# Homework #7 "preview", Spring 2013 (due May 15th)

One parameter that you'll be estimating is a population size parameter.

Suppose that you are interested in trying out an MCMC proposal that has the following behavior:

- the current value of the population size is the expected value for the proposed state,
- when the current value of the population size is large, you tend to propose more drastic changes to the parameter.

This might explore the parameter space better than a sliding window move (in which the expected distance between the current state and the proposed state does not change over the course of an MCMC run).

You decide to go with a proposal in which the proposed state, $\theta^*$, is drawn from a gamma distribution with a shape=10 and scale=$\theta/10$ (where $\theta$ is the current value for the paramter).

## First part of the assignment

1. What is the Hastings ratio for this move?
2. Implement the move in your favorite MCMC program. If you just alter your previous assignment so that the log_likelihood function simply returns 1 for any combination of parameters that are within the feasible range, then your sampler should produce a simulation of draws from whatever prior distribution you choose for $\theta$.

## Notes

Confusingly, the gamma distribution can be described using "scale" and "shape" parameters or by "scale" and "rate" parameters. For the parameterization that I used above, the density is:

$$f(x) = \left(\Gamma(\text{shape})\text{scale}^{\text{shape}}\right)^{-1} x^{\text{shape}-1} e^{-\frac{x}{\text{scale}}}$$

That is a bit hard to read with words instead of single letters. So here it is more compactly:

$$\begin{aligned}
h &= \text{shape} \\
c &= \text{scale} \\
f(x) &= \left(\Gamma(h)c^h\right)^{-1} x^{h-1} e^{-\frac{x}{c}} \\
\text{mean}[x] &= hc \\
\text{Var}[x] &= hc^2
\end{aligned}$$

(continued on next page)

To generate a random draw from the gamma distribution in python with an expectation of 2 and variance of 0.4, we'd use:

```
import random
shape = 10
scale = 0.2
x = random.gammavariate(shape, scale)
```

In R, this would be:

```
x <- rgamma(shape=10, scale=0.2, n=1)
```

I haven't actually told you how to calculate the "gamma function", which is denoted $\Gamma(h)$ of the previous page. Can you see why you won't actually need to calculate that to implement the move?