Assignment 5
Theoretical Neuroscience

Maneesh Sahani

10 November 2012

1. **Encoding Models I: a linear neuron**

   In this question you will study the properties of a simulated linear neuron provided in the accompanying matlab file. The neuron generates a spike train in response to a stimulus $\mathbf{x}$ as an inhomogenous Poisson process with rate
   $$\mathbf{r} = b + \mathbf{w} \otimes \mathbf{x},$$
   where $\otimes$ represents convolution.

   (a) Find the impulse response of the neuron by collecting 200 responses to an impulse stimulus (a one followed by many zeros). This gives an estimate of both $\mathbf{w}$ and b.

   (b) Simulate the response of the neuron with a 1-sec Gaussian white noise stimulus sampled at a framerate of 100-Hz (i.e. with the output of `randn(1, 100)`). Generate 200 responses of the neuron to the *same* noise stimulus. Compute the PSTH of these responses, and show that it matches the rate prediction given by convolving the stimulus with $\mathbf{w}$.

   (c) Simulate the response to a long Gaussian white noise stimulus, and compute the STA (spike-triggered average). Plot the STA rescaled as a unit vector, and show that it provides a reasonable match to $\mathbf{w}/\|\mathbf{w}\|$.

   (d) Stimulate the model cell with correlated Gaussian noise: take the original (white) stimulus and filter it in time with a Gaussian-impulse filter whose standard deviation is 20ms. Rescale if necessary to ensure that the standard deviation of the new stimulus is the same as the old. Now simulate the neuron and compute the STA, and compare it to $\mathbf{w}$. Compute the de-correlated STA (obtained by "whitening" with the inverse of the stimulus covariance matrix), and compare with $\mathbf{w}$. If necessary, regularize by adding a small constant to the diagonal of the stimulus covariance matrix (this corresponds to doing "ridge regression"), and examine how this affects the estimate.

2. **Bussgang.**

   Prove Bussgang's Theorem. That is, show that if we have samples $\{\mathbf{x}_i, y_i\}$, where y_i is a random variable whose expectation is given by $E[y_i|\mathbf{x}_i] = f(\mathbf{w} \cdot \mathbf{x}_i)$, then the cross-correlation $\sum_i y_i \mathbf{x}_i$ (i.e. the "spike-triggered average" if y_i is binary) provides an unbiased estimate of $\alpha \mathbf{w}$ (i.e. $\mathbf{w}$ times an unknown constant α) if:

   (a) $P(\mathbf{x})$ is spherically symmetric, where we define spherical symmetry to mean that,
   $$\forall\, \mathbf{x}_1, \mathbf{x}_2 \in \mathbb{R}^n, \|\mathbf{x}_1\| = \|\mathbf{x}_2\| \Rightarrow P(\mathbf{x}_1) = P(\mathbf{x}_2).$$

   (b) $E[y\mathbf{x}] \neq \mathbf{0}$ (i.e. the expected spike-triggered average is not zero).

3. **Encoding Models II: LNP neurons**

   Simulate the response of an LNP (Linear-Nonlinear-Poisson) model to a temporal stimulus using `lnpNeuron.m`. This neuron has the same temporal response kernel as in part I above, but a soft threshold output nonlinearity.

   (a) Find the impulse response of the neuron by collecting 200 responses to an impulse stimulus (a one followed by many zeros). Note that this estimate is clipped.

   (b) Simulate the response to a long Gaussian white noise stimulus, and compute the STA (spike-triggered average). Plot the STA rescaled as a unit vector, and show that it provides a reasonable match to $\mathbf{w}/\|\mathbf{w}\|$.

   (c) Reconstruct the nonlinearity of the cell: begin by filtering the raw stimulus with the STA. Make a histogram of the filtered stimulus values, and make another histogram of the spike-triggered filtered stimulus values, using in the same binning. Divide the latter histogram by the former and multiply by the inverse of the bin size. Plot this estimate of the nonlinearity against the true f extracted from the code.

   (d) Now use `lnpNeuron2` which has a symmetric nonlinearity. Stimulate the new model neuron with an impulse and then with Gaussian white noise. Compute the STA and largest eigenvector of the spike-triggered covariance (STC) matrix. Compare with $\mathbf{w}$ obtained by the impulse approach. Reconstruct the nonlinearity using both STA/STC-derived filters, and compare with the true f.