Homework 2

Gaurav Kandlikar

October 5, 2016

```r
library(R2jags)
```

```
## Loading required package: rjags

## Loading required package: coda

## Linked to JAGS 4.2.0

## Loaded modules: basemod,bugs

##
## Attaching package: 'R2jags'

## The following object is masked from 'package:coda':
##
##     traceplot
```

```r
library(lattice) # Needed for scatterplot matrix
# Set working directory
setwd("/home/gsk/grad/courses/UCLA/biost234/homework/lab2")
getwd()
```

```
## [1] "/home/gsk/grad/courses/UCLA/biost234/homework/lab2"
```

```r
# Save the data file to your working directory.

#READ IN DATA
housing=read.table("housingdata2.txt")
head(housing)   ## Always look at your data
```

```
##       V1   V2   V3 V4 V5
## 1 15.783 3.00 2.00  2  2
## 2 12.570 1.66 2.33  3  2
## 3 19.600 3.33 2.33  2  2
## 4  8.206 1.66 1.66  2  2
## 5 15.333 2.33 2.33  5  2
## 6 14.955 5.00 3.00  2  2
```

```r
## Anything funny about any of the columns?
## There are no column names!
str(housing)
```

```
## 'data.frame':    21 obs. of  5 variables:
##  $ V1: num  15.78 12.57 19.6 8.21 15.33 ...
##  $ V2: num  3 1.66 3.33 1.66 2.33 5 4.33 2.33 1.33 3 ...
##  $ V3: num  2 2.33 2.33 1.66 2.33 3 3 2.33 1.66 2.66 ...
##  $ V4: int  2 3 2 2 5 2 2 3 2 2 ...
##  $ V5: int  2 2 2 2 2 2 2 2 2 2 ...
```

```
colnames(housing) <- c("cost", "eaves", "windows", "yard", "roof")

#SEPARATE X & Y
y <- housing[,1]
x <- as.matrix(housing[,2:5])

## Remember: look at your data.
y
```

```
##  [1] 15.783 12.570 19.600  8.206 15.333 14.955 13.710 11.388  4.802 12.547
## [11] 13.677  9.683 16.798 25.615 15.734 13.510 13.855  3.986  5.997  9.778
## [21] 10.152
```

```
head(x)
```

```
##      eaves windows yard roof
## [1,]  3.00    2.00    2    2
## [2,]  1.66    2.33    3    2
## [3,]  3.33    2.33    2    2
## [4,]  1.66    1.66    2    2
## [5,]  2.33    2.33    5    2
## [6,]  5.00    3.00    2    2
```

Classical regression

Before beginning our Bayesian analysis, we can conduct classical multiple linear regression:

```
reg = lm(y~x)
summary(reg) # classical regression
```

```
##
## Call:
## lm(formula = y ~ x)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.8827 -1.8219 -0.9953  1.3467  7.4674
##
## Coefficients: (1 not defined because of singularities)
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -7.5055     3.9087  -1.920  0.07177 .
## xeaves        1.8981     0.9868   1.923  0.07133 .
## xwindows      3.5310     1.6703   2.114  0.04960 *
## xyard         2.5450     0.8721   2.918  0.00958 **
```

```
## xroof              NA         NA      NA       NA
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.275 on 17 degrees of freedom
## Multiple R-squared:  0.6318, Adjusted R-squared:  0.5668
## F-statistic: 9.722 on 3 and 17 DF,  p-value: 0.0005765
```

The yard and window quality seem to correlate with the contractor's estimates. We see a row of NAs for coefficients of roof, since all of the roofs in this dataset had a quality of 2- so, there is no variation in that measurement.

Bayesian approach

First we write out our model

```
# sink("housingmodel.txt")
cat("
model
{
   for(i in 1:N) {
         y[i] ~ dnorm(mu[i] , tau )
         mu[i] <- beta0 + inprod(x[i,] , beta[] )
        }

     beta0 ~ dnorm( mbeta0 , precbeta0) # mean and precision of beta0 (intercept) defined in R

for (j in 1:K) {
     beta[j] ~ dnorm( m[j] , prec[j] ) # the prior for the four different betas will be provided throug
                                     # in all models we will use mean beta values estimated in an old a

        }
       tau ~ dgamma( tau.a , tau.b )  # we tend to use gamma for tau because it is always non-negative
                                    # we will weigh the priors differently in a few different models by
                                    # fidgeting with the precision in prior
       sigma <- 1 / sqrt(tau)         # note that we use <- here since it's a simple calculation
    }
  ",fill = TRUE)
```

```
##
## model
## {
##    for(i in 1:N) {
##       y[i] ~ dnorm(mu[i] , tau )
##       mu[i] <- beta0 + inprod(x[i,] , beta[] )
##      }
##
##   beta0 ~ dnorm( mbeta0 , precbeta0) # mean and precision of beta0 (intercept) defined in R
##
## for (j in 1:K) {
##   beta[j] ~ dnorm( m[j] , prec[j] ) # the prior for the four different betas will be provided throug
##                                      # in all models we will use mean beta values estimated in an ol
##
```

```
##      }
##     tau ~ dgamma( tau.a , tau.b )  # we tend to use gamma for tau because it is always non-negative
##                                     # we will weigh the priors differently in a few different models
##                                     # fidgeting with the precision in prior
##     sigma <- 1 / sqrt(tau)         # note that we use <- here since it's a simple calculation
##   }
##
```

```
# sink()
```

Now, we set up three different models:

```
# define some variables
N = nrow(housing)
K = ncol(x) # number of coefficients to estimate betas for
m = c(1.6053, 1.2556, 2.3413, 3.6771)# mean betas from previous analysis

dataA <- list(N=N, K=4, m = m,
              prec = c(.2164, .1105, .2061, .1337), tau.a=17,
              tau.b = 1128, mbeta0= -5.682, precbeta0=.05464, x=x, y=y)

dataB <- list(N=N, K=4, m=m,
            prec=c(.02774, .014160, .02642, .01714), tau.a=2.1795,
            tau.b=144.6, mbeta0= -5.682, precbeta0=.007005, x=x, y=y)

dataC <- list(N=N, K=4, m=m,
            prec=c(.005549, .002832, .005284, .003428), tau.a=.4359,
            tau.b=28.92, mbeta0= -5.682, precbeta0=.00140, x=x, y=y)


inits <- rep(list(list(beta0=0, beta=c(1,1,1,1),tau=1)),5) # 5 equal to the n.chains in jags call

#DEFINE PARAMETERS TO MONITOR
parameters <- c("beta0", "beta" , "tau", "sigma")
```

Then we run jags:

```
#RUN THE JAGS PROGRAM, SAVING DATA TO LAB2.SIM
lab2.simA <- jags (dataA, inits, parameters, "housingmodel.txt", n.chains=5,
    n.iter=5100, n.burnin=100, n.thin=1, DIC=FALSE) # DIC = F - deviance not calcuated
```

```
## module glm loaded

## module dic loaded

## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 21
##    Unobserved stochastic nodes: 6
##    Total graph size: 191
##
## Initializing model
```

```r
lab2.simB <- jags (dataB, inits, parameters, "housingmodel.txt", n.chains=5,
    n.iter=5100, n.burnin=100, n.thin=1, DIC=FALSE)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 21
##    Unobserved stochastic nodes: 6
##    Total graph size: 191
##
## Initializing model
```

```r
lab2.simC <- jags (dataC, inits, parameters, "housingmodel.txt", n.chains=5,
    n.iter=5100, n.burnin=100, n.thin=1, DIC=FALSE)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 21
##    Unobserved stochastic nodes: 6
##    Total graph size: 191
##
## Initializing model
```

```r
lab2.sims = list(lab2.simA, lab2.simB, lab2.simC)

knitr::kable(lab2.simA$BUGSoutput$summary[,c("mean", "sd", "2.5%", "97.5%")], digits = 3)
```

	mean	sd	2.5%	97.5%
beta[1]	1.786	1.366	-0.897	4.449
beta[2]	1.787	1.997	-2.115	5.735
beta[3]	2.233	1.316	-0.351	4.796
beta[4]	2.586	2.258	-1.895	7.008
beta0	-7.010	3.830	-14.576	0.497
sigma	6.931	0.688	5.749	8.440
tau	0.021	0.004	0.014	0.030

```r
knitr::kable(lab2.simB$BUGSoutput$summary[,c("mean", "sd", "2.5%", "97.5%")], digits = 3)
```

	mean	sd	2.5%	97.5%
beta[1]	1.911	1.413	-0.892	4.710
beta[2]	3.220	2.319	-1.426	7.754
beta[3]	2.480	1.263	-0.025	4.961
beta[4]	1.155	4.992	-8.719	10.944
beta0	-8.876	9.604	-27.656	9.930
sigma	4.840	0.771	3.611	6.591

	mean	sd	2.5%	97.5%
tau	0.046	0.014	0.023	0.077

```
knitr::kable(lab2.simC$BUGSoutput$summary[,c("mean", "sd", "2.5%", "97.5%")], digits = 3)
```

	mean	sd	2.5%	97.5%
beta[1]	1.904	1.165	-0.421	4.185
beta[2]	3.480	1.972	-0.402	7.388
beta[3]	2.543	1.034	0.515	4.607
beta[4]	0.839	10.690	-20.247	21.808
beta0	-9.059	21.239	-50.290	32.592
sigma	3.829	0.689	2.758	5.437
tau	0.074	0.025	0.034	0.131

Question 2

2. Give a table of inferences for the coefficient of roofs for the three priors. Briefly explain why it comes out as it does.

```
temp3 <- t(sapply(lab2.sims, function(x) x$BUGSoutput$summary["beta[4]",c("mean", "sd")]))
rownames(temp3) <- c("Model 1", "Model 2", "Model 3")
knitr::kable(temp3)
```

	mean	sd
Model 1	2.5864595	2.257813
Model 2	1.1547660	4.992266
Model 3	0.8393065	10.689721