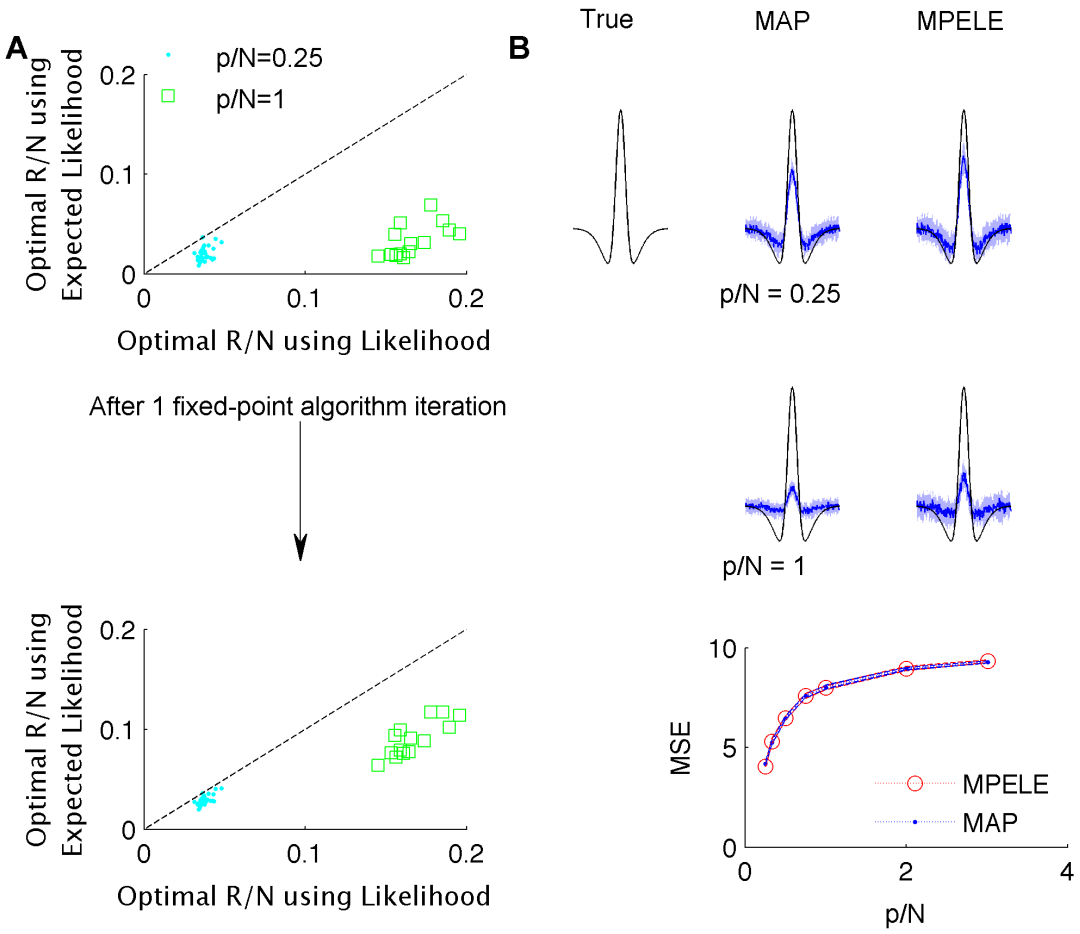

Figure 5: The EL can be used for fast model selection via approximate maximum marginal likelihood. Thirty simulated neural responses were drawn from a linear-nonlinear Poisson (LNP) model, with stimuli drawn from an independent white-noise Gaussian distribution. The true filter (shown in black in B) has $p = 250$ parameters and norm 10. A.) Optimal hyper-parameters R (the precision of the Gaussian prior distribution) which maximize the marginal likelihood using the EL (top left column, vertical axis) scale similarly to those which maximize the full Laplace-approximated marginal likelihood (top left column, horizontal axis), but with a systematic downward bias. After a single iteration of the fixed point algorithm used to maximize the full marginal likelihood (see text), the two sets of hyper-parameters (bottom left column) match to what turns out to be sufficient accuracy, as shown in (B): the median filter estimates (blue lines) ($\pm$ absolute median deviation (light blue), based on 30 replications) computed using the exact and one-step approximate approach match for a wide range of $\frac{p}{N}$. The MSE of the two approaches also matches for a wide range of $\frac{p}{N}$.

top). This difference shrinks after a single iteration of equation 56 initialized using equation 54 (Figure 5A bottom). The remaining differences in $\hat{R}$ between the two methods did not lead to differences in the corresponding estimated filters $\hat{\theta}_{MAP}(\hat{R})$ (Figure 5B). In these simulations the exact MAP typically took about 20 times longer to compute than a single iteration of equation 56 initialized using equation 54 (10 versus 0.5 seconds).

We close by noting that many alternative methods for model selection in penalized GLMs have been studied; it seems likely that the EL approximation could be useful in the context of other approaches based on cross-validation or generalized cross-validation (Golub et al., 1979), for example. We leave this possibility for future work. See (Conroy and Sajda, 2012) for a related recent discussion.

2.7 Decreasing the computation time of Markov Chain Monte Carlo methods

As a final example, in this section we investigate the utility of the EL approximation in the context of Markov chain Monte Carlo (MCMC) methods (Robert and Casella, 2005) for sampling from the posterior distribution of θ given GLM observations. Two approaches suggest themselves. First, we could simply replace the likelihood with the exponentiated EL, and sample from the resulting approximation to the true posterior using our MCMC algorithm of choice (Sadeghi et al., 2013). This approach is fast but only provides samples from an approximation to the posterior. Alternatively, we could use the EL-approximate posterior as a proposal density within an MCMC algorithm, and then use the standard Metropolis-Hastings correction to obtain samples from the exact posterior. This approach, however, is slower, because we need to compute the true log-likelihood with each Metropolis-Hastings iteration.

Figure 6 illustrates an application of the first approach to simulated data. We use a standard Hamiltonian Monte Carlo (Neal, 2012) method to sample from both the true and EL-approximate posterior given $N = 4000$ responses from a 100-dimensional LNP model, with a uniform (flat) prior. We compare the marginal median and 95% credible interval computed by both methods, for each element of θ. For most elements of the θ vector, the EL-approximate posterior matches the true posterior well. However, in a few cases the true and approximate credible intervals differ significantly; thus, it makes sense to use this method as a fast exploratory tool, but perhaps not for conclusive analyses when N/p is of moderate size. (Of course, as $N/p \to \infty$, the EL approximation becomes exact, while the true likelihood becomes relatively more expensive, and so the EL approximation will become the preferred approach in this limit.)

For comparison, we also compute the median and credible intervals using two other approaches: (1) the standard Laplace approximation (computed using the true likelihood), and (2), the profile EL-posterior, which is exactly Gaussian in this case (recall section 2.3). The intervals computed via (1) closely match the MCMC output on the exact posterior, while the intervals computed via (2) closely match the EL-approximate MCMC output; thus the respective Gaussian approximations of the posterior appear to be quite accurate for this example.

Experiments using the second approach described above (using Metropolis-Hastings to obtain samples from the exact posterior) were less successful. The Hamiltonian Monte Carlo method is attractive here, since we can in principle evaluate the EL-approximate posterior cheaply many times along the Hamiltonian trajectory before having to compute the (expensive) Metropolis-Hastings probability of accepting the proposed trajectory. However, we find (based on simulations similar to those described above) that proposals based on the fast EL-approximate approach are rejected at a much higher rate than proposals generated using the exact posterior. This lower acceptance probability in turn implies that more iterations are required to generate a sufficient number of accepted steps, sharply reducing the computational advantage of the EL-based approach. Again, as $N/p \to \infty$, the EL approximation becomes exact, and the EL approach will be preferred over exact MCMC methods — but in this limit the Laplace approximation will also be exact, obviating the need for expensive MCMC approaches in the first place.

3 Conclusion

We have demonstrated the computational advantages of using the expected log-likelihood (EL) to approximate the log-likelihood of a generalized linear model with canonical link. When making multiple calls to the GLM likelihood (or its gradient and Hessian), the EL can be computed approximately $O(Np/z)$ times faster, where N is the number of data samples collected, p is the dimensionality of the parameters and z is the cost of a one-dimensional integral; in many cases this integral can be evaluated analytically or semi-analytically, making z trivial. In addition, in some cases the EL can be analytically optimized or integrated out, making EL-based approximations even more powerful. We discussed applications to maximum penalized likelihood-based estimators, model selection, and MCMC sampling, but this list of applications is certainly far from exhaustive.

Ideas related to the EL approximation have appeared previously in a wide variety of contexts. We

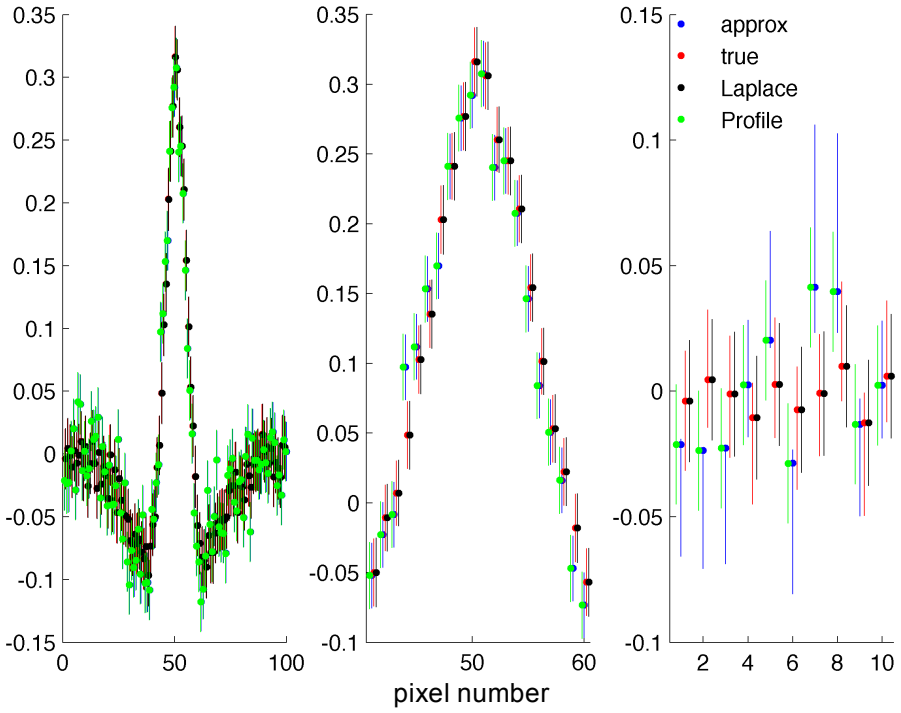

Figure 6: The EL approximation leads to fast and (usually) accurate MCMC sampling from GLM posteriors. 4000 responses were simulated from a 100-dimensional LNP model using i.i.d. white-noise Gaussian stimuli. 10^6 Markov Chain Monte Carlo samples, computed using Hybrid Monte Carlo, were then drawn from the posterior assuming a flat prior, using either the exact Poisson likelihood or the EL approximation to the likelihood. The left column displays the median vector along with 95% credible regions for each marginal distribution (one marginal for each of the 100 elements of θ); approximate intervals are shown in blue, and exact intervals in red. In the middle and right column we have zoomed in around different elements for visual clarity. Statistics from both distributions are in close agreement for most, but not all, elements of θ. Replacing the EL-approximate likelihood with the EL-approximate profile likelihood yielded similar results (green). The Laplace approximation to the exact posterior also provided a good approximation to the exact posterior (black).

have already discussed previous work in neuroscience (Paninski, 2004; Park and Pillow, 2011; Field et al., 2010; Sadeghi et al., 2013) that exploits the EL approximation in the LNP model. Similar approximations have also been used to simplify the likelihood of GLM models in the context of neural decoding (Rahnama Rad and Paninski, 2011). In the Gaussian process regression literature, the well-known "equivalent kernel" approximation can be seen as a version of the EL approximation (Rasmussen and Williams, 2005; Sollich and Williams, 2005); similar approaches have a long history in the spline literature (Silverman, 1984). Finally, the EL approximation is somewhat reminiscent of the classical Fisher scoring algorithm, in which the observed information matrix (the Hessian of the negative log-likelihood) is replaced by the expected information matrix (i.e., the expectation of the observed information matrix, taken over the responses r) in the context of approximate maximum likelihood estimation. The major difference is that the EL takes the expectation over the covariates x instead of the responses r. A potential direction for future work would be to further explore the relationships between these two expectation-based approximations.

4 Appendix: methods

4.1 Computing the mean-squared error for the MPELE and MAP in the linear-Gaussian model

In this section we provide derivations for the results discussed in section 2.4. We consider the standard linear regression model with Gaussian noise and a ridge (Gaussian) prior of the form $f(\theta) \propto \exp\left(-cp\frac{\theta^T\theta}{2}\right)$, with c a scalar. We further assume that the stimuli x are i.i.d. standard Gaussian vectors. We derive the MSE of the MPELE and MAP, then recover the non-regularized cases (i.e., the MLE and MELE) by setting c to zero. Note that we allow the regularizer to scale with the dimensionality of the problem, p, for reasons that will become clear below. The resulting MAP and MPELE are then found by

$$\begin{aligned}
\hat{\theta}_{MAP} &= \arg\max_{\theta}\ \theta^T X^T r - \frac{1}{2}\theta^T(X^TX + cpI)\theta && (57)\\
&= (X^TX + cpI)^{-1}X^Tr && (58)\\
\hat{\theta}_{\text{MPELE}} &= \arg\max_{\theta}\ \theta^T X^T r - \frac{1}{2}(N + cp)\theta^T\theta && (59)\\
&= \frac{X^Tr}{N+cp}, && (60)
\end{aligned}$$

where we consider $X_{ij} \sim \mathcal{N}(0,1)\ \forall i,j$. For convenience of notation we define the quantity $\tilde{S} = X^TX + cpI$ and therefore write the MAP as $\hat{\theta}_{MAP} = \tilde{S}^{-1}X^Tr$.

As usual the MSE can be written as the sum of a squared bias term and a variance term

$$\mathbf{E}\Big[\|\hat{\theta} - \theta\|_2^2\Big] = \|\mathbf{E}\Big[(\hat{\theta} - \theta)\Big]\|_2^2 + \mathbf{E}\Big[\|\hat{\theta} - \mathbf{E}\Big[\hat{\theta}\Big]\|_2^2\Big]. \quad (61)$$

The bias of the MAP equals

$$\begin{aligned}
\mathbf{E}\Big[(\hat{\theta}_{MAP} - \theta)\Big] &= \mathbf{E}\Big[\hat{\theta}_{MAP}\Big] - \theta && (62)\\
&= \mathbf{E}\Big[\mathbf{E}\Big[\hat{\theta}_{MAP}|X\Big]\Big] - \theta && (63)\\
&= \mathbf{E}\Big[\tilde{S}^{-1}X^T\mathbf{E}\Big[r|X\Big]\Big] - \theta && (64)\\
&= \mathbf{E}\Big[\tilde{S}^{-1}X^TX\theta\Big] - \theta, && (65)
\end{aligned}$$

The second line follows from the law of total expectation (Johnson and Wichern, 2007) and the fourth follows from the fact $\mathbf{E}[r|X] = X\theta$.

From the law of total covariance, the variance can be written as

$$\begin{aligned}
\mathbf{E}\Big[\|\hat{\theta}_{MAP} - \mathbf{E}\Big[\hat{\theta}_{MAP}\Big]\|_2^2\Big] &= \mathbf{tr}\Big(\text{Cov}(\hat{\theta}_{MAP})\Big) && (66)\\
&= \mathbf{tr}\Big(\mathbf{E}\Big[\text{Cov}(\hat{\theta}_{MAP}|X)\Big] + \text{Cov}(\mathbf{E}[\hat{\theta}_{MAP}|X])\Big). && (67)
\end{aligned}$$

The term $\text{Cov}(\hat{\theta}_{MAP}|X)$ equals

$$\begin{aligned}
\text{Cov}(\hat{\theta}_{MAP}|X) &= \tilde{S}^{-1}X^T\text{Cov}(r|X)X\tilde{S}^{-1} && (68)\\
&= \tilde{S}^{-1}X^TX\tilde{S}^{-1}. && (69)
\end{aligned}$$

In the second line we use the fact that $\text{Cov}(r|X) = \mathbf{I}$. The term $\mathbf{E}[\hat{\theta}_{MAP}|X]$ was used to derive equation 65 and equals $\tilde{S}^{-1}X^TX\theta$.

Substituting the relevant quantities into equation 61, we find that the mean squared error of the MAP is

$$\begin{aligned}\mathbf{E}\Big[\|\hat{\theta}_{MAP}-\theta\|_2^2\Big] &= \|(\mathbf{E}\Big[\tilde{S}^{-1}X^TX\Big]-I)\theta\|_2^2 + \\ &\quad \mathbf{tr}\Big(\mathbf{E}\Big[\tilde{S}^{-1}X^TX\tilde{S}^{-1}\Big]+\mathrm{Cov}(\tilde{S}^{-1}X^TX\theta)\Big). \end{aligned} \tag{70}$$

The MSE of the MPELE can be computed in a similar fashion. The bias of the MPELE equals

$$\begin{aligned}\mathbf{E}\Big[(\hat{\theta}_{\mathrm{MPELE}}-\theta)\Big] &= \mathbf{E}\Big[\hat{\theta}_{\mathrm{MPELE}}\Big]-\theta && (71)\\ &= \mathbf{E}\Big[\mathbf{E}\Big[\hat{\theta}_{\mathrm{MPELE}}|X\Big]\Big]-\theta && (72)\\ &= \mathbf{E}\Big[(N+cp)^{-1}X^T\mathbf{E}\Big[r|X\Big]\Big]-\theta && (73)\\ &= (N+cp)^{-1}\mathbf{E}\Big[X^TX\theta\Big]-\theta && (74)\\ &= (N+cp)^{-1}N\theta-\theta. && (75)\end{aligned}$$

To derive the fourth line we have again used the fact $\mathbf{E}[r|X]=X\theta$ to show that $\mathbf{E}[\hat{\theta}_{\mathrm{MPELE}}|X]=(N+cp)^{-1}X^TX\theta$. The fifth line follows by the definition $\mathbf{E}\Big[X^TX\Big]=N\mathbf{I}$. To compute the variance we again use the law of total covariance which requires the computation of a term $\mathbf{E}\Big[\mathrm{Cov}(\hat{\theta}_{\mathrm{MPELE}}|X)\Big]$,

$$\begin{aligned}\mathbf{E}\Big[\mathrm{Cov}(\hat{\theta}_{\mathrm{MPELE}}|X)\Big] &= \mathbf{E}\Big[(N+cp)^{-1}X^T\mathrm{Cov}(r|X)X(N+cp)^{-1}\Big] && (76)\\ &= \mathbf{E}\Big[(N+cp)^{-1}X^TX(N+cp)^{-1}\Big] && (77)\\ &= (N+cp)^{-2}N\mathbf{I}. && (78)\end{aligned}$$

We use the fact that $\mathrm{Cov}(r|X)=\mathbf{I}$ to derive the second line and the definition $\mathbf{E}\Big[X^TX\Big]=N\mathbf{I}$ to derive the third.

Using the bias-variance decomposition of the MSE, equation 61, we find that the mean squared error of the MPELE estimator is

$$\mathbf{E}\Big[\|\hat{\theta}_{\mathrm{MPELE}}-\theta\|_2^2\Big] = \|(\frac{N}{N+cp}-1)\theta\|_2^2+\frac{1}{(N+cp)^2}\Big(Np+\mathbf{tr}\Big(\mathrm{Cov}(X^TX\theta)\Big)\Big). \tag{79}$$

The term $\mathbf{tr}\Big(\mathrm{Cov}(X^TX\theta)\Big)$ can be simplified by taking advantage of the fact that rows of X are i.i.d normally distributed with mean zero, so their fourth central moment can be written as the sum of outer products of the second central moments (Johnson and Wichern, 2007).

$$\begin{aligned}\mathbf{E}\Big[(X^TX)_{ij}(X^TX)_{kl}\Big] &= N\mathbf{E}\Big[X_{1i}X_{1j}X_{1k}X_{1l}\Big]+N(N-1)\delta_{ij}\delta_{kl} && (80)\\ &= N(\delta_{ij}\delta_{kl}+\delta_{ik}\delta_{jl}+\delta_{il}\delta_{jk})+N(N-1)\delta_{ij}\delta_{kl}. && (81)\end{aligned}$$

We then have

$$\mathbf{E}\Big[\|\hat{\theta}_{\mathrm{MPELE}}-\theta\|_2^2\Big] = \|(\frac{N}{N+cp}-1)\theta\|_2^2+\frac{1}{(N+cp)^2}\Big(Np(1+\|\theta\|_2^2)+N\|\theta\|_2^2\Big). \tag{82}$$

Without regularization ($c=0$) the MSE expressions simplify significantly:

$$\begin{aligned}\mathbf{E}\Big[\|\hat{\theta}_{MLE}-\theta\|_2^2\Big] &= \mathbf{tr}\Big(\mathbf{E}\Big[(X^TX)^{-1}\Big]\Big) && (83)\\ \mathbf{E}\Big[\|\hat{\theta}_{MELE}-\theta\|_2^2\Big] &= \frac{\theta^T\theta+p(\theta^T\theta+1)}{N}. && (84)\end{aligned}$$

(The expression for the MSE of the MLE is of course quite well-known.) Noting that $(X^TX)^{-1}$ is distributed according to an inverse Wishart distribution with mean $\frac{1}{N-p-1}\mathbf{I}$ (Johnson and Wichern, 2007), we recover equations 38 and 39.

For $c \neq 0$ we calculate the MSE for both estimators in the limit $N, p \to \infty$, with $0 < \frac{p}{N} = \rho < \infty$. In this limit equation 82 reduces to

$$\mathbf{E}\Big[\|\hat{\theta}_{\text{MPELE}} - \theta\|_2^2\Big] \quad \to \quad \frac{\rho + \theta^T\theta(c^2\rho^2 + \rho)}{(1+c\rho)^2}. \tag{85}$$

To calculate the limiting MSE value for the MAP we work in the eigenbasis of X^TX. This allows us to take advantage of the Marchenko-Pastur law (Marchenko, 1967) which states that in the limit $N, p \to \infty$ but $\frac{p}{N}$ remains finite, the eigenvalues of $\frac{X^TX}{N}$ converge to a continuous random variable with known distribution. We denote the matrix of eigenvectors of X^TX by O:

$$X^TX \quad = \quad OLO^T, \tag{86}$$

with the diagonal matrix L containing the eigenvalues of X^TX.

Evaluating the first and last term in the MAP MSE (equation 70) leads to the result

$$\begin{aligned}\|(\mathbf{E}\Big[\tilde{S}^{-1}X^TX\Big] - I)\theta\|_2^2 + \mathbf{tr}\Big(\text{Cov}(\tilde{S}^{-1}X^TX\theta)\Big) \quad &= \quad \|\theta\|_2^2 - 2\theta^T\mathbf{E}\Big[\tilde{S}^{-1}X^TX\Big]\theta \\ &\quad + \mathbf{E}\Big[\|\tilde{S}^{-1}X^TX\theta\|_2^2\Big].\end{aligned} \tag{87}$$

To evaluate the last term in the above equation, first note that

$$\tilde{S}^{-1}X^TX\theta \quad = \quad O(L + cpI)^{-1}LO^T\theta. \tag{88}$$

Abbreviate $D = (L + cpI)^{-1}L$, for convenience. Now we have

$$\begin{aligned}\mathbf{E}\Big[\|\tilde{S}^{-1}X^TX\theta\|_2^2\Big] \quad &= \quad \theta^T\mathbf{E}\Big[OD^2O^T\Big]\theta && (89)\\ &= \quad \mathbf{E}\Big[\theta^TOD^2O^T\theta\Big] && (90)\\ &= \quad \mathbf{E}\Big[\mathbf{tr}(D^2O^T\theta\theta^TO)\Big] && (91)\\ &= \quad \mathbf{tr}\ \mathbf{E}\Big[D^2O^T\theta\theta^TO\Big] && (92)\\ &= \quad \mathbf{tr}\ \mathbf{E}\Big[D^2\mathbf{E}\Big[O^T\theta\theta^TO\Big|D\Big]\Big] && (93)\end{aligned}$$

In the last line we have used the law of total expectation. Since the vector $O^T\theta$ is uniformly distributed on the sphere of radius $\|\theta\|_2^2$ given L, $\mathbf{E}[O^T\theta\theta^TO|D] = \frac{\|\theta\|_2^2}{p}\mathbf{I}$. Thus

$$\mathbf{E}\Big[\|\tilde{S}^{-1}X^TX\theta\|_2^2\Big] \quad = \quad \frac{\|\theta\|_2^2}{p}\mathbf{tr}\Big(\mathbf{E}\Big[\Big((L + cpI)^{-1}L\Big)^2\Big]\Big). \tag{94}$$

We can use similar arguments to calculate the second term in equation 87.

$$\begin{aligned}\mathbf{E}\Big[\theta^T\tilde{S}^{-1}X^TX\theta\Big] \quad &= \quad \mathbf{E}\Big[\theta^TODO^T\theta\Big] && (95)\\ &= \quad \mathbf{tr}\ \mathbf{E}\Big[D\Big]\frac{\|\theta\|_2^2}{p} && (96)\end{aligned}$$

Substituting this result and equation 94 into equation 87 we find

$$\begin{aligned}\|(\mathbf{E}\Big[\tilde{S}^{-1}X^TX\Big] - I)\theta\|_2^2 + \mathbf{tr}\Big(\text{Cov}(\tilde{S}^{-1}X^TX\theta)\Big) \quad &= \quad \|\theta\|_2^2 - 2\ \mathbf{tr}\ \mathbf{E}\Big[D\Big]\frac{\|\theta\|_2^2}{p} + \frac{\|\theta\|_2^2}{p}\mathbf{tr}\mathbf{E}\Big[D^2\Big] && (97)\\ &= \quad \frac{\|\theta\|_2^2}{p}\mathbf{E}\Big[\mathbf{tr}(D - \mathbf{I})^2\Big)\Big] && (98)\\ &= \quad \frac{\|\theta\|_2^2}{p}\mathbf{E}\Big[\mathbf{tr}\Big((L + cpI)^{-1}L - \mathbf{I})^2\Big)\Big]. && (99)\end{aligned}$$

Using the result given above and noting that $\mathbf{tr}\Big(\mathbf{E}\Big[\tilde{S}^{-1}X^TX\tilde{S}^{-1}\Big]\Big) = \mathbf{tr}\Big(\mathbf{E}\Big[(L+cp\mathbf{I})^{-2}L\Big]\Big)$, the MAP MSE can be written as

$$\mathbf{E}\Big[\|\hat{\theta}_{MAP}-\theta\|_2^2\Big] = \mathbf{tr}\Big(\mathbf{E}\Big[(L+cp\mathbf{I})^{-2}L\Big]\Big) + \frac{\|\theta\|_2^2}{p}\mathbf{E}\Big[\mathbf{tr}\Big((L+cpI)^{-1}L-\mathbf{I})^2\Big)\Big]. \tag{100}$$

Taking the limit $N, p \to \infty$ with $\frac{p}{N}$ finite,

$$\mathbf{E}\Big[\|\hat{\theta}_{MAP}-\theta\|_2^2\Big] \to \rho\mathbf{E}\Big[\frac{l}{(l+c\rho)^2}\Big] + \|\theta\|_2^2\mathbf{E}\Big[\Big(\frac{l}{(l+c\rho)}-1\Big)^2\Big], \tag{101}$$

where l is a continuous random variable with probability density function $\frac{d\mu}{dl}$ found by the Marchecko-Pastur law

$$\frac{d\mu}{dl} = \frac{1}{2\pi l\rho}\sqrt{(b-l)(l-a)}\mathcal{I}_{[a,b]}(l) \tag{102}$$

$$a\Big(\rho\Big) = (1-\sqrt{\rho})^2 \tag{103}$$

$$b\Big(\rho\Big) = (1+\sqrt{\rho})^2. \tag{104}$$

Using equation 102 we can numerically evaluate the limiting MAP MSE. The results are plotted in figure 2.

Figure 7 evaluates the accuracy of these limiting approximations for finite N and p. For the range of N and p used in our real data analysis ($\frac{p}{N} \sim 0.01, N \sim 10000$), the approximation is valid.

4.2 Computing the MPELE for an LNP model with Gaussian stimuli

The MPELE is given by the solution of equation 30, which in this case is

$$\hat{\theta}_{\text{MPELE}} = \arg\max_{\theta,\theta_0}\ \theta_0\sum_{n=1}^{N} r_n + (X\theta)^T r - N\mathbf{E}\Big[\exp(\theta_0)\exp(x^T\theta)\Big] - \log f(\theta). \tag{105}$$

Since $x\theta$ is Normally distributed, $x\theta \sim \mathcal{N}(0, \theta^TC\theta)$, we can analytically calculate the expectation, yielding

$$\hat{\theta}_{\text{MPELE}} = \arg\max_{\theta,\theta_0}\ \theta_0\sum_{n=1}^{N} r_n + (X\theta)^T r - N\exp(\theta_0)\exp\Big(\frac{\theta^TC\theta}{2}\Big) - \log f(\theta). \tag{106}$$

Optimizing with respect to θ_0 we find

$$\exp(\theta_0^*) = \frac{\sum_{n=1}^{N} r_n}{N}\exp(-\frac{\theta^TC\theta}{2}). \tag{107}$$

Inserting θ_0^* into equation 106 leaves the following quadratic optimization problem

$$\hat{\theta}_{\text{MPELE}} = \arg\max_{\theta}\ -\frac{\theta^T\Big(C\sum_{n=1}^{N} r_n\Big)\theta}{2} + \theta^TX^Tr - \log f(\theta). \tag{108}$$

Note that the first two terms here are quadratic; i.e., the EL-approximate profile likelihood is Gaussian in this case. If we use a Gaussian prior, $f(\theta) \propto \exp(\theta^TR\theta/2)$, we can optimize for $\hat{\theta}_{\text{MPELE}}$ analytically to obtain equation 31.

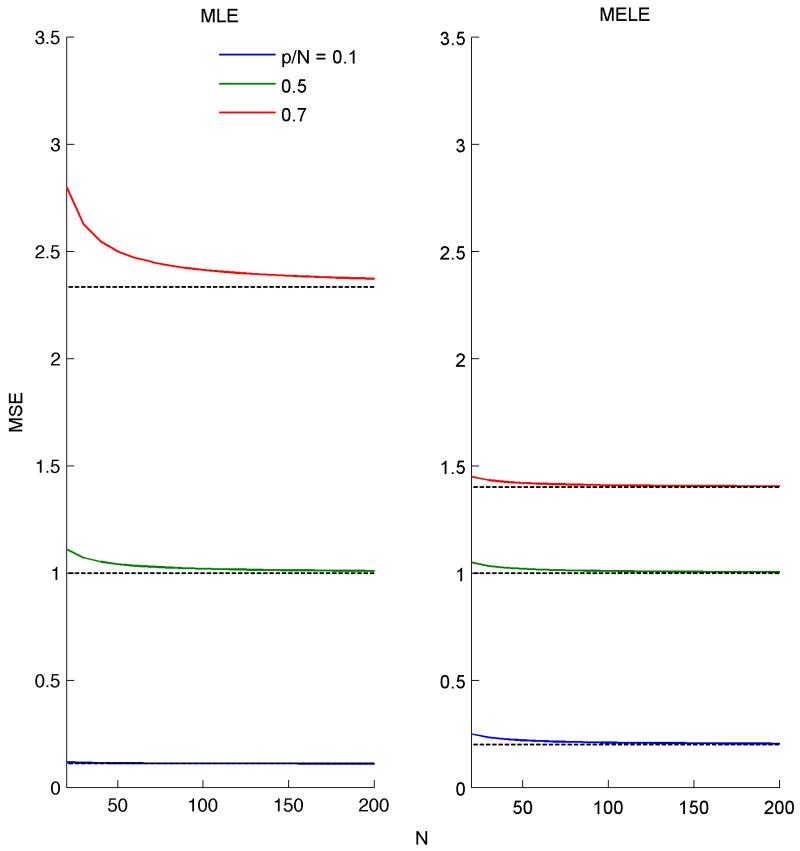

Figure 7: Finite sample comparisons to the limiting values of the MLE and MELE MSE as $p, N \to \infty$, $\frac{p}{N} \to \rho$. We plot the true mean-squared error (MSE) (colored lines) as a function of finite sample size for different values of $\frac{p}{N}$ for both the MLE (left column) and MELE (right column). Black dashed lines show limiting MSE when $p, N \to \infty$, $\frac{p}{N} \to \rho$. The quality of the approximation does not seem to depend on $\frac{p}{N}$ for the MELE and is within 1% accuracy after about 100 samples. For the MLE, this approximation depends on $\frac{p}{N}$. However, while the quality of the approximation depends on the estimator used, for data regimes common in neuroscience applications ($\frac{p}{N} \sim 0.01$, $N \sim 10000$) the limiting approximation is acceptable.

4.3 Calculating $\hat{R}$ for the LNP model

For the LNP model with $\mathbf{E}\Big[G(x^T\theta)\Big]$ approximated as in equation 16 and $f(\theta|R) = \mathcal{N}(0, R^{-1})$, the Laplace approximation (52) to the marginal likelihood yields

$$\begin{aligned}
\log F(R) &\approx \sum_{n=1}^{N} \hat{\theta}_0^* r_n + (X\hat{\theta}_{\text{MPELE}})^T r - N\Delta t \exp(\hat{\theta}_0^*) \exp\Big(\frac{(\hat{\theta}_{\text{MPELE}})^T C \hat{\theta}_{\text{MPELE}}}{2}\Big) + \log\Big(f(\hat{\theta}_{\text{MPELE}}|R)\Big) \\
&\quad -\frac{1}{2}\log(\det(-H(\hat{\theta}_{\text{MPELE}}))) \qquad (109) \\
&= -\frac{(\hat{\theta}_{\text{MPELE}})^T C \hat{\theta}_{\text{MPELE}}}{2} \sum_{n=1}^{N} r_n + (X\hat{\theta}_{\text{MPELE}})^T r - \frac{(\hat{\theta}_{\text{MPELE}})^T R \hat{\theta}_{\text{MPELE}}}{2} + \frac{1}{2}\log(\det(R)) \\
&\quad -\frac{1}{2}\log(\det(-H(\hat{\theta}_{\text{MPELE}}))) \qquad (110) \\
&= -\frac{(\hat{\theta}_{\text{MPELE}})^T (CN_s + R) \hat{\theta}_{\text{MPELE}}}{2} + (X\hat{\theta}_{\text{MPELE}})^T r + \frac{1}{2}\log(\det(R)) \\
&\quad -\frac{1}{2}\log(\det(-H(\hat{\theta}_{\text{MPELE}}))), \qquad (111)
\end{aligned}$$

where the second line follows from substituting in equation 107 and we have denoted $\sum_{n=1}^{N} r_n$ as N_s in the third line. Note that from the definition of the L2 regularized MPELE (equation 31) we can write

$$X^T r = (CN_s + R)\hat{\theta}_{\text{MPELE}}, \qquad (112)$$

and simplify the first two terms

$$\begin{aligned}
-\frac{(\hat{\theta}_{\text{MPELE}})^T (CN_s + R) \hat{\theta}_{\text{MPELE}}}{2} + (X\hat{\theta}_{\text{MPELE}})^T r &= \frac{(\hat{\theta}_{\text{MPELE}})^T (CN_s + R) \hat{\theta}_{\text{MPELE}}}{2} \\
&= \frac{(X^T r)^T (CN_s + R)^{-1} X^T r}{2}. \qquad (113)
\end{aligned}$$

Noting that $C = \mathbf{I}, R = \beta\mathbf{I}$ equation 113 simplifies further to $\frac{1}{2}\Big(N_s + \beta\Big)^{-1} q$ with $q \equiv \|X^T r\|_2^2$. Using the fact that the profile Hessian is $-H(\hat{\theta}_{\text{MPELE}}) = CN_s + R$ and the assumptions $C = \mathbf{I}, R = \beta\mathbf{I}$ the last two terms in equation 111 reduce to

$$\frac{1}{2}\log(\det(R)) - \frac{1}{2}\log(\det(-H(\hat{\theta}_{\text{MPELE}}))) = \frac{p}{2}\log(\beta) - \frac{p}{2}\log(N_s + \beta). \qquad (114)$$

Combining these results we find

$$\log F(R) \approx \frac{1}{2}\Big(N_s + \beta\Big)^{-1} q + \frac{p}{2}\log(\beta) - \frac{p}{2}\log(N_s + \beta). \qquad (115)$$

Taking the derivative of equation 115 with respect to β we find that the critical points, $\hat{\beta}_c$ obey

$$\begin{aligned}
0 &= -\frac{q}{(N_s + \hat{\beta}_c)^2} + p\frac{N_s}{(N_s + \hat{\beta}_c)\hat{\beta}_c}, \qquad (116) \\
\hat{\beta}_c &= \Big(\frac{N_s^2 p}{q - pN_s}, \infty\Big). \qquad (117)
\end{aligned}$$

If $p \geq \frac{q}{N_s}$, the only critical point is ∞ since β is constrained to be positive. When $p < \frac{q}{N_s}$, the critical point $\hat{\beta}_c = \frac{p}{\frac{q}{N_s^2} - \frac{p}{N_s}}$ is the maximum since $\log F$ (equation 115) evaluated at this point is greater than $\log F$ evaluated at ∞:

$$\log F\left(\frac{N_s^2 p}{q - pN_s}\right) = \frac{1}{2}\Big(\frac{q}{N_s} - p + p\log\Big(\frac{p}{q}N_s\Big)\Big) \geq 0 = \lim_{\beta\to\infty} \log F. \qquad (118)$$

Therefore $\hat{R}$ satisfies equation 54 in the text.

We can derive similar results for a more general case if C and R are diagonalized by the same basis. If we denote this basis by M, we then have the property that the profile Hessian $CN_s + R = M(D^c N_s + D^r)M^T$ where D^c and D^r are diagonal matrices containing the eigenvalues of C and R. In this case the last two terms of equation 111 reduce to

$$\frac{1}{2}\log(\det(R)) - \frac{1}{2}\log(\det(-H(\hat{\theta}_{\text{MPELE}}))) \quad = \quad \frac{1}{2}\sum_{i=1}^{p}\log\Big(\frac{D^r_{ii}}{D^c_{ii}N_s + D^r_{ii}}\Big). \tag{119}$$

Defining $X^T r$ rotated in the coordinate system specified by M as $\tilde{q} = M^T X^T r$, equation 113 simplifies to

$$\frac{(X^T r)^T (CN_s + R)^{-1} X^T r}{2} \quad = \quad \frac{1}{2}\sum_{i=1}^{p}\tilde{q}_i^2 (D^c_{ii}N_s + D^r_{ii})^{-1}. \tag{120}$$

Combining terms we find

$$\log F(R) \quad \approx \quad \frac{1}{2}\Big(\sum_{i=1}^{p}\tilde{q}_i^2 (D^c_{ii}N_s + D^r_{ii})^{-1} + \log\Big(\frac{D^r_{ii}}{D^c_{ii}N_s + D^r_{ii}}\Big)\Big). \tag{121}$$

Taking the gradient of the above equation with respect to the eigenvalues of R, D^{r*}_{jj}, we find that the critical points obey

$$0 \quad = \quad -\frac{\tilde{q}_j^2}{(D^c_{jj}N_s + D^{r*}_{jj})^2} + \frac{D^c_{jj}N_s}{D^{r*}_{jj}(D^c_{jj}N_s + D^{r*}_{jj})} \tag{122}$$

$$D^{r*}_{jj} \quad = \quad \Big(\frac{(D^c_{jj}N^s)^2}{\tilde{q}_j^2 - D^c_{jj}N_s}, \infty\Big). \tag{123}$$

If $D^c_{jj}N_s \geq \tilde{q}_j^2$, the only critical point is ∞ since D^{r*}_{jj} is constrained to be positive (R is constrained to be positive definite).

4.4 Real neuronal data details

Stimuli are refreshed at a rate of 120 Hz and responses are binned at this rate (figure 3) or at 10 times (figure 4) this rate. Stimulus receptive fields are fit with 81 (figure 3) or 25 (figure 4) spatial components and ten temporal basis functions, giving a total of 81x10 = 810 or 25x10=250 stimulus filter parameters. Five basis functions are delta functions with peaks centered at the first 5 temporal lags while the remaining 5 are raised cosine 'bump' functions (Pillow et al., 2008). The self-history filter shown in figure 4 is parameterized by 4 cosine 'bump' functions and a refractory function that is negative for the first stimulus time bin and zero otherwise. The coupling coefficient temporal components are modeled with a decaying exponential of the form, $\exp(-b\tau)$, with b set to a value which captures the time-scale of cross-correlations seen in the data. The errorbars of the spike-history functions in figure 4A show an estimate of the variance of spike-history function estimates. These are found by first estimating the covariance of the spike-history basis coefficients. Since the L1 penalty is non-differentiable we estimate this covariance matrix using the inverse log-likelihood Hessian of a model without coupling terms, say H_0^{-1}, evaluated at the MAP and MPELE solutions, which are found using the full model that assumes non-zero coupling weights. The covariance matrix of the spike-history functions are then computed using the standard formula for the covariance matrix of a linearly transformed variable. Denoting the transformation matrix from spike-history coefficients to spike-history functions as B, the covariance of spike-history function estimates is $B^T H_0^{-1} B$. Figure 4 plots elements off the diagonal of this matrix. We use the activity of 100 neighboring cells yielding a total of 100 coupling coefficient parameters, 5 self-history parameters, 250 stimulus parameters, and 1 offset parameter (356 parameters in total). The regularization coefficients used in figure 3B and 4 are found via cross-validation

on a novel two minute (14,418 samples) data set. Model performance is evaluated using 2 minutes of data not used for determining model parameters or regularization coefficients. To report the log-likelihood in bits per second, we take the difference of the log-likelihood under the model and log-likelihood under a homogeneous Poisson process, divided by the total time.

The covariance of the correlated stimuli was spatiotemporally separable, leading to a Kronecker form for C. The temporal covariance was given by a stationary AR(1) process; therefore this component has a tridiagonal inverse (Paninski et al., 2009). The spatial covariance was diagonal in the two-dimensional Fourier basis. We were therefore able to expoit fast Fourier and banded matrix techniques in our computations involving C.

Acknowledgements

We thank the Chichilnisky lab for kindly sharing their retinal data, C. Ekanadham for help obtaining the data, and E. Pnevmatikakis, A. Pakman, and W. Truccolo for helpful comments and discussions. We thank Columbia University Information Technology and the Office of the Executive Vice President for Research for providing the computing cluster used in this study. LP is funded by a McKnight scholar award, an NSF CAREER award, NEI grant EY018003, and by the Defense Advanced Research Projects Agency (DARPA) MTO under the auspices of Dr. Jack Judy, through the Space and Naval Warfare Systems Center, Pacific Grant/Contract No. N66001-11-1-4205.

References

Behseta, S., Kass, R., and Wallstrom, G. (2005). Hierarchical models for assessing variability among functions. *Biometrika*, 92:419–434.

Bickel, P. J. and Doksum, K. A. (2007). *Mathematical Statistics: Basic Ideas and Selected Topics*, volume 1. Pearson Prentice Hall, second edition.

Bishop, C. (2006). *Pattern Recognition and Machine Learning*. Springer.

Bottou, L. (1998). Online algorithms and stochastic approximations. In Saad, D., editor, *Online Learning and Neural Networks*. Cambridge University Press, Cambridge, UK.

Boyd, S. and Vandenberghe, L. (2004). *Convex Optimization*. Oxford University Press.

Boyles, L., Balan, A. K., Ramanan, D., and Welling, M. (2011). Statistical tests for optimization efficiency. In *NIPS*, pages 2196–2204.

Brillinger, D. (1988). Maximum likelihood analysis of spike trains of interacting nerve cells. *Biological Cyberkinetics*, 59:189–200.

Brown, E., Kass, R., and Mitra, P. (2004). Multiple neural spike train data analysis: state-of-the-art and future challenges. *Nature Neuroscience*, 7(5):456–461.

Calabrese, A., Schumacher, J. W., Schneider, D. M., Paninski, L., and Woolley, S. M. N. (2011). A generalized linear model for estimating spectrotemporal receptive fields from responses to natural sounds. *PLoS One*, 6(1):e16104.

Conroy, B. and Sajda, P. (2012). Fast, exact model selection and permutation testing for l2-regularized logistic regression. *Journal of Machine Learning Research - Proceedings Track*, 22:246–254.

Cossart, R., Aronov, D., and Yuste, R. (2003). Attractor dynamics of network up states in the neocortex. *Nature*, 423:283–288.

David, S., Mesgarani, N., and Shamma, S. (2007). Estimating sparse spectro-temporal receptive fields with natural stimuli. *Network*, 18:191–212.

Diaconis, P. and Freedman, D. (1984). Asymptotics of graphical projection pursuit. *The Annals of Statistics*, 12(3):793–815.

Donoghue, J. P. (2002). Connecting cortex to machines: recent advances in brain interfaces. *Nat Neurosci*, 5 Suppl:1085–8.

Fang, K. T., Kotz, S., and Ng, K. W. (1990). *Symmetric multivariate and related distributions*. CRC Monographs on Statistics and Applied Probability. Chapman and Hall.

Field, G. D., Gauthier, J. L., Sher, A., Greschner, M., Machado, T. A., Jepson, L. H., Shlens, J., Gunning, D. E., Mathieson, K., Dabrowski, W., Paninski, L., Litke, A. M., and Chichilnisky, E. J. (2010). Functional connectivity in the retina at the resolution of photoreceptors. *Nature*, 467(7316):673–7.

Friedman, J. H., Hastie, T., and Tibshirani, R. (2010). Regularization Paths for Generalized Linear Models via Coordinate Descent. *Journal of Statistical Software*, 33(1):1–22.

Gelman, A., Carlin, J. B., Stern, H. S., and Rubin, D. B. (2003). *Bayesian data analysis*. Chapman and Hall/CRC, 2nd ed. edition.

Golub, G. and van Van Loan, C. (1996). *Matrix Computations*. (Johns Hopkins Studies in Mathematical Sciences). The Johns Hopkins University Press, 3rd edition.

Golub, G. H., Heath, M., and Wahba, G. (1979). Generalized Cross-Validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223.

Johnson, R. A. and Wichern, D. W. (2007). *Applied multivariate statistical analysis*. Pearson Prentice Hall.

Kass, R. and Raftery, A. (1995). Bayes factors. *Journal of the American Statistical Association*, 90:773–795.

Lehmann, E. L. and Casella, G. (1998). *Theory of Point Estimation*. Springer, second edition.

Lewi, J., Butera, R., and Paninski, L. (2009). Sequential optimal design of neurophysiology experiments. *Neural Computation*, 21:619–687.

Lütcke, H., Murayama, M., Hahn, T., Margolis, D. J., Astori, S., Zum Alten Borgloh, S. M., Göbel, W., Yang, Y., Tang, W., Kügler, S., Sprengel, R., Nagai, T., Miyawaki, A., Larkum, M. E., Helmchen, F., and Hasan, M. T. (2010). Optical recording of neuronal activity with a genetically-encoded calcium indicator in anesthetized and freely moving mice. *frontiers in Neural Circuits*, 4(9):1–12.

Marchenko, V.A, P. L. (1967). Distribution of eigenvalues for some sets of random matrices. *Mat.Sb*, 72.

McCullagh, P. and Nelder, J. A. (1989). *Generalized Linear Models*. Chapman and Hall/CRC, second edition.

Minka, T. (2001). *A Family of Algorithms for Approximate Bayesian Inference*. PhD thesis, MIT.

Mishchenko, Y. and Paninski, L. (2011). Efficient methods for sampling spike trains in networks of coupled neurons. *The Annals of Applied Statistics*, 5(3):1893–1919.

Neal, R. (2012). MCMC using Hamiltonian dynamics. In Brooks, S., Gelman, A., Jones, G., and Meng, X., editors, *Handbook of Markov Chain Monte Carlo*. Chapman and Hall/CRC Press.

Nesterov, Y. (2004). *Introductory Lectures on Convex Optimization: A Basic Course*. Kluwer Academic Publishers, 1 edition.

Ohki, K., Chung, S., Ch'ng, Y., Kara, P., and Reid, C. (2005). Functional imaging with cellular resolution reveals precise micro-architecture in visual cortex. *Nature*, 433:597–603.

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

Paninski, L., Ahmadian, Y., Ferreira, D., Koyama, S., Rahnama, K., Vidne, M., Vogelstein, J., and Wu, W. (2009). A new look at state-space models for neural data. *Journal of Computational Neuroscience*, 29(1-2):107–126.

Paninski, L., Pillow, J., and Lewi, J. (2007). Statistical models for neural encoding, decoding, and optimal stimulus design. In Cisek, P., Drew, T., and Kalaska, J., editors, *Computational Neuroscience: Progress in Brain Research*. Elsevier.

Park, I. M. and Pillow, J. W. (2011). Bayesian spike-triggered covariance analysis. *NIPS*, 24.

Pillow, J. W., Shlens, J., Paninski, L., Sher, A., Litke, A. M., Chichilnisky, E. J., and Simoncelli, E. P. (2008). Spatio-temporal correlations and visual signalling in a complete neuronal population. *Nature*, 454(7207):995–999.

Rahnama Rad, K. and Paninski, L. (2011). Information rates and optimal decoding in large neural populations. In Shawe-Taylor, J., Zemel, R. S., Bartlett, P. L., Pereira, F. C. N., and Weinberger, K. Q., editors, *NIPS*, pages 846–854.

Rasmussen, C. and Williams, C. (2005). *Gaussian Processes for Machine Learning (Adaptive Computation and Machine Learning series)*. The MIT Press.

Robert, C. and Casella, G. (2005). *Monte Carlo Statistical Methods*. Springer.

Sadeghi, K., Gauthier, J., Greschner, M., Agne, M., Chichilnisky, E. J., and Paninski, L. (2013). Monte carlo methods for localization of cones given multielectrode retinal ganglion cell recordings. *Network*, 24:27–51.

Santhanam, G., Ryu, S. I., Yu, B. M., Afshar, A., and Shenoy, K. V. (2006). A high-performance brain–computer interface. *Nature*, 442(7099):195–198.

Shaffer, J. P. (1991). The gauss-markov theorem and random regressors. *The American Statistician*, 45(4):269–273.

Shewchuk, J. R. (1994). An introduction to the conjugate gradient method without the agonizing pain. Technical report, Carnegie Mellon University, Pittsburgh, PA, USA.

Shlens, J., Field, G. D., Gauthier, J. L., Grivich, M. I., Petrusca, D., Sher, A., Litke, A. M., and Chichilnisky, E. J. (2006). The structure of multi-neuron firing patterns in primate retina. *Journal of Neuroscience*, 26(32):8254–66.

Silverman, B. W. (1984). Spline smoothing: The equivalent variable kernel method. *The Annals of Statistics*, 12(3):pp. 898–916.

Simoncelli, E., Paninski, L., Pillow, J., and Schwartz, O. (2004). Characterization of neural responses with stochastic stimuli. In *The Cognitive Neurosciences*. MIT Press, 3rd edition.

Sollich, P. and Williams, C. K. I. (2005). Understanding gaussian process regression using the equivalent kernel. In *Proceedings of the First international conference on Deterministic and Statistical Methods in Machine Learning*, pages 211–228, Berlin, Heidelberg. Springer-Verlag.

Stevenson, I. H. and Kording, K. P. (2011). How advances in neural recording affect data analysis. *Nature Neuroscience*, 14(2):139–142.

Truccolo, W., Eden, U., Fellows, M., Donoghue, J., and Brown, E. (2005). A point process framework for relating neural spiking activity to spiking history, neural ensemble and extrinsic covariate effects. *Journal of Neurophysiology*, 93:1074–1089.

Truccolo, W., Hochberg, L. R., and Donoghue, J. P. (2010). Collective dynamics in human and monkey sensorimotor cortex: predicting single neuron spikes. *Nat Neurosci*, 13(1):105–11.

van der Vaart, A. (1998). *Asymptotic statistics*. Cambridge University Press, Cambridge.

Vidne, M., Ahmadian, Y., Shlens, J., Pillow, J. W., Kulkarni, J., Litke, A. M., Chichilnisky, E. J., Simoncelli, E., and Paninski, L. (2011). Modeling the impact of common noise inputs on the network activity of retinal ganglion cells. *Journal of Computational Neuroscience*.

Zhang, T. (2011). Adaptive Forward-Backward Greedy Algorithm for Learning Sparse Representations. *IEEE Transactions on Information Theory*, 57(7):4689–4708.