Introduction to MCMC using `RevBayes`

Wade Dismukes, Tracy Heath, Walker Pett

1   Overview

This tutorial is intended to provide a introduction to the basics of Markov chain Monte Caro (MCMC) using the Metropolis-Hastings algorithm. This will provide a brief introduction to MCMC moves as well as prior distributions. We begin with a simple example of estimating the probability distribution of an archer's ability to shoot at a target, and the distance those arrows land from the center. We will simulate data using this example and attempt to estimate the posterior distribution using a variety of MCMC moves.

1.1   Learning Outcomes

- Understand and implement the Metropolis-Hastings MCMC algorithm
- Understand the difference and utility of various MCMC moves
- Begin to develop an intuition regarding the use of different priors

1.2   Required Software

This tutorial requires that you download and install the latest release of `RevBayes` (**?**), which is available for Mac OS X, Windows, and Linux operating systems. Directions for downloading and installing the software are available on the program webpage: http://revbayes.com.

The exercise provided also requires additional programs for editing text files and visualizing output. The following are very useful tools for working with `RevBayes`:

- A good text editor – if you do not already have one that you like, we recommend one that has features for syntax coloring, easy navigation between different files, line numbers, etc. Good options include `Sublime Text` or Atom, which are available for Mac OSX, Windows, and Linux.
- `Tracer` – for visualizing and assessing numerical parameter samples from `RevBayes`

2   Introduction

3 Modeling an archer's shots on a target

Introduction to problem. Include within here the distribution the shots are fired under and the first prior distribution. Might be worth showing some similar figures to the ones in Paul's Woods Hole powerpoint here to illustrate points. Also would be a a place to explain the idea of a conjugate prior and why they are useful.

3.1 Tutorial Format

This tutorial follows a specific format for issuing instructions and information.

> The boxed instructions guide you to complete tasks that are not part of the RevBayes syntax, but rather direct you to create directories or files or similar.

Information describing the commands and instructions will be written in paragraph-form before or after they are issued.

All command-line text, including all Rev syntax, are given in `monotype font`. Furthermore, blocks of Rev code that are needed to build the model, specify the analysis, or execute the run are given in separate shaded boxes. For example, we will instruct you to create a constant node called `rho` that is equal to `1.0` using the `<-` operator like this:

```
rho <- 1.0
```

It is important to be aware that some PDF viewers may render some characters given as `Rev commands` differently. Thus, if you copy and paste text from this PDF, you may introduce some incorrect characters. Because of this, we recommend that you type the instructions in this tutorial or copy them from the scripts provided.

3.2 Data and Files

> On your own computer or your remote machine, create a directory called RB_MyFirstMCMC_Tutorial (or any name you like).

In this tutorial we will be simulating our own data using RevBayes. Explain how to simulate data in Rev.

3.3 Getting Started with MCMC

Explain what MCMC does here.

3.3.1 Metropolis-Hastings algorithm

Go through outline of algorithm steps. explain proposal distributions. Then go into a step-by-step on how to write a MH-algorithm in Rev. First write the functions for the likelihood and the prior. Set the number of iterations, draw intitial values for prior. And then write the for loop defining the algorithm itself.

3.4 Using different priors

What happens when we change our priors. Why would we want to do that? What exactly are priors? Should explain these things here. Running under the prior will be explained here.

3.5 Defining different MCMC moves

Previously, we had been using random draws from (whatever) distribution for the prior. There are lots of possibilities for doing this. We can test how efficient these moves are at sampling our target distributions. Some are far more efficient than others. Code up all of the moves in RevBayes.

3.5.1 Random draw

introduce weights here. (maybe have an aside explaining using a 2 parameter model?)

3.5.2 Sliding move

Explain about δ (the tuning parameter). What values are good/bad for this? Explain about how to tune and the purpose of it.

3.5.3 Scaling move

Similarly, explain the tuning parameter for this move. Also, this move is asymmetric so explain more about the calculation of the Hastings ratio.

Compare the three different moves in tracer.

Version dated: August 23, 2017