Neural Information Processing: 2008-2009
Assignment 1

School of Informatics, University of Edinburgh

Instructor: Mark van Rossum

16th February 2009

Remember that plagiarism is a university offence. Please read the policy at http://www.inf.ed.ac.uk/admin/ITO/DivisionalGuidelinesPlagiarism.html .

Practical information

You should produce a digital document for your assignment answers (e.g. with latex) and submit this electronically using the `submit` command on a DICE machine. The format is e.g.

```
submit msc nip 1 nipasst1.pdf
```

You can check the status of your submissions with the `show_submissions` command. NOTE: postscript or pdf formats are acceptable, other formats are not. Make sure that the file you submit prints ok on the DICE system, in particular when you produced it on a non-Unix machine.

Late submissions:

Late submissions will receive a zero mark. Only evidence for illness or other serious reasons can prevent this at the discretion of the instructor. See

```
http://www.inf.ed.ac.uk/teaching/years/msc/courseguide08.html\#exam
```

The handout "Introduction to MATLAB" available from the PMR web-page may be helpful if you are not very familiar with MATLAB. Recall that the current figure window can be saved as an encapsulated postscript file `myplot.eps` using the command `print -deps2 myplot.eps`.

LNP models with refractoriness

In this assignment we analyse LNP models of neurons. Instead of using real data, the neuron that we try to model is an integrate-and-fire model, the code of which is on the website. The input of the neuron is a one dimensional temporal stimulus. The output is the spike train.

In LNP models the probability of a spike in a time interval between t and $t + \Delta t$ is given by $p_{spike}(t) = f(K \star s)$, where $K \star s$ is the convolution of the stimulus with a to be fitted kernel, and $f()$ is a non-linearity. We a use rectifying non-linearity with adjustable threshold $f(x) = x - \theta$ for $x > \theta$ and $f(x) = 0$ otherwise. In theory the timebin $\Delta t \to 0$, but in practice we uses a finite value (1ms) as otherwise the problem is ill-posed.

In the lectures we used the cross-correlation between a white Gaussian stimulus and the response to determine the kernel. Here we use that for certain classes of the non-linearity, the optimal kernel can be found by maximising the log-likelihood. This is worked out in [Paninski, 2004] (paper on course website). Study section 1,2 and 6 of this paper.

Given a small enough time bin, the probability for a spike is $p_{spike}(t) = f(K \star s)$, and thus the probability $p_{no\,spike}(t) = 1 - f(K \star s)$. This leads to the log-likelihood given by the last equation on p. 245.

Question 1: Generate data using the integrate-and-fire code. Numerically find the optimal kernel that maximises the log-likelihood (last equation on p. 245) by varying the kernel elements. Use a timebin of 1ms and kernel length of about 10 time bins. Include one additional variable to be optimised that sets the threshold.
Plot the kernel found and comment on its shape.
Note: 1) You can use the Octave/Matlab function 'fminunc' to do multidimensional optimisation Take a reasonable initial condition for K. 2) You will not only need to restrict f to be between 0 and 1, but because of the logs the values 0 and 1 are also problematic. Recitify f such that it cannot reach those values.

Next, we include a refractory effect in the model. See section 6 of the paper.

Question 2: Derive the one before last equation on page 253, in a discrete time formulation (note, there is a typo in this equation...) assuming very small timebins such that $\log(1 - f) \approx f$.
Also derive an equation for r in the discrete time formulation without making this approximation.

Question 3: Use the one before last equation on p.253 to find the (time-discretized) refractory function r. Use about 20 1ms timebins.

Plot it and comment on its shape. How is it possible to see a refractory effect, while the integrate-and-fire neuron does not have a refractory period built in?
Note, the integrate-and-fire simulation script creates an array with Interspike Intervals, and an array with the time since the last spike. These can be helpful.

Question 4: Improve the model parameters by alternating finding the kernel K and determining r a few times (see bottom of p.254). Report the final values found for K and r.

Question 5: Plot the output of the LNP model averaged across a number of trials. Compare the result of the predicted firing rate with and without refractoriness to the actual spike train.

References

[Paninski, 2004] Paninski, L. (2004). Maximum likelihood estimation of cascade point-process neural encoding models. *Network*, 15(4):243–262.