Finding a Basis for the Neural State

Chris Cueva
ccueva@stanford.edu

I. INTRODUCTION

How is information represented in the brain? For example, consider arm movement. Neurons in dorsal premotor cortex (PMd) are selective for reach target [1], the type of grasp [2], reach speed [3], reach curvature [4], and the required accuracy [5]. Based on these experiments we might conclude that arm movement is represented in PMd as a combination of these features (along with some others that we haven't accounted for).

Example 2: Neurons in temporal cortex, especially the fusiform gyrus in humans [6] and superior temporal sulcus portion of IT cortex in macaque monkeys [7, 8], preferentially fire when the subject views pictures of faces . So the features that the neurons are encoding seem to be "face" and "not-face".

But how do we know what features the neurons are selecting for? The paradigm can be summarized as follows; "...one day...having failed to drive a unit with any light stimulus, we waved a hand at the stimulus screen and elicited a very vigorous response from the previously unresponsive neuron. We then spent the next 12 h testing various paper cutouts in an attempt to find the trigger feature for this unit" [9]. In other words, the experimenter presents many stimuli while simultaneously monitoring neuronal activity and then concludes by identifying similarities amongst stimuli that elicit a response [10]. This approach runs the risk of picking features that are conceptually simple to the experimenter but that have little to do with the features encoded by neurons. In other words, the representation used by neurons may not be simple or readily apparent.

The goal of this project is to tackle the question, "How is information represented in the brain?" using machine learning algorithms that remove experimental bias in feature selection. Let's make some assumptions about neuron firing rates and then state the goal more clearly.

II. MODELING

Modeling Assumption 1: The neural state only depends on the instantaneous firing rates, not relative spike times, of each of the neurons. So at any point in time we can completely specify the state by the instantaneous firing rates of each of the neurons.

Modeling Assumption 2: Each neuron fires with a Poisson distribution based on some underlying firing rate.

The goal of this project is to take spike time data from simultaneously recorded neurons and find the underlying rates that gave rise to these firing patterns. The number of underlying firing rates tell us approximately how many features are being represented. It will be left for a future project to determine what exactly the underlying firing rates represent.

If all of the neurons in the trial have the same underlying firing rate we can use various smoothing techniques to find this rate as shown in Figure 1. The acausal methods use information from the whole spike train, across all times, to compute the frequency at a specific time.

But what happens if the neurons do not have the same underlying firing rate? The normalized variance [3], defined in Equation (1) where r_{trial} is the firing rate on that trial and $\bar{r}$ is the mean firing rate across all trials, is 1 if poisson neurons have the same underling rate and greater than 1 if the underlying rates are different. The normalized variance is shown in the bottom panel of Figure 2 for 10,000 neurons simulated with the same underlying rate (black line, top panel) and 10,000 neurons each having different noisy versions of this underlying rate (red lines, top panel). Starting at 900 ms the noisy rates decay exponentially, with a time constant of 200 ms, to the underlying rate shown in black. Only 10 out of the 10,000 noisy rates are shown in the top panel of Figure 2. The spike times for 10 neurons from each group are shown in the middle panel.

$$\mathrm{NV}(t) = \frac{c}{\bar{r}(t)} \times \sum_{\text{trial}=1}^{n} \frac{(r_{\text{trial}}(t) - \bar{r}(t))^2}{n-1} \tag{1}$$

The normalized variance tells us when the neurons have different underlying rates but does not tell us what those rates are. Consider two neurons that have different underlying rates that are related at all points in time as shown by the black bar in Figure 3 [11]. The blue dots are the measured/noisy rates for the two neurons. The firing rate

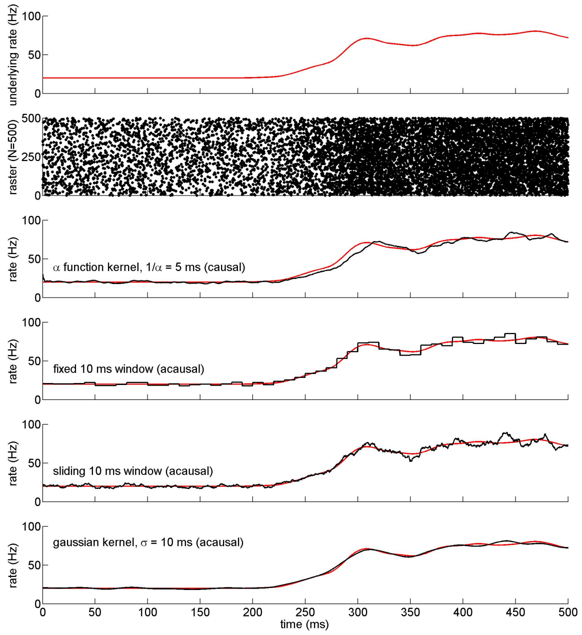

FIG. 1:

relationship between the two neurons is predicted by principle component analysis (PCA), red line, and factor analysis (FA), green line. The dotted black line shows the two SD covariance ellipse.

Neuron 1 has a higher mean firing rate and so has a higher variance than neuron 2 [12]. FA, which allows for different noise variances along different dimensions and seeks the dimension of greatest covariance, is better able to predict the firing rate relationship than PCA which seeks the dimension of greatest variance and so is biased towards neurons with higher firing rates.

A. Initial Conditions

FA Notation:

$$\mathbf{x} \sim \mathcal{N}(\mathbf{0}, I) \tag{2}$$

$$\mathbf{y}|\mathbf{x} \sim \mathcal{N}(\mu + \Lambda\mathbf{x}, \Psi) \tag{3}$$

where $\mathbf{x} \in \mathbb{R}^p$ and $\mathbf{y} \in \mathbb{R}^q$ such that $q > p$.

The slope of the FA fit is very sensitive to the initial values of the matrices Ψ and Λ. For example, in Figure 3 we can match the covariance matrix $\Lambda\Lambda^T + \Psi$ perfectly to the data even if we fix one of the parameters at an arbitrary value. For poisson spikes counted over a given interval the mean number of spikes equals the variance of spike counts. Each element of the diagonal matrix Ψ is the variance of one of the neurons so we initialize these values to the mean. If we initialize Λ with random normal elements EM requires ~3 times more iterations to converge than when initializing with an optimum value of Λ. We can reduce the number of iterations to ~1.5 times the number of optimum iterations if we provide a relevant scale by multiplying these random numbers by a factor of $\sqrt{\det(S)^{1/q}/p}$ where S is the covariance matrix of the data. However, the real problem is that Λ and Ψ converge to a local optimum that doesn't accurately predict the underlying firing rate. So given our value of Ψ let's take the matrix derivative of the log likelihood with respect to Λ, set it equal to zero, and find an optimum initial Λ. The method for doing this is borrowed from the first step of Jöreskog's algorithm [13]. There is not sufficient space to write the whole procedure but, to start, we note that maximizing the log likelihood

$$\ell(\mu, \Lambda, \Psi) = \ln \prod_{i=1}^{m} \frac{1}{(2\pi)^{q/2}|\Lambda\Lambda^T + \Psi|^{1/2}} \exp\left(-\frac{1}{2}(\mathbf{y}^{(i)} - \mu)^T(\Lambda\Lambda^T + \Psi)^{-1}(\mathbf{y}^{(i)} - \mu)\right) \tag{4}$$

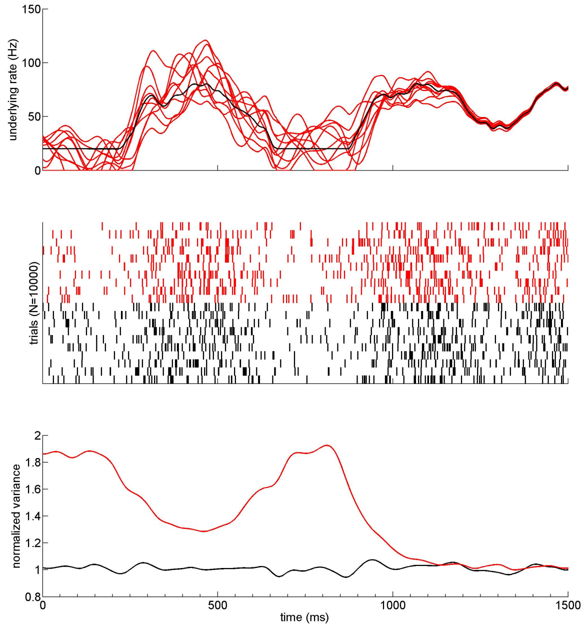

FIG. 2:

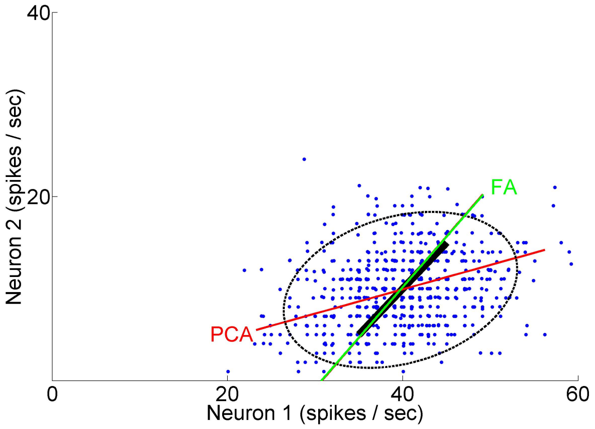

FIG. 3:

is equivalent to minimizing

$$\ell' = \ln|\Lambda\Lambda^T + \Psi| + \operatorname{tr}\left(S(\Lambda\Lambda^T + \Psi)^{-1}\right) - \ln|S| - q \tag{5}$$

where S is the covariance matrix of the data. The log likelihood for the EM algorithm implemented with this optimum initialization is shown in Figure 4.

Even given this optimum initialization I wasn't sure that the local optimum that the EM algorithm converged to was ideal for predicting the underlying rate so I also solved FA using the Newton-Raphson algorithm.

$$\Psi^{(t+1)} = \Psi^{(t)} - H^{-1}\nabla_\Psi \ell' \tag{6}$$

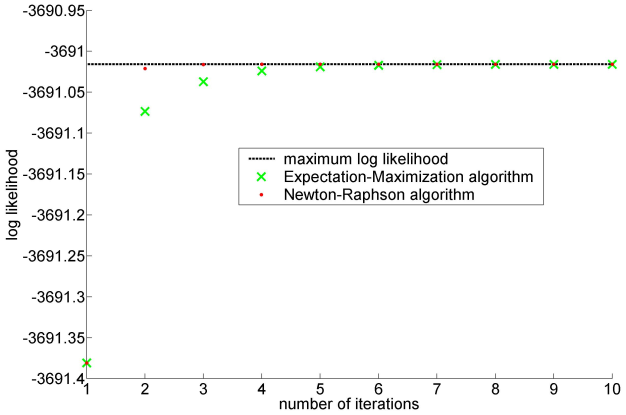

FIG. 4:

where H is the Hessian and

$$\frac{\partial \ell'}{\partial \Psi_i} = \mathrm{tr}\left[(\Lambda\Lambda^T+\Psi)^{-1}(\Lambda\Lambda^T+\Psi-S)(\Lambda\Lambda^T+\Psi)^{-1}\frac{\partial(\Lambda\Lambda^T+\Psi)}{\partial\Psi_i}\right] \tag{7}$$

$$\frac{\partial^2 \ell'}{\partial \Psi_i \Psi_j} = \mathrm{tr}\left[(\Lambda\Lambda^T+\Psi)^{-1}\frac{\partial(\Lambda\Lambda^T+\Psi)}{\partial\Psi_i}(\Lambda\Lambda^T+\Psi)^{-1}\frac{\partial(\Lambda\Lambda^T+\Psi)}{\partial\Psi_j}\right] \tag{8}$$

Both the EM and Newton-Raphson predictions for the underlying rate are plotted in Figure 3; given the same initial conditions both converge to the same Λ and Ψ. The log likelihood for the Newton-Raphson algorithm as a function of iteration number is shown in Figure 4. It converges faster than EM but is more computationally intensive.

III. GAUSSIAN-PROCESS FACTOR ANALYSIS [11]

If the underlying rate in Figure 4 increases/decreases continuously with time then FA has also solved the problem of picking out a “neural trajectory”, i.e. finding how the underlying rate changes over time.

For rates that change over time we would like to perform FA at each time point with the additional modeling assumption that the underlying firing rates vary smoothly. We will use Yu et al.’s gaussian-process factor analysis (GPFA) algorithm [11].

Assume we have spike counts from q neurons. Divide the time interval into T nonoverlapping bins. Let $\mathbf{y}_{i,t}$ be the square root of the number of spike counts for neuron i in time bin $t = 1, \ldots, T$. $\mathbf{y}_{:,t} \in \mathbb{R}^{q\times 1}$ is related to a low-dimensional latent neural state $\mathbf{x}_{:,t} \in \mathbb{R}^{p\times 1}$, $p < q$, via

$$\mathbf{y}_{:,t}|\mathbf{x}_{:,t} \sim \mathcal{N}(C\mathbf{x}_{:,t} + \mathbf{d}, R) \tag{9}$$

where the model parameters are $C \in \mathbb{R}^{q\times p}$, $\mathbf{d} \in \mathbb{R}^{q\times 1}$, and $R \in \mathbb{R}^{q\times q}$ which is taken to be diagonal with elements that are the independent variances of each neuron. To ensure that the neural state, $\mathbf{x}_{:,t}$ varies smoothly over time let

$$\mathbf{x}_{i,:} \sim \mathcal{N}(\mathbf{0}, K_i) \tag{10}$$

where $\mathbf{x}_{i,:} \in \mathbb{R}^{1\times T}$ and element (t_1, t_2) of the covariance matrix $K_i \in \mathbb{R}^{T\times T}$ is given by

$$K_i(t_1,t_2) = \sigma^2_{f,i}\exp\left(-\frac{(t_1-t_2)^2}{2\tau_i^2}\right) + \sigma^2_{n,i}\cdot\delta_{t_1,t_2}. \tag{11}$$

$\sigma^2_{f,i} \in \mathbb{R}_+$ and $\sigma^2_{n,i} \in \mathbb{R}_+$ are fixed constants and δ is the Kroneker delta function. The model parameters $C, \mathbf{d}, R, \tau_1, \ldots, \tau_p$ are found using the expectation maximization (EM) algorithm [14]. We can then vary p to find the optimum dimension for the underlying neural state. The neural states from all time points are grouped into a matrix $X = [\mathbf{x}_{:,1}, \ldots, \mathbf{x}_{:,T}] \in \mathbb{R}^{q\times T}$ which can be thought of as the “neural trajectory”. For example, the neural trajectory from the previous FA example is shown in Figure 5 in black.

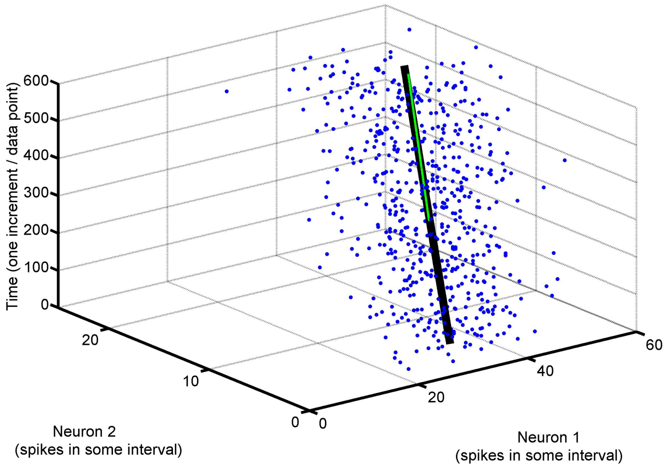

FIG. 5:

IV. TO DO

1. GPFA works because sufficiently small portions of the neural trajectory can be approximated by neurons having the same variance and tracing out straight paths moving in one direction. One way to extend this would be to allow different GPFA approximations to be stitched together by letting $C \rightarrow C_t, \mathbf{d} \rightarrow \mathbf{d}_t, R \rightarrow R_t$ in equation (9).

2. It can take several hours to learn the GPFA parameters. One possibility for improving speed is to implement GPFA using the algorithm recently proposed by Zhao et al. [15].

[1] J. Messier, J. F. Kalaska (2000) Covariation of primate dorsal premotor cell activity with direction and amplitude during a memorized-delay reaching task, *Journal of Neurophysiology*, 84, 152-165.

[2] M. Godschalk, R. N. Lemon, H. G. Kuypers, J. van der Steen (1985) The involvement of monkey premotor cortex neurones in preparation of visually cued arm movements, *Behav Brain Res*, 18, 143-157.

[3] M. M. Churchland, B. M. Yu, S. I. Ryu, G. Santhanam, K. V. Shenoy (2006) Neural variability in premotor cortex provides a signature of motor preparation, *Journal of Neuroscience*, 26 (14), 3697-3712.

[4] S. Hocherman, S. P. Wise (1991) Effects of hand movement path on motor cortical activity in awake, behaving rhesus monkeys, *Exp Brain Res*, 83, 285-302.

[5] J. E. Gomez, Q. Fu, D. Flament, T. J. Ebner (2000) Representation of accuracy in the dorsal premotor cortex, *Eur J Neurosci*, 12, 3748-37600.

[6] J. G. Ojemann, G. A. Ojemann, E. Lettich (1992) Neuronal activity related to faces and matching in human right nondominant temporal cortex, *Brain*, 115, 1-13.

[7] R. Desimone, T. D. Albright, C. G. Gross, C. Bruce (1984) Stimulus-selective properties of inferior temporal neurons in the macaque, *Journal of Neuroscience*, 4, 2051-2062.

[8] D. Y. Tsao, W. A. Freiwald, R. B. H. Tootell, M. S. Livingstone (2006) A Cortical Region Consisting Entirely of Face-Selective Cells, *Science*, 311, 670-674.

[9] C. G. Gross, C. E. Rocha-Miranda, D. B. Bender (1972) Visual Properties of Neurons in Inferotemporal Cortex of the Macaque, *Journal of Neurophysiology*, 35, 96-111.

[10] A. H. Bell, F. Hadj-Bouziane, J. B. Frihauf, R. B. H. Tootell, L. G. Ungerleider (2009) Object Representations in the Temporal Cortex of Monkeys and Humans as Revealed by Functional Magnetic Resonance Imaging), *Journal of Neurophysiology*, 101, 688-700.

[11] B. M. Yu, J. P. Cunningham, G. Santhanam, S. I. Ryu, K. V. Shenoy, M. Sahani (2009) Gaussian-Process Factor Analysis for Low-Dimensional Single-Trial Analysis of Neural Population Activity, *Journal of Neurophysiology*, 102, 614-635.

[12] P. Dayan, L. F. Abbott (2001) , *Theoretical Neuroscience*, cambridge, MA: MIT Press.

[13] K. G. Jöreskog (1967) Some contributions to maximum likelihood factor analysis, *Psychometrika*, 32, 443-82.

[14] A. P. Dempster, N. M. Laird, D. B. Rubin (1977) Maximum Likelihood from Incomplete Data via the EM Algorithm (with discussion), *Journal of the Royal Statistical Society. Series B*, 39 (1), 1-38.

[15] J. H. Zhao, P. L. H. Yu (2008) Fast ML Estimation for the Mixture of Factor Analyzers via an ECM Algorithm, *IEEE Transactions on Neural Networks*, 19 (11), 1956-1961.