

UNIVERSITY OF WOLLONGONG  
School of Mathematics and Applied Statistics  
**STAT902. Advanced Data Analysis**  
**ASSIGNMENT 10**

**Due:** 5:00pm Monday 24th May, 2010.

Local students: please put under lecturer's door.

Remote students: please e-mail or post by this date.

1. Use Laplace's Method to approximate

(a)

$$J = \int_{-\infty}^{\infty} \frac{1}{x^4 - 12x^3 + 56x^2 - 120x + 100} dx,$$

(b)

$$K = \int_{-\infty}^{\infty} e^{3x - \frac{1}{4}x^2 - e^x} dx.$$

Hint: Laplace's method for (a) can be solved purely analytically (although an algebraic package such as `Maple` or `Mathematica` may help). For (b) numerical numerical root-finding methodology such as Newton-Raphson is required. (In addition,  $J$  also has an exact analytic solution, although you're not required to obtain it – unless you really want to, e.g. for checking purposes).

2. This question is concerned with approximate inference for GLMM based on Penalised Quasi-Likelihood (PQL).

- (a) Make sure that the file `seizure.txt` containing the seizure data, analysed in Assignment 9, is available.
- (b) Start an R session and issue the following commands:

```
library(nlme) ; library(MASS)
seizure <- read.table("seizure.txt",header=TRUE)
sszcntbas <- as.vector(scale(seizure$szcntbas))
sage <- as.vector(scale(seizure$sage))
fitPQL <- glmmPQL(szcnt~progabide+sszcntbas+sage,
                  random=~1|idnum,family="poisson",data=seizure)
print(summary(fitPQL))
print(intervals(fitPQL))
```

This performs a PQL-approximate fit of the model

$$[y_{ij}|U_i] \stackrel{\text{ind.}}{\sim} \text{Poisson}\{\exp(\beta_0 + U_i + \beta_1 \text{progabide}_{ij} + \beta_2 \text{sszcntbas}_{ij} + \beta_3 \text{sage}_{ij})\},$$
$$U_i \stackrel{\text{ind.}}{\sim} N(0, \sigma_U^2).$$

- (c) Compare the approximate 95% confidence intervals for  $\beta_1$  and  $\sigma_U$  with those obtained using `BRugs` and `WinBUGS` (Assignment 9, Question 3).

- (d) Issue the commands to obtain the PQL-approximate Fisher information matrix for  $\beta$ :

$$\mathbf{X}^T (\mathbf{W}^{-1} + \hat{\sigma}_U^2 \mathbf{Z} \mathbf{Z}^T)^{-1} \mathbf{X}$$

where  $\mathbf{W} = \text{diag}\{\exp(\mathbf{X}\hat{\beta} + \mathbf{Z}\hat{\mathbf{u}})\}$ .

```
betaHat <- fitPQL$coef$fixed
uHat <- unlist(fitPQL$coef$random)
sigmaSqHat.U <- (fitPQL$sigma^2)*exp(2*unlist(fitPQL$modelStruct))
X <- cbind(rep(1,nrow(seizure)),seizure$progabide,sszcntbas,sage)
Z <- kronecker(diag(59),rep(1,4))
etaHat <- as.vector(X%*%betaHat + Z%*%uHat)
invW <- diag(exp(-etaHat))
FishInfoPQL <- t(X)%*%solve(invW+sigmaSqHat.U*Z%*%t(Z),X)
print(FishInfoPQL)
```

- (e) Next, issue the commands:

```
invFishInfoPQL <- solve(FishInfoPQL)
print(invFishInfoPQL)
print(sqrt(diag(invFishInfoPQL)))
```

Identify a column of numbers in the output from (b) that most resembles the last set of numbers printed. (Unfortunately, I'm not getting an exact match here, so `glmPQL()` may be using a slightly different Fisher information matrix approximation compared with that calculated in (d). In 2008 I noted that this discrepancy may be due to estimation of the dispersion parameter  $\phi$ , but have not yet pursued this. )

3. Markov Chain Monte Carlo (MCMC) is a delicate art and research into its convergence performance has been ongoing since its introduction to Statistics in 1990. Consider the random intercept model used in the Assignment 9 for the pig weights data:

$$[y_{ij} | \beta_0, \beta_1, U_i, \sigma_U^2, \sigma_\varepsilon^2] \stackrel{\text{ind.}}{\sim} N(\beta_0 + U_i + \beta_1 x_{ij}, \sigma_\varepsilon^2)$$

$$[U_i | \sigma_U^2] \stackrel{\text{ind.}}{\sim} N(0, \sigma_U^2); \quad \beta_0, \beta_1 \stackrel{\text{ind.}}{\sim} N(0, 10^8), \quad \sigma_\varepsilon^2, \sigma_U^2 \stackrel{\text{ind.}}{\sim} \text{IG}(0.01, 0.01).$$

The journal article:

Gelfand, A.E., Sahu, S.K. and Carlin, B.P. (1995). Efficient parametrisations for normal linear mixed models. *Biometrika*, **82**, 479–488.

argues that MCMC convergence can be better if one works with the re-parametrised model:

$$[y_{ij} | \beta_1, \gamma_i, \sigma_U^2, \sigma_\varepsilon^2] \stackrel{\text{ind.}}{\sim} N(\gamma_i + \beta_1 x_{ij}, \sigma_\varepsilon^2)$$

$$[\gamma_i | \sigma_U^2] \stackrel{\text{ind.}}{\sim} N(\beta_0, \sigma_U^2); \quad \beta_0, \beta_1 \stackrel{\text{ind.}}{\sim} N(0, 10^8), \quad \sigma_\varepsilon^2, \sigma_U^2 \stackrel{\text{ind.}}{\sim} \text{IG}(0.01, 0.01).$$

Mathematically, the models are identical. The first one works with

$$\beta_0 + U_i, \quad [U_i | \sigma_U^2] \stackrel{\text{ind.}}{\sim} N(0, \sigma_U^2)$$

while the second one replaces  $U_i$  with  $\gamma_i = \beta_0 + U_i$  and works with

$$\gamma_i, \quad [\gamma_i | \sigma_U^2] \stackrel{\text{ind.}}{\sim} N(\beta_0, \sigma_U^2).$$

However, differences in the full conditional distributions implies differences in the MCMC fitting.

The second parametrisation is known as *hierarchical centring*.

- (a) Re-run the MCMC fitting of the original model using the code in `pigwtsBayes.Rs` and `pigwtsBayesModel.txt` from Assignment 9.
  - (b) Download the files `pigwtsBayesHC.Rs` and `pigwtsBayesModelHC.txt` from the **Computer Code and Data** page on the course web-site. These correspond to the hierarchically centred version of the model. Type `source("pigwtsBayesHC.Rs")` to fit this version of the model via MCMC.
  - (c) Comment on similarities and differences in the MCMC output, and include relevant graphics in your submission.
4. Consider, again, the random intercept model for the pigweights data. In Assignment 6 the random slope extension was fitted using maximum likelihood. The Bayesian version of the model is:

$$[y_{ij} | \beta_0, \beta_1, U_i, V_i, \boldsymbol{\Omega}] \stackrel{\text{ind.}}{\sim} N(\beta_0 + U_i + (\beta_1 + V_i) x_{ij}, \sigma_\varepsilon^2)$$

$$\left[ \begin{array}{c} U_i \\ V_i \end{array} \middle| \boldsymbol{\Omega} \right] \sim N \left( \begin{bmatrix} 0 \\ 0 \end{bmatrix}, \boldsymbol{\Omega}^{-1} \right)$$

$$\beta_0, \beta_1 \stackrel{\text{ind.}}{\sim} N(0, 10^8), \quad [\sigma_\varepsilon^2] \sim \text{IG}(0.01, 0.01) \quad \boldsymbol{\Omega} \sim \text{Wishart} \left( \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}, 2 \right).$$

Here  $\boldsymbol{\Omega}$  is a symmetric positive definite  $2 \times 2$  matrix and the Wishart distribution is defined by (for  $2 \times 2$  matrices  $\boldsymbol{\Sigma}$ ):

$$[\boldsymbol{\Sigma}; \mathbf{R}, k] \propto |\mathbf{R}|^{k/2} |\boldsymbol{\Sigma}|^{(k-3)/2} \exp\{-\frac{1}{2}\text{tr}(\mathbf{R}\boldsymbol{\Sigma})\}, \quad \boldsymbol{\Sigma} \text{ positive definite.}$$

- (a) Obtain the files `pigwtsBayesModelRS.txt` and `pigwtsBayesRS.Rs` from the **Computer Code and Data** page on the course web-site and use these to obtain MCMC-based fits of the above model.
- (b) Use `BRugs` and `WinBUGS` to fit the hierarchically centred version of the random intercepts and slopes model:

$$[y_{ij} | \beta_0, \beta_1, U_i, V_i, \boldsymbol{\Omega}] \stackrel{\text{ind.}}{\sim} N(\gamma_i + \delta_i x_{ij}, \sigma_\varepsilon^2)$$

$$\left[ \begin{array}{c} \delta_i \\ \gamma_i \end{array} \middle| \boldsymbol{\Omega} \right] \sim N \left( \begin{bmatrix} \beta_0 \\ \beta_1 \end{bmatrix}, \boldsymbol{\Omega}^{-1} \right)$$

$$\beta_0, \beta_1 \stackrel{\text{ind.}}{\sim} N(0, 10^8), \quad [\sigma_\varepsilon^2] \sim \text{IG}(0.01, 0.01) \quad \mathbf{\Omega} \sim \text{Wishart} \left( \begin{bmatrix} 0.01 & 0 \\ 0 & 0.01 \end{bmatrix}, 2 \right).$$

- (c) Is there any evidence that the random slope term significantly improves the fit? If so, then summarise it.

Hint: If templating from `pigwtsBayesModelRS.txt` and `pigwtsBayesRS.Rs` then make sure you change the `parInits` object to reflect the new parametrisation.