Exercises for the DINA Research School Workshop on Monte Carlo methods for hierarchical models

Rasmus Waagepetersen
Institute of Mathematical Sciences
Aalborg University

October 25, 2003

1 Introduction

The exercises are supposed to be solved using either `R` or `BUGS`. Many of the exercises are one- or two-dimensional "toy examples" in order to keep the programming simple. This creates a bit of a paedagogical conflict in a course on Monte Carlo methods since in one dimension basically any method works while Monte Carlo methods prevail in high-dimensional problems.

2 Some comments on `R`

`R` is a free-ware version of Splus. A wide range of statistical methods are implemented in `R` and `R` is furthermore a very flexible programming language where new applications can easily be developed.

2.1 Entering commands

In your pc version commands can be executed via the command line. Longer sequences of commands or implementations of new functions can be written in a separate file. If the commands are stored in a file, `cmd.R` say, then the command line statement `source(‘‘cmd.R’’)` will execute the commands in `cmd.R` and load function code contained in `cmd.R`. You can also open `cmd.R` in your favourite text editor and use copy/paste to paste commands or sequences of commands into the command line.

2.2 Introduction and help

You may find it useful to run the commands in `basic.R` to learn more about some basic features of R. Help can be obtained by clicking the help-button. For help on a specific topic, say the function `scan`, you may alternatively just type `help(scan)` on the command line.

3 Exercises

You may either program the solutions from scratch or consider the programs in the `.R` or `.odc` files associated to each exercise. However, to encourage you to work actively with the code you will often need to either first correct bugs in the programs or modify the programs before you get the correct solution.

3.1 Exercises for Lecture I

1 (`exercise1.R`)
Suppose $U \sim N(0,1)$ and $Y|U = u \sim \text{Poisson}(\exp(\beta + U))$ (recall that $\text{Poisson}(\lambda)$ has density $f(y;\lambda) = \exp(-\lambda)\lambda^y/y!$). Assume that $Y = 8$ is observed.

a) In the same plot draw the density $f(u) = \exp(-u^2/2)/\sqrt{2\pi}$ of $N(0,1)$ and $f(8;\exp(\beta + u))$ as a function of u and for various β-values - why can the variance of $f(8;\exp(\beta + U)$ be large for some β ?. Also plot the product $f(8;\exp(\beta + u)f(u)$ of the two densities. Can you tell which values of β yield small likelihoods ?
b) Compute and plot the marginal likelihood of β using numerical quadrature (Use e.g. the `R` function `integrate`).
c) Compute and plot a Laplace-approximation of the likelihood (note that you obtain as a biproduct a normal distribution with mean and variance approximately equal to the conditional mean and variance of U given $Y = y$).
d) Compute and plot a simple Monte Carlo approximation of the likelihood and compute an estimate of the Monte Carlo error. Compare results obtained

with different numbers of simulations.
e) Repeat d) - c) but in the situations where 10 observations 8, 18, 5, 7, 10, 9, 9, 6, 7, 10 are available (i.e. $f(8; \exp(\beta + u))$ is replaced by the product

$$\prod_{i=1}^{10} [f(y_i; \exp(\beta + u))$$

of conditional densities for these observations). Note that we now have to take care of numerical problems since the product of 10 Poisson densities may take very small values. One possibility is to stabilize by dividing with the product of Poisson densities evaluated at $\hat{\lambda} = \bar{y}$ (the MLE). Does the accuracy of the Laplace approximation improve or worsen when more observations are available ?

2 (`exercise2.R`) (Pig growth)

The *R*-package `nlme` can be used to compute maximum likelihood estimates of parameters in non-linear mixed models. This package use a specialized numerical integration method called Gaussian quadrature.

a) Consider a Gompertz model

$$y = \exp(a - (a - \log(v0)) \exp(-k * t)) + \epsilon$$

for pig growth where k is the growth rate, t is time in days, $v0$ corresponds to the initial weight of pig, and ϵ is $N(0, \sigma^2)$ residual noise. Fit a Gompertz model with a random growth coefficent to the growth data (`growth.txt`). Also fit a Gompertz model with a non random growth coefficient. Why is the estimated residual variance larger for the latter model ?
b) Use the fitted models to simulate and plot growth curves for the five first pigs. Does the simulated curves resemble the data ?

3.2 Exercises for Lecture II

3 (`exercise3.R`) (continuation of Exercise 1)

This exercise considers importance sampling approximation of the likelihood using a t-distribution with mean and variance from the Laplace approximation obtained in Exercise 1 e).

a) In the same plot draw the density of the importance t-distribution and of the joint density

$$\exp(g(u)) = \prod_{i=1}^{10} [f(y_i; \exp(\beta + u))/f(y_i; \bar{y})] f(u)$$

of the observations and the random effect (here we have for numerical stability divided with $f(y_i; \bar{y})$). How does the importance sampling distribution depend on the value of β ?
b) Compute an importance sampling approximation of the likelihood. What happens if you use the importance sampling t-distribution obtained for, say $\beta = 4$ to compute the likelihood for all the other values of β ?
b) Compute the first and second order derivatives of the importance sampling approximation of the log likelihood and maximize the log likelihood using Newton-Raphson.

4 (`exercise4.R`) (continuation of Exercise 1)

Construct a random walk Metropolis sampler which simulates from $U|Y = y$. Compare the Monte Carlo estimate of the conditional distribution with the approximate normal distribution obtained in Exercise 1 c). A rule of thumb states that the optimal acceptance rate is around 25 %. Use different proposal variances (leading to different acceptance rates) and compare the estimated autocorrelations (use `acf` to compute autocorrelations).

5 (`exercise5.R`) (Gibbs sampler for bivariate normal)
Consider a bivariate normal distribution

$$(U_1, U_2) \sim N\big((0,0), \begin{bmatrix} 1 & \rho \\ \rho & 1 \end{bmatrix}\big)$$

Note that the conditional distributions of $U_1|U_2 = u_2$ and $U_2|U_1 = u_1$ are

$$N(\rho u_2, 1 - \rho^2) \text{ and}$$
$$N(\rho u_1, 1 - \rho^2),$$

respectively.

Run a Gibbs sampler for (U_1, U_2) for values of $\rho = 0, 0.5$, and 0.95. Plot the trajectory of the simulated pairs of (U_1, U_2). Compute autocorrelation for the simulated chain using `acf`. Why are the chains more auto-correlated (slowly mixing) when ρ is large ?

3.3 Exercises for Lecture III

6 (`simple_ex.odc`) (continuation of Exercise 1)

As a warm up for `BUGS`, try to simulate from the conditional distribution of $U|Y = y$ using `BUGS`.

7 (`cherries.odc,cherriesdat.odc`) (Bayesian analysis for cherries)

Perform a Bayesian analysis of the cherry data using `BUGS`. Assess convergence by considering timeseries and compute posterior means and variances of treatment effects and variance components.

3.4 Exercises for Lecture IV

8 (`exercise8.R`) (sensitivity analysis)

Use importance sampling to asses the sensitivity of the posterior results from Exercise 7 to the choice of prior.

9 (`cherriesbinary.odc`) (non-Gaussian random effects, continuation of Exercise 7)

Try to replace the Gaussian random branch effects with binary random effects. Try also to replace an observation with `NA` (so that it is missing) and then predict that observation with `BUGS`.