

Bayesian Data Analysis

Dr Niamh Cahill (she/her)

Just Another Gibbs Sampler (JAGS)

JAGS

- ▶ JAGS is a system that automatically builds Markov chain Monte Carlo (MCMC) samplers for Bayesian Models. It was developed by Martin Plummer.
- ▶ JAGS takes a user's description of a Bayesian model and returns an MCMC sample of the posterior distribution for the monitored parameters.
- ▶ It's very easy to implement (m)any Bayesian models as long as you can write down the model specification. The user does not need to worry about the MCMC algorithm (you just have to check MCMC output for convergence/mixing).
- ▶ Conveniently, there are R packages that let users run JAGS from R.
- ▶ As a simple example we've considered the estimation of a Binomial probability (with the happiness example). Now we'll implement this simple example using JAGS.

How to use JAGS?

- ▶ Install JAGS and the R packages `rjags` and `R2jags`.
- ▶ Minimum things to do:
 1. Write down your model specification and its corresponding JAGS model.
 2. Specify data list and the names of the parameters for which you want to save the posterior samples.
 3. GO: Run JAGS!
 4. Get the posterior MCMC samples and use those for further analysis.

Step 1 - Model Specification

Recall for the Happiness example out of $n = 20$ women, $y = 14$ women reported being happy.

- Model specification (data model and priors):

$$y|\theta \sim \text{Binomial}(\theta, N = 20)$$
$$\theta \sim \text{Be}(1, 1)$$

- JAGS model specification:

```
binomialmodel = "  
model{  
  
  # data model (likelihood)  
  y ~ dbinom(theta, N)  
  
  # prior  
  theta ~ dbeta(1, 1)  
}  
"
```

Step 2 - JAGS Data

- ▶ Input data and parameter names:

```
jags.data <- list(y = 14,  
                 N = 20)
```

- ▶ Specify the names of the parameters for which you want to save the posterior samples:

```
parnames <- c("theta")
```

Step 3 - Run JAGS

```
library(rjags)
library(R2jags)
mod <- jags(data = jags.data,
            parameters.to.save = parnames,
            model.file = textConnection(binomialmodel))
```

- By default, JAGS runs 3 chains with 2,000 iterations per chain of which 1,000 iterations are excluded as burnin.

Step 4 - Output

- ▶ MCMC samples can be obtained as follows:

```
mcmc.array <- mod$BUGSoutput$sims.array  
dim(mcmc.array)
```

where the dimension of the mcmc.array is given by niterations x nchains x (nparameters+deviance).

- ▶ We can also get output summaries

```
mod$BUGSoutput$summary
```

More on Output

Here's some useful functions from the coda package.

```
library(coda)

# turn the model into an mcmc object
mod_mcmc <- as.mcmc(mod)

plot(mod_mcmc) # get trace plot and density

autocorr.plot(mod_mcmc) # get autocorrelation plot

effectiveSize(mod_mcmc) # get ESS

gelman.diag(mod_mcmc) # get Rhats
```

Or use shinystan to look at output (nicer)

```
library(shinystan)
shiny.array <- as.shinystan(mcmc.array)
launch_shinystan(shiny.array)
```