Quantitative Methods in Neuroscience

(NEU 466M)

Homework 3

Due: Thursday 02/15 by 5 pm in NHB 3.128 (or by 12:30pm in class if preferred)

In this assignment you will explore convolution and the STA. *General guidelines:* Read through each complete problem carefully before attempting any parts. Feel free to collaborate in groups of size 2-3, but always note the names of your collaborators on your submitted homework. For graphs: clearly label your axes and use good color and symbol choices. Print out your matlab code (in the form of a script file). For derivations you're asked to do 'by hand' (in other words, analytically, using paper and pencil) feel free to turn in handwritten or typed-out work.

1) **Identities for convolution.**

   a. Prove (by hand) that convolution is commutative, specifically, that $f * g(n) = g * f(n)$, where f, g are two infinite time-series. Use the definition of convolution provided in class. Remember that n is not a dummy index, while the summed-over index is. You will need to do a change-of-variables in the middle.

   b. By contrast, prove that correlation is not commutative, specifically, that $C_{fg}(n) = C_{gf}(-n)$.

   c. Prove that convolution is associative, specifically, that $f * (g * h)(n) = ((f * g) * h)(n)$.

2) **From spikes to rates: interspike intervals and convolution.** Sometimes we assume that detailed spike times don't carry information and that information is carried in the modulation of the underlying rate with which the neuron emits spikes instead. (The quality of this assumption varies from area to area in the brain: auditory neurons fire highly precisely timed action potentials locked to the phase of the impinging sound wave, while other sensory neurons like the stretch or touch receptors seem to convey stimulus intensity information in the overall modulation of the firing rates.) In such cases, we would like to estimate the firing rate of the neuron. In this problem, we'll consider different ways to generate firing rates from neural spikes. Download file `c1p8.mat` from the course webpage and assign `spks = rho(1:10000)`.

   a. Instantaneous spike rate: The instantaneous spike rate of a neuron equals the inverse of the inter-spike interval. To be specific, we define the instantaneous firing rate at any time as the inverse of the interval between the immediately preceding spike and the spike that immediately follows. For the spike train spks, generate the instantaneous spike rate (appropriately rescaled to Hz) for each time-point. (Hint: diff gives the list of intervals between spikes. The inverse of this value, rescaled to units of Hz, gives the instantaneous interspike rate for the full time-interval between corresponding pairs of spikes. Now simply assign this value to every time-point between those two spikes.) Plot the resulting rate, rate_isi, together with spks.

b. Rates from boxcar convolution. Construct a 25-ms wide boxcar kernel. Compute the convolution rate_boxcar_manual between spks and the kernel by writing a for loop, using the definition of convolution from class. Next, use the conv function from Matlab to do the same, and generate rate_boxcar. Rescale the two rates so they represent a firing rate in Hz. Plot spks, rate_boxcar_manual, and rate_boxcar on the same plot. Is the alignment correct?

c. Rates from convolution with a smooth Hanning window. Construct a 25-ms wide Hanning kernel (see hanning). Compute the convolution rate_hanning between spks using conv, convert the resulting rate to units of Hz. Plot it on top of spks: is the alignment correct? Why or why not? Use the argument same for conv and replot. Finally, make a new plot with spks, rate_isi rate_boxcar, and rate_hanning (different colors for each). Compare and contrast the features of the different resulting rate estimates.

3) **Using the spike-triggered average to estimate a neural kernel/solve the encoding problem.**

a. Download the file generate_STAdata.m from the course webpage. This .m file generates a time-varying stimulus stim the resulting spike train spks of a model neuron. This model neuron applies its own kernel h to the stimulus, to generate an underlying spike rate rsub, and finally outputs Poisson spikes based on the rate. Generate the STA of the spikes from the model neuron. Plot the STA (i.e., the estimated kernel of the neuron) and the true kernel h of the neuron on top of each other (making sure that you have chosen the correct part of the cross-correlation for the STA, the indices run in the correct direction, the time units are the same, and the heights of the curves are scaled appropriately). Describe what you see, and discuss why.

b. Double the length of the generated data in generate_STAdata.m and re-do the analysis. Does anything change in the STA estimate? Modify the kernel h in generate_STAdata.m, and re-do the analysis: is the STA (or inferred neural kernel) again a good match to the kernel you chose for the model neuron?

c. Now edit `generate_STAdata.m`, to set `whitenoise = 0`. How has the stimulus changed? Plot the autocorrelation of the new stimulus. Contrast it with the autocorrelation of the stimulus in part a. Re-do the STA analysis from part a. Is the estimated kernel a better or worse approximation to the true kernel? Show in a plot. Think about why this might be, and explain.