

00. Finish what remains to be done from the previous exercise session.

0. In any of the below examples: try some of the (limited) graphical features in WinBUGS/OpenBUGS when making e.g. density plots. Right-click the figure and choose 'Properties...' for editing. Try also 'Save State' from the 'Model' menu, to get current values for all stochastic nodes. This could be useful for restarting the MCMC from exactly this same state again by inserting these as initial values. In OpenBUGS, try 'Latex' from the 'Model' menu to get the documentation of the BUGS model in latex. Try 'Attributes' menu to edit the fonts and colors of the BUGS code to make it more readable. (Save the model code as '.odc' or '.txt' file. In the latter format you get the simple plain text only).

4. Using the following data (generated from $\text{dlnorm}(1,0.5)$), fit $\text{dlnorm}(\mu, \tau)$ and $\text{dgamma}(\alpha, \beta)$ models in WinBUGS. You may use these priors: $\mu \sim N(0,0.001)$, $\tau \sim \Gamma(0.001,0.001)$, $\alpha \sim \exp(0.001)$ and $\beta \sim \exp(0.001)$. Another possible parametrization would be $\text{dgamma}(\mu\tau, \tau)$ with priors $\mu \sim LN(0,0.001)$ and $\tau \sim \Gamma(0.001,0.001)$ which enables to directly model the mean μ for the gamma distribution. Check the posterior of model parameters. Compute also predictive distribution for x and compare predictions. Compute DIC and compare models.

```
list(x=c(3.162, 0.5266, 1.245, 2.679, 1.781, 2.245,
        2.3, 9.147, 2.122, 12.53, 4.2, 18.95, 86.27,
        15.91, 6.062, 4.757, 4.911, 1.667, 2.749, 3.101))
```

```
model{
# priors for lognormal parameters:
a[1] ~ dnorm(0,0.001); b[1] ~ dgamma(0.001,0.001)
# priors for gamma parameters:
a[2] ~ dexp(0.001); b[2] ~ dexp(0.001)
# priors for alternative gamma parametrization:
a[3] <- aa*bb; b[3] <- bb
aa ~ dlnorm(0,0.001); # =mean
bb ~ dgamma(0.001,0.001); # 1/bb=var/mean
for(i in 1:20){
x[i] ~ dlnorm(a[1],b[1])
y[i] <- x[i]
y[i] ~ dgamma(a[2],b[2])
z[i] <- x[i]
z[i] ~ dgamma(a[3],b[3])
}
# predictions:
xp[1] ~ dlnorm(a[1],b[1])
xp[2] ~ dgamma(a[2],b[2])
xp[3] ~ dgamma(a[3],b[3])
# moments for x:
m[1] <- exp(a[1]+0.5/b[1]); v[1] <- (exp(1/b[1])-1)*exp(2*a[1]+1/b[1])
m[2] <- a[2]/b[2]; v[2]<-a[2]/pow(b[2],2)
m[3] <- a[3]/b[3]; v[3]<-a[3]/pow(b[3],2)
}
```

Note that BUGS will not generate suitable initial values for parameters a,b (from the priors), so we need to give them:

```
list(a=c(1,1,NA),b=c(1,1,NA),aa=1,bb=1)
```

7. Microbial samples are collected from individual animals and analyzed as pooled samples of 3 sub-samples. (3 animal specific samples put together). Each pooled sample then results to either negative or positive test. A positive result is obtained if any of the 3 sub-samples were colonized. A negative result is obtained if none of the 3 sub-samples were colonized. Assume 50 pooled samples were analyzed and 1 was positive. Compute the posterior of pooled sample population prevalence, and the posterior of sub-sample (animal) population prevalence, using WinBUGS. Uninformative prior of animal prevalence can be used. What is the posterior probability that animal prevalence is $>1\%$? Calculate posterior of both pooled and population prevalence for all possible outcomes: 0,1,2,...,50 positives in 50.

```
model{
  X ~ dbin(ppool,N)
  ppool <- 1-pow(1-panimal,3)
  panimal ~ dunif(0,1)
  pr <- step(panimal-0.01)
}
list(X=1,N=50)
```

for all 51 different sample results, the posteriors can be computed using for-loop. Data for X could be given in data list, but to be lazy, we can also write it within the loop.

```
model{
  for(i in 1:51){
    X[i] <- i-1
    X[i] ~ dbin(ppool[i],N)
    ppool[i] <- 1-pow(1-panimal[i],3)
    panimal[i] ~ dunif(0,1)
    pr[i] <- step(panimal[i]-0.01)
  }
  list(N=50)
```

Then, choose 'Compare...' from 'Inference' menu, and write ppool or panimal for the 'node', and X for the 'axis', to see how the result behaves as a function of $X \in [0, 1, \dots, 50]$.

13. Use Multinomial model with Dirichlet prior to model the 2×2 table (Cornfield, 1962):

	Heart disease		
	yes	no	
Serum cholesterol < 260	51	992	1043
Serum cholesterol > 260	41	245	286
	92	1237	1329

If the cell probabilities are written $\begin{bmatrix} p_{11} & p_{12} \\ p_{21} & p_{22} \end{bmatrix}$, the odds in the low chol group is $o_- = [p_{11}/(p_{11} +$

$p_{12}]/[1 - p_{11}/(p_{11} + p_{12})] = p_{11}/p_{12}$. Likewise, the odds in the high chol group is $o_+ = p_{21}/p_{22}$. The odds ratio is o_+/o_- . The risk ratio is $RR = [p_{21}/(p_{21} + p_{22})]/[p_{11}/(p_{11} + p_{12})]$. Compute with BUGS the posterior distribution of odds ratio and risk ratio.

```
model{
  for(i in 1:4){a[i]<-1}
  p[1:4]~ddirch(a[1:4]) # use ddirch in OpenBUGS!
  x[1:4]~dmulti(p[1:4],N)
  ominus <- p[1]/p[2]; oplus <- p[3]/p[4]
  OR <- oplus/ominus
  RR <- (p[3]/(p[3]+p[4]))/(p[1]/(p[1]+p[2]))
}
list(x=c(51,992,41,245),N=1329)
```