OM 386 Assignment 5

By: Callie Gilmore (cgg756)

Due: May 5th, 11:59pm

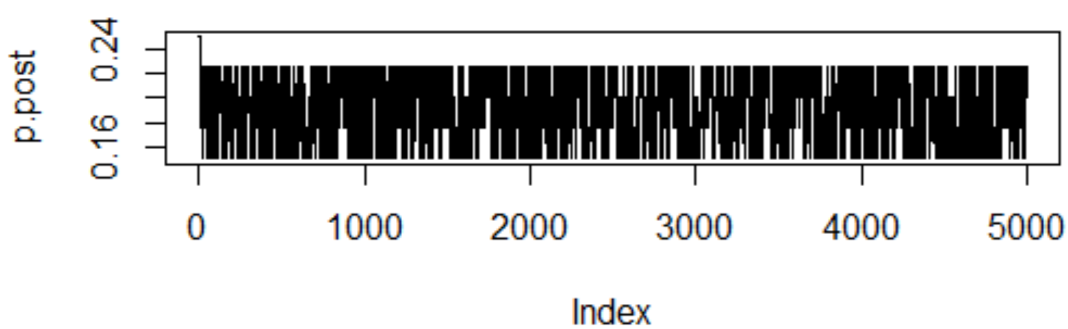
Metropolis-Hastings Algorithm for Bernoulli Estimation

1). We will practice coding the Metropolis-Hastings (MH) algorithm, which is Markov Chain Monte Carlo method, for estimating the probability of a binary variable being equal to 1. The binary variable used in this exercise is “Latepay” in the dataset "CreditCard_LatePayment_Data.csv".

Assume the probability of the binary variable can only take values from a set of discretized values between 0 and 1. For example, this probability p can only be (0, 0.1, 0.2, 0.3,…, 0.9, 1.0). The MH algorithm uses a random walk to generate a proposal p value. The random walk allows the p value to jump to right of the current value with the probability=0.5 and jump to the left of the current value with the probability=0.5. For example, if the current p value is 0.2, it can jump to 0.1 with the probability=0.5 and jump to 0.3 with the probability=0.5. It cannot jump to any other values that are not the neighbors the current value. If the current value is 0, it can jump to the next higher value with the probability=0.5 or stay at 0 with the probability=0.5. If the current value is 1, it can jump to the next lower value with the probability=0.5 or stay at 1 with the probability=0.5. The MH algorithm compares the likelihood at the proposal p value and the current p value and decides which value is accepted as the posterior sample.

Please complete the code in “Assignment-5_MH-Bernoulli-code_blanks.r” to implement this random walk Metropolis Hastings MCMC algorithm. Use the plot() function to plot the posterior sampling chains and calculate the posterior mean of the probability of the binary variable. Please copy and paste the plot and result here.

```
> plot(p.post, type="l")
```




```
> mean(p.post[1001:n.step])
[1] 0.1902375
```

2). Next, we will practice coding the Metropolis-Hastings (MH) algorithm for estimating the coefficients in a logistic regression model. We will still use the dataset "CreditCard_LatePayment_Data.csv".

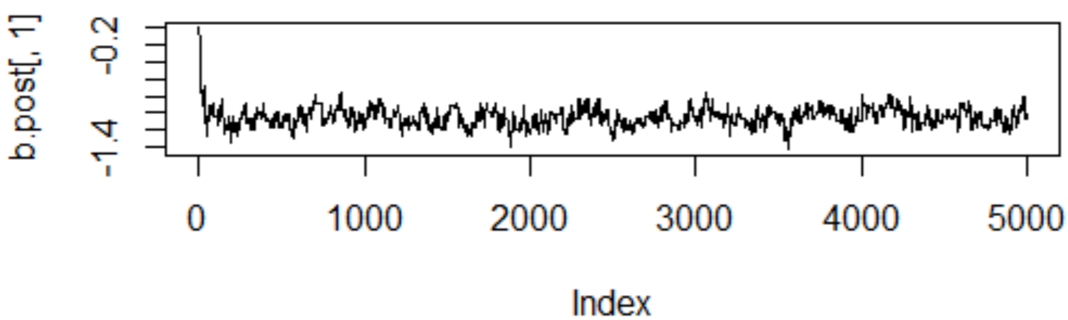
First, please use the glm( ) function in R run a logistic regression:
glm(Latepay~Usage+Balance, family=binomial(link= "logit "))
to find the MLE estimate of the regression coefficients for "Usage" and "Balance". Please copy and paste the results here.

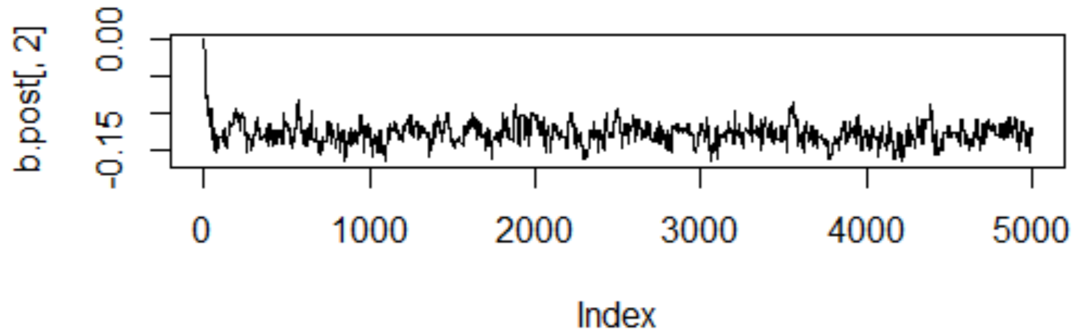
Now, we will fit this logistic model using a Metropolis-Hastings algorithm. Parts of the R code are in "Assignment-5_ MH-logistic-code_blanks.r". Please read the code carefully and fill in the code in the blanks in the file.

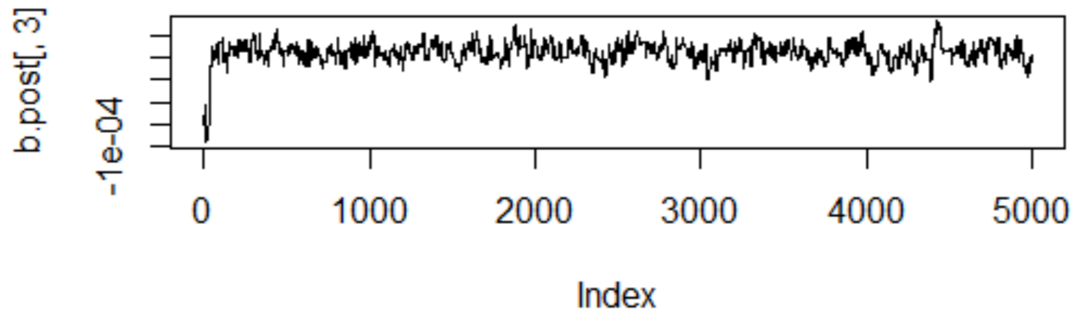
Please run the completed code. Use the plot() function to plot the posterior sampling chains and hist() to plot posterior histograms for the regression coefficients for "Usage" and "Balance". Copy and paste the results here.

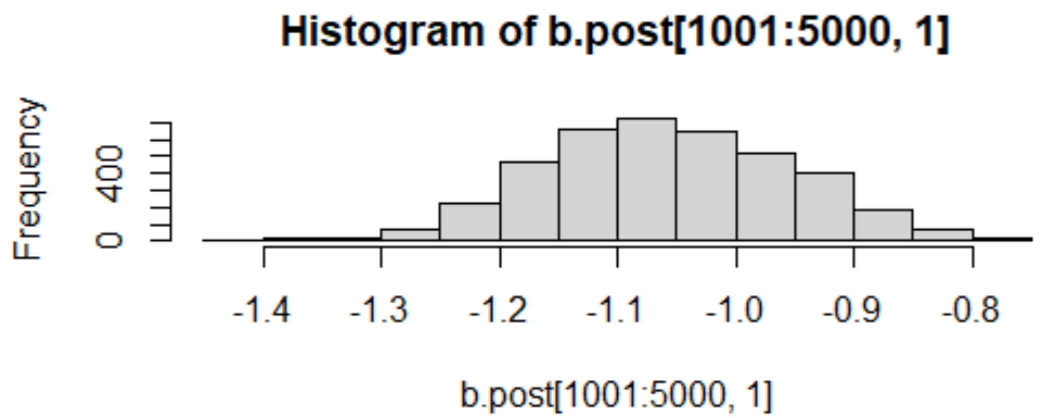
```
> mean(b.post[1001:5000,1])
[1] -1.057092
> mean(b.post[1001:5000,2])
[1] -0.1281701
> mean(b.post[1001:5000,3])
[1] 0.0003239612
```

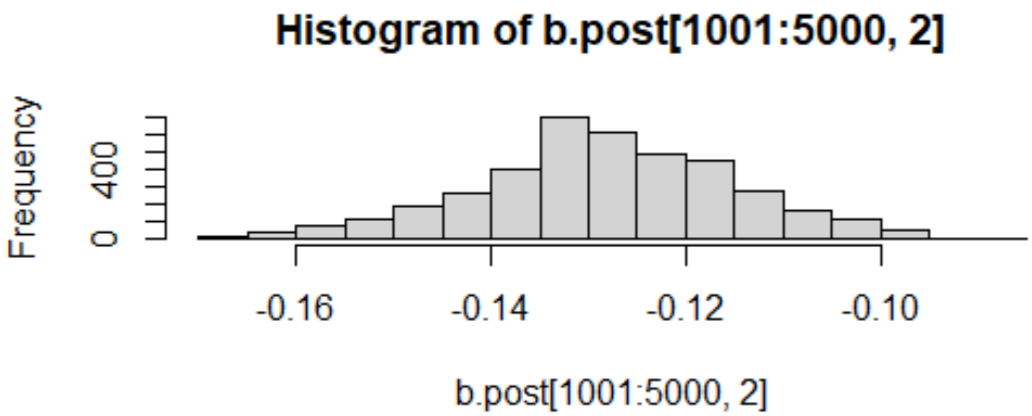
```
> plot(b.post[,1], type="l")
> plot(b.post[,2], type="l")
> plot(b.post[,3], type="l")
> hist(b.post[1001:5000,1])
> hist(b.post[1001:5000,2])
> hist(b.post[1001:5000,3])
```

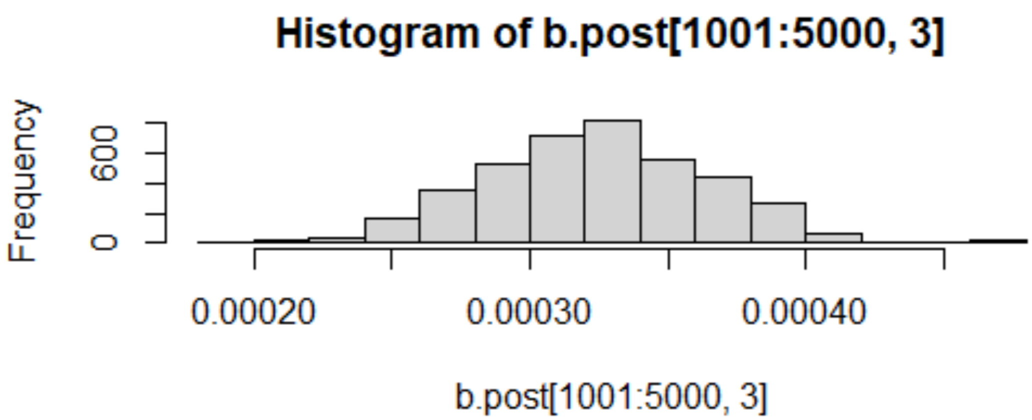



b.post[, 2]
0.00
-0.15
0
1000
2000
3000
4000
5000
Index


b.post[, 3]
-1e-04
0
1000
2000
3000
4000
5000
Index


Histogram of b.post[1001:5000, 1]
Frequency
0
400
-1.4
-1.3
-1.2
-1.1
-1.0
-0.9
-0.8
b.post[1001:5000, 1]


Histogram of b.post[1001:5000, 2]
Frequency
0
400
-0.16
-0.14
-0.12
-0.10
b.post[1001:5000, 2]


Histogram of b.post[1001:5000, 3]
Frequency
0
600
0.00020
0.00030
0.00040
b.post[1001:5000, 3]