Robust learning of low dimensional dynamics from large neural ensembles

David Pfau[1], **Eftychios A. Pnevmatikakis**[1], **and Liam Paninski**[1]

[1]Columbia University

Summary

Activity in large neural populations can often be modeled in terms of simpler low-dimensional dynamics. The advent of large-scale neural recording techniques has motivated the development of algorithms for dimensionality reduction and latent dynamics estimation from single-trial data. However, many previous algorithms are slow or suffer from local optima. In this work we present a novel, more robust approach to simultaneous dimensionality reduction and linear dynamical system estimation from single-trial population spiking data.

Our approach is based on recent work in control theory and convex optimization and can be broken into two stages. In the first stage, an instantaneous firing rate is estimated directly from spike counts by use of convex *nuclear norm* penalized regression; this trades off between the likelihood of the data and the dimensionality of the latent dynamics. In the second stage, the firing rates are used to estimate the parameters of a linear dynamical system via standard regression-based approaches. The full model can be viewed as a low-dimensional latent linear system with point process outputs.

Because this approach is based on a convex optimization, it is not prone to local optima (unlike previous Expectation-Maximization approaches). Explicit assumptions on the state noise (e.g., Gaussianity) are avoided. Finally, this approach can incorporate any output model with a smooth concave loglikelihood, including generalized linear models, maximum-entropy models, dichotomized Gaussian models, and other population spike train models. We believe that the generality and broad applicability of our method makes it suitable for a wide variety of large-scale neuroscience problems.

Additional Details

In the simplest case, our model is of the form

$$\begin{aligned} \vec{x}_{t+1} &= A\vec{x}_t + B\vec{u}_t + \vec{\epsilon}_t \\ \vec{y}_t &= C\vec{x}_t \\ s_{i,t}|y_{i,t}, b_i &\sim \mathrm{Poiss}(f(y_{i,t} + b_i)) \end{aligned}$$

where $\vec{x}_t$ is a low-dimensional latent state, $\vec{u}_t$ is an optional (known) input, f is a monotonic non-negative nonlinearity, b_i is a bias term, and $s_{i,t}$ is the spike count for neuron i at time t. We use Poisson noise for the sake of simplicity, but emphasize that this can be replaced by any point process with a smooth concave loglikelihood. This model has been applied to neural data by a number of groups $[3, 6, 7]$, and has close connections other approaches such as jPCA $[2]$ and GPFA $[8]$. In $[5]$ the authors found that this model trained with EM was an excellent empirical model of neural data from monkey motor cortex.

The first stage of our method estimates the pre-nonlinearity firing rates $\vec{y}_t$ and $\vec{b}$ by performing the following optimization:

$$\min_{\vec{y}_{1:T}, \vec{b}} \lambda||\mathcal{A}(\vec{y}_{1:T})||_* - \log p(\vec{s}_{1:T}|\vec{y}_{1:T}, \vec{b}),$$

where $||\cdot||_*$ is the sum of singular values, or nuclear norm, and $\mathcal{A}(\cdot)$ is the linear operator that takes a matrix and stacks time-shifted copies of itself underneath, known as a *block-Hankel* matrix in the engineering literature.

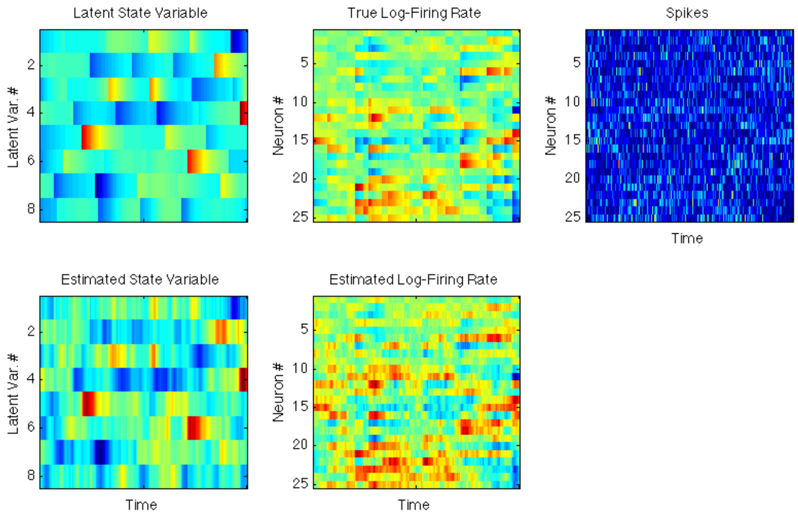

Figure 1: Latent dynamics with sparse (highly non-Gaussian) driving noise recovered from artificial data from 25 neurons driven by a 8-dimensional linear dynamical system. A subset of the observed population spiking data is shown on the right, with corresponding estimates of the log-rate (middle) and latent dynamics (left).

In the case where we have input data, $\mathcal{A}$ also includes right-multiplication by a matrix that depends on the input. For low-dimensional dynamics, the matrix $\mathcal{A}(y)$ is low-rank, and in particular when the innovation terms $\vec{\epsilon}_t$ are 0 (and the data has been generated from the above model) the rank of $\mathcal{A}(y)$ equals the dimensionality of the latent state. The minimization is performed by *alternating direction method of multipliers*, which was first applied to this problem with a quadratic term in place of the loglikelihood by [4]. In the absence of inputs, each iteration of ADMM scales linearly in the number of time bins and quadratically in the number of neurons. When inputs are present we utilize tools from numerical linear algebra (Kronecker products, Woodbury lemma), to keep our algorithm tractable. We believe this is the first application of this method to linear-nonlinear-Poisson outputs.

Another subspace identification method to fitting this model was proposed recently in [1], using a moment-matching approach. While the method proposed in [1] is faster than our proposed algorithm, it relies on the assumption that the state noise is Gaussian and thus appears less general. Another important difference is that our nuclear norm based method introduces an implicit bias towards linear dynamics with fewer states, which we believe could add robustness in the presence of high state noise power.

References

[1] Buesing L. et al, *Advances in Neural Information Processing Systems*, 25, 2012.
[2] Churchland M.M. et al, *Nature*, 2012.
[3] Kulkarni J.E. and Paninski L., *Network: Computation in Neural Systems*, 18(4):375-307, 2007.
[4] Liu Z. et al, *Systems and Control Letters*, submitted.
[5] Macke J.H. et al, *Advances in Neural Information Processing Systems*, 24, 2011.
[6] Smith A.C. and Brown E.N., *Neural Computation*, 15(5):965-991, 2003.
[7] Vidne M. et al, Journal of Computational Neuroscience, 33(1):97-121, 2012.
[8] Yu B.M. et al, Journal of Neurophysiology, 102(1):614-635, 2009.