STA 4503, Spring 2013 — Programming Assignment #2

Due in class on March 22. Please hand it in on 8 1/2 by 11 inch paper, stapled in the upper left, with no other packaging.

This assignment is to be done by each student individually. You may discuss it in general terms with other students, but the work you hand in should be your own. In particular, you should not leave any discussion with someone else with any written notes (either on paper or in electronic form).

For this assignment, you will write a program to sample from the posterior distribution of the Bayesian model you worked with for the first programming assignment using Hamiltonian Monte Carlo, and try it out on the same three datasets that you used for that assignment.

You can use the simple R implementation of a Hamiltonian Monte Carlo (HMC) update from my review paper (available from the course web page) as a starting point, if you wish. However, in this assignment, you will need to extend it in two ways. First, you should use a kinetic energy of the form $K(p) = \sum_i p_i^2/2m_i$, where m_i is the "mass" for coordinate i. These masses will be set manually, in order to provide better exploration of the distribution. Second, you should randomly choose a leapfrog stepsize for each update from the uniform distribution on the interval $(0.9\epsilon, 1.1\epsilon)$, where ϵ is the mean stepsize, which you will need to set manually. Randomly changing the stepsize avoids the possibility that the trajectories simulated will produce periodic updates (or nearly periodic updates) that do not move around the whole distribution.

To tune HMC, you will need to set the masses, m_i for $i = 1, \ldots, 5$, the mean stepsize, ϵ, and the number of leapfrog steps, L, by trial and error. You should consider whether the proposal standard deviations that you found worked well for the first programming assignment can be used to suggest masses for use with HMC. Whatever masses you use, you should then set ϵ so that a fairly high acceptance rate of about 0.7 is achieved.

As for the first programming assignment, for each dataset, you should look at trace plots of each of the five components and perhaps other quantities, to see how well HMC is sampling the posterior distribution. As in the first assignment, you should try several initial states, using different random seeds, in order to test whether the runs are actually sampling from the whole distribution, within the length of run that you do.

You should compare the performance of HMC updates with the single-variable Metropolis updates you did in the first assignment. For this to be a fair comparison, you will need to adjust for the relative computation time of the methods. You could do this by looking at actual computation time, or by counting evaluations of $\log \pi$ or its gradient (which one might consider to take about the same time). The latter method has the advantage of not depending on the details of how you wrote your program.

You should discuss the results you found with the three datasets, handing in suitable plots to illustrate your conclusions. You should also hand in your program and test scripts. The program may be in any language, though the simple HMC example program I provide is written in R.