MT4531/5731: (Advanced) Bayesian Inference

The BUGS language for fitting statistical models (Nimble)

Nicolò Margaritella

School of Mathematics and Statistics, University of St Andrews

–

University of St Andrews

Outline

Outline

Introduction

- In the last chapter we described the ideas behind Bayesian statistics and saw a number of (simple) examples.
- However, real statistical problems are typically significantly more complex, most notably in terms of the number of unknown parameters to be estimated, resulting in high dimensional posterior distributions of complex and often of non-standard form.
- This was essentially the reason why the classical approach dominated statistics until the 1990s – Bayesian statistics were fine in theory, but intractable in practice.
- However, the increase in computational power, coupled with computational algorithms introduced to the statistical literature has made Bayesian analyses feasible (and often easier than alternative classical approaches!).

The BUGS language

- In this half of the course we describe two of these algorithms (Gibbs sampling and Metropolis-Hastings) and the computer language `BUGS` (Bayesian inference Using Gibbs Sampling).
- `Nimble` is an R package that allows users to fit complex statistical models using the BUGS language. (It does more, beyond the scope of this module).
- One of the aims of this module is for you to learn how to fit models using the BUGS language. `Nimble` will be demonstrated in lectures, with relevant material uploaded in Moodle.

Check Appendix B for downloading instructions etc.

- Appendix B contains instructions on what is required for running `Nimble` within R, other than R studio of course.
- **Task**: during the ILW you are required to install `Nimble` on your laptop and familiarise with the examples that will be discussed in class today.
- Note that your in course assignment will require `Nimble`!

The NIMBLE code

The NIMBLE code is composed of 4 separate components:

1. Model specification - containing the likelihood and prior specification in distribution form written in BUGS language;
2. Constants - model constants;
3. Data - input data; and
4. Initial starting value of the Markov chain, i.e. an initial starting value for each parameter.

Outline

Simple example - Exponential likelihood - Gamma prior

- It is perhaps easiest to demonstrate the BUGS language via examples.
- We start with a very simple conjugate analysis.
- Consider the 'Exponential likelihood - Gamma prior' example in Section 1.2, where...
- ... we observe 20 laptop lifetimes, $\boldsymbol{x} = \{x_1, \ldots, x_n\}$, such that, given λ, each $X_i \overset{iid}{\sim} Exp(\lambda)$.
- Note that $E(X_i|\lambda) = 1/\lambda$.
- In the relevant lecture we considered either a $\Gamma(0.2, 0.6)$ prior for λ, or a more informative $\Gamma(10, 30)$.
- The observed data are, $\boldsymbol{x} = \{4, 4, 4, 4, 4, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 6, 6, 6, 6, 6\}$, with $\bar{x} = 5$.

Example demonstration

- Please see the video demonstration of the relevant BUGS code during the lecture, for an implementation in `Nimble`.
- See also the relevant code that is uploaded in Moodle.

Example2 - Rats

- The data correspond to the weight of 30 rats on 5 different days (http://www.openbugs.net/Manuals/Contents.html). We can display the data in the form given in Table 1.

	Day (x_j)				
Rat	8	15	22	29	36
1	151	199	246	283	320
2	145	199	249	293	354
3	147	214	263	312	328
4	155	200	237	272	297
⋮					
30	153	200	244	286	324

Table: The weights of the rats at each time.

Rats - the model

- Let Y_{ij} denote the weight of rat $i = 1, \dots, 30$ at time $j = 1, \dots, 5$. We assume a very simple linear model, where weight is linearly regressed on time,

$$Y_{ij} \sim N(\alpha + \beta z_j, \sigma^2),$$

where α, β and σ^2 are parameters to be estimated; and z_j denotes the jth (normalised) time given by

$$z_j = \frac{x_j - \text{mean}(x)}{\text{sd(x)}}.$$

- (This is not a very realistic model! We will see different models later in the module).
- **Note:** *Centering* the data decreases dependence between α and β and improves interpretability; *normalising* the data helps also with parameters comparability in multivariate regression.

Rats - The priors

- We assume that we have no prior information on the parameters, and specify:

$$\alpha \sim N(0, 10^5); \qquad \beta \sim N(0, 10^5); \qquad \sigma^2 \sim \Gamma^{-1}(0.001, 0.001).$$

- Note that in BUGS the Normal distribution is parameterised via the precision ($\tau = 1/\sigma^2$).

Rats - demonstration

- See the relevant code uploaded to Moodle, and Section 2.1 of the lecture notes (part 2).