Blocker: random effects meta-analysis of clinical trials

A random effects meta-analysis of 22 trials of beta-blockers to prevent mortality after myocardial infarction is available from the WinBUGS manual (http://www.mrc-bsu.cam.ac.uk/bugs/winbugs/Vol1.pdf).

In this meta-analysis, the treatment effect on a log-odds scale is \( \delta_i \). The number of events in the control group in trial \( i \) is denoted as \( r^C_i \), and \( r^T_i \) denote events under treatment. The model provided in the winBUGS manual is described below, where we want to make inferences about the population effect \( d \), and the predictive distribution for the effect \( \delta_{\text{new}} \) in a new hypothetical trial.

\[
\begin{align*}
    r^C_i & \sim \text{Binomial}(p^C_i, n^C_i) \\
    r^T_i & \sim \text{Binomial}(p^T_i, n^T_i) \\
    \text{logit} (p^C_i) & = \mu_i \\
    \text{logit} (p^T_i) & = \mu_i + \delta_i \\
    \delta_i & \sim \text{Normal}(d, \tau)
\end{align*}
\]

In some circumstances it might be reasonable to assume that the population distribution has heavier tails, for example a t distribution with low degrees of freedom. This is easily accomplished in BUGS by using the \( \text{dt} \) distribution function instead of \( \text{dnorm} \) for \( d \) and \( \delta_{\text{new}} \). Then \( \delta_i \sim \text{t-dist}(d, \tau, \text{degrees of freedom}) \)

2 Exploratory analysis

The Stata code for the exploratory analysis are given in BLOCK A of section 6. The meta-analysis data are given in the WinBUGS manual in the form of R style lists. These were copied and pasted into the text file data.txt in the folder d:/Blocker. This file was read into Stata using \texttt{wbdecode} and the meta-analysis was plotted using the Stata command \texttt{meta} in figure 1.
3 WinBUGS analysis

The WinBUGS model as shown in the manual is:

model
{
    for ( i in 1 : Num ) {
        rc[i] ~ dbin(pc[i], nc[i])
        rt[i] ~ dbin(pt[i], nt[i])
        logit(pc[i]) <- mu[i]
        logit(pt[i]) <- mu[i] + delta[i]
        mu[i] ~ dnorm(0.0,1.0E-5)
        delta[i] ~ dt(d, tau, 4)
    }
    d ~ dnorm(0.0,1.0E-6)
    tau ~ dgamma(0.001,0.001)
    delta.new ~ dt(d, tau, 4)
    sigma <- 1 / sqrt(tau)
}

This model was copied to the file model.txt and BLOCK B includes the three text files of initial values that were saved as sav(d:\Blocker\inits#.txt,replace) were # represents the file number. The Bayesian fitting was carried out from within Stata using the code in BLOCK C, which automatically creates a script using wbscript and then runs the script using wbrun. The three chains were read into Stata using wbcoda as shown in BLOCK D.

4 Convergence Assessment

The Stata code for assessing convergence is given in BLOCK E where all plots have been saved by using the option export("d:\Blocker\plot#.wmf",replace). The trace plot for the first chain was calculated using wbtrace and is shown in as figure 2. All three parameters seem to mix acceptably well except for some reduced periods of poor mixing. It seems that when sigma is stuck at values very close to zero, the model switches from a random effects meta-analysis to a fixed effects model. Since delta[i] is set equal to d and there is not between-
study variation, d also gets stuck. In fact his is an inherent problem with Gibbs sampling in random effects models. And since sigma and d are both mixing poorly, delta_new will also.

**wbintervals** is similar to **wbtrace** in that it plots the history of a chain. However, instead of plotting individual simulations, **wbintervals** plots medians and 80% intervals for ordered sections of the simulations. In this example, the chain has been divided into 5 sections and plotted in figure 3.

Figure 4 shows the posterior densities for the first and second halves of chain 1 created using **wbsection**. Here we can observe the impact of the poor mixing on the stability of the density curves peaks, particularly on sigma. One half seems to have its peak close the zero value implying a fixed effects meta-analysis. The other half implies a random effect meta-analysis as intended in the model code.

The autocorrelations and partial autocorrelations for chain 1 created using **wbac** are shown in figures 5 and 6 and again show the slightly poorer mixing for sigma. The unusual aspect of delta_new is possibly caused by the way the dataset was ranked (sample size) for the analysis. If we look at the meta-analysis plotted in figure 1 we can observe from approximately half the way downwards that the mean effect size of these studies with larger weights in the pooled size effect seem to jump from larger than the pooled size effect to smaller and then back to larger indefinitely. This seems to cause the autocorrelation to jump from a positive to a negative value and then back to positive. This effect is even more evident in the partial autocorrelation. Further inspection by examining the **wbgeweke** test doesn’t indicate any drift in the mean of parameter sigma when the first 10% and last 50% of the chains are compared in every chain. The p-value doesn’t indicate any statistically significant difference between the mean value last 5,000 simulations and the first 1,000 simulations. This might be a sign of all 3 chains converging independently of the initial values used.

<table>
<thead>
<tr>
<th>Parameter: sigma</th>
<th>first 10.0% (n=1000) vs last 50.0% (n=5000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Means (se)</td>
<td>0.1351 (0.0107) vs 0.1255 (0.0065)</td>
</tr>
<tr>
<td>Autocorrelations</td>
<td>0.8908 0.9293</td>
</tr>
<tr>
<td>Mean Difference</td>
<td>-0.0096 (0.0126) z = 0.765 p = 0.4446</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Parameter: sigma</th>
<th>first 10.0% (n=1000) vs last 50.0% (n=5000)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Means (se)</td>
<td>0.1617 (0.0113) vs 0.1583 (0.0042)</td>
</tr>
<tr>
<td>Autocorrelations</td>
<td>0.8817 0.8754</td>
</tr>
<tr>
<td>Mean Difference</td>
<td>-0.0034 (0.0121) z = 0.278 p = 0.7813</td>
</tr>
</tbody>
</table>
Parameter: sigma 
first 10.0% (n=1000) vs last 50.0% (n=5000)

Means (se) 
0.1298 ( 0.0138)  0.1422 ( 0.0056)

Autocorrelations
0.9262  0.9127

Mean Difference (se) 
0.0125 ( 0.0149)  z = 0.836 p = 0.4031

The Brooks-Gelman-Rubin plots by variance and 80% interval are shown as figures 7 and 8. If we observe the y-axis scale, we will realise that R deviates very little from 1 after 2,000 simulations which suggest convergence. When looking at the lower plot of the same figures, we would like to see how the dotted and straight lines become close in a flat horizontal fashion after a few simulations. Although the small scaled employed may prevent the flat appearance of the lines, some concern about the difficulty of the lines to stabilise still exists.

Using **wbsection** for the whole series of three chains with the option m=3 superimposes the three densities for the separate chains as shown in figure 9 with dotted lines where the continuous lines represents the density for the three chains combined. The densities of the parameters in the three chains do not differ sufficiently to make one concerned about convergence. Re-running the analysis for longer should produce improved convergence, that is, a burn-in of 10,000 and a length of 50,000 produced the densities shown in figure 10. They are in much better agreement and show that sigma is slightly bimodal. The measure D included in the plot represents the maximum difference of two densities as a percentage of the maximum height of the density of the whole chain. D < 20 is usually considered as reasonable agreement.

There appears to be two competing scenarios for these data, one with sigma near zero implying a fixed effects model and the other with a sigma between 0.1 and 0.2. A long run like the one just undertaken is believed to be sufficient to obtain the correct mix of these two scenarios.

Figure 11 shows the variance based Brooks-Gelman-Rubin plots for the longer run which suggests a higher degree of convergence in the longer simulation compared to the previous figures 7. Here the feeling is that the lines in the lower plot stabilise into horizontal lines implying that the chains have finally converged for sigma.
5 Summarising the results

The simulations from the 3 chains of length 50,000 were summarised using the code in BLOCK F. *wbstats* produced the following estimates for chain 1:

Parameter | n  | mean  | sd   | se   | median | 95% CrI
----------|----|-------|------|------|--------|-----------------
\(d\)      | 10000 | -0.247 | 0.064 | 0.0016 | -0.249 ( -0.376, -0.115 )
delta_new  | 10000 | -0.249 | 0.165 | 0.0018 | -0.252 ( -0.600, 0.101 )
sigma      | 10000 | 0.131  | 0.083 | 0.0039 | 0.122 ( 0.006, 0.316 )

Parameter | n  | mean  | sd   | se   | median | 95% CrI
----------|----|-------|------|------|--------|-----------------
\(d\)      | 10000 | -0.247 | 0.067 | 0.0017 | -0.249 ( -0.378, -0.113 )
delta_new  | 10000 | -0.249 | 0.178 | 0.0020 | -0.257 ( -0.619, 0.127 )
sigma      | 10000 | 0.142  | 0.083 | 0.0037 | 0.136 ( 0.007, 0.319 )

Parameter | n  | mean  | sd   | se   | median | 95% CrI
----------|----|-------|------|------|--------|-----------------
\(d\)      | 10000 | -0.250 | 0.071 | 0.0020 | -0.255 ( -0.378, -0.113 )
delta_new  | 10000 | -0.251 | 0.176 | 0.0020 | -0.257 ( -0.613, 0.125 )
sigma      | 10000 | 0.135  | 0.086 | 0.0042 | 0.126 ( 0.006, 0.324 )

Parameter | n  | mean  | sd   | se   | median | 95% CrI
----------|----|-------|------|------|--------|-----------------
\(d\)      | 30000 | -0.248 | 0.068 | 0.0010 | -0.250 ( -0.378, -0.113 )
delta_new  | 30000 | -0.250 | 0.173 | 0.0011 | -0.255 ( -0.611, 0.118 )
sigma      | 30000 | 0.136  | 0.084 | 0.0023 | 0.128 ( 0.006, 0.319 )

<table>
<thead>
<tr>
<th>(d)</th>
<th>delta_new</th>
<th>sigma</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.0000</td>
<td>0.3827</td>
<td>1.0000</td>
</tr>
<tr>
<td>0.1061</td>
<td>0.0299</td>
<td>1.0000</td>
</tr>
</tbody>
</table>
The probability densities are shown in figure 12 for the three parameters of interest. The density of sigma is drawn with a lower boundary of zero and shows the slight bimodality pointed out earlier. The correlations between the estimates are computed by the command `corr` and show that, as expected, there is some correlation between d and delta_new. That is, the mean pooled treatment effect is a predictor of the effect of a hypothetical new study. All the other correlations seem to be week implying that d and delta_new are actually independent of sigma.

Therefore, reasonable estimates for them two would have been obtained with fewer runs since they seem to be independent on the accuracy of sigma. This impression may not be fully assured by looking at the contours of figure 13 when delta_new is plotted against sigma. Here we can observe that there is some clear relationship between sigma and delta_new. This correlation wasn’t apparent from the table of correlations but it has a reasonable interpretation by looking at how the model calculates delta_new. Delta_new is modelled by a t distribution with 4 degrees of freedom, mean d and standard error sigma. Therefore, it is obvious that delta_new will depend on d, as indicated earlier, but also on sigma. This observation makes us recommend not analysing this meta-analysis using less than 10,000 simulations at the very least in order to obtain a good approximation to parameters d and delta_new. However, an accurate estimate of sigma will require a longer run of around 50,000 simulations.
6 Stata Code

6.1 BLOCK A: Exploratory analysis

```
wbdecode , file(d:/Blocker/data.txt) clear
meta log_or se_log_or, graph(r) cline xline(0) b2title(logit treatment effect) ///
title("Random Effects Meta-analysis") xlab(-2,-1.5,-1,-0.5,0,0.5,1,1.5,2)
```

6.2 BLOCK B: Initial values

```
list(list(d = 0, delta.new = 0, sigma=1,
    mu = c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0),
    delta = c(0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0))

list(list(d = -2, delta.new = -2, sigma=5,
    mu = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1),
    delta = c(-1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1))

list(list(d = 2, delta.new = 2, sigma=.1,
    mu = c(-1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1, -1),
    delta = c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1))
```

6.3 BLOCK C: Model fitting

```
wbscript, model(d:\Blocker\model.txt) ///
data(d:\Blocker\data.txt) ///
inits(d:\Blocker\init.txt) ///
coda(d:\Blocker\data_out) ///
set(d delta.new sigma) ///
burn(1000) update(10000) chain(3) ///
quit ///
noprint ///
dic ///
logfile("d:\Blocker\wb_log.txt") ///
saving("d:\Blocker\script.txt",replace)
wbrun, script("d:\Blocker\script.txt") ///
w("K:\Software\winbugs14\winbugs14.exe")
```
6.4 BLOCK D: Reading the MCMC results

wbcoda, root("d:\Blocker\data_out") clear multichain chain(3) id(chain)

6.5 BLOCK E: Convergence checking

wbtrace d delta_new sigma if chain==1, cgoptions(cols(1))
wbintervals d delta_new sigma, m(5) bychain(chain)
wbsection d delta_new sigma if chain==1
wbac d delta_new sigma if chain==1
wbac d delta_new sigma if chain==1, pac
wbsection d delta_new sigma, m(3)
wbac d delta_new sigma if chain==1
wbsection d delta_new sigma, m(3)
wbsection d delta_new sigma, m(3)
wbbgr sigma, id(chain) var
wbbgr sigma, id(chain)
wbgeweke sigma if chain==1, p(10 50)
wbgeweke sigma if chain==2, p(10 50)
wbgeweke sigma if chain==3, p(10 50)

6.6 BLOCK F: Summarising the results

wbstats d delta_new sigma if chain==1
wbstats d delta_new sigma if chain==2
wbstats d delta_new sigma if chain==3
wbstats d delta_new sigma
wbdensity d delta_new sigma if chain==1, low(. . 0)
corr d delta_new sigma
wbhull d delta_new sigma if chain==1, peel(0, 10, 25, 50, 75, 100)
7 Figures

Figure 1: Random effects meta-analysis

Figure 2: Traces for chain 1
Figure 3: Interval plots for 3 parameters and 3 chains

Figure 4: Density estimates of chain selections for chain 1
Figure 5: Autocorrelations for chain 1

Figure 6: Partial autocorrelations for chain 1
Figure 7: Variance based Brooks-Gelman-Rubin plot for sigma

Figure 8: Interval based Brooks-Gelman-Rubin plot for sigma
Figure 9: Density estimates comparing the three chains of length 10,000

Figure 10: Density estimates comparing the three chains of length 50,000
Figure 11: variance based Brooks-Gelman-Rubin plots for sigma for a long run of 50,000 simulations

Figure 12: Density estimates from chain 1 based on 50,000 simulations. Sigma density estimated with a lower boundary at zero
Figure 13: Peels 0, 10, 25, 50, 75, and 100 for the three parameters