STAT 578 - Fall 2019 - Assignment 6

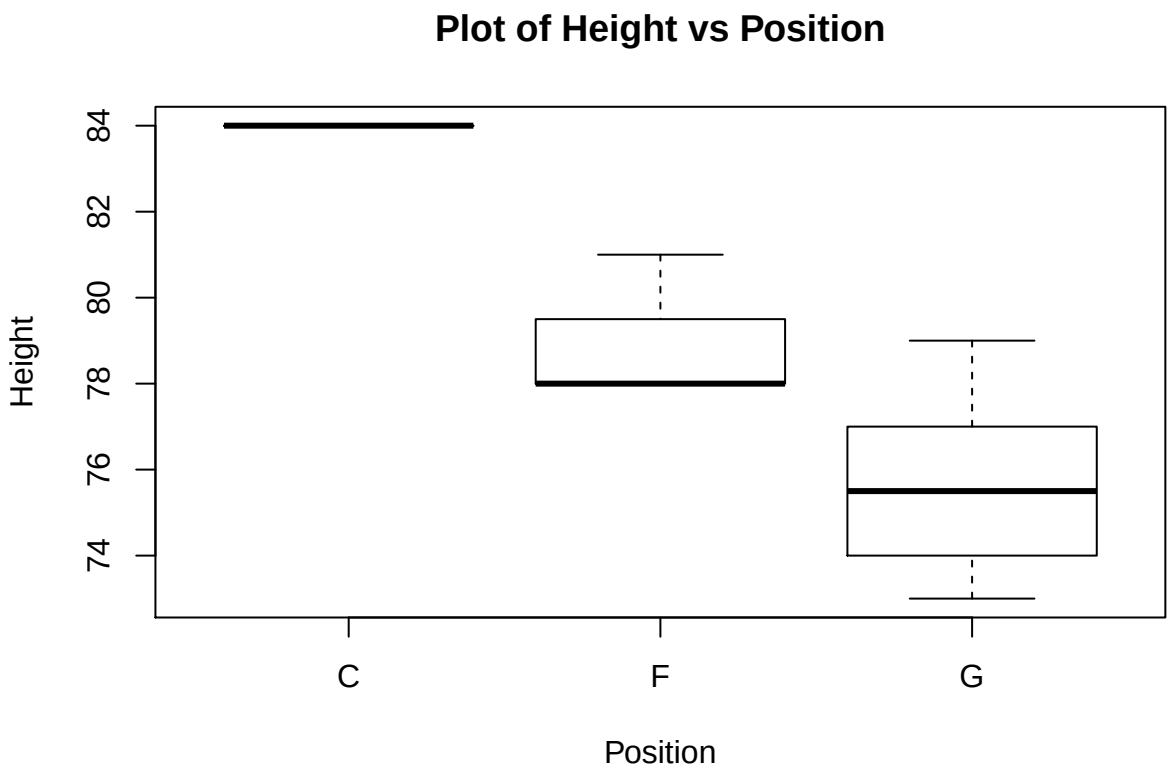
Frederick (Eric) Ellwanger - fre2

December 7, 2019

Exercise 1

Using plot(Ht ~ Pos, data= $\cdots$), display box plots of height by position. Is there a relationship between height and position? (Such a relationship might cause substantial posterior correlations between regression coeffcients if both height and position are used as explanatory variables.)

```
illinibb <- read.csv("illinimensbb.csv")
plot(Ht ~ Pos, data = illinibb, main = "Plot of Height vs Position",
     xlab = "Position", ylab = "Height")
```



There appears to be a relationship between height and position, the taller players play Center (with what appears to be no overlap with the other positions), while the shorter players tend to play Guard (although there is some overlap with the Forward position). Forwards heights tend to be between Center and Guard.

Exercise 2

Let y_i be the number of field goals made by player i out of n_i attempts (i = 1, ..., 15). Consider the following logistic regression (with implicit intercept) on player position and

height:

$$y_i|p_i \sim indep\ Bin(n_i, p_i)$$

$$logit(p_i) = \beta_{Pos(i)} + \beta_{Ht} H_i$$

where:

$Pos(i)$ **= player** i **position (C, F, G)**
H_i **= player** i **height after centering and scaling to sample standard dev. 0.5**

Consider the prior:

$\beta_C, \beta_F, \beta_G \sim iid \quad t1(0, 10^2) \qquad \beta_{Ht} \sim t1(0, 2.5^2)$

(2)(a) List an appropriate JAGS model. Include nodes for the vector of binomial probabilities pi and a vector yrep of replicate responses.

```
model {
   for (i in 1:length(fgm)) {
      fgm[i] ~ dbin(prob[i], fga[i])
      logit(prob[i]) <- betapos[pos[i]] + betaht*htscaled[i]

      yrep[i] ~ dbin(prob[i], fga[i])
   }
   for (j in 1:max(pos)) {
      betapos[j] ~ dt(0, 0.01, 1)
   }
   betaht ~ dt(0, 0.16, 1)
}
```

Now run your model using rjags. Make sure to use multiple chains with overdispersed starting points, check convergence, and monitor the regression coeffcients, probabilities, and replicate responses (after convergence) long enough to obtain effective sample sizes of at least 4000 for each regression coeffcient.

```
d1 <- list(fgm = illinibb$FGM,
           fga = illinibb$FGA,
           pos = unclass(illinibb$Pos),
           htscaled = as.vector(scale(illinibb$Ht, scale=2*sd(illinibb$Ht))))
inits1 <- list(list(betapos=c(10,10,10), betaht=10),
               list(betapos=c(10,10,-10), betaht=-10),
               list(betapos=c(10,-10,10), betaht=-10),
               list(betapos=c(10,-10,-10), betaht=10))
library(rjags)
m1 <- jags.model("ilinnibb.bug", d1, inits1, n.chains=4, n.adapt=1000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 15
##    Unobserved stochastic nodes: 19
##    Total graph size: 110
##
## Initializing model
```

```
#Burn-in
update(m1, 1000)

#Get samples
x1 <- coda.samples(m1, c("betapos","betaht"), n.iter=2000)

#Check convergence
gelman.diag(x1, autoburnin=FALSE)
```

```
## Potential scale reduction factors:
##
##            Point est. Upper C.I.
## betaht              1       1.00
## betapos[1]          1       1.00
## betapos[2]          1       1.00
## betapos[3]          1       1.01
##
## Multivariate psrf
##
## 1
```

```
#Add additional variable - prob and yrep
x1 <- coda.samples(m1, c("betapos","betaht","prob","yrep"), n.iter=8000)

#Check for effective sample size > 4000
effectiveSize(x1[,1:4])
```

```
##     betaht betapos[1] betapos[2] betapos[3]
##       4323       6561       6935       5812
```

(2)(b) Display the coda summary of the results for the monitored regression coeffcients.

```
summary(x1[,1:4])
```

```
##
## Iterations = 4001:12000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 8000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##               Mean     SD Naive SE Time-series SE
## betaht      0.1397 0.1800 0.001006       0.002740
## betapos[1] -0.4546 0.2910 0.001627       0.003615
## betapos[2] -0.0639 0.1111 0.000621       0.001335
## betapos[3] -0.3341 0.0717 0.000401       0.000942
##
## 2. Quantiles for each variable:
##
##              2.5%     25%     50%     75%  97.5%
## betaht     -0.211  0.0168  0.1398  0.2620  0.491
## betapos[1] -1.026 -0.6496 -0.4565 -0.2592  0.119
## betapos[2] -0.282 -0.1400 -0.0635  0.0113  0.154
## betapos[3] -0.474 -0.3825 -0.3341 -0.2859 -0.193
```

(2)(c) With your posterior samples, display scatterplots of (i) β_C versus β_{Ht}, (ii) β_F versus β_{Ht}, and (iii) β_G versus β_{Ht}. Do you see (posterior) correlations?

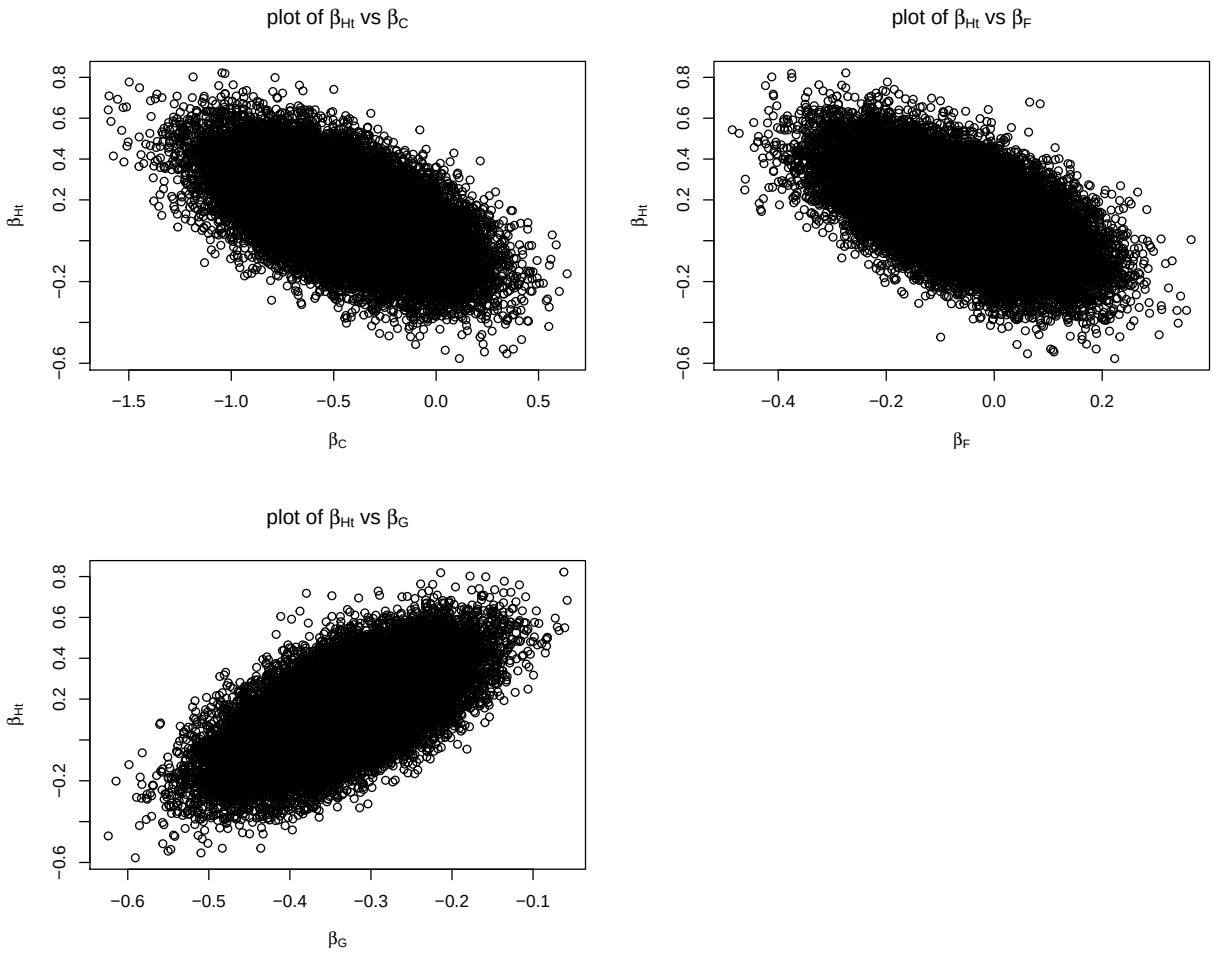
```
betaposs = as.matrix(x1)[, paste("betapos[",1:3,"]", sep="")]
betahts = as.matrix(x1)[, paste("betaht", sep="")]

par(mfrow=c(2, 2))

#betaC ~ betaHt
plot(betahts ~ betaposs[,1], main = expression(paste("plot of ", beta[Ht], " vs ", beta[C])),
     xlab = expression(beta[C]), ylab = expression(beta[Ht]))

#betaF ~ betaHt
plot(betahts ~ betaposs[,2], main = expression(paste("plot of ", beta[Ht], " vs ", beta[F])),
     xlab = expression(beta[F]), ylab = expression(beta[Ht]))

#betaG ~ betaHt
plot(betahts ~ betaposs[,3], main = expression(paste("plot of ", beta[Ht], " vs ", beta[G])),
     xlab = expression(beta[G]), ylab = expression(beta[Ht]))
```

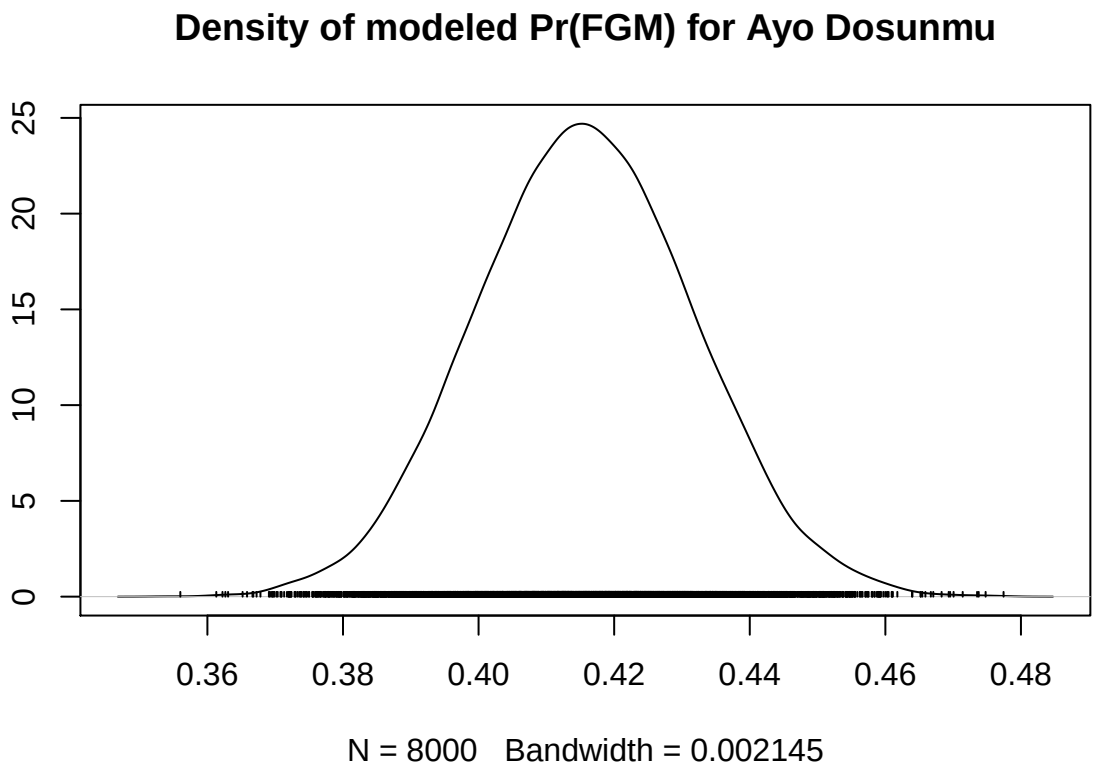


The above graphs seem to indicate there are correlations between Ht and Position.

(2)(d) Consider the modeled probability that Ayo Dosunmu (No. 11) successfully makes an

attempted field goal. Plot the (approximate) posterior density of this probability.

```
densplot(x1[,"prob[4]"], main = "Density of modeled Pr(FGM) for Ayo Dosunmu")
```



**(2)(e) Approximate the posterior probability that $\beta_F > \beta_G$ (i.e., that forwards have a higher probability of successfully making an attempted field goal than guards, after adjusting for height). Also, approximate the Bayes factor favoring $\beta_F > \beta_G$ versus $\beta_F < \beta_G$. (Note that, by symmetry, $\beta_F > \beta_G$ and $\beta_F < \beta_G$ have equal prior probability.) What can you say about the data evidence that $\beta_F > \beta_G$?

```
betasx1 = as.matrix(x1)[, paste("betapos[",1:3,"]", sep="")]
(fgrtg = mean(betasx1[,2] > betasx1[,3]))
```

```
## [1] 0.961
```

The approximate posterior probability that $\beta_F > \beta_G$ is 0.961

```
#Since PR_prior_h1 = PR_prior_h2 we divide by 1
(BF = (mean(betasx1[,2] > betasx1[,3])/mean(betasx1[,2] < betasx1[,3]))/1)
```

```
## [1] 24.4
```

The Bayes Factor is approximately 24.397. This is strong evidence on the Bayes Factor scale to support that $\beta_F > \beta_G$.

(2)(f) Use the chi-square discrepancy to compute an approximate posterior predictive p-value. Does it indicate any evidence of problems (such as overdispersion)?

```
probs = as.matrix(x1)[, paste("prob[",1:nrow(illinibb),"]", sep="")]
yreps = as.matrix(x1)[, paste("yrep[",1:nrow(illinibb),"]", sep="")]
Tchi = numeric(nrow(yreps))
```

```
Tchirep = numeric(nrow(yreps))
for(s in 1:nrow(yreps)){
  Tchi[s] = sum((illinibb$FGM-illinibb$FGA*probs[s,])^2/(illinibb$FGA*probs[s,]*(1-probs[s,])))
  Tchirep[s] = sum((yreps[s,]-illinibb$FGA*probs[s,])^2/(illinibb$FGA*probs[s,]*(1-probs[s,])))
}
(chi2 = mean(Tchirep >= Tchi))
```

```
## [1] 0.0472
```

The chi-square discrepency value is 0.047 suggesting the possibility that there may be a problem with overdispersion.

(2)(g) Now consider expanding the model to allow for overdispersion.

(2)(g)(i) List an appropriately modified JAGS model. Then run it using rjags, with all of the usual steps.

```
model {
   for (i in 1:length(fgm)) {
      fgm[i] ~ dbin(prob[i], fga[i])
      logit(prob[i]) <- betapos[pos[i]] + betaht*htscaled[i] + epsilon[i]
      epsilon[i] ~ dnorm(0,1/sigmaepsilon^2)

      yrep[i] ~ dbin(prob[i], fga[i])
   }
   for (j in 1:max(pos)) {
      betapos[j] ~ dt(0, 0.01, 1)
   }
   betaht ~ dt(0, 0.16, 1)

   sigmaepsilon ~ dunif(0,10)
}
```

```
d2 <- list(fgm = illinibb$FGM,
           fga = illinibb$FGA,
           pos = unclass(illinibb$Pos),
           htscaled = as.vector(scale(illinibb$Ht, scale=2*sd(illinibb$Ht))))
inits2 <- list(list(betapos=c(10,10,10), betaht=10, sigmaepsilon = 0.01),
               list(betapos=c(10,10,-10), betaht=-10, sigmaepsilon = 9),
               list(betapos=c(10,-10,10), betaht=-10, sigmaepsilon = 0.01),
               list(betapos=c(10,-10,-10), betaht=10, sigmaepsilon = 9))
m2 <- jags.model("ilinnibb2.bug", d2, inits2, n.chains=4, n.adapt=1000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 15
##    Unobserved stochastic nodes: 35
##    Total graph size: 142
##
## Initializing model
```

```
#Burn-in
update(m2, 4000)

#Get samples
```

```
x2 <- coda.samples(m2, c("betapos","betaht"), n.iter=16000)

#Check convergence
gelman.diag(x2, autoburnin=FALSE)
```

```
## Potential scale reduction factors:
##
##            Point est. Upper C.I.
## betaht              1       1.01
## betapos[1]          1       1.00
## betapos[2]          1       1.01
## betapos[3]          1       1.01
##
## Multivariate psrf
##
## 1
```

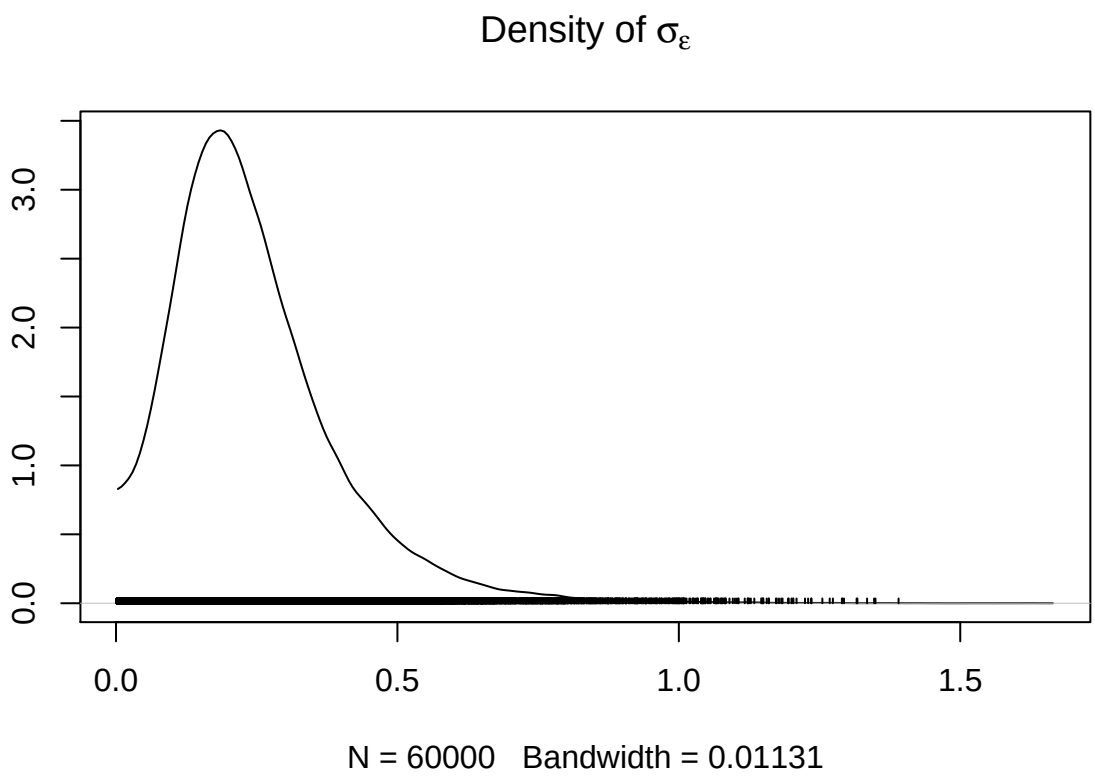
```
#Add additional variable - prob, yrep, and sigmaepsilon
x2 <- coda.samples(m2, c("betapos","betaht","prob","yrep", "sigmaepsilon"), n.iter=60000)

#Check for effective sample size > 4000
effectiveSize(x2[,1:4])
```

```
##     betaht betapos[1] betapos[2] betapos[3]
##       5044       7010       5969       6785
```

(2)(g)(ii)Plot the (approximate) posterior density of σ_ϵ.

```
densplot(x2[,"sigmaepsilon"], main = expression(paste("Density of ", sigma[epsilon])))
```



(2)(g)(iii) Repeat part (e) under this expanded model. Does your conclusion change?

```
betasx2 = as.matrix(x2)[, paste("betapos[",1:3,"]", sep="")]
(fgrtg = mean(betasx2[,2] > betasx2[,3]))
```

```
## [1] 0.79
```

The approximate posterior probability that $\beta_F > \beta_G$ is 0.79

```
#Since PR_prior_h1 = PR_prior_h2 we divide by 1
(BF = (mean(betasx2[,2] > betasx2[,3])/mean(betasx2[,2] < betasx2[,3]))/1)
```

```
## [1] 3.76
```

The Bayes Factor is 3.755. This is lower than (e) above, changing the conclusion slightly in the fact that there is only positive evidence that $\beta_F > \beta_G$ rather than strong evidence.

Exercise 3

(3)(a) List an appropriate JAGS model. Include nodes for the vector of Poisson means $\lambda_i = t_i r_i$ and a vector y^{rep} of replicate responses.

```
model {
   for (i in 1:length(blk)) {
      blk[i] ~ dpois(lambda[i])
      log(lambda[i]) <- logminutes[i] + beta.p[pos[i]] + beta.ht*htscaled[i]

     yrep[i] ~ dpois(lambda[i])
   }
```

```
   for (j in 1:max(pos)) {
      beta.p[j] ~ dnorm(0, 1/100^2)
   }
   beta.ht ~ dnorm(0, 1/100^2)
}
```

Now run your model using rjags. Make sure to use multiple chains with overdispersed starting points, check convergence, and monitor the regression coeffcients, Poisson means, and replicate responses (after convergence) long enough to obtain effective sample sizes of at least 4000 for each regression coeffcient.

```
d3 <- list(blk = illinibb$BLK,
           logminutes = log(illinibb$MIN),
           pos = unclass(illinibb$Pos),
           htscaled = as.vector(scale(illinibb$Ht)))
inits3 <- list(list(beta.p=c(100,100,100), beta.ht=10),
               list(beta.p=c(100,100,-100), beta.ht=-10),
               list(beta.p=c(100,-100,100), beta.ht=-10),
               list(beta.p=c(100,-100,-100), beta.ht=10))
m3 <- jags.model("ilinnibbpois.bug", d3, inits3, n.chains=4, n.adapt=1000)
```

```
## Compiling model graph
##    Resolving undeclared variables
##    Allocating nodes
## Graph information:
##    Observed stochastic nodes: 15
##    Unobserved stochastic nodes: 19
##    Total graph size: 124
##
## Initializing model
```

```
#Burn-in
update(m3, 1000)

#Get samples
x3 <- coda.samples(m3, c("beta.p","beta.ht"), n.iter=4000)

#Check convergence
gelman.diag(x3, autoburnin=FALSE)
```

```
## Potential scale reduction factors:
##
##           Point est. Upper C.I.
## beta.ht            1       1.01
## beta.p[1]          1       1.01
## beta.p[2]          1       1.01
## beta.p[3]          1       1.00
##
## Multivariate psrf
##
## 1
```

```
#Add additional variable - prob, yrep, and sigmaepsilon
x3 <- coda.samples(m3, c("beta.p","beta.ht","lambda","yrep"), n.iter=20000)
```

```
#Check for effective sample size > 4000
effectiveSize(x3[,1:4])
```

```
##   beta.ht beta.p[1] beta.p[2] beta.p[3]
##      4492      5185      5736     16839
```

(3)(b) Display the coda summary of the results for the monitored regression coeffcients.

```
summary(x3[,1:4])
```

```
##
## Iterations = 6001:26000
## Thinning interval = 1
## Number of chains = 4
## Sample size per chain = 20000
##
## 1. Empirical mean and standard deviation for each variable,
##    plus standard error of the mean:
##
##             Mean    SD Naive SE Time-series SE
## beta.ht     1.01 0.275 0.000973        0.00411
## beta.p[1] -5.29 0.606 0.002143        0.00843
## beta.p[2] -4.51 0.287 0.001014        0.00379
## beta.p[3] -4.45 0.178 0.000629        0.00138
##
## 2. Quantiles for each variable:
##
##             2.5%    25%   50%   75% 97.5%
## beta.ht     0.48  0.816  1.00  1.19  1.56
## beta.p[1] -6.51 -5.695 -5.28 -4.88 -4.14
## beta.p[2] -5.09 -4.699 -4.50 -4.31 -3.97
## beta.p[3] -4.81 -4.568 -4.45 -4.33 -4.12
```

(3)(c) The sampling model implies that $e^{\beta_{Ht}}$ represents the factor by which the mean rate of blocking shots changes for each increase in height of one standard deviation (here, about 3.5 inches). (Under the model, this factor is the same for all positions.) Form an approximate 95% central posterior credible interval for this factor. According to your interval, does it seem that greater height is associated with a higher rate of blocking shots?

```
(cpci95 = exp(summary(x3)$quantiles["beta.ht", c("2.5%", "97.5%")]))
```

```
##  2.5% 97.5%
##  1.62  4.77
```

The approximate 95% central posterior credible interval for this factor is (1.616, 4.771)

Acording to this interval it would appear that greater height is associated with a higher rate of blocking shots.

(3)(d) Use the chi-square discrepancy to compute an approximate posterior predictive p-value. Does it indicate any evidence of problems?

```
lambdas <- as.matrix(x3)[, paste("lambda[",1:nrow(illinibb),"]", sep="")]
yreps = as.matrix(x3)[, paste("yrep[",1:nrow(illinibb),"]", sep="")]
Tchi = numeric(nrow(yreps))
Tchirep = numeric(nrow(yreps))
for(s in 1:nrow(yreps)){
   Tchi[s] <- sum((illinibb$BLK - lambdas[s,])^2 / lambdas[s,])
```

```
   Tchirep[s] <- sum((yreps[s,] - lambdas[s,])^2 / lambdas[s,])
}
(chi2 = mean(Tchirep >= Tchi))
```

```
## [1] 0.00694
```

The chi-square discrepency value is 0.007 suggesting the possibility that there may be a problem with overdispersion.

(3)(e) For each player (i), approximate $Pr(y_i^{rep} \geq y_i|y)$, which is a kind of marginal predictive p-value.

(3)(e)(i) Show your R code, and display a table with the player names and their values of this probability.

```
#Setup results vector
pryrep_grt_y = numeric(nrow(illinibb))

#Loop through each player
for (i in 1:nrow(illinibb)){
  pryrep_grt_y[i] = mean(yreps[,i] >= illinibb$BLK[i])
}
pframe = data.frame(name=illinibb$Player, p_value=pryrep_grt_y)
library(knitr)
library(kableExtra)
kable(pframe) %>%
  kable_styling(bootstrap_options = c("striped", "condensed"))
```

name	p_value
Bezhanishvili, Giorgi	0.590
Cayce, Drew	1.000
De La Rosa, Adonis	0.998
Dosunmu, Ayo	0.800
Feliz, Andres	0.956
Frazier, Trent	0.950
Griffin, Alan	0.021
Griffith, Zach	1.000
Jones, Tevian	0.977
Jordan, Aaron	0.197
Kane, Samba	0.005
Nichols, Kipper	0.323
Oladimeji, Samson	1.000
Underwood, Tyler	1.000
Williams, Da'Monte	0.088

(3)(e)(ii) Name any players for whom this probability is less than 0.05. (Any such player blocked notably more shots than the model would suggest, for his position and height.)

```
df <- data.frame(name = character(), p_value = numeric())
for (i in 1:length(pryrep_grt_y)) {
  if (pframe$p_value[i] < 0.05) {
    df = rbind(df, data.frame(name = pframe$name[i], p_value = pframe$p_value[i]))
  }
}
```

```r
kable(df) %>%
  kable_styling(bootstrap_options = c("striped", "condensed"))
```

name	p_value
Griffin, Alan	0.021
Kane, Samba	0.005

(3)(e)(iii) Notice that the probability equals 1 for some players. Why is that actually not surprising? (Hint: How many shots were actually blocked by those players? How much playing time did they have?)

It is not surprising that the probability equals 1 for players that had 0 blocked shots. The yrep will always be >= 0 since we are dealing with numbers >= 0. This will lead to an approximate $Pr(y_i^{rep} \geq y_i|y) = 1$**.**