MCMC and Applied Bayesian Statistics: Practical Session HT 2007

1. Getting started

WinBUGS is installed on the departmental machines. If you would like to download the software on your private machine, you can get the package from the BUGS home page,

http://www.mrc-bsu.cam.ac.uk/bugs/

Once you have downloaded the files you need to email the BUGS project for a key that will let you use the full version.

Manuals are available online at the BUGS home page. The BUGS website also contains some slides of a tutorial introduction by David Spiegelhalter and many other useful links.

To analyze WinBUGS output, in addition to the tools provided in WinBUGS there are two packages that can be used with R. They are CODA and BOA. You can get both from the BUGS web site. We will not have time to discuss these today, but do have a look at them.

Starting WinBUGS

- Click on the WinBUGS icon. You will get a message about the license conditions that you can read, and then close. Now explore the menus
- HELP - you see manuals and examples. When you click on examples, you can open them in something called a *compound document* that allows programs graphs and explanations to be there together. The file surgical: institution ranking would be a good one to look at first, as it is not too complex.
- FILE - allows you to open existing files, or to start a new file to program your own example in the BUGS language.
- DOODLE - NEW is what you need to use to start a file for a new model specified in graphical format. We shall not discuss graphical models here.

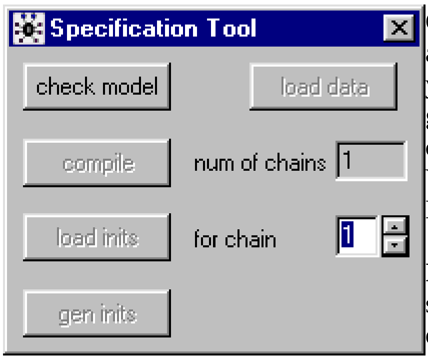
2. Model fitting

2..1 *Specifying the model*

The first stage in model fitting is to specify your model in the BUGS language. The BUGS language is rather similar to R, when we ignore the graphical models approach.

First create a new file with the code. Then open a file with the data.

You then go to the **model** menu and you will get a **specification tool** with buttons.



Click on the **check model** button. Any error messages will appear in the grey bar at the bottom of the screen. If all is well, you will get the message ”model is syntactically correct” in the grey bar at the bottom of the screen. You will also see that the **compile** and **load data** buttons will have their text darkened You are now ready for the next steps using these buttons. Leave this tool, and the other tools, open.

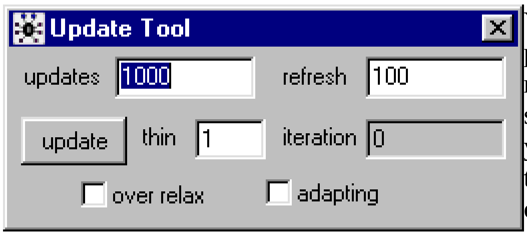
Important: First click on the file with the model, then on the specification tool. The file with the model has to be active in order for the model check to work.

Data can be entered either in list format or in rectangular format. Once you have typed in your data or opened the data file, highlight either the word list or the whole set of data. You can use the **load data** button on the specification tool to read the data in to WinBUGS.

You are now ready to compile your model using **compile** button on the specification tool. Again, check error messages. If all is well you will get a "model compiled" message.
All nodes that are not given as data, or derived from other nodes, need to be given initial values. This is done with the specification tool menu either by setting them specifically from values you type in (**load inits** button) or by generating a random value from the prior (**gen inits** button). WinBUGS allows you to run nore than one chain at the same time; see specification tool above. If you want more chains you will need to set different initial values for each one.

2..2 Generating the samples - updating

You are now ready to generate samples and to examine the simulated output. To start the sampler, go to **model** and then **update** and you will get an **updating tool**.

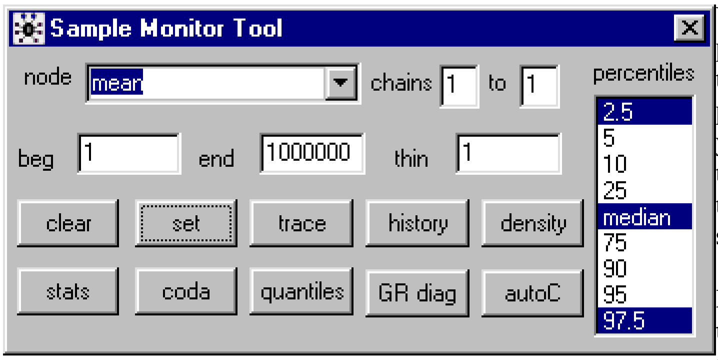


updates 1000
refresh 100
thin 1
iteration 0
□ over relax
□ adapting

You can select how many updates you get for each press of the update button and how often the screen is refreshed to show how sampling is proceeding. For some methods of updating WinBUGS will not permit you to store samples until after an 'adapting' phase. In these cases the 'adapting' button will be ticked. The over-relax button only applies to some methods of updating – see BUGS manual.
Updating does the sampling, but does not store any values. In MCMC methods you usually want to run the sampler for some time (perhaps 1000 iteration) to be sure it is stable before you start storing values.
Leave your updating tool open – you will be needing it again.

2..3 Storing values and summarising results

Updating does the sampling, but does not store any values. In MCMC methods you usually want to run the sampler for some time (perhaps 1000 iterations) to be sure it is stable before you start storing values. After an initial run values go to the **inference menu** and **samples** and you will get a **sample monitoring tool.**



node mean
chains 1 to 1
beg 1
end 1000000
thin 1
percentiles 2.5 5 10 25 median 75 90 95 97.5

You start by entering the parameters you want to monitor in the node box, and for each one press **set**. If you also press **trace** you will see a plot of the samples as they are generated. Now go back to the **updating tool** and generate some samples.

Now go back to the your **sampling tool** to look at the various ways of displaying your results or summary statistics. The most useful buttons are:-
history - shows you a plot of all the samples you have generated.
density - gives a kernel density estimate of the posterior.
stats - gives summary statistics including mean, s.d., median and percentiles that can be set with the panel on the right. These can be used for **credible intervals.** You will also get a **MonteCarlo error** for the mean that will indicate how well the mean of the posterior has been estimated from your number of samples.
Auto cor – a plot of the autocorrelation in the chains.
Coda – provides output that can be loaded into R and analyzed further.

The WinBUGS manual explains the other buttons.

3. Example – The Titanic Data

You can download the Titanic survival data at http://www.stats.ox.ac.uk/~reinert/mcmc/mcmc07.html. Use the following WinBUGS code.

```
{
  # sampling distribution........note # symbol implies comment line
         for (i in 1:1315) {

           # i'th passenger survival follows a Bernoulli density with prob q[i]
                 survived[i] ~ dbern(q[i])

                 # ..therefore death prob. is 1 - q[i]
                 q[i] <- 1 - p[i]

                 # use GLM model to relate p[i], prob. of not surviving,...
                 # to a set of covariates via
                 logit(p[i]) <- alpha + beta[class[i]] + beta[4]*age[i] + beta[5]*sex[i]
         }

         # priors
         for (j in 2:5) {
           # assume normal priors for beta
                 beta[j] ~ dnorm(0.0, 10)
         }
         beta[1] <- 0

         alpha ~ dnorm(0.0,0.001)

  # transformations and contrasts
  for (j in 1:5) {
     exp_beta[j] <- exp(beta[j])
  }

  contrast1 <- exp(beta[3] + beta[4] + beta[5])
  # contrast1 compares {first class, young, female} with.....
  # {third class, old, male}.......oh dear!

  contrast2 <- exp(beta[3]-beta[2])
  # what does contrast 2 compare?
}

# Initial values
list(alpha=0.0, beta=c(NA, 0.0, 0.0, 0.0, 0.0))
list(alpha=10.0, beta=c(NA, 2.0, 2.0, 2.0, 2.0))
```

In the **file** menu, click on new and copy the code into the window. Then carry out step 2.1 above. Open the data file and then click on the **load data** button. Note that the cursor will have to be on the file that you are loading (or compiling), otherwise you get an error message.

Follow steps 2.2 and 2.3. to run the code in WinBUGS. Which nodes would you like to monitor? Nodes could be parameters, or functions such as the contrasts). For each node that you monitor, record your posterior mean, posterior median, and 95% credible interval, and plot the posterior density.

How would you use WinBUGS to assess whether the odds of a first class passenger surviving are larger than the odds of a third class passenger to survive? How would you compare the odds of survival for children to that for adults?

Try out different initial values and priors. If you change your prior on alpha to have mean 10 and variance 10, how would your results be affected? If you chose an exponential prior for alpha, how would your results be affected?

Health warning:

Things can go wrong in using WINbugs. Always look critically at your results to see that they make sense, and use the diagnostics too.