

Practical 1. Monte Carlo in BUGS

Monday, 20 June 2022

[Lecture 1](#)
[PDF version](#)

1. Coins example

- Start OpenBUGS
- Load the file `coins.odc` from the appropriate folder — for example, this could look something `C:\bayes-hecourse\1_monte-carlo`. This program will simulate throws of 10 balanced coins and record which give 8 or more heads.
- First run the program from a script. Load the file `coins-script.odc` and check that the path to the working directory is appropriate. Check the script makes sense. With this script window open, click on `Info` → `Open Log`.

! NB: we use the notation `Command1` → `Command2` to indicate that you need to click on the menu labelled as `Command1` and then on the sub-menu labelled as `Command2`).

This opens a new window (containing the log of your BUGS session) — if you do not then you will not be able to see the results. Then `Model` → `Script`.

- Now try running using the interactive interface.
 - Open up the `Model` → `Specification` window.
 - Making sure that `coins.odc` is open, click on check model.
 - Then `compile` and `gen inits`.
 - Open up the `Model` → `Update` window and generate 1000 iterations.
 - Then open up the `Inference` → `Samples` window, type `Y` in the node window and then click `set`. This sets the monitor. Repeat for `P8`.
 - Type `*` in the node window to indicate all monitored quantities.
 - Click `trace` to generate traces of the simulated values.
 - Then do another 1000 updates.
 - `stats` then gives summary statistics.
- Assume now that the coin is actually unbalanced and the probability of a head is 0.7. Find the probability that 30 throws will show 15 heads or fewer.

i Hint: you could use the notation `step(15.5 - Y)` or `1 - step(Y - 15.5)`

2. Drug example

- Open the file `drug-MC.odc` and carry out a BUGS run for this model, obtaining the results shown in the lectures. You should be able to run it using the previous instructions (question 1) and the short list given in the lectures. Otherwise full details are given in *Running a model in OpenBUGS* of the “Hints on using OpenBUGS” chapter of this handout. If stuck, a script `drug-MC-script.odc` is available.
- Edit the model code to specify a `Uniform(0, 1)` prior on the response rate `theta`, and re-run the analysis.


i Note: the syntax for the uniform prior in BUGS is `dunif(a,b)` where `a` and `b` are the lower and upper bounds. The values of `a` and `b` can either be specified in the data file, or directly in the BUGS code (e.g. `a <- 1`), or just replace `a` and `b` by their values in the `dunif` statement).

- Plot the predictive distribution for the number of successes.
- What is now the predictive probability that 15 or more patients will experience a positive response out of 20 new patients affected?

3. Simulating functions of random quantities

- Write a model for a variable with a normal distribution with mean 0 and standard deviation 1 (remember that BUGS parameterises the normal in terms of precision = 1/Variance).
- Simulate 10000 values and plot their density.
- Simulate 10000 values of a variable Y with a normal distribution with mean 1 and standard deviation 2.
- For the same Y , create a new variable $Z = Y^3$, simulate 10000 values of Z , and find the mean and variance of Z , and the probability that Z is greater than 10. Are these results surprising?

- Plot the density of Z .

 If you are using the [binder VM](#), you will also have access to a series of **R** scripts that essentially replicate these analyses using **R** only (without accessing **BUGS**).

It may be helpful to use them, particularly to get even more familiar with the underlying concepts of simulations (which underpin Monte Carlo and Markov Chain Monte Carlo methods). If you want to use them, you can download them from [here](#).

NB: Instructions to install all the relevant packages that are available in the binder VM on your local machine are available [here](#).

