Scientific Programming Assignment 2

MPhil in Computational Biology

October 28, 2010

If there are errors found, I will update the assignment on the web at `http://www.damtp.cam.ac.uk/user/sje30/teaching/r`

Due date: 2010-11-10 17:00

Please submit your report to the CamTools website as a Sweave document, i.e. the Rnw file: NO OTHER FORMATS ARE ACCEPTED. Name your file `spa2_XXX.Rnw`, where XXX is your CRSid. (For example, I would save my file as `spa2_sje30.Rnw`.).

I will then run the commands to generate the PDF from the Sweave document.

This course work will consist of 20% towards your overall mark for this module.

1 Roulette [20 marks]

Complete question 22.2 (playing Roulette) from (Jones, Maillardet & Robinson, 2009), answering the questions that they set. (The question is appended at the end of this assignment.) In particular, your Sweave document must dynamically generate the tables requested in the question. (Tables can be included in Sweave using the *xtable* package; see `http://www.stat.umn.edu/~charlie/Sweave/foo.pdf` for examples. This package is installed on PWF Linux.

(You should assume that there is only one zero on the roulette wheel; i.e. there is no double zero.)

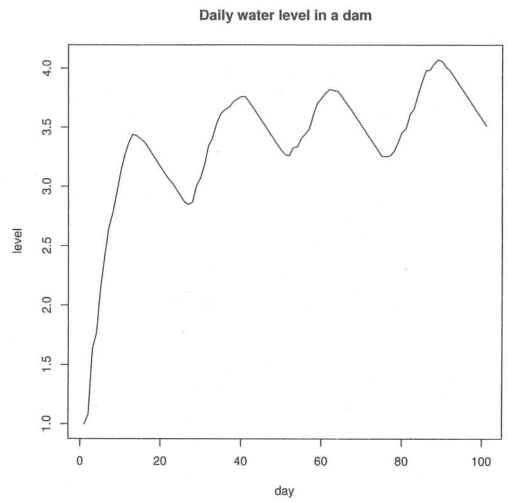

Figure 22.3 *Simulated time trace of water level for dam,* $h(1) = 1$.

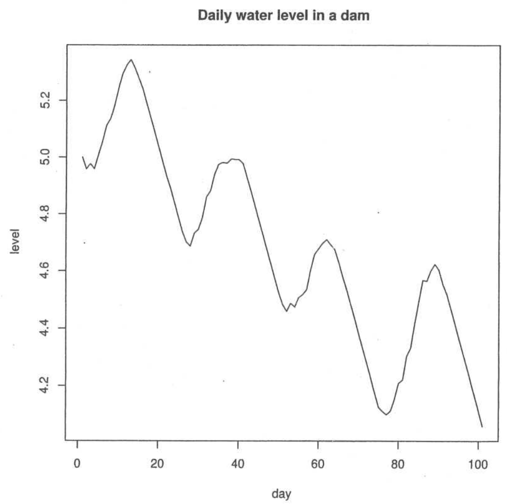

Figure 22.4 *Simulated time trace of water level for dam,* $h(1) = 5$.

22.2 Roulette

At the Crown Casino in Melbourne, Australia, some roulette wheels have 18 slots coloured red, 18 slots coloured black, and 1 slot (numbered 0) coloured green. The red and black slots are also numbered from 1 to 36. (Note that some of the roulette wheels also have a double zero, also coloured green, which nearly doubles the house percentage.)

You can play various 'games' or 'systems' in roulette. Four possible games are:

- A. Betting on Red

  This game involves just one bet. You bet \$1 on red. If the ball lands on red you win \$1, otherwise you lose.

- B. Betting on a Number

  This game involves just one bet. You bet \$1 on a particular number, say 17; if the ball lands on that number you win \$35, otherwise you lose.

- C. Martingale System

  In this game you start by betting \$1 on red. If you lose, you double your previous bet; if you win, you bet \$1 again. You continue to play until you have won \$10, or the bet exceeds \$100.

- D. Labouchere System

  In this game you start with the list of numbers (1, 2, 3, 4). You bet the sum of the first and last numbers on red (initially \$5). If you win you delete the first and last numbers from the list (so if you win your first bet it becomes (2,3)), otherwise you add the sum to the end of your list (so if you lose your first bet it becomes (1, 2, 3, 4, 5)). You repeat this process until your list is empty, or the bet exceeds \$100. If only one number is left on the list, you bet that number.

Different games offer different playing experiences, for example some allow you to win more often than you lose, some let you play longer, some cost more to play, and some risk greater losses. The aim of this assignment is to compare the four games above using the following criteria:

1. The expected winnings per game;
2. The proportion of games you win;
3. The expected playing time per game, measured by the number of bets made;
4. The maximum amount you can lose;
5. The maximum amount you can win.

22.2.1 Simulation

For each game write a function (with no inputs) that plays the game once and returns a vector of length two consisting of the amount won/lost and how many bets were made. Then write a program that estimates 1, 2, and 3, by simulating 100,000 repetitions of each game. Note that a game is won if you make money and lost if you lose money.

22.2.2 Verification

For games A and B, check your estimates for 1 and 2 by calculating the exact answers. What is the percentage error in your estimates for 100,000 repetitions?

For each game, work out the exact answers for 4 and 5. Of course, if this is not close to the answer given by your simulation, then you should suspect that either your calculation or your program is erroneous.

22.2.3 Variation

Repeat the simulation experiment of Part 22.2.1 five times. Report the minimum and maximum values for 1, 2, and 3 in a table as follows:

Game	Exp. winnings min–max	Prop. wins min–max	Exp. play time min–max
A			
B			
C			
D			

Modify your program from Part 22.2.1 so that in addition to estimating the expected winnings, expected proportion of wins, and expected playing time, it also estimates the *standard deviation* of each of these values. (You may use the built-in function `sd(x)` to do this.) For a single run, consisting of 100,000 repetitions of each game, report your results in a table as follows:

Game	Winnings mean, std dev	Prop. wins mean, std dev	Play time mean, std dev
A			
B			
C			
D			

For which game is the amount won most variable?

For which game is the expected playing time most variable?