

Bayesian Statistical Software

Carrie Groth (Caroline)
PUBH 7440: Spring 2015

March 3, 2015

Preview

- Computer programs available
 - Winbugs vs. Openbugs
 - JAGS
- R packages for these programs
- Examples

Winbugs vs. Openbugs

- Both Winbugs and Openbugs specialize in a Gibbs sampling framework. However, other forms of sampling may still be performed
- **Openbugs** is the new updated version of Winbugs that continues to be updated from time to time
- **Winbugs** is no longer being updated. Therefore you need an immortality key to keep using it.
- Openbugs uses the same syntax and set-up as Winbugs but has additional features that allow for better stability of the model in particular circumstances
- For more information on Openbugs, visit <http://www.openbugs.net/w/FrontPage>

Winbugs Immortality Key

- This key can be found at this [website](#). You must run this key before using Winbugs

Winbugs vs. Openbugs

- In both cases, these programs run in Windows environments.
 - Using a Windows Bootcamp or other converter is recommended for MAC users
 - WINE, a new version of Openbugs for MAC, is under development but currently giving errors
- We have already talked some about Winbugs, and will talk more about it in the next lectures

JAGS

- JAGS stands for Just Another Gibbs Sampler
- This program uses the Bugs language with a few minor differences
 - By Bugs language, we mean the same statement for the model (likelihood and priors)
 - Example: Censoring and Truncation differ between the programs
 - **Censoring** occurs when observations are known to fall before a given point, after a given point, or in between two points but no other information is known; example LOD censoring
 - Winbugs or Openbugs: use $I(a, b)$ which is an indicator function following the distribution
 - JAGS uses a $indicator[i] \sim dinterval(a, b)$

JAGS (continued)

- This software does work in a MAC interface
- Most commonly used as part of the RJAGS package in R (example to be shown soon)

R Packages

- R allows us to create graphics and run programs efficiently instead of using a point-click nature
- Therefore, R programs have been developed for the programs above: The programs include:
 - [R2WinBUGS](#)
 - [R2OpenBUGS](#)
 - [RJAGS](#)
 - and others
- For these packages to work, you will need to install them in your library. You must also have Winbugs, Openbugs, or JAGS installed as they will call them from within R
- Other R packages that may be required include: coda, lattice, and MASS

Example Winbugs Code

```
model{
# standardise x's and coefficients
for (j in 1 : p) {
  b[j] <- beta[j] / sd(x[, j ])
  for (i in 1 : N) {z[i,j] <- (x[i,j] - mean(x[,j])) / sd(x[,j])}
}
b0 <- beta0 - b[1]*mean(x[,1]) - b[2]*mean(x[,2]) - b[3]*mean(x[,3])

# Likelihood
for (i in 1 : N) {
  Y[i] ~ dnorm(mu[i], tau)
  mu[i] <- beta0 + beta[1]*z[i,1]
+ beta[2]*z[i,2] + beta[3]*z[i,3]

}

# priors
beta0 ~ dnorm(0, 0.00001)
for (j in 1 : p) {beta[j] ~ dnorm(0, 0.00001)}
sigma ~ dunif(0.01,100)
# vague Gelman prior for sigma
tau<- 1/(sigma*sigma)
}
```

Example Winbugs Code Continued

```
#Data
list(p = 3,    N = 21,
     Y = c(42, 37, 37, 28, 18, 18, 19, 20, 15, 14, 14, 13, 11, 12, 8, 7, 8, 8, 9, 15, 15),
     x = structure(.Data = c(      80, 27, 89,
                                80, 27, 88,
                                75, 25, 90,
                                62, 24, 87,
                                62, 22, 87,
                                62, 23, 87,
                                62, 24, 93,
                                62, 24, 93,
                                58, 23, 87,
                                58, 18, 80,
                                58, 18, 89,
                                58, 17, 88,
                                58, 18, 82,
                                58, 19, 93,
                                50, 18, 89,
                                50, 18, 86,
                                50, 19, 72,
                                50, 19, 79,
                                50, 20, 80,
                                56, 20, 82,
                                70, 20, 91), .Dim = c(21, 3)))

#Initial Values
list(beta0 = 10, beta=c(0,0, 0), sigma = 0.1)
list(beta0 = -10, beta=c(10,10, 10), sigma = 10)
```

Example RJAGS Code

Model Statement:

```
model{
  # standardise x's and coefficients
  for (j in 1 : p) {
    b[j] <- beta[j] / sd(x[, j ])
    for (i in 1 : N) {z[i,j] <- (x[i,j] - mean(x[,j])) / sd(x[,j])}
  }
  b0 <- beta0 - b[1]*mean(x[,1]) - b[2]*mean(x[,2]) - b[3]*mean(x[,3])

  # Likelihood
  for (i in 1 : N) {
    Y[i] ~ dnorm(mu[i], tau)
    mu[i] <- beta0 + beta[1]*z[i,1]
    + beta[2]*z[i,2] + beta[3]*z[i,3]
  }

  # priors
  beta0 ~ dnorm(0, 0.00001)
  for (j in 1 : p) {beta[j] ~ dnorm(0, 0.00001)}
  sigma ~ dunif(0.01,100)
  # vague Gelman prior for sigma
  tau<- 1/(sigma*sigma)
}
```

Example RJAGS Code

See R Code and `samplemodel.txt`

Additional Resources

- Winbugs
 - A Step by Step Movie from MRC Biostatistics Unit:
<http://www.mrc-bsu.cam.ac.uk/software/bugs/the-bugs-project-winbugs/the-bugs-project-winbugs-the-movie/>
- RJAGS
 - Andrew Gelman's tutorial
<http://blue.for.msu.edu/lwf/exercises/exercise-jags-1/ex-1.pdf>