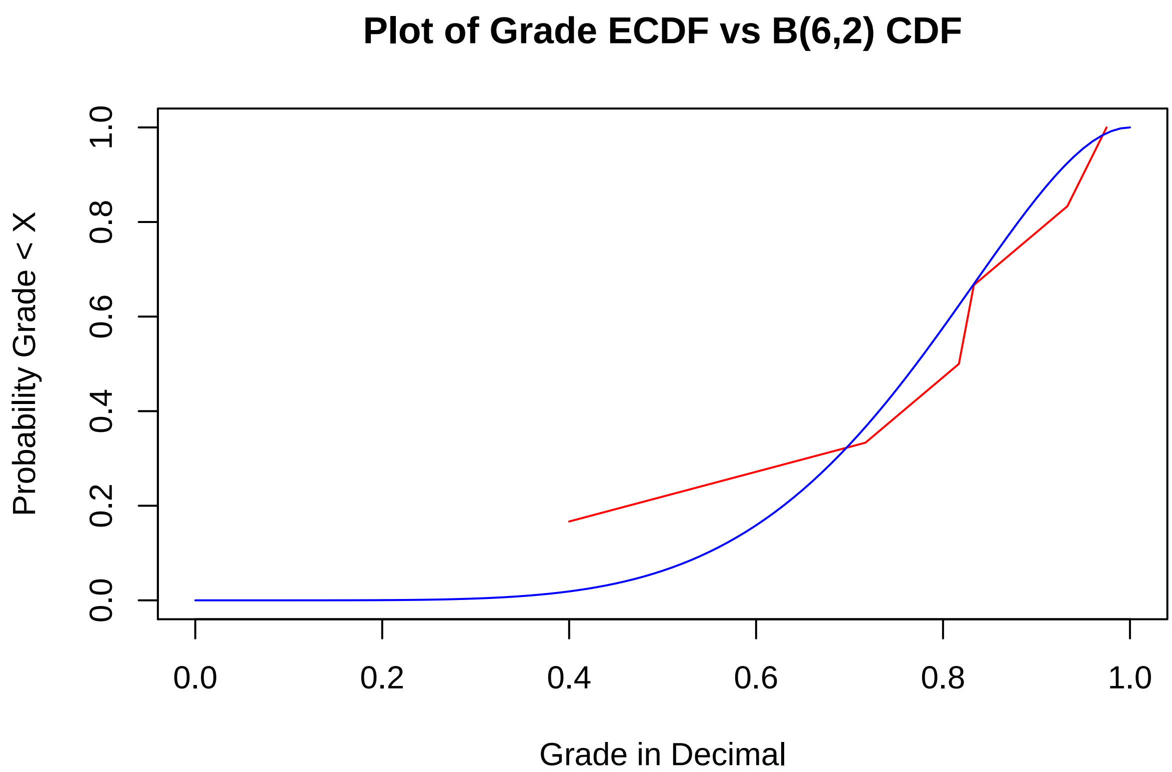
Pstat 105 Lab C

Kendall Brown

Fall 2017

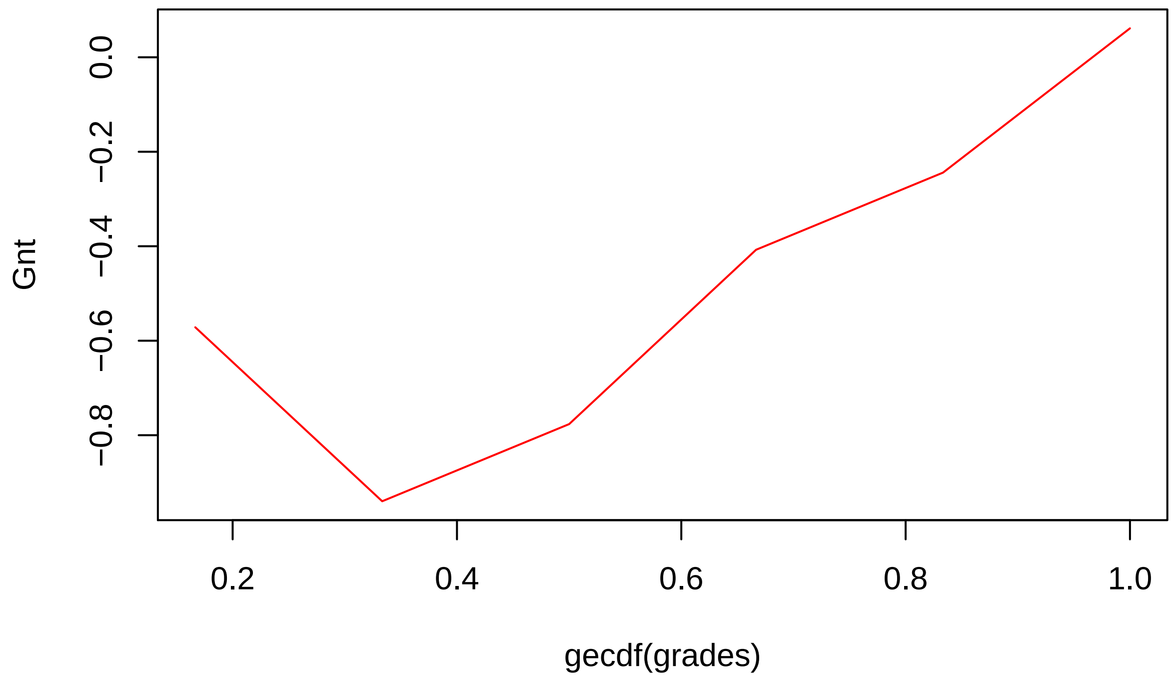
Q1a. Plot of Beta(6,2) CDF vs Grades Empirical CDF

```
grades=c(40,71.7,81.7,83.3,93.3,97.5)/100
quant=seq(1/6,1,1/6)
plot(grades, quant, type="l",xlim=c(0,1),ylim = c(0,1),xlab="Grade in Decimal",ylab = "Probability Grad
curve(pbeta(x,6,2),add=T,col="blue")
```



Q1b.Plot of Gn(t)

```
gecdf=ecdf(grades)
Gnt=(sqrt(6))*(gecdf(grades)-grades)
plot(gecdf(grades),Gnt,col="red",type="l")
```



Q1c.Ks test for beta distribution of grades.

```
ks.test(grades,"pbeta",6,2)
```

```
##
##  One-sample Kolmogorov-Smirnov test
##
## data:  grades
## D = 0.2906, p-value = 0.5955
## alternative hypothesis: two-sided
```

```
#Taking 0<U1<U2<U3<...<U6<1
#from the D statistic D=.2906, D+=max(1-U6,5/6-U5,2/3-U4,1/2-U3,1/3-U2,1/6-U1) or D-=max(U1,U2-1/6,U3-1
#since 0<Ui<1, D+ implies that if U6<.7094 then Ui<.7094, similarly if U1>.2906 then Ui>.2906
#From this we can calculate the exact P-value
epv=(.7094^5)+(.7094)^5
epv #Exact P-value
```

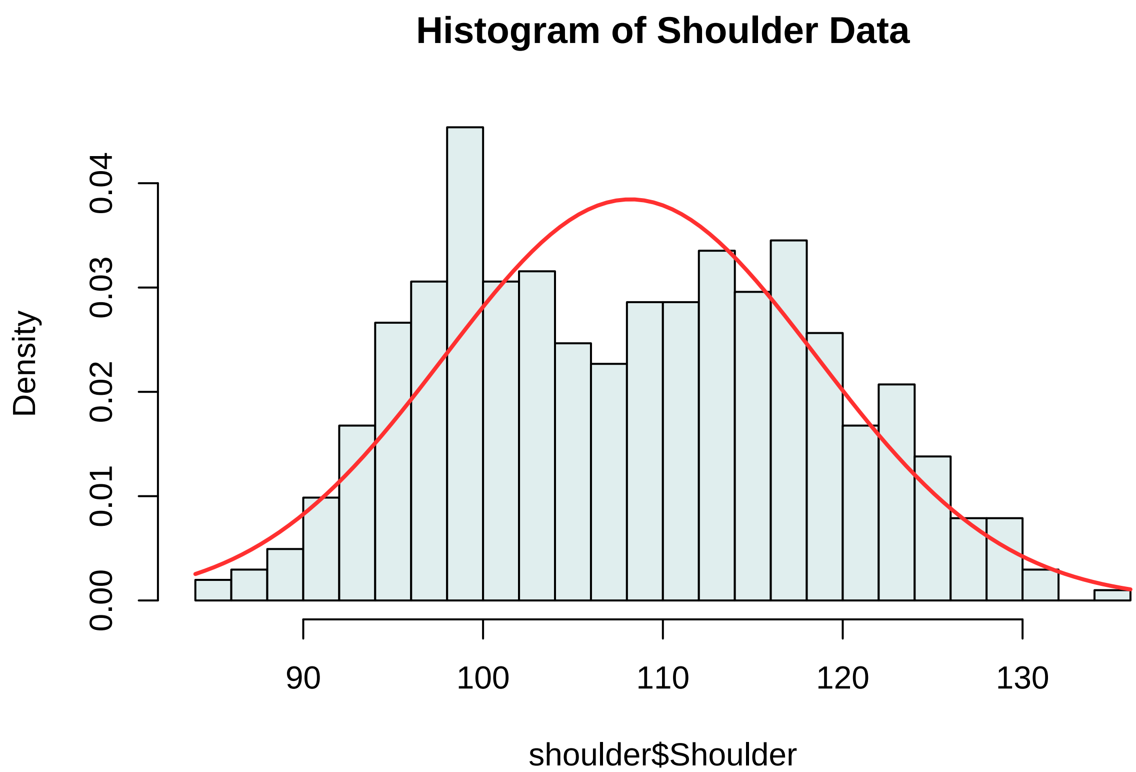
```
## [1] 0.3593237
```

Q1d.From this P-value we fail to reject the null hypothesis, there is not enough evidence to suggest the data does not follow B(6,2).

Q1e. It would be difficult to preform a chi-squared test here because we have very few samples to distribute enough amongst bins to calculate the X2 test statistic.

Q2a.Histogram and Normal Curve of Shoulder Data

```r
shoulder=read.table("C:/Users/kebro/Desktop/PSTAT 105/shoulder.txt",header = TRUE)
hist(shoulder$Shoulder,main="Histogram of Shoulder Data",breaks = 25,probability = TRUE,col="azure2")
ssd=sd(shoulder$Shoulder)
sm=mean(shoulder$Shoulder)
curve(dnorm(x,mean=sm,sd=ssd),add = TRUE,col="firebrick1",lw=2)
```



Q2b.Tests of Shoulder Data

```r
library("nortest")
lillie.test(shoulder$Shoulder)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  shoulder$Shoulder
## D = 0.077918, p-value = 8.838e-08
```

```r
cvm.test(shoulder$Shoulder)
```

```
##
##  Cramer-von Mises normality test
##
## data:  shoulder$Shoulder
## W = 0.62429, p-value = 2.321e-07
```

```r
ad.test(shoulder$Shoulder)
```

```
##
##  Anderson-Darling normality test
```

```
##
## data:  shoulder$Shoulder
## A = 3.6469, p-value = 4.112e-09
```

From the observer P-Values we reject the notion of normality in this dataset.

Q2c.Tests of Shoulder Data by Gender

```r
mshoulder=subset(shoulder,Gender=="Male")
fshoulder=subset(shoulder,Gender=="Female")
#Male Shoulders
lillie.test(mshoulder$Shoulder)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  mshoulder$Shoulder
## D = 0.041117, p-value = 0.3921
```

```r
cvm.test(mshoulder$Shoulder)
```

```
##
##  Cramer-von Mises normality test
##
## data:  mshoulder$Shoulder
## W = 0.035563, p-value = 0.7605
```

```r
ad.test(mshoulder$Shoulder)
```

```
##
##  Anderson-Darling normality test
##
## data:  mshoulder$Shoulder
## A = 0.2263, p-value = 0.8158
```

```r
#Female Shoulders
lillie.test(fshoulder$Shoulder)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  fshoulder$Shoulder
## D = 0.076669, p-value = 0.0008286
```

```r
cvm.test(fshoulder$Shoulder)
```

```
##
##  Cramer-von Mises normality test
##
## data:  fshoulder$Shoulder
## W = 0.28647, p-value = 0.0004713
```

```r
ad.test(fshoulder$Shoulder)
```

```
##
##  Anderson-Darling normality test
##
## data:  fshoulder$Shoulder
## A = 1.6058, p-value = 0.0003889
```

From the observed P-values we conclude that thereis not enough evidence to reject normality amongst male shoulderwidth, but there is enough to reject normality amongst female shoulder width.

Q2d. Normaility Tests for Adjusted Shoulder Width

```
amshoulder=mshoulder$Shoulder-mean(mshoulder$Shoulder)
afshoulder=fshoulder$Shoulder-mean(fshoulder$Shoulder)
ashoulder=c(amshoulder,afshoulder)
lillie.test(ashoulder)
```

```
##
##  Lilliefors (Kolmogorov-Smirnov) normality test
##
## data:  ashoulder
## D = 0.041835, p-value = 0.03391
```

```
cvm.test(ashoulder)
```

```
##
##  Cramer-von Mises normality test
##
## data:  ashoulder
## W = 0.16851, p-value = 0.01357
```

```
ad.test(ashoulder)
```

```
##
##  Anderson-Darling normality test
##
## data:  ashoulder
## A = 0.94516, p-value = 0.01668
```

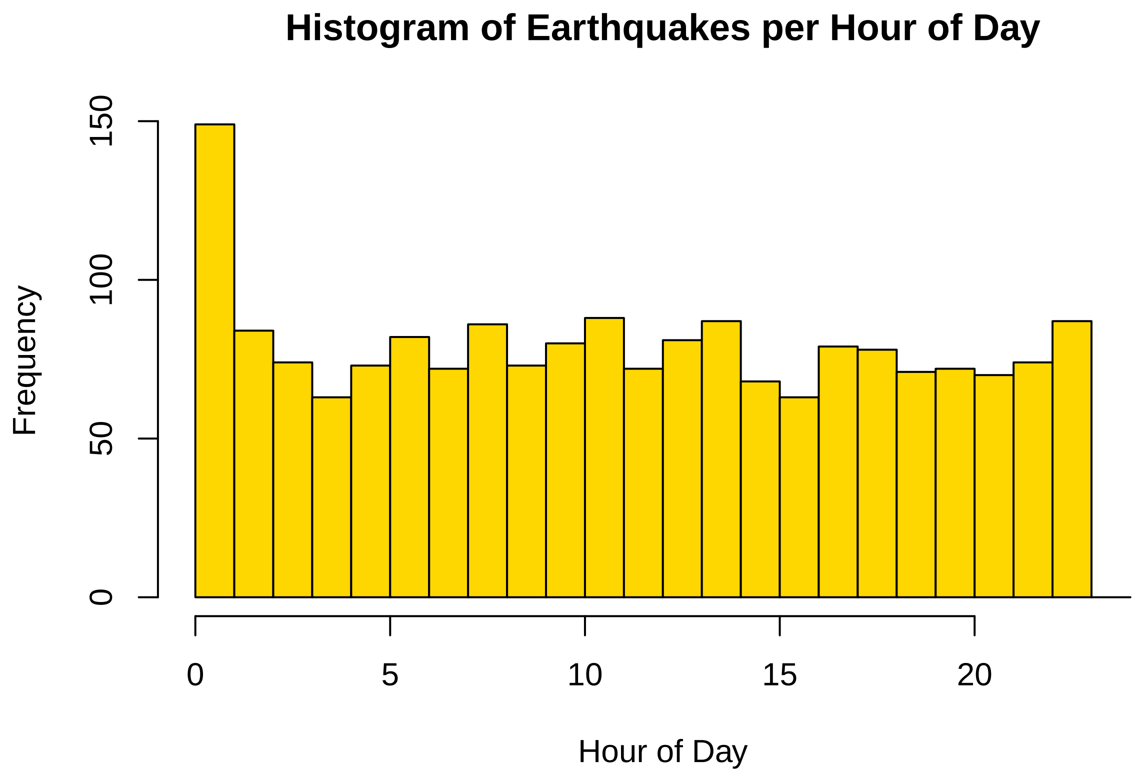
From the observed P-Values, we reject normality at the .05 level after adjusting for the population means. However, we fail to reject normaility at the .01 level.

Q2e. Based on previous results, I do not believe it would be wise to assume normality and preform a two sampled t-test.

```
EarthquakeData <- read.csv("C:\\Users\\kebro\\Desktop\\PSTAT 105\\EarthquakeData.htm", header = TRUE, s
eq.hour <- as.integer(substr(EarthquakeData$Time,1,2))
eq.min <- as.integer(substr(EarthquakeData$Time,3,4))
eq.sec <- as.numeric(substr(EarthquakeData$Time,5,8))
```

Q3a. Chi-Squared Test for Uniformity Across Hours of Day.

```
pph=rep((1/24),24)
disteqh=(hist(eq.hour,breaks = seq(0,24,1),xlab="Hour of Day", main="Histogram of Earthquakes per Hour
```




```
chisq.test(disteqh$counts,p=pph)
```

```
##
##  Chi-squared test for given probabilities
##
## data:  disteqh$counts
## X-squared = 161.43, df = 23, p-value < 2.2e-16
```

Q3b. KS-Test of Hourly Data.

```
eq.time=(3600*eq.hour)+(60*eq.min)+eq.sec #Converting time of day to seconds past midnight.
ecdfeq=sort(eq.time)/86400 #With 86400 secs/day we can get the ecdf of the time data.
ecdfuni=seq(0,1,length=1826) #Uniform CDF of seconds past in a day with 1826 intervals
Dp=(ecdfuni-ecdfeq)
Dm=(ecdfeq-ecdfuni)
max(Dp) #Dplus value
```

```
## [1] 0.01319614
```

```
max(Dm) #Dminus value
```

```
## [1] 0.01251513
```

```
Dmax=max(max(Dp),max(Dm)) #D test statistic
```

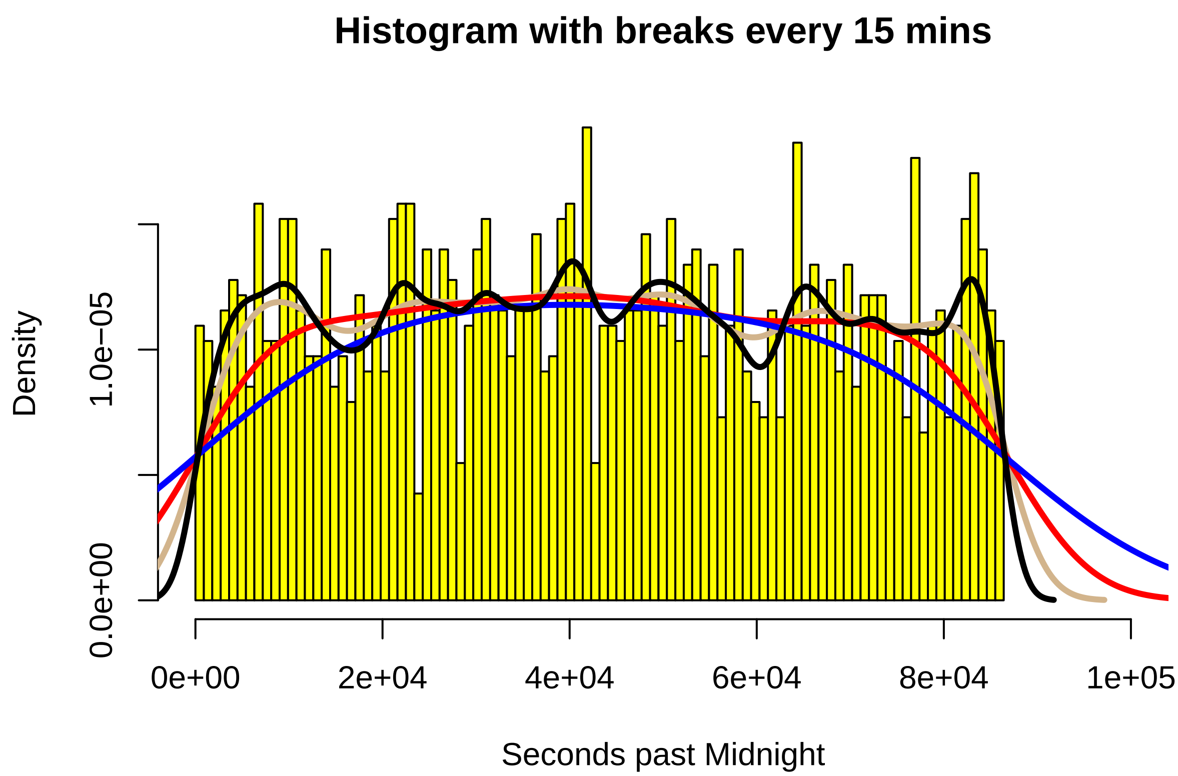
Q3c. P-value of KS.Test

```
n=c(1,2,3,4,5,6,7,8,9,10)
pv=2*sum(((-1)^(n-1))*exp(-2*1826*(n^2)*(Dmax^2)))#Approximating P-Value with 10 sums.
pv#Approximated p-Value given D statistic
```

```
## [1] 0.908189
```

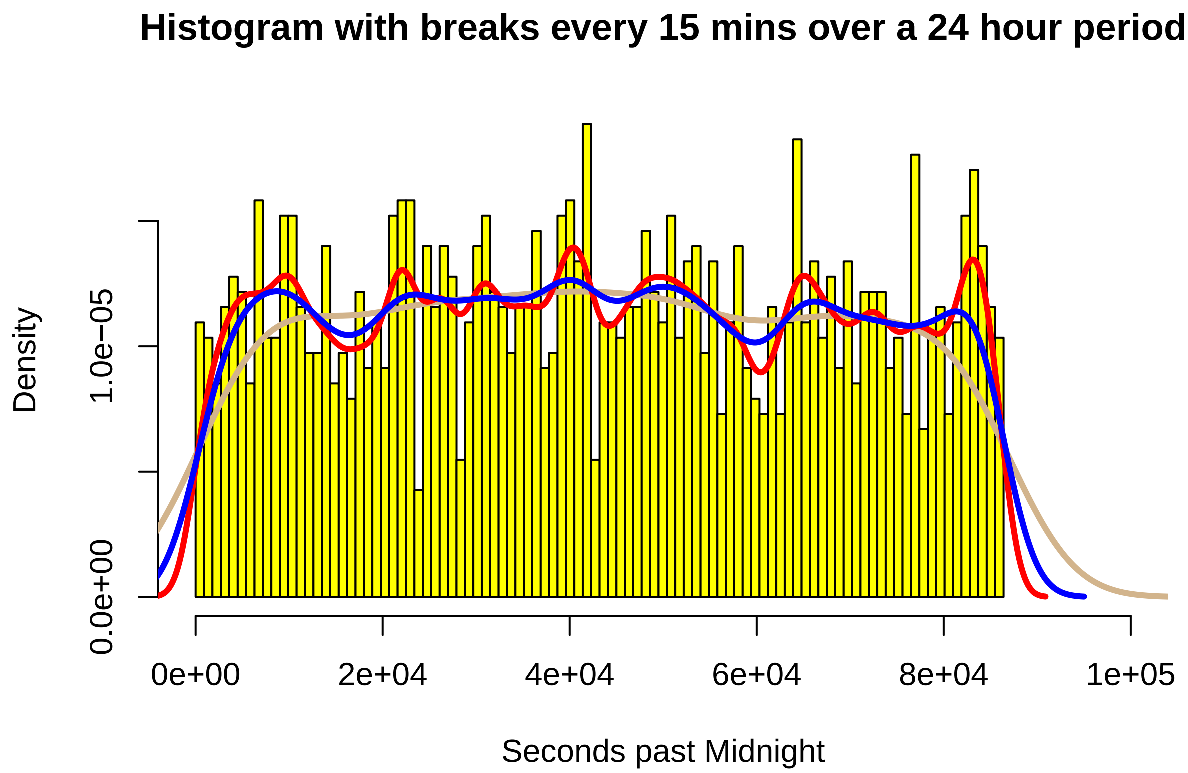
Q3d&e.Histogram with breaks every 15 mins with

```
hist(eq.time,breaks = seq(0,86400,900),main="Histogram with breaks every 15 mins",xlab="Seconds past Mic
lines(density(eq.time,bw=3600,kernel="gaussian"),col="tan",lw=3) #BW is an hour
lines(density(eq.time,bw=3600*2,kernel="gaussian"),col="red",lw=3) #BW is two hours
lines(density(eq.time,bw=3600*4,kernel="gaussian"),col="blue",lw=3) #BW is four hours
lines(density(eq.time,bw=3600/2,kernel="gaussian"),col="black",lw=3) #BW is 30 mins
```



I belive that the 2 and 4 hour bandwidths gave kernels that were too smooth. As such I believe the 1 hour bandwidth to be bettwe than both the 2 and 4 hour bandwidths, but not as good as the 30 min bandwidth which. It is a bit rough but it matches the underlying histogram a bit better. Q3f.

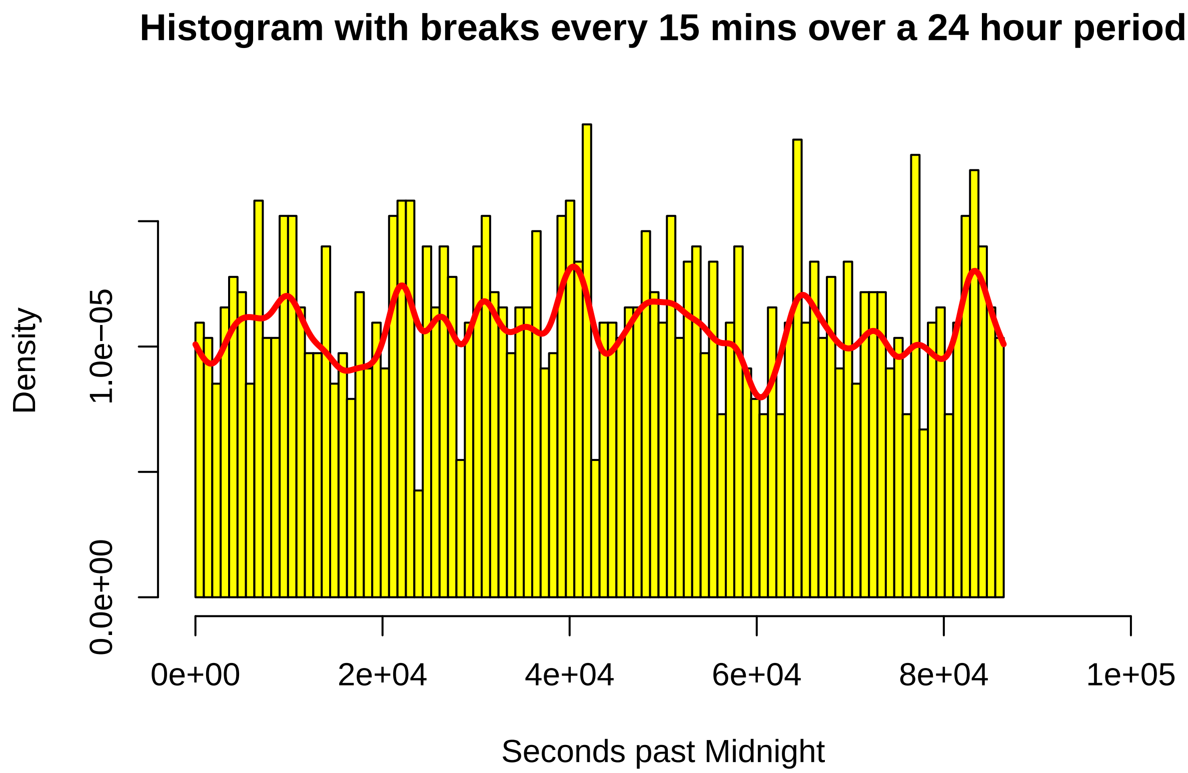
```
hist(eq.time,breaks = seq(0,86400,900),main="Histogram with breaks every 15 mins over a 24 hour period"
lines(density(eq.time,bw="nrd",kernel="gaussian"),col="tan",lw=3) #BW is the nrd
lines(density(eq.time,bw="ucv",kernel="gaussian"),col="red",lw=3) #BW is the ucv
lines(density(eq.time,bw="SJ",kernel="gaussian"),col="blue",lw=3) #BW is the SJ
```



I would say that the UCV bandwidth method is the most accurate bandwidth of NRD, UCV, and SJ as it appears to match the histogram's distribution well.

Q3g.Fixing cyclical data.

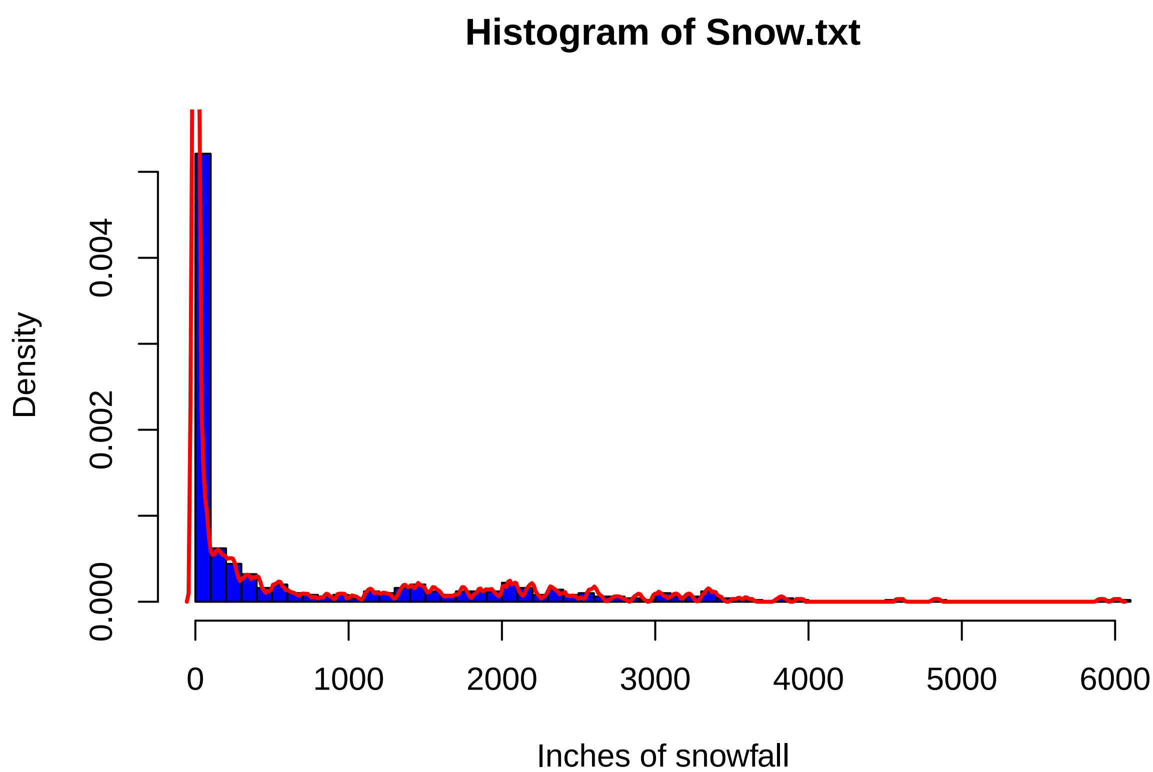
```
seq.time=sort(eq.time)
histeq=hist(eq.time,breaks = seq(0,86400,900),main="Histogram with breaks every 15 mins over a 24 hour p
deneq=density(c(eq.time,seq.time[1739:1826]-86400,seq.time[0:68]+86400),bw="ucv",kernel="gaussian",from=
#Took the first's and last's hour of datapoints and affixed them to the ends of the set used them to app
#the density at hour 0 and hour 24.
lines(deneq,col="red",lw=3)
```




```
snow <- scan("snow.txt")
```

Q4a.Histogram of Snow.txt

```
hists=hist(snow,breaks=50,main="Histogram of Snow.txt",col="blue",probability = T,xlab="Inches of snowfa
dsnow=density(snow,bw="SJ",kernel="rectangular")
lines(dsnow,lw=2,col="red")
```



Q4b.Estimation at 2000

```
c2000=sum(hists$counts[20:22])
c2000
```

```
## [1] 25
```

```
density(snow,bw="SJ",kernel="rectangular",from=1900,to=2100)
```

```
##
## Call:
##  density.default(x = snow, bw = "SJ", kernel = "rectangular",     from = 1900, to = 2100)
##
## Data: snow (499 obs.);   Bandwidth 'bw' = 18.62
##
##        x               y
##  Min.   :1900   Min.   :3.108e-05
##  1st Qu.:1950   1st Qu.:1.243e-04
##  Median :2000   Median :1.865e-04
##  Mean   :2000   Mean   :1.686e-04
##  3rd Qu.:2050   3rd Qu.:2.175e-04
##  Max.   :2100   Max.   :3.108e-04
```

```
c2100=sum(hists$counts[1:22])
c1900=sum(hists$counts[1:20])
Fht=(c2100-c1900)/499/(2*18.62)
sv=.0001686/(2*499*(18.62^2))
d2000=.0001686
```

```
cint=c(d2000-1.96*sqrt(sv/499),d2000+1.96*sqrt(sv/499))
cint
```

```
## [1] 0.0001666632 0.0001705368
```

With a bandwidth of 18.62 and a rectangular Kernel, we have a sample density of .0001686 based off of 25 observations. 95% confidence interval {0.0001666632, 0.0001705368}

Q4c. Prob of snowfall in a year

```
sort(snow)[183]
```

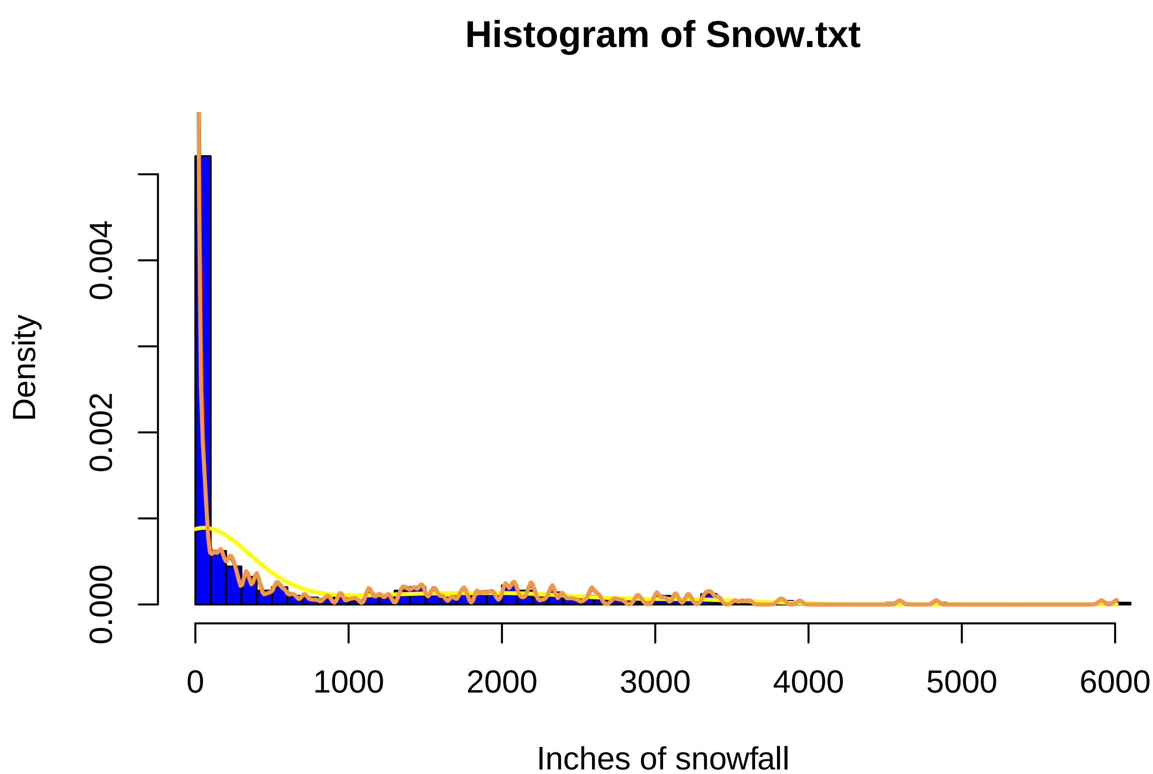
```
## [1] 0
```

```
psnow=183/499
psnow #Probability a station sees a warm winter
```

```
## [1] 0.3667335
```

Q4d. Density where snow!=0.

```
hists=hist(snow,breaks=50,main="Histogram of Snow.txt",col="blue",probability = T,xlab="Inches of snowfa
lines(density(snow,bw="SJ",kernel="gaussian",from=1,to=6009),lw=2,col="red")
lines(density(snow,bw="nrd",kernel="gaussian",from=1,to=6009),lw=2,col="yellow")
lines(density(snow,bw=15,kernel="gaussian",from=1,to=6009),lw=2,col="tan2")
```



The 15 bandwidth looks nice. Density for snow>0 is approximately 1.4*10^-4

Q4e.As a guess I would say that the bias of the estimate is rather significant as the data is quite obviously skewed towards lower values.