Homework7_Hwang

Charles Hwang

4/28/2022

Charles Hwang

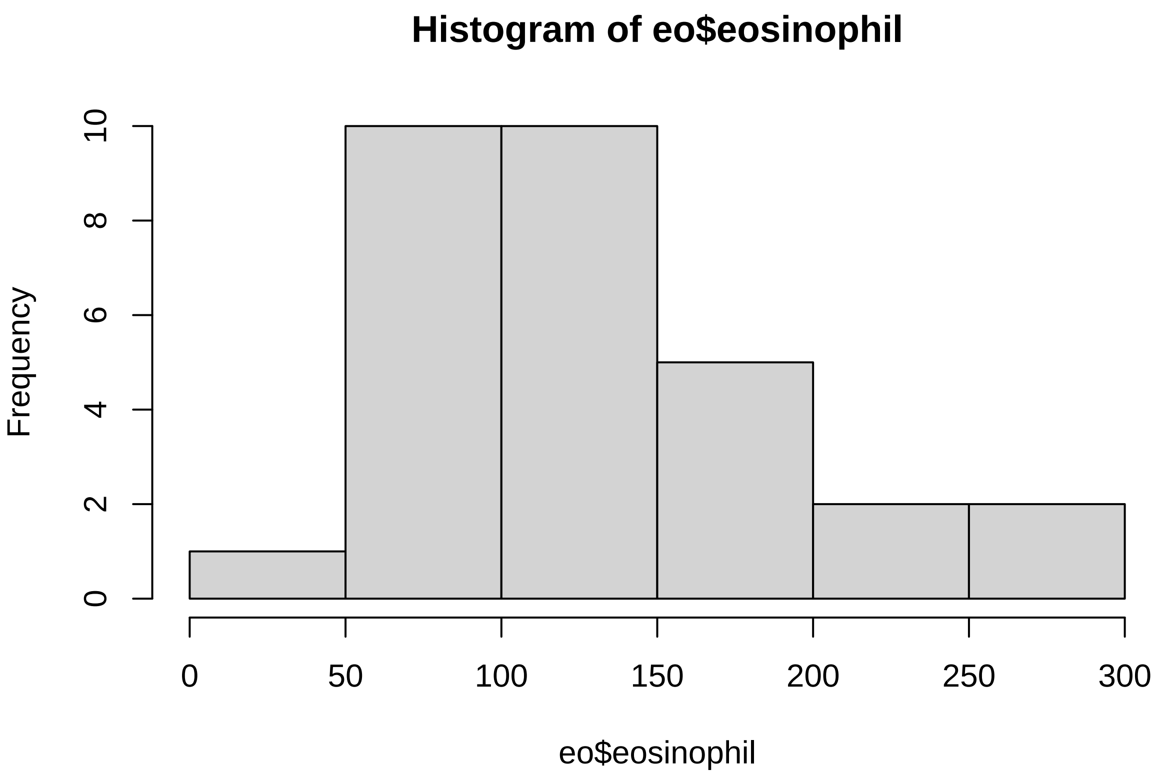
Dr. Perry

STAT 451-001

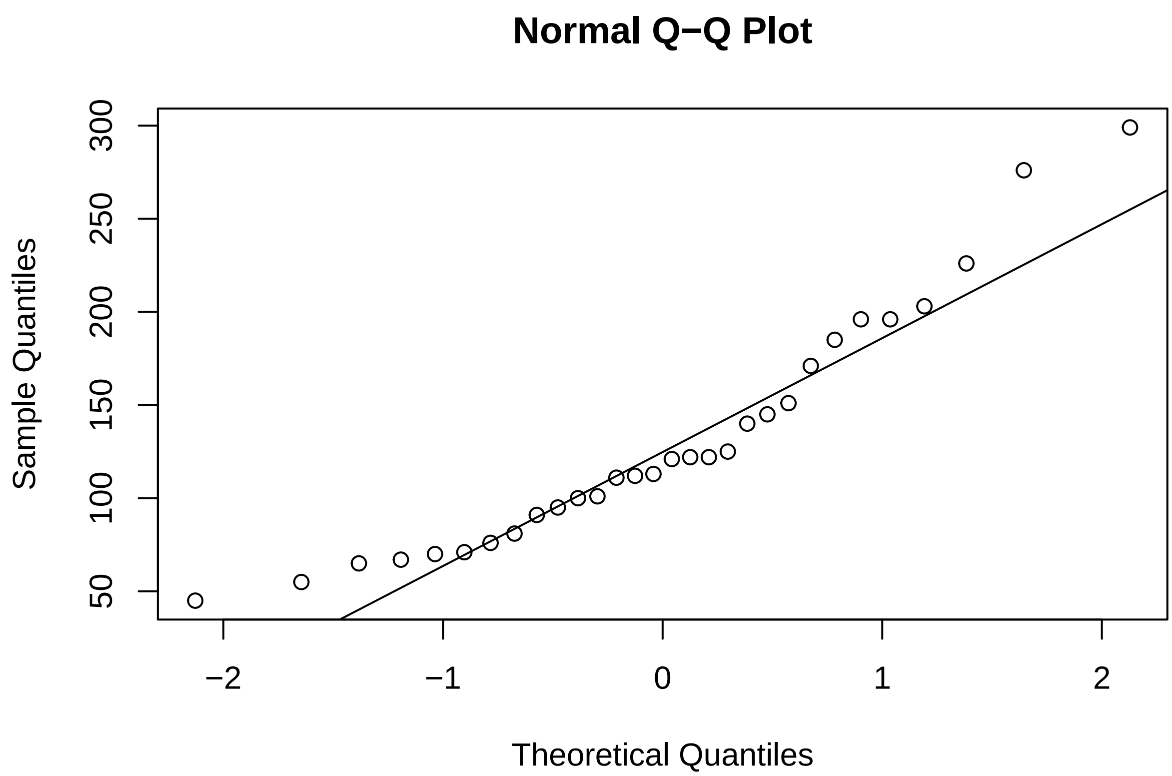
28 April 2022

Problem 1

```
rm(list=ls())
eo<-read.table("/Users/newuser/Desktop/Notes/Graduate/STAT 451 - Nonparametric Statistical Methods/eosi
hist(eo$eosinophil)                                                                    # Problem 1(a)
```




```
qqnorm(eo$eosinophil)
qqline(eo$eosinophil)
```




```
# The distribution appears to be right-skew. It is possible that a standard normal
# confidence interval may not be appropriate.
set.seed(2804)                                                              # Problem 1(b)
x<-rnorm(nrow(eo),mean(eo$eosinophil),sd(eo$eosinophil))
Ex<-tb<-Xb<-rb<-kb<-rep(NA,5000)
for (i in 1:5000){bs=x[sample(1:nrow(eo),nrow(eo),replace=TRUE)]
Ex[i]<-mean(bs)}
mean(Ex)
```

```
## [1] 142.6827
```

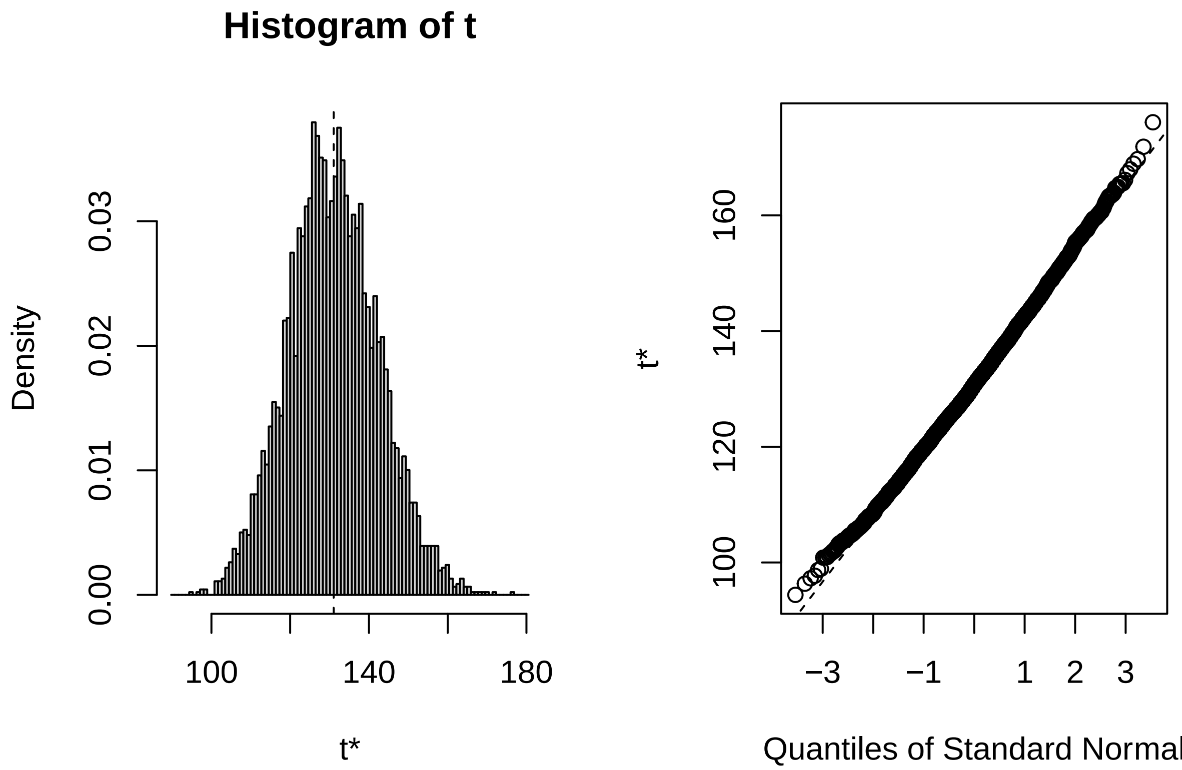
```
mean((Ex-mean(x))^2)                                                        # Problem 1(c)
```

```
## [1] 90.39709
```

```
for (i in 1:5000){bst<-eo$eosinophil[sample(1:nrow(eo),nrow(eo),replace=TRUE)] # Problem 1(d)
tb[i]<-(mean(bst)-mean(eo$eosinophil))/(sd(bst)/sqrt(nrow(eo)))}
mean(eo$eosinophil)-quantile(tb,c(0.95,0.05))*sd(eo$eosinophil)/sqrt(nrow(eo))
```

```
##      95%       5%
## 112.9448 154.2529
```

```
library(boot)                                                               # Problem 1(e)
boot<-boot(eo,function(x,n){return(mean(x[n,]))},5000)
plot(boot)
```




```r
boot.ci(boot,0.9,c("norm","perc","bca"))
```

```
## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 5000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boot, conf = 0.9, type = c("norm", "perc",
##     "bca"))
##
## Intervals :
## Level      Normal              Percentile            BCa
## 90%   (112.3, 149.9 )   (112.5, 150.3 )   (113.7, 151.9 )
## Calculations and Intervals on Original Scale
```

```r
# I believe the BCa confidence interval is the most appropriate for this data. It is
# second-order accurate as opposed to the percentage interval which is only first-order
# accurate (https://blogs.sas.com/content/iml/2017/07/12/bootstrap-bca-interval.html and
# page 425 (33) of Hollander, Wolfe, and Chicken textbook).
for (i in 1:5000){bsX=eo$eosinophil[sample(1:nrow(eo),nrow(eo),replace=TRUE)]  # Problem 1(f)
Xb[i]<-(nrow(eo)-1)*var(bsX)/var(eo$eosinophil)}
(nrow(eo)-1)*var(eo$eosinophil)/quantile(Xb,c(0.95,0.05))
```

```
##      95%       5%
## 2813.476 7573.896
```

Problem 2

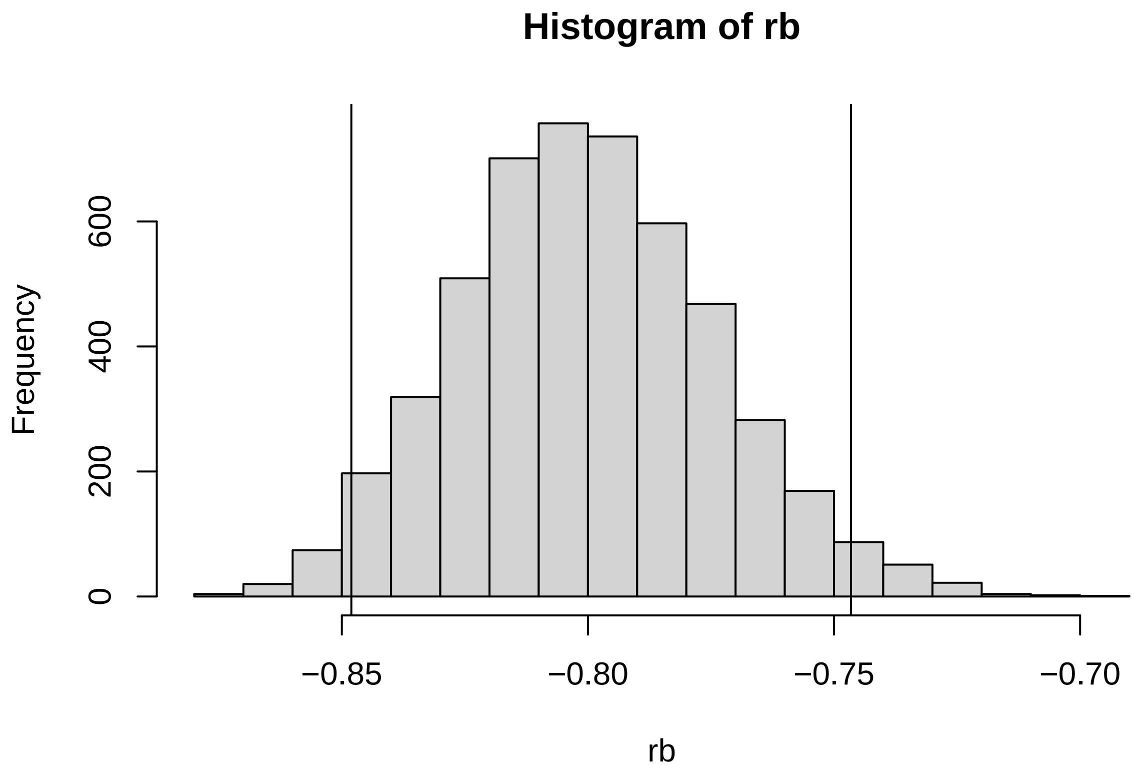
```r
fle<-read.delim("/Users/newuser/Desktop/Notes/Graduate/STAT 451 - Nonparametric Statistical Methods/fer
r<-cor(fle$Fertility,fle$Life_expectancy)                                  # Problem 2(a)
r
```

```
## [1] -0.7993762
```

```
cor.test(fle$Fertility,fle$Life_expectancy)$conf.int                    # Problem 2(b)
```

```
## [1] -0.8444612 -0.7430434
## attr(,"conf.level")
## [1] 0.95
```

```
for (i in 1:5000){bsr<-fle[sample(1:nrow(fle),nrow(fle),replace=TRUE),-1] # Problem 2(c)
rb[i]<-cor(bsr)[1,2]}
hist(rb)
abline(v=quantile(rb,c(0.025,0.975)))
```



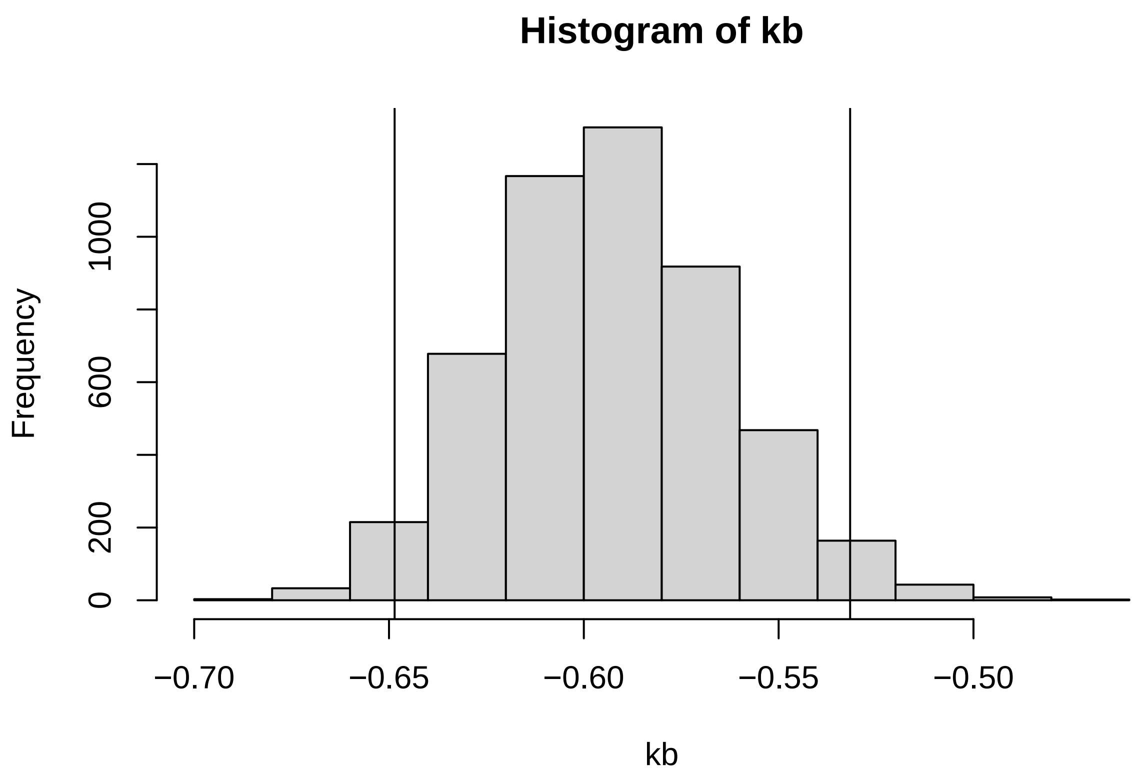

```
quantile(rb,c(0.025,0.975))
```

```
##       2.5%      97.5%
## -0.8480681 -0.7465520
```

```
# We can see the two confidence intervals are very similar. The interval from the
# percentile method appears to be leaning towards a stronger correlation coefficient.
c<-2*qnorm(0.975)/sqrt(nrow(fle)-3)                                     # Problem 2(d)
t<-cor(fle$Fertility,fle$Life_expectancy,method="kendall")
((1+t)*exp(c(-c,0,c))-(1-t))/((1+t)*exp(c(-c,0,c))+(1-t))
```

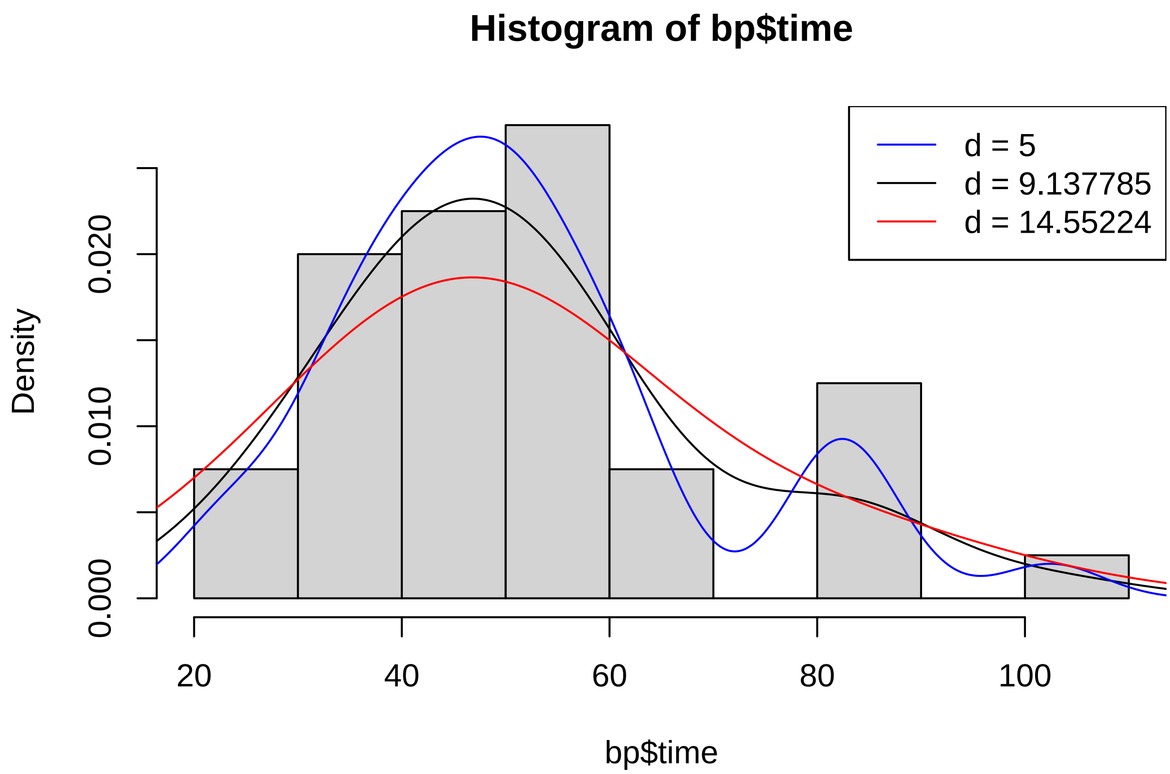
```
## [1] -0.6756426 -0.5924361 -0.4943264
```

```
for (i in 1:5000){bsr<-fle[sample(1:nrow(fle),nrow(fle),replace=TRUE),-1] # Problem 2(e)
kb[i]<-cor(bsr,method="kendall")[1,2]}
hist(kb)
abline(v=quantile(kb,c(0.025,0.975)))
```



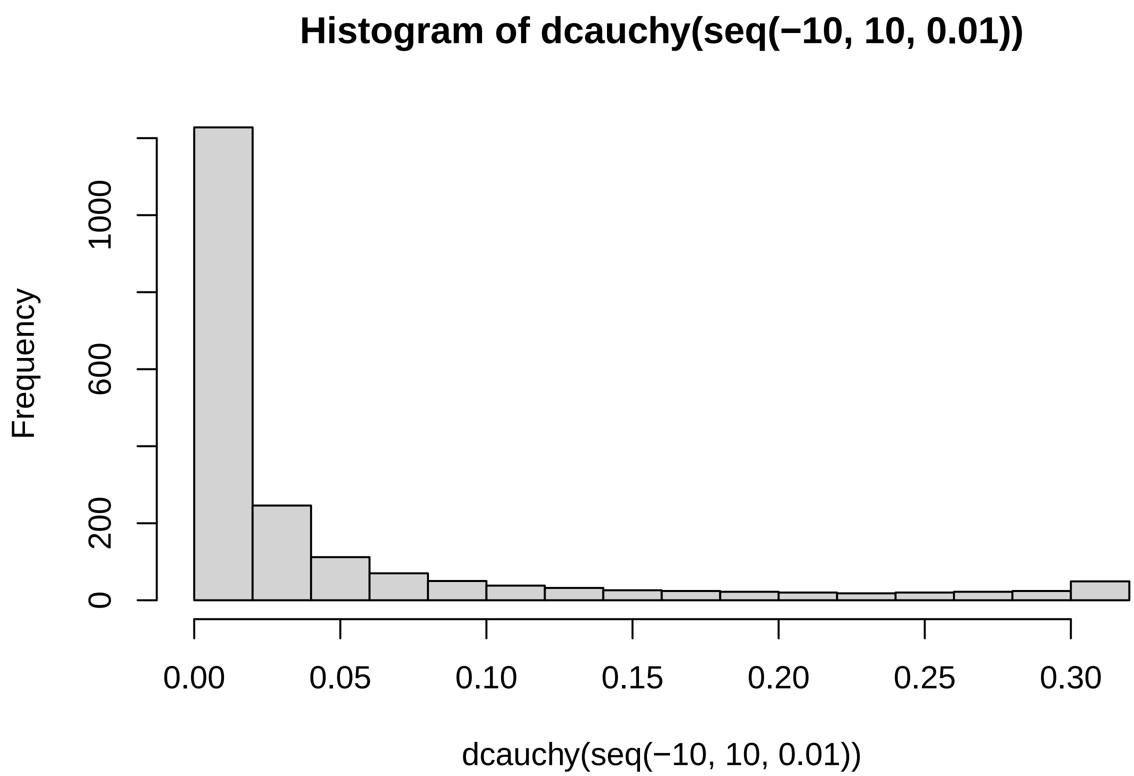
Problem 3

```
bp<-read.table("/Users/newuser/Desktop/Notes/Graduate/STAT 451 - Nonparametric Statistical Methods/brake
hist(bp$time,freq=FALSE)                                                    # Problem 3(a)
k<-density(bp$time,1.06*sd(bp$time)/nrow(bp)^0.2)
points(k$x,k$y,type="l")
points(density(bp$time,5)$x,density(bp$time,5)$y,type="l",col="blue") # Problem 3(b)
IQR<-density(bp$time,IQR(bp$time)/1.34) # Hardle (1991) suggests using IQR/1.34
points(IQR$x,IQR$y,type="l",col="red")
legend("topright",c("d = 5", "d = 9.137785","d = 14.55224"),lwd="1",col=c("blue","black","red"))
```

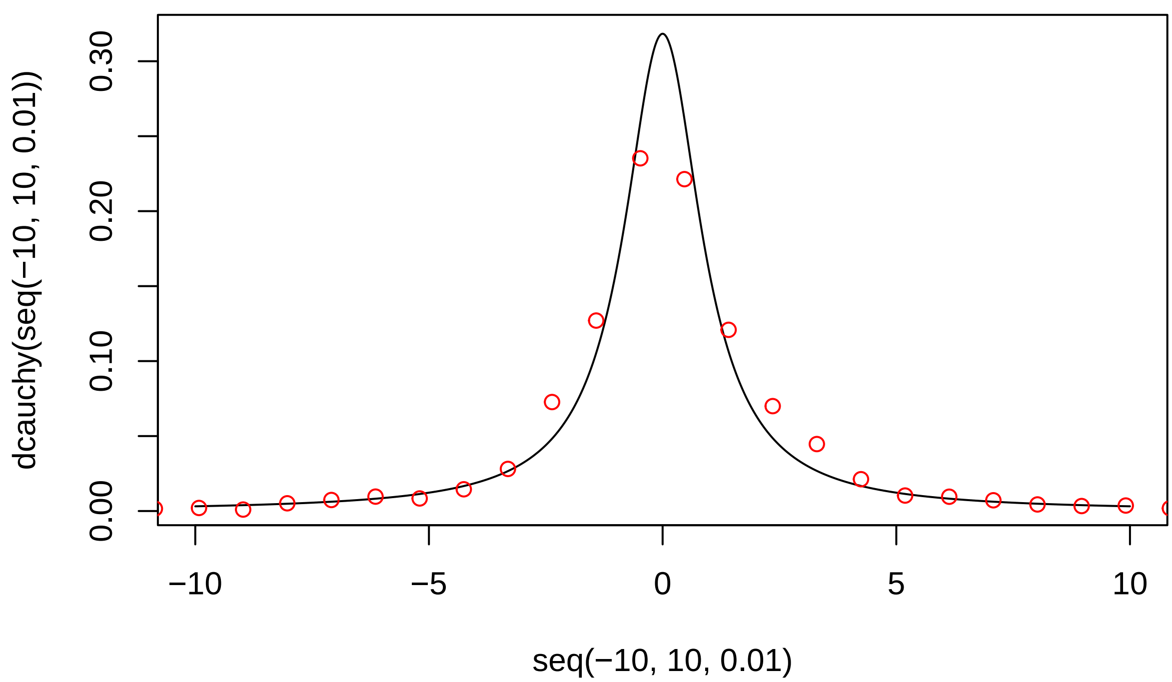



```
# Delta (d) is the bandwidth of the kernel density estimate.
# I believe the original value of d = 1.06*S/n^0.2 = 9.137785 is the best
# kernel density function. We can see the two functions less than d = 5
# and greater than d = 1.06*S/n^0.2 = 14.55224 are a little too rigid
# and a little too smooth respectively (Goldilocks principle).
library(VGAM)                                                     # Problem 3(c)

hist(dcauchy(seq(-10,10,0.01)))
```




```
set.seed(2804)
plot(seq(-10,10,0.01),dcauchy(seq(-10,10,0.01)),type="l")
points(density(rcauchy(500)),col="red")
```




```
#points(k$x,k$y,type="l")
#points(density(bp$time,5)$x,density(bp$time,5)$y,type="l",col="blue")
#points(IQR$x,IQR$y,type="l",col="red")
#legend("topright",c("d = 5",  "d = 9.137785","d = 14.55224"),lwd="1",col=c("blue","black","red"))
```