Computational methods in Statistics

Yukang Xu (002462280)
Dept of Math and Stat, Georgia State University, Atlanta, GA 30302-4110, USA

1 Introduction

Bayesian Analysis is a formal method for combining prior beliefs with observed information. It can't very realistic but complicated models. In this project we will do Bayesian data analysis using Markov Chain Monte Carlo to speculate whether the difference between points of Stephen Curry and points of James Harden increased or decreased in NBA from 2014 to 2019. The data sets we use are points they get in NBA regular season from year 2014 to year 2019. The data sets are downloaded from http://www.stat-nba.com. For each year, we use the difference (in ten) between the points Stephen Curry and James Harden. For every season, the games they attended are different. There are more than 60 observations every year.
Now we formulate our questions as:

1.Is the mean of differences in year 2014 greater than the mean of differences in year 2015?
If so, how much?

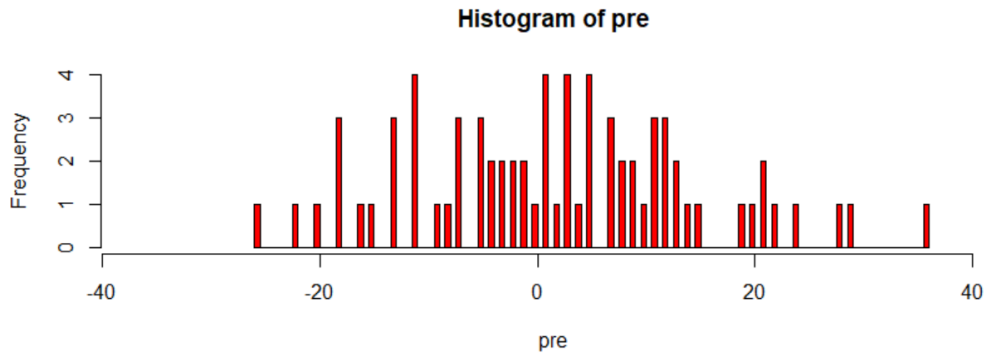
2.Is the mean of differences in year 2016 greater than the mean of differences in year 2017?
If so, how much?

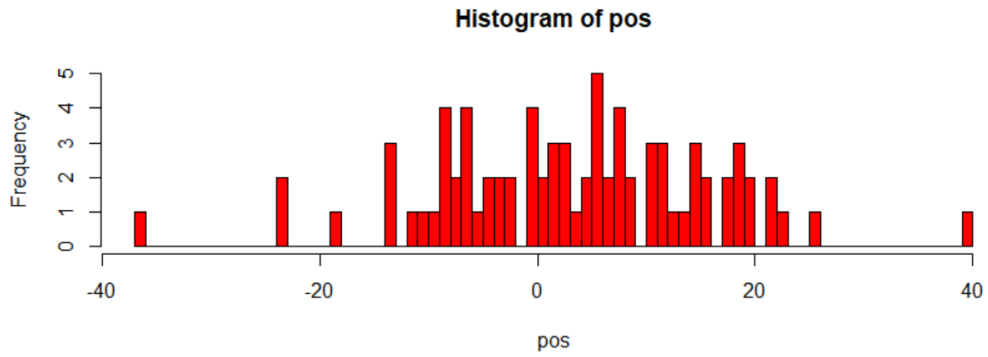
3.Is the mean of differences in year 2018 greater than the mean of differences in year 2019?
If so, how much?

2 Data Analysis

2.1 Comparison of differences in 2014 and 2015

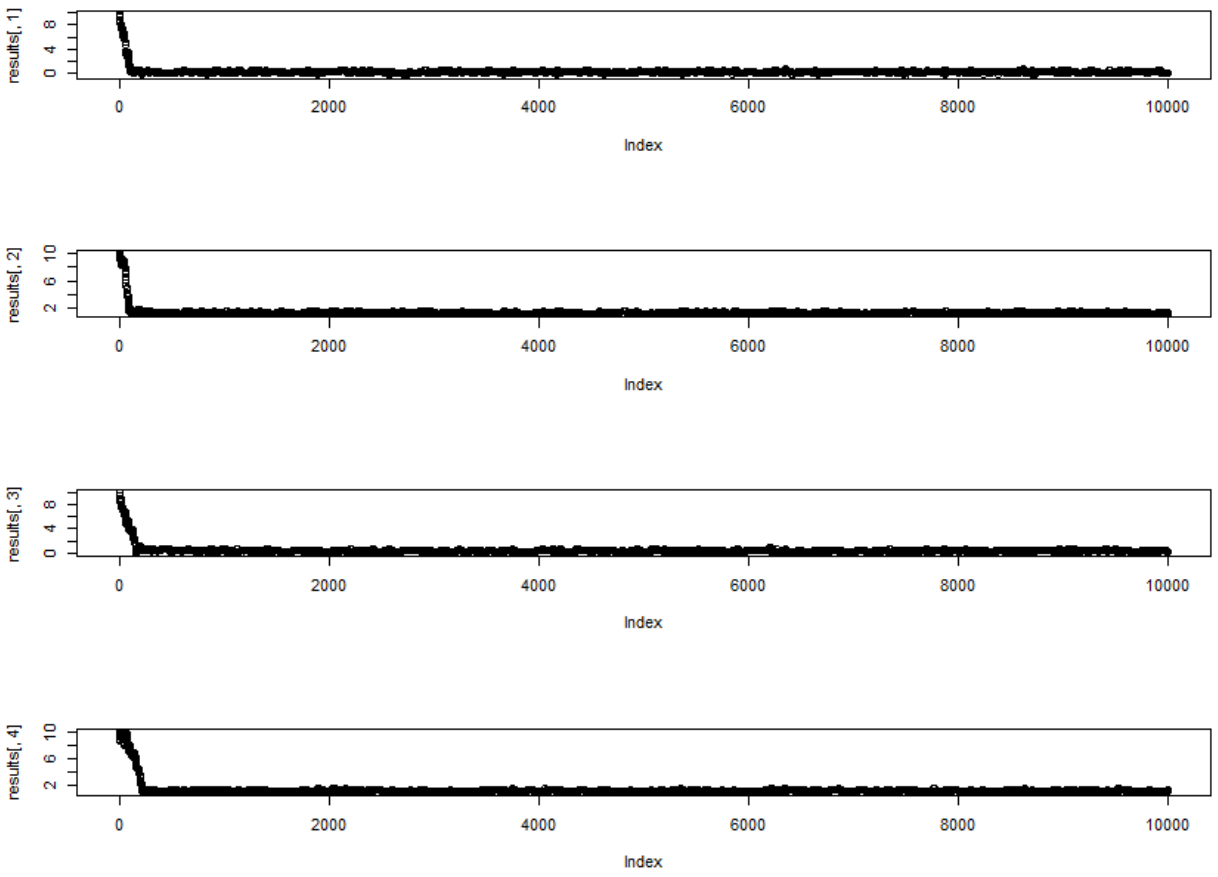
The histograms of differences in 2014 and 2015 are given below.



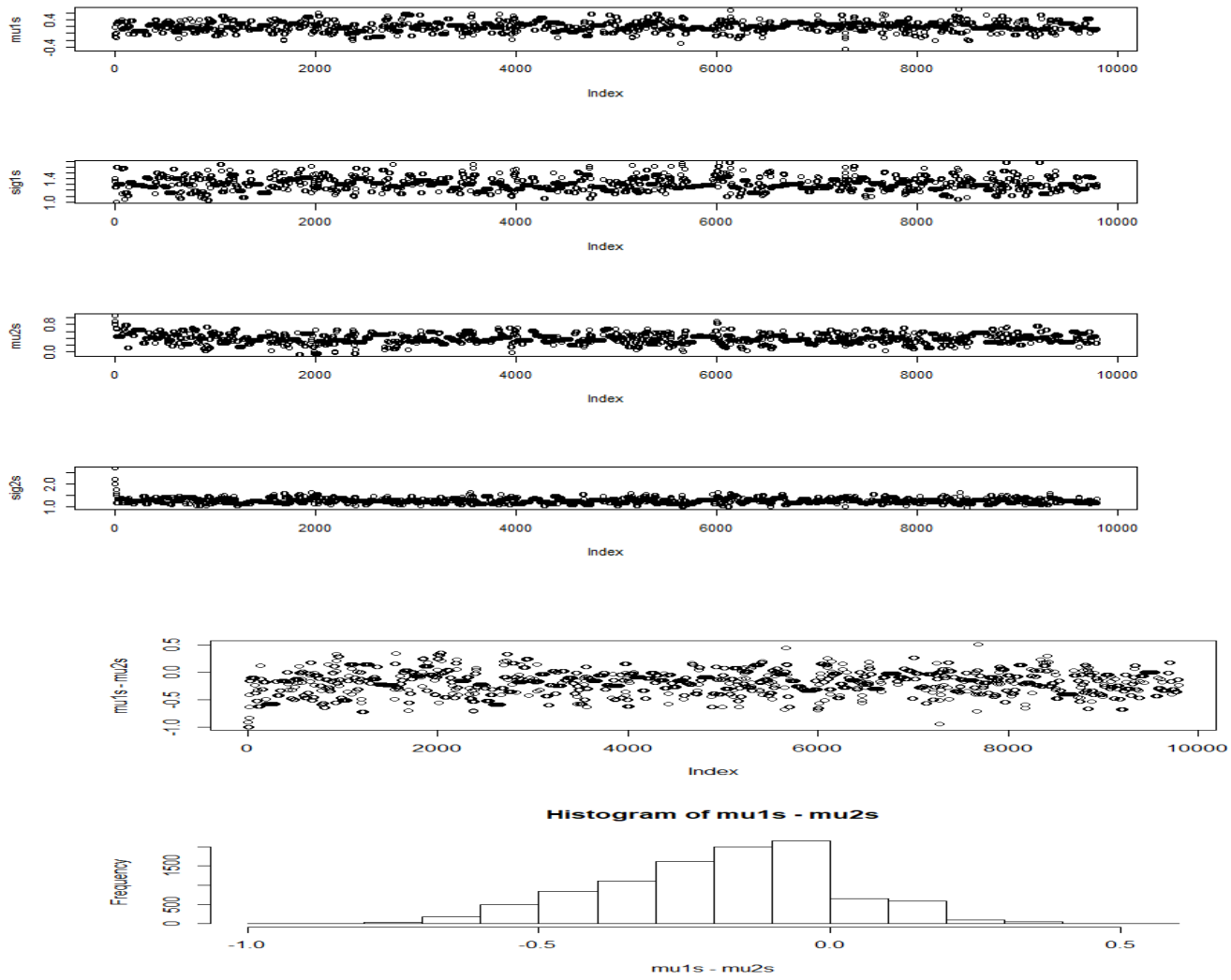


The conclusion is not so obvious from histograms. We may assume that the differences in each year follows normal distribution $N(\mu, \delta^2)$. Let us assume that the difference in 2014 come from $N(\mu_1, \delta_{1q}^2)$ and the differences in 2015 come from $N(\mu_2, \delta_2^2)$. So our parameter vector is $\Theta = (\mu_1, \delta_1, \mu_2, \delta_2)$.
We will construct a likelihood function and a prior and then multiply them together to get posterior $P(\Theta|Data)$.
Let x denote the difference in 2014 and let y denote the difference in 2015. Then the likelihood is given by the formula: $P(Data|\Theta) = P(x|\Theta)P(y|\Theta)$
where the second equation holds since x_i and y_j are independent for i = 1, 2,… 68 and j = 1, 2, …., 51. We can write the likelihood function according to the above formula.
Then we need to decide on a prior density for $\Theta = (\mu_1, \delta_1, \mu_2, \delta_2)$, and write it as a function.
The basic thinking is simply to choose priors for the μ_1 and μ_2 that do not seem to prejudge which is bigger, and to choose priors that are sort of uniform over all even remotely plausible values, with the aim of letting the data produce any interesting features in the posterior distribution.
We choose the distribution of μ_1 as N(1.9, 1.9^2) and choose the distribution of μ_2 as N(3.8, 3.8^2), where 1.9 is the sample mean of x and 3.8 is the sample mean for y. We choose the distributions like this because we want the distribution as at as possible, otherwise the prior will control the posterior. Then we choose the distribution of δ_1 as exponential with mean 1.9 and choose the distribution of δ_2 as exponential with mean 3.8. Thus all 4 variables are independent.

Next, we multiply prior times likelihood to get the posterior. We will run a Markov chain using Metropolis for 10000 iterations to simulate a sample. First, we choose a starting value Θ_0 = (1.9, 1.9, 3.8, 3.8). Given a current state, we need to decide on a way to propose a “candidate" move, then evaluate the posterior of candidate move and the posterior of current state and take the ratio. We will record our results in a big matrix with 4 columns and 10000 rows. The first row is the starting value and each successive row will record the next Θ as we run the chain. Now we have got a bunch of parameter vectors. Let's firstly take a look at how the chain ran.



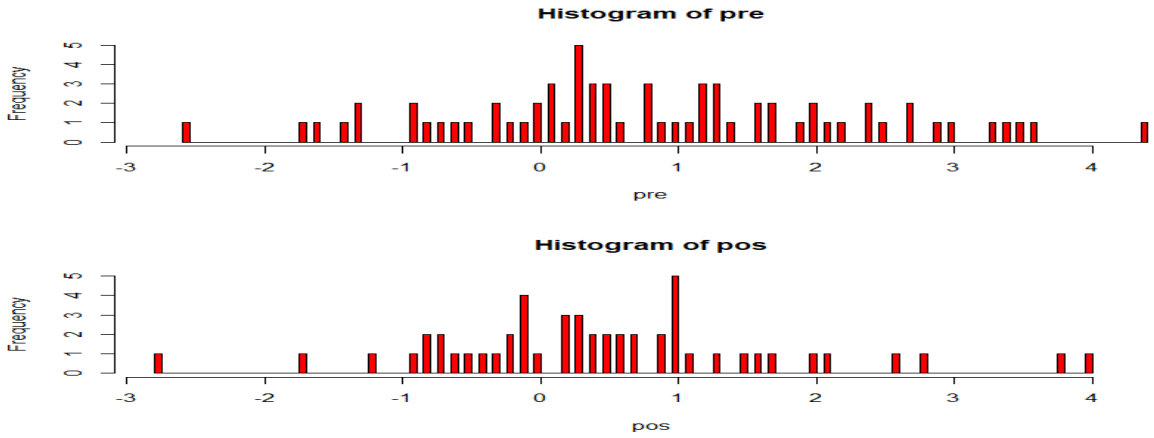
It looks like the First few hundred iterations may be noticeably influenced by our starting values. Then we throw away the first 200 iterations and let the remaining be our new results. Then we could base our inferences on the new results instead of the whole results matrix.



Our estimated posterior probability that $\mu_1 - \mu_2 < 0$ is about -0.1. Thus with high probability the mean of differences in year 2015 is smaller than the mean of difference in year 2014.

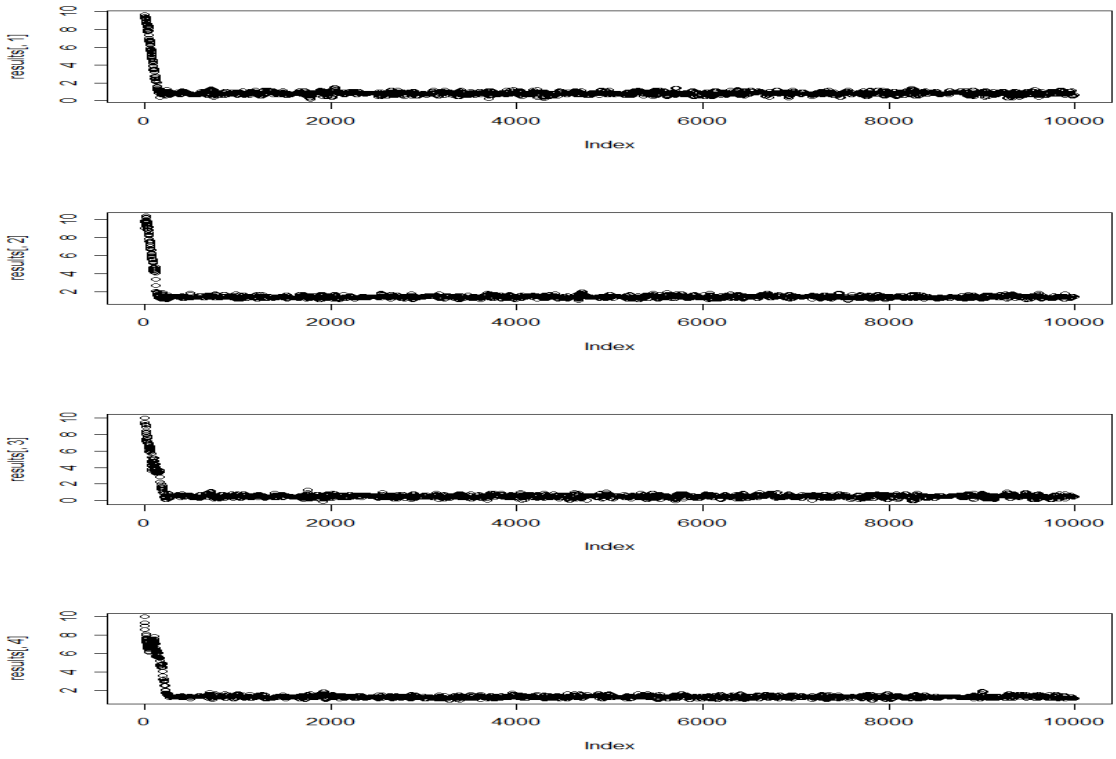
2.2 Comparison of differences in 2016 and 2017

The histograms of differences in 2016 and 2017 are given below.
The conclusion is not so obvious from histograms. We may assume that the difference in each year follow normal distribution $N(\mu; \delta^2)$. Let us assume that the difference in 2016 come from $N(\mu_1, \delta_{1q}^2)$ and the difference in 2017 come from $N(\mu_2, \delta_{2q}^2)$. So our parameter vector is $\Theta = (\mu_1, \delta_1, \mu_2, \delta_2)$. To get posterior, we still need to construct a likelihood function and a prior and then multiply them together. The likelihood function we use is exactly the same as the likelihood function in the previous subsection. To decide the prior density for $\Theta = (\mu_1, \delta_1, \mu_2, \delta_2)$, we choose the



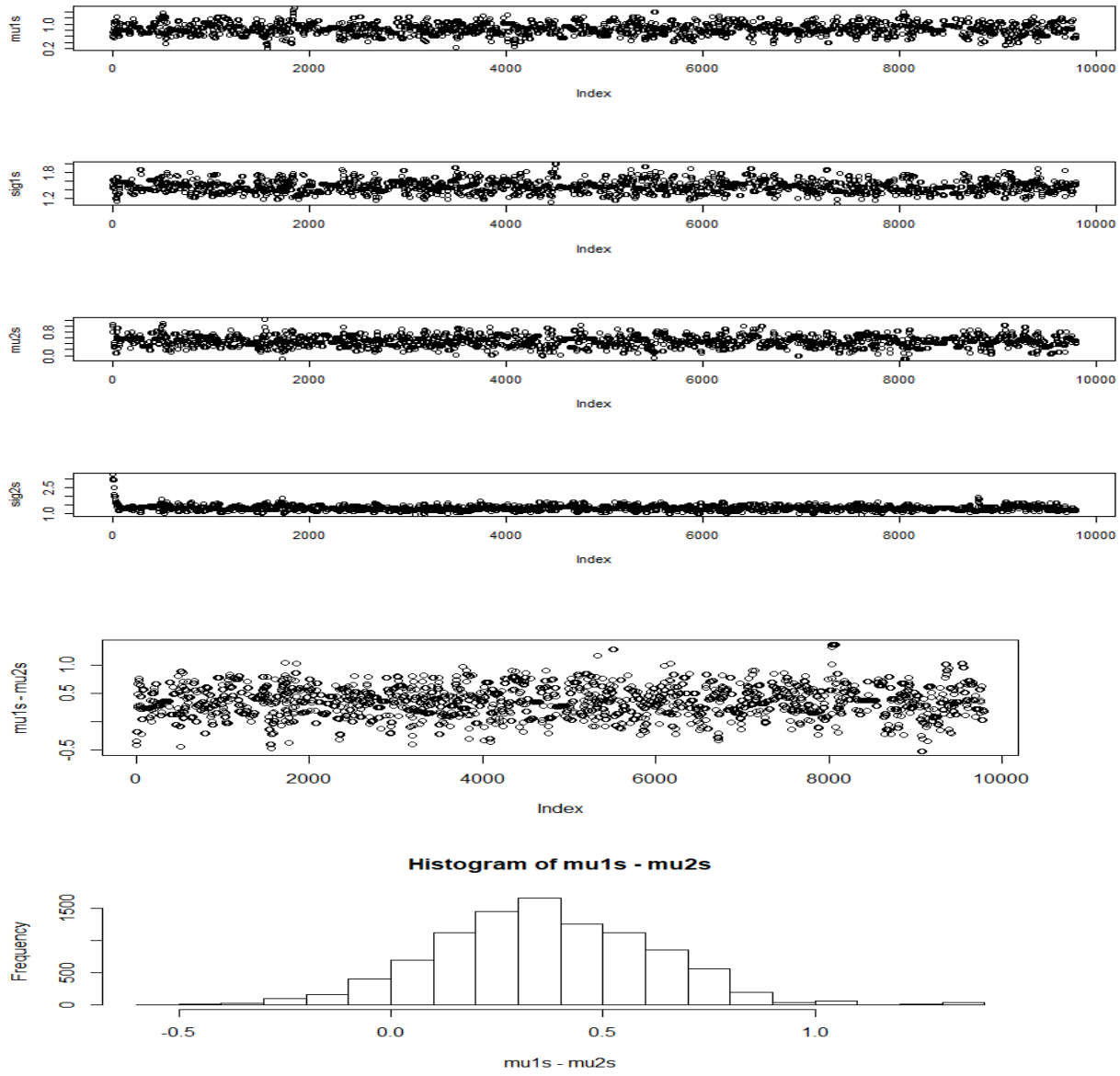
distribution of μ1 as N (1.4, 1.4) and choose the distribution of μ2 as N (4, 4), where 1.4 is the sample mean of differences in 2006 and 4 is the sample mean for differences in 2017. Then we choose the distribution of σ1 as exponential with mean 1.4 and choose the distribution of σ2 as exponential with mean 4. Next, we multiply prior times likelihood to get the posterior.

We still run a Markov chain using Metropolis for 10000 iterations to simulate a sample. This time we choose a starting value θ0 = (1.4,1.4,4,4). The results are shown below.



It looks like the first few hundred iterations may be noticeably influenced by our starting values. Then we throw away the first 200 iterations and let the remaining be our new results.

Then we could base our inferences on the new results instead of the whole results matrix..

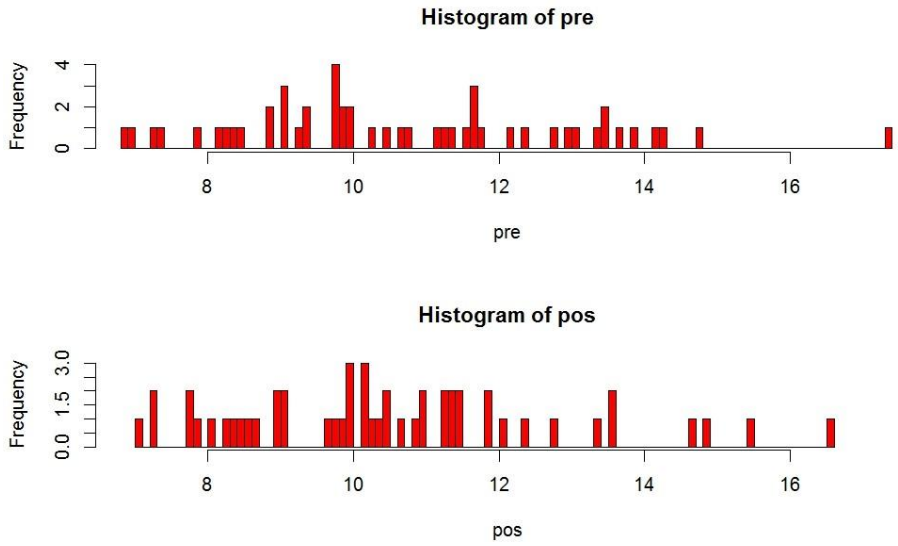


Our estimated posterior probability that $\mu 1 - \mu 2 < 0$ is about 0.3. Thus with high proba- bility the mean of differences in year 2007 is greater than the mean of difference in year 2006.

2.3 Comparison of differences in 2018 and 2019

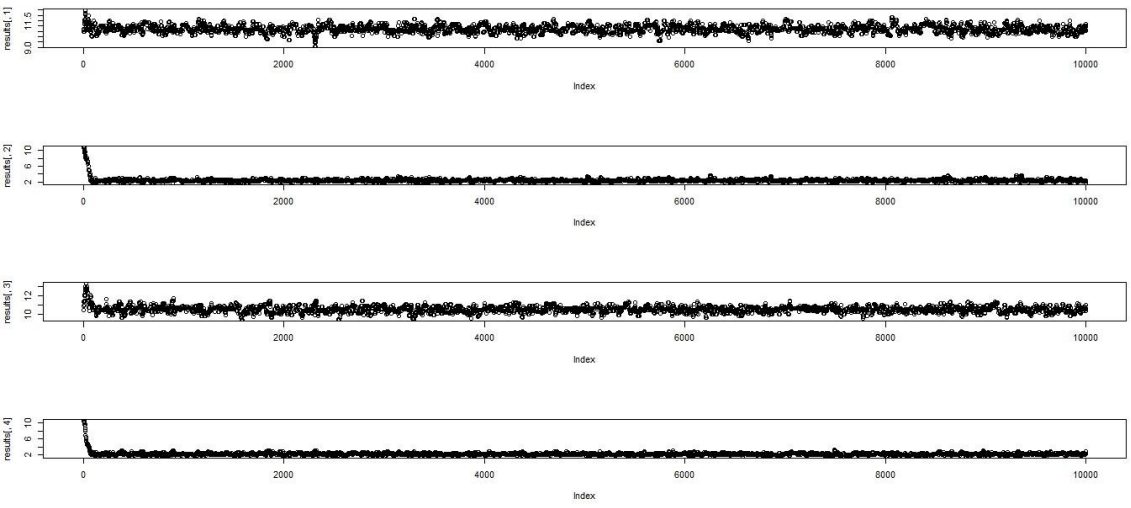
The histograms of differences in 2018 and 2019 are given below.

The conclusion is not so obvious from histograms. We may assume that the difference in each year follow normal distribution N (μ, $\sigma 2$). Let us assume that the difference in 2018 come from N ($\mu 1$, $\sigma 2$) and the difference in 2019 come from N ($\mu 2$, $\sigma 2$). So our parameter vector is $\theta = (\mu 1, \sigma 1, \mu 2, \sigma 2)$. To decide the prior density for $\theta = (\mu 1, \sigma 1, \mu 2, \sigma 2)$, we choose the distribution of $\mu 1$ as N (4.9, 4.9) and choose the distribution of $\mu 2$ as N (5.2, 5.2), where 4.9 is the sample mean of differences in 2018 and 5.2 is the sample mean for differences in 2019. Then we



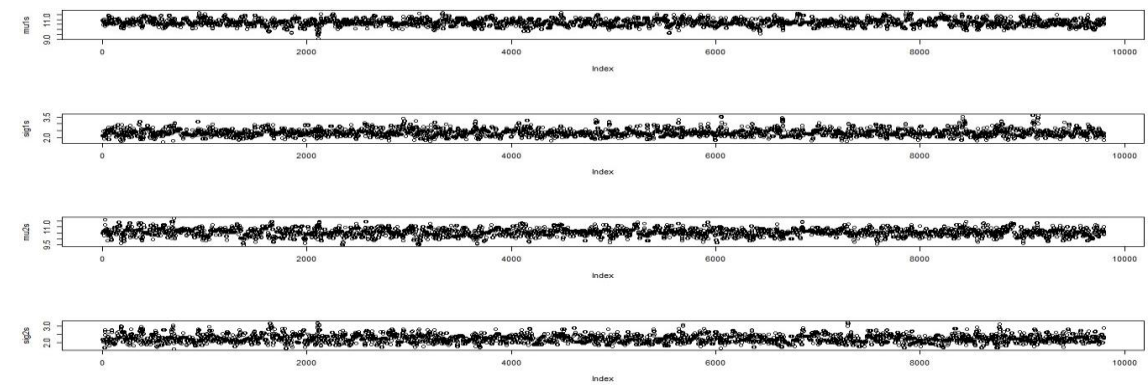
choose the distribution of σ_1 as exponential with mean 4.9 and choose the distribution of σ_2 as exponential with mean 5.2. Next, we multiply prior times likelihood to get the posterior.

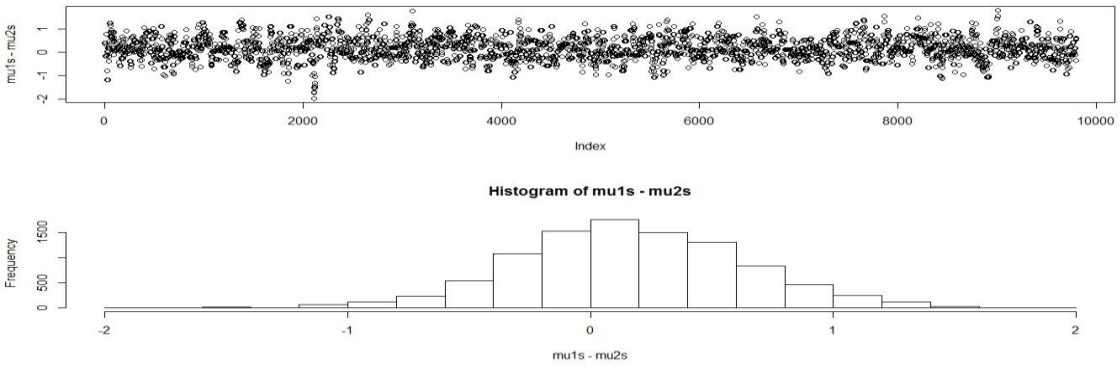
We still run a Markov chain using Metropolis for 10000 iterations to simulate a sample. This time we choose a starting value θ_0 = (4.9,4.9,5.2,5.2). The results are shown below.



It looks like the first few hundred iterations may be noticeably influenced by our starting values. Then we throw away the first 200 iterations and let the remaining be our new results.

Then we could base our inferences on the new results instead of the whole results matrix.





Our estimated posterior probability that $\mu_1 - \mu_2 < 0$ is about 0.2. Thus with high probability the mean of differences in year 2019 is less than the mean of difference in year 2018.

3. Conclusion

According to our discussion in section 2, we get the conclusion that

- The mean of differences in year 2015 is smaller than the mean of differences in year 2014.
- The mean of differences in year 2017 is greater than the mean of differences in year 2016.
- The the mean of differences in year 2019 is less than the mean of differences in year 2018.

Thus overall, the difference between Stephen Curry and James Harden was increasing at the beginning and then decreasing.

4.Code:

```
#2.1
data=read.csv('C:/Users/xuyuk/Desktop/final.csv',header=TRUE)
summary(data)
pre=(data[1:73,12]-data[1:73,11])
pos=(data[1:80,10]-data[1:80,9])
dif=rbind(c(pre,pos))
mean(pre)
mean(pos)

### draw a p i c t u r e
xlim=c(min(dif),max(dif))
par(mfrow=c(2,1))
hist(pre,100,col="red",xlim=xlim)
hist(pos,100,col="red",xlim=xlim)

### l i k e l i h o o d f unc t i on
lik=function(th){
mu1=th[1];sig1=th[2];mu2=th[3];sig2=th[4]
prod(dnorm(pre,mean=mu1,sd=sig1))*prod(dnorm(pos,mean=mu2,sd=sig2))
}

### p r i o r f unc t i on
prior=function(th){
mu1=th[1];sig1=th[2];mu2=th[3];sig2=th[4]
if(sig1<=0|sig2<=0)return(0)
dnorm(mu1,1.9,1.9)*dnorm(mu2,3.8,3.8)*dexp(sig1,rate=1/1.9)*dexp(sig2,rate=1/3.8)
}
### p o s t e r i o r f unc t i on
```

```
post=function(th){prior(th)*lik(th)}
#S t a r t i n g v a l u e s
mu1=1.9;sig1=1.9;mu2=3.8;sig2=3.8
th0=c(mu1,sig1,mu2,sig2)
# Here i s what does the MCMC (Me t ropol i s method ) :
nit=10000
results=matrix(0,nrow=nit,ncol=4)
th=th0
results[1,]=th0
for(it in 2:nit){
cand=th+rnorm(4,sd=.3)
ratio=post(cand)/post(th)
if(runif(1)<ratio)th=cand
results[it,]=th
}
# Take a peek at what we got
edit(results)
par(mfrow=c(4,1))
plot(results[,1])
plot(results[,2])
plot(results[,3])
plot(results[,4])
res=results[201:10000,]
mu1s=res[,1]
sig1s=res[,2]
mu2s=res[,3]
sig2s=res[,4]
par(mfrow=c(4,1))
plot(mu1s)
```

```
plot(sig1s)
plot(mu2s)
plot(sig2s)
par(mfrow=c(2,1))
plot(mu1s−mu2s)
hist(mu1s−mu2s)
mean(mu1s−mu2s<0)



#2.2
data=read.csv('C:/Users/xuyuk/Desktop/final.csv',header=TRUE)
summary(data)
pre=(data[1:79,8]-data[1:79,7])
pos=(data[1:79,6]-data[1:79,5])
dif=rbind(c(pre,pos))
mean(pre)
mean(pos)

### draw a p i c t u r e
xlim=c(min(dif),max(dif))
par(mfrow=c(2,1))
hist(pre,100,col="red",xlim=xlim)
hist(pos,100,col="red",xlim=xlim)

### l i k e l i h o o d f u n c t i o n
lik=function(th){
mu1=th[1];sig1=th[2];mu2=th[3];sig2=th[4]
prod(dnorm(pre,mean=mu1,sd=sig1))∗prod(dnorm(pos,mean=mu2,sd=sig2))
}
```

```
### p r i o r f u n c t i o n
prior=function(th){
mu1=th[1];sig1=th[2];mu2=th[3];sig2=th[4]
if(sig1<=0|sig2<=0)return(0)
dnorm(mu1,9.9,9.9)∗dnorm(mu2,10.5,10.5)∗dexp(sig1,rate=1/9.9)∗dexp(sig2,rate=1/10.5)
                                        }

### p o s t e r i o r f u n c t i o n
post=function(th){prior(th)∗lik(th)}

#S t a r t i n g v a l u e s
mu1=9.9;sig1=9.9;mu2=10.5;sig2=10.5
th0=c(mu1,sig1,mu2,sig2)
# Here i s what d o e s t h e MCMC ( M e t r o p o l i s method ) :
nit=10000
results=matrix(0,nrow=nit,ncol=4)
th=th0
results[1,]=th0
for(it in 2:nit)
{
cand=th+rnorm(4,sd=0.5)
ratio=post(cand)/post(th)
if(runif(1)<ratio)
th=cand
results[it,]=th
}
# Take a p e e k a t what we g o t
edit(results)
```

```
par(mfrow=c(4,1))
plot(results[,1])
plot(results[,2])
plot(results[,3])
plot(results[,4])
res=results[201:10000,]
mu1s=res[,1]
sig1s=res[,2]
mu2s=res[,3]
sig2s=res[,4]

par(mfrow=c(4,1))
plot(mu1s)
plot(sig1s)
plot(mu2s)
plot(sig2s)

par(mfrow=c(2,1))
plot(mu1s−mu2s)
hist(mu1s−mu2s)
mean(mu1s−mu2s<0)



#2.3
data=read.csv('C:/Users/xuyuk/Desktop/final.csv',header=TRUE)
summary(data)
pre=(data[1:51,4]-data[1:51,3])
pos=(data[1:68,2]-data[1:68,1])
```

```
dif=rbind(c(pre,pos))
mean(pre)
mean(pos)

### draw a p i c t u r e
xlim=c(min(dif),max(dif))
par(mfrow=c(2,1))
hist(pre,100,col="red",xlim=xlim)
hist(pos,100,col="red",xlim=xlim)

### l i k e l i h o o d f u n c t i o n
lik=function(th){
mu1=th[1];sig1=th[2];mu2=th[3];sig2=th[4]
prod(dnorm(pre,mean=mu1,sd=sig1))*prod(dnorm(pos,mean=mu2,sd=sig2))
}

### p r i o r f u n c t i o n
prior=function(th){
mu1=th[1];sig1=th[2];mu2=th[3];sig2=th[4]
if(sig1<= 0|sig2<= 0)return(0)
dnorm(mu1,10.7,10.7)*dnorm(mu2,10.5,10.5)*dexp(sig1,rate=1/10.7)*dexp(sig2,rate=1/10.5)
}

### p o s t e r i o r f u n c t i o n
post=function(th){prior(th)*lik(th)}

#S t a r t i n g v a l u e s
mu1=10.7;sig1=10.7;mu2=10.5;sig2=10.5
th0=c(mu1,sig1,mu2,sig2)
```

```
# Here i s what d o e s t h e MCMC ( M e t r o p o l i s method ) :
nit=10000
results=matrix(0,nrow=nit,ncol=4)
th=th0
results[1,]=th0
for(it in 2:nit){
cand=th+rnorm(4,sd=.4)
ratio=post(cand)/post(th)
if(runif(1)<ratio)th=cand
results[it,]=th
}
# Take a p e e k a t what we g o t
edit(results)
par(mfrow=c(4,1))
plot(results[,1])
plot(results[,2])
plot(results[,3])
plot(results[,4])
res=results[201:10000,]
mu1s=res[,1]
sig1s=res[,2]
mu2s=res[,3]
sig2s=res[,4]
par(mfrow=c(4,1))
plot(mu1s)
plot(sig1s)
plot(mu2s)
plot(sig2s)
par(mfrow=c(2,1))
```

```
plot(mu1s−mu2s)

hist(mu1s−mu2s)

mean(mu1s−mu2s<0)
```