

# Durham/Newcastle Statistics Graduate Course

## Introduction to OpenBUGS *via* BRugs

M. Farrow  
School of Mathematics and Statistics  
Newcastle University

January 2011

## 1 Introduction

### 1.1 Introduction

The software BUGS (**B**ayesian **I**nference **U**sing **G**ibbs **S**ampling) was developed to allow users to specify models and priors, connect these with data and compute samples of unknowns from the posterior distribution using a Gibbs sampler (Spiegelhalter *et al*, 1995). Later a menu-driven version to run under MS Windows, called WinBUGS (Lunn *et al*, 2000) was developed. This eventually incorporated new features not found in the original, or ‘Classic’, BUGS. There is now also OpenBUGS, developed at the University of Helsinki, and various other implementations of the basic “BUGS” idea. In particular we will be using BRugs which is a R package which implements OpenBUGS within R. All of these use (apart from a few small differences) the same *Model Specification Language* and, in this part of this week’s course, it is this language which is of particular interest.

Various resources are available at the following Web page.

<http://www.mas.ncl.ac.uk/~nmf16/teaching/mas8303/>

This is the Web page for a module taught at Newcastle in which similar material is covered. References to the “Module Web Page” in what follows refer to this page.

The WinBUGS manual is available from the Module Web Page. The details of how you tell BRugs to do things are different from WinBUGS but the model specification language and many other features are the same.

A good overview is given by Lunn *et al* (2009).

#### 1.1.1 Potted history

**1989** BUGS Project starts at MRC Biostatistics Unit, Cambridge.

**1993** Unix version available. At first it used only Gibbs sampling (with adaptive rejection sampling where necessary) – hence the name. Metropolis-Hastings was added later.

**1996** Project moved to Imperial College.

**1997** Windows version: WinBUGS.

**2004** Work began at Helsinki University on the open-source version, OpenBUGS. This led to BRugs – interface which allows BUGS to run from R. (There are also interfaces to other statistical software now).

**2007** WinBUGS development work stopped. The future lies with OpenBUGS.

**2008** Competition: JAGS (“**J**ust **A**nother **G**ibbs **S**ampler”).

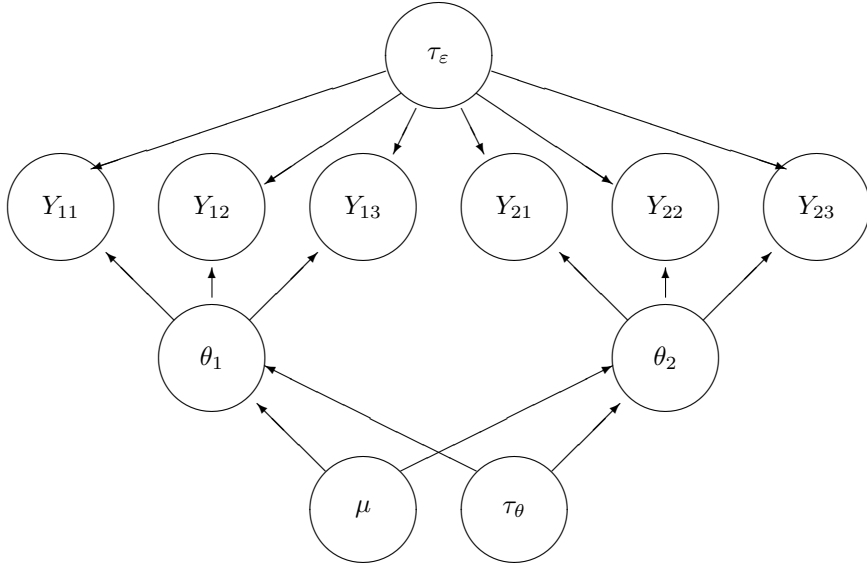


Figure 1: DAG: Repeated measures example.

## 1.2 Graphical models and the BUGS model specification language

We will be using BRugs to do practical work. This is a R package which implements OpenBUGS. Models and priors are specified using the BUGS language. We will need to make a *model specification* file using the BUGS language. The BUGS language is closely linked to the idea of *directed acyclic graphs* (DAG). At each node, we only need to specify the conditional distribution of the variable given the variables at the parent nodes (direct predecessors).

**Example 1** (Adapted from Gelfand *et al*, 1990).

Suppose we have  $k$  samples of observations and observation  $j$  in sample  $i$  is  $Y_{ij}$  where, given the values of  $\theta_i$  and  $\tau_\epsilon$ ,

$$Y_{ij} \sim N(\theta_i, \tau_\epsilon^{-1})$$

for  $j = 1, \dots, J$ , and, given the values of  $\mu$  and  $\tau_\theta^{-1}$ ,

$$\theta_i \sim N(\mu, \tau_\theta^{-1})$$

(all independent).

(This could be a repeated measure model, for example).

Suppose we have independent priors for the three parameters:

$$\mu \sim N(\mu_0, \tau_0^{-1})$$

$$\tau_\theta \sim \text{Ga}(a_1, b_1)$$

$$\tau_\epsilon \sim \text{Ga}(a_2, b_2)$$

where  $\text{Ga}(a, b)$  stands for a gamma distribution.

Figure 1 shows the model with  $k = 2$  and  $J = 3$ .

The BUGS model specification is shown in Figure 2.

**Example 2** Patients in four groups are observed for various lengths of time. During this time tumours may develop. The dependent variable is the number of tumours observed for each

```

model Gelfand

{for (i in 1:k)
  {for (j in 1:J)
    { y[i,j]~dnorm(theta[i],p.eps)
    }
    theta[i]~dnorm(mu,p.theta)
  }

mu~dnorm(mu0,p0)
p.theta~dgamma(a1,b1)
p.eps~dgamma(a2,b2)

sigmasq.theta<-1/p.theta
sigmasq.eps<-1/p.eps

}

```

Figure 2: BUGS model specification, repeated measures example.

patient. The mean number of tumours for a patient in group  $g$  is  $\lambda_g t = \exp(\beta_g + \ln t)$  where  $t$  is the time observed in weeks. Thus the parameters are  $\beta_1, \dots, \beta_4$  where  $\beta_g = \ln(\lambda_g)$ . There is no intercept here and the coefficient of  $\ln t$  is known to be 1.

Suppose that we observe  $n$  patients (written as `N` in the BUGS code). For each patient we have a group number  $g$  (`group`), the time  $t$  for which the patient was observed (`t`) and the number  $y$  of tumours observed (`y`).

Figure 3 shows some suitable BUGS code.

Note that the code in Figures 2 and 3 are *not* programs with commands to be executed. They are model specifications. We are defining the joint distribution of the unknowns and the data, mostly by specifying conditional distributions. For example

$$y[i] \sim \text{dpois}(\text{mean}[i])$$

might be written in standard mathematical notation as

$$Y_i \mid m_i \sim \text{Po}(m_i).$$

The code `dpois` represents the Poisson distribution and the symbol  $\sim$  has its usual meaning of “has the following distribution.” Similarly `dnorm` stands for a normal distribution. Note however that the parameters are mean and *precision*, not mean and variance. So, in the example, we are saying that

$$\beta_g \mid \mu \sim N(\mu, 0.1).$$

```

model tumours

{
  for (i in 1:N)
    {y[i]~dpois(mean[i])
     mean[i]<-lambda[group[i]]*t[i]
    }

  for (g in 1:4)
    {lambda[g]<-exp(beta[g])
     beta[g]~dnorm(mu,10)
    }
mu~dnorm(0,5)
}

```

Figure 3: BUGS code for the tumours example

## 2 BUGS functions

Function	Usage	Definition
Complementary log log	$cloglog(p) \leftarrow -a + b * x$ $y \leftarrow cloglog(p)$	$\log[-\log(1-p)] = a + bx$ $y = \log[-\log(1-p)]$
Logical equals	$y \leftarrow equals(x, z)$	$y = 1$ if $x = z$ $y = 0$ if $x \neq z$
Exponential	$y \leftarrow exp(x)$	$y = e^x$
Inner product	$y \leftarrow inprod(a[], b[])$	$y = \sum_i a_i b_i$
Matrix inverse	$y[, ] \leftarrow inverse(x[, ])$	$y = x^{-1}$ $y, x$ both $n \times n$ matrices
Natural logarithm	$log(lambda) \leftarrow -a + b * x$ $y \leftarrow log(x)$	$\log(\lambda) = a + bx$ $y = \log x$
Log determinant	$y \leftarrow logdet(x[, ])$	$y = \log  x $ $x$ is a $n \times n$ matrix
Log factorial	$y \leftarrow logfact(x)$	$y = \log(x!)$
Log(gamma function)	$y \leftarrow loggam(x)$	$y = \log[\Gamma(x)]$
Logit	$y \leftarrow logit(p)$ $logit(p) \leftarrow -a + b * x$	$y = \log[p/(1-p)]$ $\log[p/(1-p)] = a + bx$
Maximum	$c \leftarrow max(a, b)$	$c = \max(a, b)$
Mean	$x.bar \leftarrow mean(x[])$	$\bar{x} = \sum_i x_i / n$
Minimum	$c \leftarrow min(a, b)$	$c = \min(a, b)$
Standard Gaussian distribution function	$p \leftarrow phi(x)$	$p = \int_{-\infty}^x (2\pi)^{-1/2} e^{-t^2/2} dt$ i.e. $p = \Phi(x)$
Power	$z \leftarrow pow(x, y)$	$z = x^y$
Probit	$y \leftarrow probit(p)$ $probit(p) \leftarrow -a + b * x$	$y = \Phi^{-1}(p)$ $\Phi^{-1}(p) = a + bx$
Standard deviation	$s \leftarrow sd(x[])$	$s = \sqrt{\sum_i (x_i - \bar{x})^2 / n}$
Square root	$sigma \leftarrow sqrt(tau)$	$\sigma = \sqrt{\tau}$
Unit step	$y \leftarrow step(x)$	$y = 0$ if $x < 0$ $y = 1$ if $x \geq 0$
Sum	$x.sum \leftarrow sum(x[])$	$x_{sum} = \sum_i x_i$

### 3 BUGS distributions

Distribution	Usage	Definition
Bernoulli	<code>r~dbern(p)</code>	$f(r   p) = p^r(1-p)^{1-r};$ $r = 0, 1$
beta	<code>p~dbeta(a,b)</code>	$f(p   a, b) = \frac{\Gamma(a+b)}{\Gamma(a)\Gamma(b)}p^{a-1}(1-p)^{b-1};$ $0 < p < 1$
binomial	<code>r~dbin(p,n)</code>	$f(r   p, n) = \frac{n!}{r!(n-r)!}p^r(1-p)^{n-r};$ $r = 0, \dots, n$
categorical	<code>r~dcat(p[])</code>	$f(r   p_1, \dots, p_R) = p_r;$ $r = 1, 2, \dots, R$ where $R = \dim(\mathbf{p})$
chi-squared	<code>x~dchisq(k)</code>	$f(x   k) = 2^{-k/2}x^{k/2-1}e^{-x/2}/\Gamma(\frac{k}{2});$ $x > 0$
double exponential	<code>x~ddexp(mu,tau)</code>	$f(x   \mu, \tau) = \frac{\tau}{2}e^{-\tau x-\mu };$ $-\infty < x < \infty$
Dirichlet	<code>p[]~ddirch(alpha[])</code>	$f(\mathbf{p}   \alpha) = \frac{\Gamma(\sum_i \alpha_i)}{\prod_i \Gamma(\alpha_i)} \prod_i p_i^{\alpha_i-1};$ $0 < p_i < 1, \sum_i p_i = 1$
exponential	<code>x~dexp(lambda)</code>	$f(x   \lambda) = \lambda e^{-\lambda x};$ $x > 0$
gamma	<code>x~dgamma(r,mu)</code>	$f(x   r, \mu) = \mu^r x^{r-1} e^{-\mu x} / \Gamma(r);$ $x > 0$
lognormal	<code>x~dlnorm(mu,tau)</code>	$f(x   \mu, \tau) = \sqrt{\frac{\tau}{2\pi}}x^{-1} \exp[-\frac{\tau}{2}(\log x - \mu)^2];$ $x > 0$
logistic	<code>x~dlogis(mu,tau)</code>	$f(x   \mu, \tau) = \tau e^{\tau(x-\mu)} / (1 + e^{\tau(x-\mu)})^2;$ $-\infty < x < \infty$
multivariate normal	<code>x[]~dmnorm(mu[],T[,])</code>	$f(\mathbf{x}   \mu, \mathbf{T}) = (2\pi)^{-N/2}  \mathbf{T} ^{1/2} \exp[-\frac{1}{2}(\mathbf{x} - \mu)' \mathbf{T}(\mathbf{x} - \mu)];$ $-\infty < x_i < \infty$
multinomial	<code>x[]~dmulti(p[],N)</code>	$f(\mathbf{x}   \mathbf{p}, N) = \frac{(\sum_i x_i)!}{\prod_i x_i!} \prod_i p_i^{x_i};$ $\sum_i x_i = N$
negative binomial	<code>x~dnegbin(p,r)</code>	$f(x   p, r) = \frac{(x+r-1)!}{x!(r-1)!} p^r (1-p)^x;$ $x = 0, 1, 2, \dots$
normal	<code>x~dnorm(mu,tau)</code>	$f(x   \mu, \tau) = \sqrt{\frac{\tau}{2\pi}} \exp[-\frac{\tau}{2}(x - \mu)^2];$ $-\infty < x < \infty$
Pareto	<code>x~dpar(alpha,c)</code>	$f(x   \alpha, c) = \alpha c^\alpha x^{-(\alpha+1)};$ $x > c$
Poisson	<code>r~dpois(lambda)</code>	$f(r   \lambda) = e^{-\lambda} \frac{\lambda^r}{r!};$ $r = 0, 1, \dots$
Student's <i>t</i>	<code>x~dt(mu,tau,k)</code>	$f(x   \mu, \tau, k) = \frac{\Gamma((k+1)/2)}{\Gamma(k/2)} \sqrt{\frac{\tau}{k\pi}} [1 + \frac{\tau}{k}(x - \mu)^2]^{-(k+1)/2};$ $-\infty < x < \infty$
uniform	<code>x~dunif(a,b)</code>	$f(x   a, b) = 1/(b - a);$ $a < x < b$
Weibull	<code>x~dweib(v,lambda)</code>	$f(x   v, \lambda) = v \lambda x^{v-1} \exp(-\lambda x^v);$ $x > 0$
Wishart	<code>x[, ]~dwish(R[, ],k)</code>	$f(\mathbf{x}   \mathbf{R}, k) \propto  \mathbf{R} ^{k/2}  \mathbf{x} ^{(k-p-1)/2} \exp(-\frac{1}{2} \text{tr}[\mathbf{R}\mathbf{x}]);$ $\mathbf{x}$ symmetric and positive definite

## 4 Some BRugs (and other) Commands

```
library(BRugs)

modelCheck("modelfile.txt")

modelData("datafile.txt")

modelCompile()
modelCompile(2)

modelInits("initsfile.txt")

modelGenInits()

modelUpdate(5000)

samplesSet("theta")
samplesSet(c("alpha", "beta", "gamma"))

samplesHistory("theta")

samplesDensity("theta")

samplesStats("theta")
samplesStats(c("alpha", "beta", "gamma"))

thetavalues<-samplesSample("theta")

iter<-1:length(thetavalues)
plot(iter, thetavalues, type="l", xlab="Iteration", ylab=expression(theta))

thetadens<-density(theta, adjust=1.5)
plot(thetadens$x, thetadens$y, type="l", xlab=expression(theta), ylab="Density")
```

## References

- Box, G.E.P. and Jenkins, G.M., 1976. *Time Series Analysis: Forecasting and Control*, Revised edition. San Francisco: Holden-Day.
- Cooper, C., Shah, S., Hand, D.J., Adams, J., Compston, J., Davie, M. and Woolf, A., 1991. Screening for vertebral osteoporosis using individual risk factors. *Osteoporosis International*, **2**, 48-53.
- Cowburn, G.J., 2003. *Bayesian mixture modelling with application to road traffic flow*. Ph.D. thesis, University of Sunderland.
- Cowburn, G.J. and Farrow, M., 2007. Mixtures and diagnostic plots in modelling road traffic vehicle headways. *Statistical Modelling*, **7**, 73-89.
- Davies, O.L. and Goldsmith, P.L. (eds), 1972. *Statistical Methods in Research and Production*, 4th edition. Edinburgh: Oliver and Boyd.
- Freeman, D.H., 1987. *Applied Categorical Data Analysis*. New York: Marcel Dekker.

- Gelfand, A.E., Hills, S.E., Racine-Poon, A. and Smith, A.F.M., 1990. Illustration of Bayesian inference in normal data models using Gibbs sampling. *Journal of the American Statistical Association*, **85**, 972-85.
- Krzanowski, W.J., 1988. *Principles of Multivariate Analysis*. Oxford: Oxford University Press.
- Lunn, D.J., Thomas, A., Best, N.G., and Spiegelhalter, D.,J., 2000. WinBUGS – a Bayesian modelling framework: concepts, structure, and extensibility. *Statistics and Computing*, **10**, 325-337.
- Lunn, D.J., Spiegelhalter, D.J., Thomas, A. and Best, N.G., 2009. The BUGS project: Evolution, critique and future directions. *Statistics in Medicine*, **28**, 3049-3067.
- Phillips, D.P., 1978. Airplane accident fatalities increase just after newspaper stories about murder and suicide. *Science*, **201**, 748-750.
- Spiegelhalter, D.J., Thomas, A., Best, N.G. and Gilks, W.R., 1995. *BUGS: Bayesian inference Using Gibbs Sampling, Version 0.50*. MRC Biostatistics Unit, Cambridge.