STAA575\_HW5

Jeremy Primus

2/18/2018

Results

1. From an initial exploration of the data, it appears that men are paid more than women at this bank. The beginning salary for a male is higher than the average for a female. However, the men on average also have more education and experience, so further analysis must be conducted to determine if there are discrepancies in pay between women and men with the same qualifications.

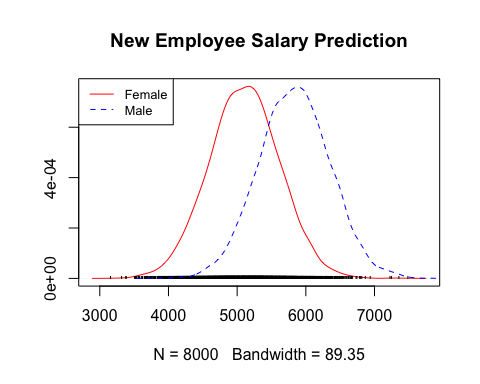
2.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Mean | Median | Lower 95% CI | Upper 95% CI | Standard Deviation |
| Beta[1] | 3528.72 | 3528.53 | 3333.09 | 3710.29 | 96.40 |
| Beta[2] | 722.30 | 722.30 | 577.69 | 871.45 | 75.29 |
| Beta[3] | 89.54 | 89.64 | 68.45 | 111.03 | 10.80 |
| Beta[4] | 1.28 | 1.28 | 0.12 | 2.36 | 0.57 |
| Beta[5] | 23.59 | 23.52 | 13.96 | 32.94 | 4.88 |
| Sigma | 502.69 | 504.45 | 588.49 | 442.45 | 1312.77 |

3. MCMC Diagnostics indicate no obvious issues with convergence or correlation.

4. The mean prediction for a future female employee’s salary is about $5100, while for a male with the same qualifications the mean prediction is about $5800.

5.



The male posterior predictive density is shifted nearly $1000 higher.

6. print(femaleProb)

## [1] 0.088625

print(maleProb)

## [1] 0.5275

The male probability of getting paid $5800 is nearly 6 times higher given the same qualifications.

7. Conclusions

The data suggests a significant difference between the pay for men and women. First, the coefficient attached to the 'sex' data is actually the most heavily weighted coefficient. Secondly future predictions of the model based upon the same qualifications for men and women indicate that men are likely to be paid higher for the same work.

STAA575\_HW5 Code and Output

## Model Definition

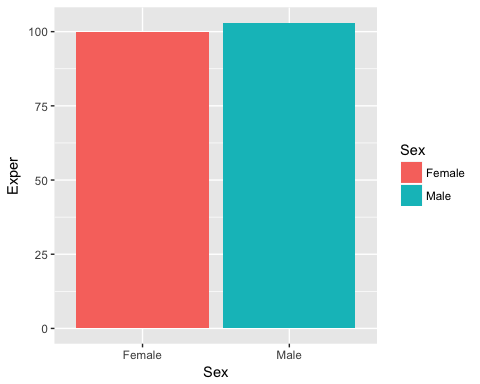
modelString = "  
model {  
 # Sampling distribution  
 for(i in 1:n)  
 {  
 salary[i] ~ dnorm(mu[i], tau)   
 mu[i] <- inprod(beta[], X[i,])  
 }  
   
 # Prior distributions:   
 for(j in 1:5){ beta[j] ~ dnorm(beta.mu[j], 0.0001) }  
 tau ~ dgamma(1, 250000)  
}  
"  
model.txt<-textConnection(modelString)

## Import, Clean, Explore Data

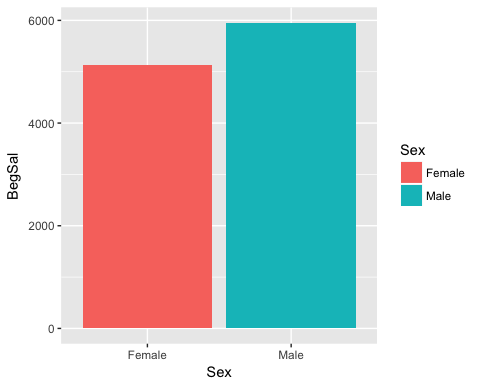
# Make sure directory is set correctly  
bank <- read.table("BankSalaryData.csv",   
 header = TRUE, sep = ",")  
# print(bank)  
  
bankbySex <- bank %>% group\_by(Sex) %>% summarise\_all(funs(mean)) %>% mutate(Sex = ifelse(Sex > 0, "Male", "Female"))  
  
# bar plot  
ggplot(data=bankbySex, aes(x = Sex, y = Educ)) +  
geom\_bar(aes(fill = Sex), stat="identity", position=position\_dodge())



ggplot(data=bankbySex, aes(x = Sex, y = Exper)) +  
geom\_bar(aes(fill = Sex), stat="identity", position=position\_dodge())



ggplot(data=bankbySex, aes(x = Sex, y = BegSal)) +  
geom\_bar(aes(fill = Sex), stat="identity", position=position\_dodge())



## Conclusions

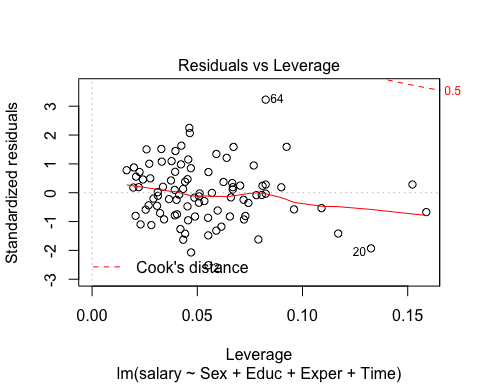
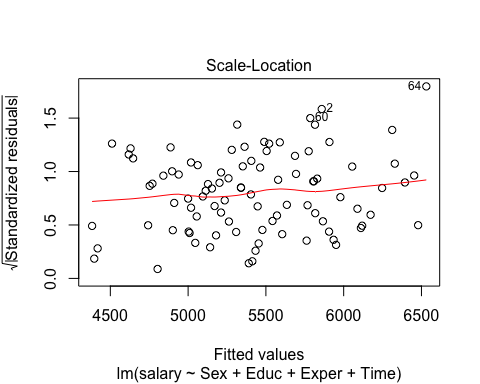
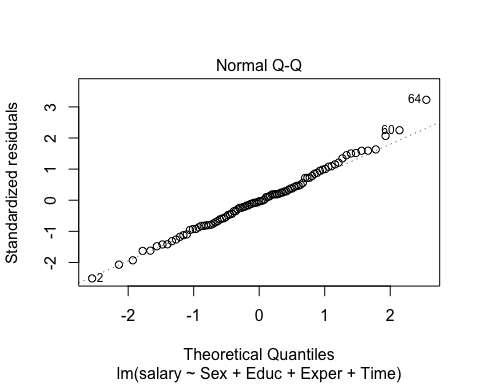
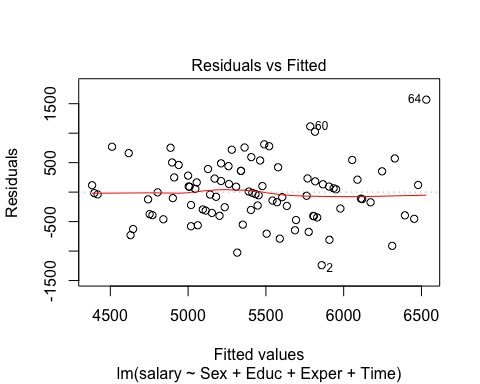
From an initial exploration of the data, it appears that men are paid more than women at this bank. The beginning salary for a male is higher than the average for a female. However, the men on average also have more education and experience, so further analysis must be conducted to determine if there are discrepancies in pay between women and men with the same qualifications.

## Linear Model Statement

# initialize parameters  
salary <- bank$BegSal  
x1 <- bank$Sex  
x2 <- bank$Educ  
x3 <- bank$Exper  
x4 <- bank$Time  
n <- length(salary)  
X <- cbind(1, x1, x2, x3, x4)  
  
# linear regression   
beta.mu <- lm(salary ~ x1 + x2 + x3 + x4)$coef # what does this line of code do?  
vanilla.reg=lm(salary ~ Sex + Educ + Exper + Time,data=bank)

## Linear Model Diagnostics

# Investigate the residuals for the basic model  
plot(vanilla.reg)



## Compile the model

dataList = list(salary = salary, X = X, n = n, beta.mu = beta.mu) # combining the data for JAGS  
monitor = c("beta", "tau") # The parameter(s) to be monitored.  
nChains = 3 # Number of chains to run.  
nIter = 10000 # Steps to save per chain.   
  
# Create, initialize, and adapt the model, burnin, and save steps from MCMC chain:  
jagsModel = jags.model(model.txt, data = dataList, n.chains = nChains, quiet = TRUE)  
codaSamples = coda.samples(jagsModel, var = monitor, n.iter = nIter)  
  
#Create a table of the results  
#Compute mean, sd, and 95% HPD interval:  
means=apply(codaSamples[[1]][2001:10000,],2,mean)  
sds=apply(codaSamples[[1]][2001:10000,],2,sd)  
hpds=HPDinterval(mcmc(codaSamples[[1]][2001:10000,]))  
meds=apply(codaSamples[[1]][2001:10000,],2,median)  
df <- data.frame(means, meds, hpds, sds, row.names = c("Beta[1]", "Beta[2]", "Beta[3]", "Beta[4]", "Beta[5]", "Sigma"))  
df[6,] = 1/sqrt(df[6,])  
  
colnames(df) <- c("Mean", "Median", "Lower 95% CI", "Upper 95% CI", "Standard Deviation")  
kable(round(df,2))

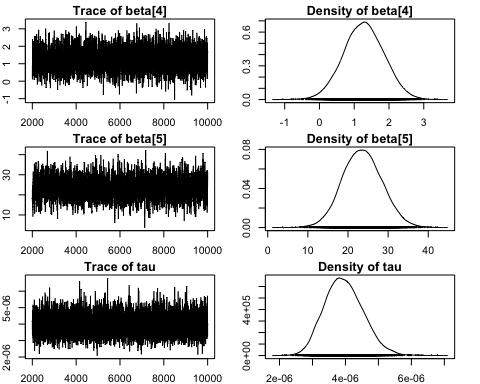
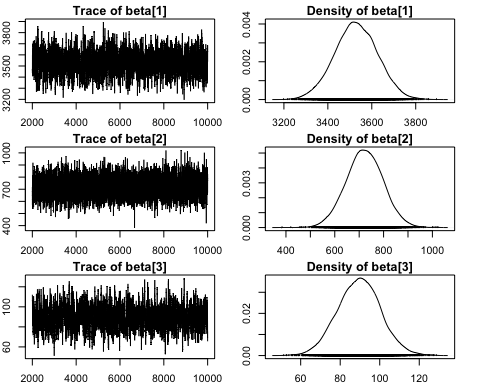
|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Mean | Median | Lower 95% CI | Upper 95% CI | Standard Deviation |
| Beta[1] | 3528.72 | 3528.53 | 3333.09 | 3710.29 | 96.40 |
| Beta[2] | 722.30 | 722.30 | 577.69 | 871.45 | 75.29 |
| Beta[3] | 89.54 | 89.64 | 68.45 | 111.03 | 10.80 |
| Beta[4] | 1.28 | 1.28 | 0.12 | 2.36 | 0.57 |
| Beta[5] | 23.59 | 23.52 | 13.96 | 32.94 | 4.88 |
| Sigma | 502.69 | 504.45 | 588.49 | 442.45 | 1312.77 |

## MCMC Model Diagnostics

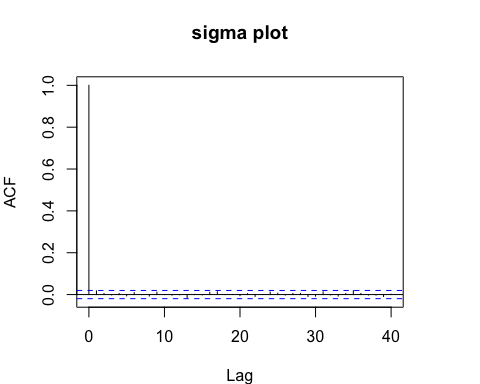
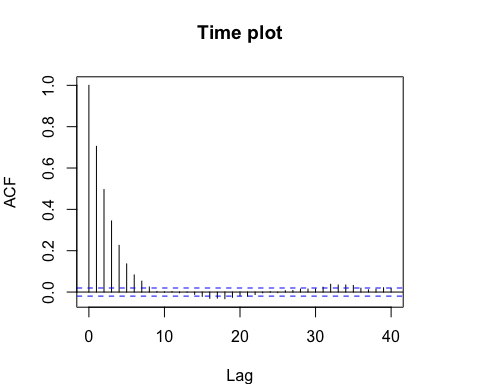
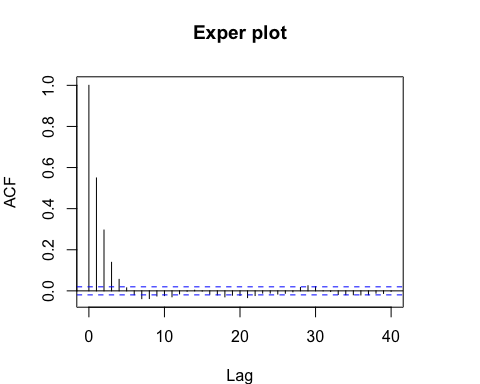
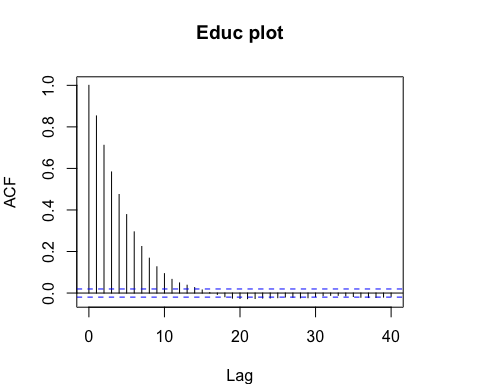
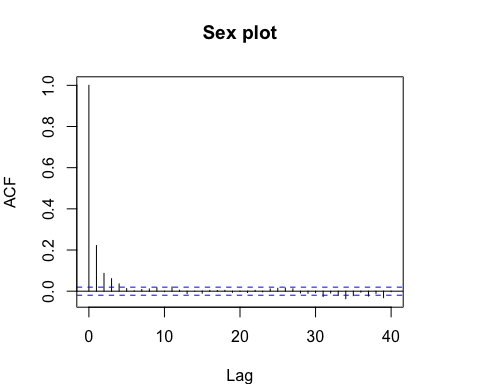
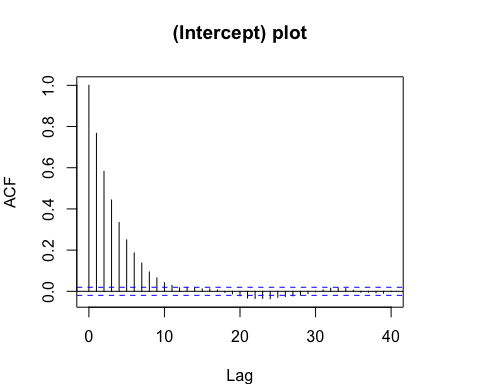
#Quick examination of the output: are the results sensible?  
#head(codaSamples)  
  
#Examine the summary output   
burn.in <- 2001  
#Examining one chain  
summary(window(codaSamples[[1]],start=burn.in))

##   
## Iterations = 2001:10000  
## Thinning interval = 1   
## Number of chains = 1   
## 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  
## beta[1] 3.529e+03 9.640e+01 1.078e+00 2.930e+00  
## beta[2] 7.223e+02 7.529e+01 8.417e-01 1.147e+00  
## beta[3] 8.954e+01 1.080e+01 1.208e-01 3.592e-01  
## beta[4] 1.276e+00 5.745e-01 6.423e-03 1.071e-02  
## beta[5] 2.359e+01 4.880e+00 5.456e-02 1.240e-01  
## tau 3.957e-06 5.803e-07 6.487e-09 6.487e-09  
##   
## 2. Quantiles for each variable:  
##   
## 2.5% 25% 50% 75% 97.5%  
## beta[1] 3.337e+03 3.463e+03 3.529e+03 3.594e+03 3.716e+03  
## beta[2] 5.740e+02 6.722e+02 7.223e+02 7.730e+02 8.695e+02  
## beta[3] 6.838e+01 8.227e+01 8.964e+01 9.677e+01 1.110e+02  
## beta[4] 1.477e-01 8.888e-01 1.277e+00 1.666e+00 2.404e+00  
## beta[5] 1.425e+01 2.021e+01 2.352e+01 2.680e+01 3.330e+01  
## tau 2.921e-06 3.547e-06 3.930e-06 4.338e-06 5.165e-06

# Some convergence diagnostics  
par(mar=c(2.0,2,1.5,2))  
plot(window(codaSamples[[1]],start=burn.in)) # plot sample path and density



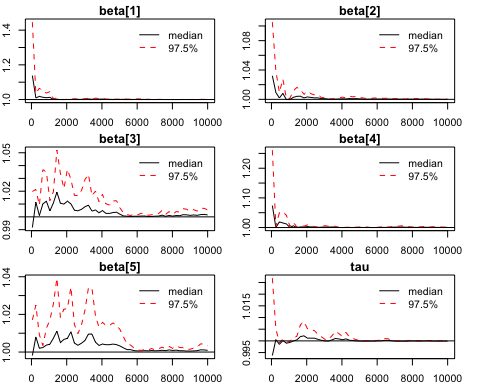
# Auto-correlation function  
k <- 6 # number of parameters  
paramNames = c(names(vanilla.reg$coefficients), "sigma")  
par(mar=rep(4,4))  
for (i in seq(1,k)){  
 acf(codaSamples[[1]][,i], main =paste0(paramNames[i], " plot"))  
}



# Caculate GR Statistic  
par(mar=c(2.0,2,1.5,2))  
gelman.diag(codaSamples)

## Potential scale reduction factors:  
##   
## Point est. Upper C.I.  
## beta[1] 1 1.00  
## beta[2] 1 1.00  
## beta[3] 1 1.01  
## beta[4] 1 1.00  
## beta[5] 1 1.00  
## tau 1 1.00  
##   
## Multivariate psrf  
##   
## 1

gelman.plot(codaSamples)

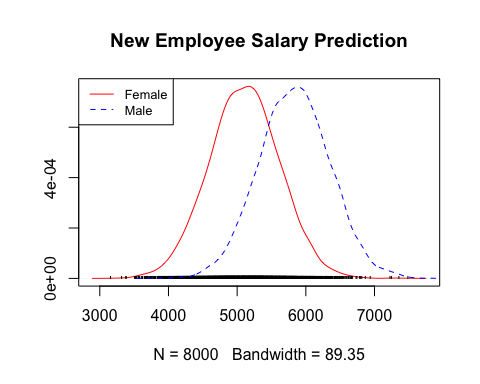
 MCMC Diagnostics indicate no obvious issues with convergance or correlation.

## Compare predictions for future employees

monitorNew = c("salary") # The parameter(s) to be monitored.  
nChains = 3 # Number of chains to run.  
nIter = 10000 #Steps to save per chain.   
salary[94:95] <- NA  
x1[94:95] <- c(0,1)  
x2[94:95] <- c(12,12)  
x3[94:95] <- c(100,100)  
x4[94:95] <- c(16,16)  
n <- length(salary)  
X <- cbind(1, x1, x2, x3, x4)  
model.txt<-textConnection(modelString)  
burn.in <- 2001  
  
beta.mu.New <- lm(salary ~ x1 + x2 + x3 + x4)$coef # what does this line of code do?  
dataList.New = list(salary = salary, X = X, n = n, beta.mu = beta.mu.New)  
  
jagsModelNew = jags.model(model.txt, data = dataList.New, n.chains = nChains, quiet = TRUE)  
codaSamplesNew = coda.samples(jagsModelNew, var = monitorNew, n.iter = nIter)  
summary(window(codaSamplesNew[[1]][,94:95],start=burn.in))

##   
## Iterations = 2001:10000  
## Thinning interval = 1   
## Number of chains = 1   
## 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  
## salary[94] 5105 513.3 5.739 5.739  
## salary[95] 5834 516.8 5.778 5.655  
##   
## 2. Quantiles for each variable:  
##   
## 2.5% 25% 50% 75% 97.5%  
## salary[94] 4109 4764 5106 5446 6098  
## salary[95] 4832 5486 5835 6180 6847

plot(window(codaSamplesNew[[1]][,94],start=burn.in), main = "New Employee Salary Prediction", trace = FALSE, col = "red" )  
legend("topleft", legend=c("Female", "Male"),  
 col=c("red", "blue"), lty=1:2, cex=0.8)  
lines(density(window(codaSamplesNew[[1]][,95],start=burn.in)), lty = 2, col = "blue")



TFFemale = window(codaSamplesNew[[1]][,94],start=burn.in) > 5800  
femaleProb = length(which(TFFemale =="TRUE" ))/length(TFFemale)  
  
TFMale = window(codaSamplesNew[[1]][,95],start=burn.in) > 5800  
maleProb = length(which(TFMale =="TRUE" ))/length(TFMale)  
  
print(femaleProb)

## [1] 0.086625

print(maleProb)

## [1] 0.52675

## Conclusions

The data suggests a significant difference between the pay for men and women. First, the coefficient attached to the 'sex' data is actually the most heavily weighted coefficient. Secondly future predictions of the model based upon the same qualifications for men and women indicate that men are likely to be paid higher for the same work.