# MCMC problems

Ben Lambert

## 1 Ticked off

Imagine you are investigating the occurrence of Lyme disease in the UK. This is a vector-borne disease caused by bacteria of species *Borrelia* which is carried by ticks. (The ticks pick up the infection by blood-feeding on animals/humans that are infected with *Borrelia*.) As such, you decide to estimate the prevalence of this bacteria in ticks you collect from the grasslands and woodlands around Oxford.

You decide to use sample sizes of 100 ticks, out of which you count the number of ticks testing positive for *Borrelia*. You decide to use a binomial likelihood since you assume that the presence of *Borrelia* in one tick is independent of that in other ticks. Also because you sample a relatively small area you assume that the presence of *Borrelia* can be assumed to be identically-distributed across ticks.

**Problem 1.1** *In a single sample you find that there are 6 ticks that test positive for Borrelia. Assuming a Beta(1,1) prior analytically calculate the posterior distribution. (Hint: by analytically here I mean look up the result on Google/in the lecture notes.) Graph this distribution.*

**Problem 1.2** *Generate 100 independent samples from this distribution using your software's inbuilt (pseudo-)random number generator. Graph this distribution. How does it compare to the pdf of the exact posterior? (Hint: in R the command is "rbeta"; in Matlab it is "betarnd"; in Mathematica it is "RandomVariate[BetaDistribution...]"; in Python it is "numpy.random.beta".)*

**Problem 1.3** *Evaluate the effect of increasing the sample size for your independent sampler on the estimate of the mean of the distribution. (Hint: for each sample you are essentially comparing the sample mean with the true mean of the posterior.)*

**Problem 1.4** *Estimate the variance of the posterior using independent sampling for a sample size of 100. How does your sample estimate compare with the exact solution?*

**Problem 1.5** *Create a proposal function for this problem that takes as input a current value of $\theta$, along with a step size, and outputs a proposed value. For a proposal distribution here we use a normal distribution centred on the current $\theta$*

*value with a standard deviation (step size) of 0.1. This means you will need to generate a random $\theta$ from a normal distribution using your statistical software's inbuilt random number generator. (Hint: the only slight modification you need to make here is to ensure that we don't get $\theta < 0$ or $\theta > 1$ is to use periodic boundary conditions. To do this we use modular arithmetic. In particular we set $\theta_{proposed} = mod(\theta_{proposed}, 1)$. The command for this in R is $x\%\%1$; in Matlab the command is $mod(x, 1)$; in Mathematica it is $Mod[x, 1]$; in Python it is $x\%1$.)*

**Problem 1.6** *Create the "accept/reject" function of Random Walk Metropolis that accepts as input $\theta_{current}$ and $\theta_{proposed}$ and outputs the next value of $\theta$. This is done based on a ratio:*

$$r = \frac{p(X|\theta_{proposed}) \times p(\theta_{proposed})}{p(X|\theta_{current}) \times p(\theta_{current})} \tag{1}$$

*and a uniformly-distributed random number between 0 and 1, which we call $a$. If $r > a$ then we update our current value of $\theta_{current} \rightarrow \theta_{proposed}$; alternatively we remain at $\theta_{current}$.*

**Problem 1.7** *Create a function that is a combined version of the previous two functions; so it takes as input a current value of $\theta_{current}$, generates a proposed $\theta_{proposed}$, and updates $\theta_{current}$ in accordance with the Metropolis accept/reject rule.*

**Problem 1.8** *Create a full-working Random Walk Metropolis sampler! (Hint: you will need to iterate the last function repeatedly. As such, you will need to decide on a starting position for $\theta$. I would recommend that you use a uniformly-distributed random number between 0 and 1.)*

**Problem 1.9** *For a sample size of 100 from your Metropolis sampler compare the sampling distribution to the exact posterior. How does the estimated posterior compare with that obtained via independent sampling using the same sample size?*

**Problem 1.10** *Run 1000 iterations, where in each iteration you run a single chain for 100 iterations. Store the results in a 1000 x 100 matrix. For each iterate calculate the sample mean. Graph the resultant distribution of sample means. How does MCMC do at estimating the posterior mean?*

**Problem 1.11** *Graph the distribution of the sample mean estimates of the for the second 50 observations of each chain. How does this result compare with that of the previous question? Why is there a difference?*

**Problem 1.12** *Decrease the standard deviation (step size) of the proposal distribution to 0.01. For a sample size of 200, how the posterior for a step size of 0.01 compare to that obtained for 0.1?*

**Problem 1.13** *Increase the standard deviation (step size) of the proposal distribution to 1. For a sample size of 200, how the posterior for a step size of 1 compare to that obtained for 0.1?*

**Problem 1.14** *Suppose we collect data for a number of such samples (each of size 100), and find the following numbers of ticks that test positive for Borrelia: (3,2,8,25). Either calculate the new posterior exactly, or use sampling to estimate it. (Hint: in both cases make sure you include the original sample of 6!)*

**Problem 1.15** *Generate samples from the posterior predictive distribution, and use these to test your model. What do these suggest about your model's assumptions?*