Homework: encoding models
Theoretical Neuroscience
Jonathan Pillow
pillow@gatsby.ucl.ac.uk
October 27, 2006

1. Suppose we have a set of data $\{\vec{x}_i, y_i\}$ and we wish to fit a model that looks like:

$$y_i = k_0 + \vec{k}_1 \cdot \vec{x}_i + \vec{x}_i^T K_2 \vec{x}_i + \sigma N_i,$$

where the system kernels $\{k_0, \vec{k}_1, K_2\}$ represent a constant, a weight vector, and a square, symmetric weight matrix, respectively, σ is a scalar and N_i is standard i.i.d. Gaussian white noise (zero-mean, unit variance). Compute the maximum likelihood estimator for the kernels $\{k_0, \vec{k}_1, K_2\}$.

Optional: suppose that we know only that the N_i are independent and zero-mean (but not necessarily Gaussian). Now we present $\{\vec{x}_i\}$ drawn from a standard Gaussian distribution (zero-mean, identity covariance), and record corresponding values $\{y_i\}$. Show how to use the 0th, 1st and 2nd-order moments of $\{\vec{x}_i\}$ correlated against $\{y_i\}$ to obtain unbiased estimators for the system kernels (where the relevant moments are $\frac{1}{n}\sum_{1=1}^{n} y_i$, and $\frac{1}{n}\sum_{i=1}^{n} y_i \vec{x}_i$ and $\frac{1}{n}\sum_{i=1}^{n} y_i \vec{x}_i \vec{x}_i^T$).

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

1. $P(x)$ is spherically symmetric. (Where we define spherical symmetry to mean that, $\forall \vec{x}_1, \vec{x}_2 \in \mathbf{R}^n$, $||\vec{x}_1|| = ||\vec{x}_2|| \implies P(\vec{x}_1) = P(\vec{x}_2)$).
2. $E[xy] \neq 0$. (i.e. the expected spike-triggered average is not zero).

3. Simulate the response of an LNP (Linear-Nonlinear-Poisson) model to a temporal stimulus. Let $\vec{k}$ be a 20-tap filter (sampled in 10-msec bins) with biphasic temporal structure (i.e. a short, large-amplitude peak and a longer, smaller-amplitude trough). Choose the nonlinear response function f to be a sigmoid that saturates at 200 spikes/sec. Recall that the instantaneous rate for an LNP neuron is given by

$$R_i = f(\vec{k} \cdot \vec{x}_i)$$

and that Poisson spikes can be generated by flipping a biased coin in each (suitably small) time bin with the probability of "heads" equal to $(\Delta t)R_i$.

(A) Simulate the neuron with a 1-sec Gaussian white noise stimulus sampled at a framerate of 100-Hz. Generate 200 responses of the neuron to this stimulus. Compute the PSTH of these responses, and show that it matches the rate prediction given by convolving the stimulus with $\vec{k}$ and passing the output through f.

(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 $\vec{k}/||\vec{k}||$.

(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.

(D) Stimulate the model cell with correlated Gaussian white noise: take the original (Gaussian white noise) stimulus and filter it with a Gaussian 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 $\vec{k}$. Compute the de-correlated STA (obtained by "whitening" with the inverse of the stimulus covariance matrix), and compare with $\vec{k}$. If necessary, "regularize" by adding a small constant to the diagonal of the stimulus covariance matrix (this corresponds to "ridge regression"), and examine how this affects the estimate.

(E) Change f to be a symmetric function, such as $f(x) = \alpha x^2$. Simulate the new model neuron with Gaussian white noise, and compute the STA and largest eigenvector of the spike-triggered covariance (STC) matrix. Compare with $\vec{k}$. Reconstruct the nonlinearity using both filters, and compare with the true f.

(F) Decoding: estimate a linear decoder for the first LNP model described above (i.e. with sigmoidal nonlinearity). You should find a decoding filter $\vec{k}_d$ and a scalar b such that we can estimate the stimulus via: $\hat{x} = y * \vec{k}_{decode} + b$ (where '$*$' indicates linear convolution). Once you have estimated the filter and constant, apply them to responses to a novel stimulus segment, and plot the true stimulus x alongside the reconstructed stimulus $\hat{x}$. Create a temporally correlated stimulus whose temporal frequency structure is better matched to that of the encoding filter k. Re-estimate the decoding filter and constant, and show reconstruction of a novel stimulus segment with the same temporal correlation.

Optional: simulate several spike trains in response to the same stimulus. Compute a multi-neuron linear decoder that estimates the stimulus using the responses of all the neurons simultaneously. How does reconstruction with multiple responses compare to that obtained from a single response?

4. Grab the data file c1p8.mat from http://people.brandeis.edu/∼abbott/book/exercises.html. This data comes from a (visual motion sensitive) fly H1 neuron, and is described more thoroughly in questions 8-9 of the neural coding exercises from the Theoretical Neuroscience book, online at http://people.brandeis.edu/∼abbott/book/exercises/c1/c1.pdf.

Compute the STA, correlation-corrected STA, regularized STA, and the vectors corresponding to the largest and smallest eigenvalues of the spike-triggered covariance. Plot the nonlinearities corresponding to to each of these vectors. Compute the likelihood of the data under the LNP models constructed by appending each of these linear filters and corresponding nonlinearities. Which model assigns the highest likelihood to a set of held-out test data?