HW 3 Markdown

Steve Harms

September 16, 2017

Exercise 1

a)

```
#Assuming the package is already installed
library(readxl)

#Read the data into a data frame using the read_excel function
wind <- data.frame(read_excel("wind.xls"))
```

b)

```
#I make my own function using the base R statistics functions to save some time
sumstats.function <- function(x){
  c(mean = mean(x), median = median(x), sd = sd(x), quartile = quantile(x), IQR =
IQR(x))
}

#Apply the function above to the wind data
apply(wind, 2, FUN = sumstats.function)
```

```
##                 Spring    Summer    Autumn    Winter
## mean          176.6667  80.83333 200.00000 238.33333
## median        185.0000  35.00000 215.00000 255.00000
## sd            123.9746  72.92067  94.00193  86.63752
## quartile.0%     0.0000  10.00000  30.00000  50.00000
## quartile.25%   55.0000  20.00000 155.00000 205.00000
## quartile.50%  185.0000  35.00000 215.00000 255.00000
## quartile.75%  275.0000 150.00000 260.00000 297.50000
## quartile.100% 350.0000 190.00000 350.00000 340.00000
## IQR           220.0000 130.00000 105.00000  92.50000
```

c)

To me, all of the summary statistics can make sense in this context. The mean and median are measures of average wind direction, so if I wanted to know which direction to expect the wind to be blowing in a certain season, I would use those measurements to make a guess. The standard deviation and quartiles are measures of how erratic the wind direction would be, so a large s.d. or IQR would suggest that the wind changes directions often (or to more extreme directions). So yes, as long as we have a reference point (i.e., the center of the circle for which these angular measurements are taken), they have some use. However, note that there is no magnitude measurements so we have no idea how strong the wind is actually blowing in each direction (so it's not as useful as it might seem)

d)

```
attach(wind)

#Use the reshape package to melt the data frame into something easier
library(reshape)

#melt the rows into an additional column then name our new columns
windmlt <- melt(wind)
```

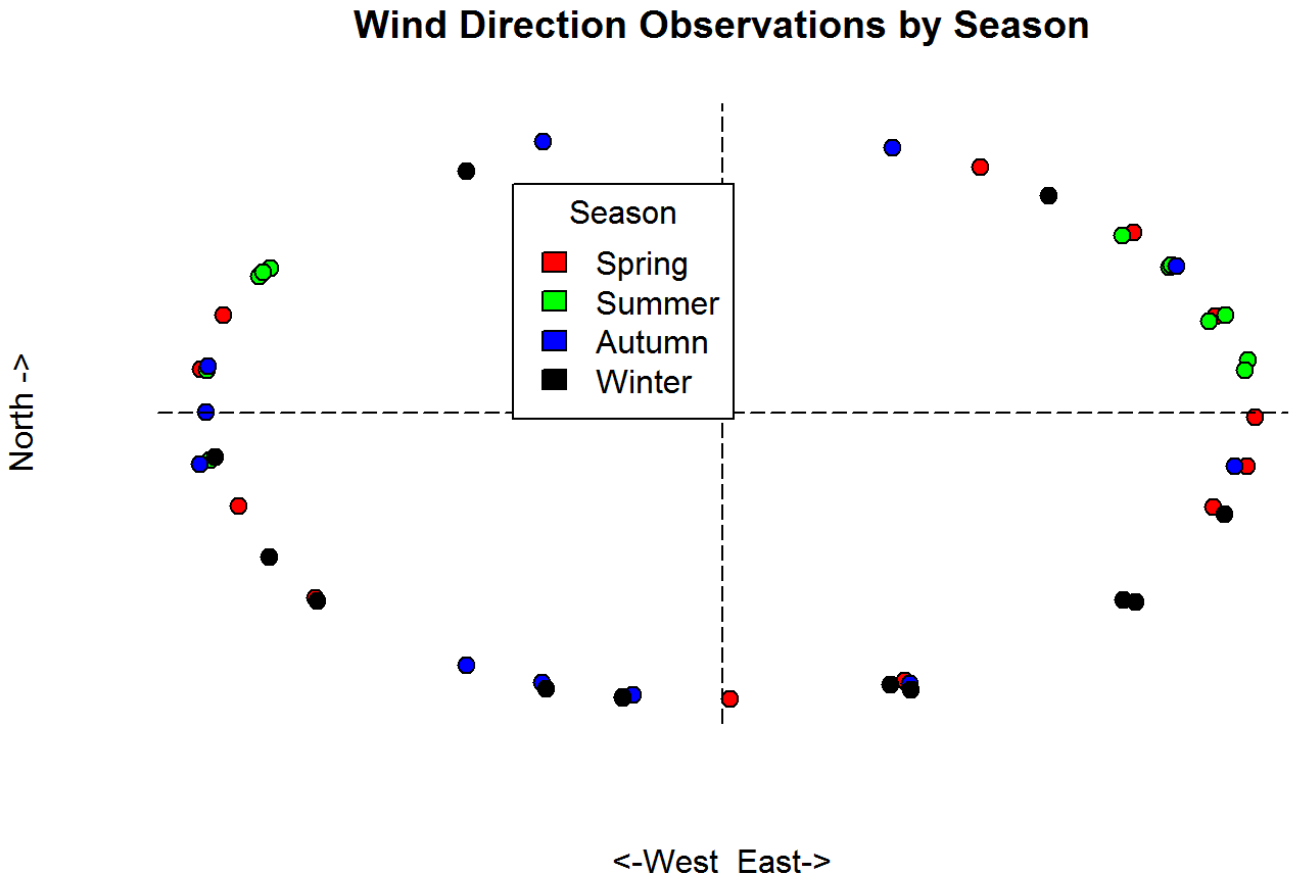
```
## Using  as id variables
```

```
names(windmlt) <- c("Season", "Direction")

#A new function to convert degrees to radians, and then get unit circle coordinates
convert.f <- function(z){
  rad = z*pi/180
  x = cos(rad)
  y = sin(rad)
  return(data.frame(x,y))
}

#Create a new data frame to plot our data using our seasons and new coordinate data
windplot <- data.frame(windmlt$Season, convert.f(windmlt$Direction))

#Plot the direction coordinates around the circle
#I used jitter() to make sure repeated points are still visible on the plot
plot(x = jitter(windplot$x, factor = 8), y = jitter(windplot$y, factor = 8),
     xlim = c(-1,1), xlab = "<-West  East->",
     ylim = c(-1,1), ylab = "North ->", cex = 1.2,
     main = "Wind Direction Observations by Season", axes = F,
     pch=21, bg = c("red", "green", "blue", "black")[(windmlt$Season)], oma = c(5,
20,5,5)
     )
abline(h = 0, v = 0, lty = 5)
legend(x = -.4, y= .8, legend = c("Spring", "Summer", "Autumn", "Winter"),
       fill=c("red", "green", "blue", "black"), title = "Season")
```



There is not much to notice in the measurements. It is clear that most of the winter observations (in black) were blowing to the south (below the x axis), while in summer most were blowing slightly to the north and east. Autumn had a few clustered to the South, either straight South or West/Southwest. The most erratic was Spring without much obvious trend at all, although most are to the East, somewhat similar to the Summer observations. In total, it is hard to see visually much of a trend across seasons, especially without any magnitude data.

Exercise 2

a)

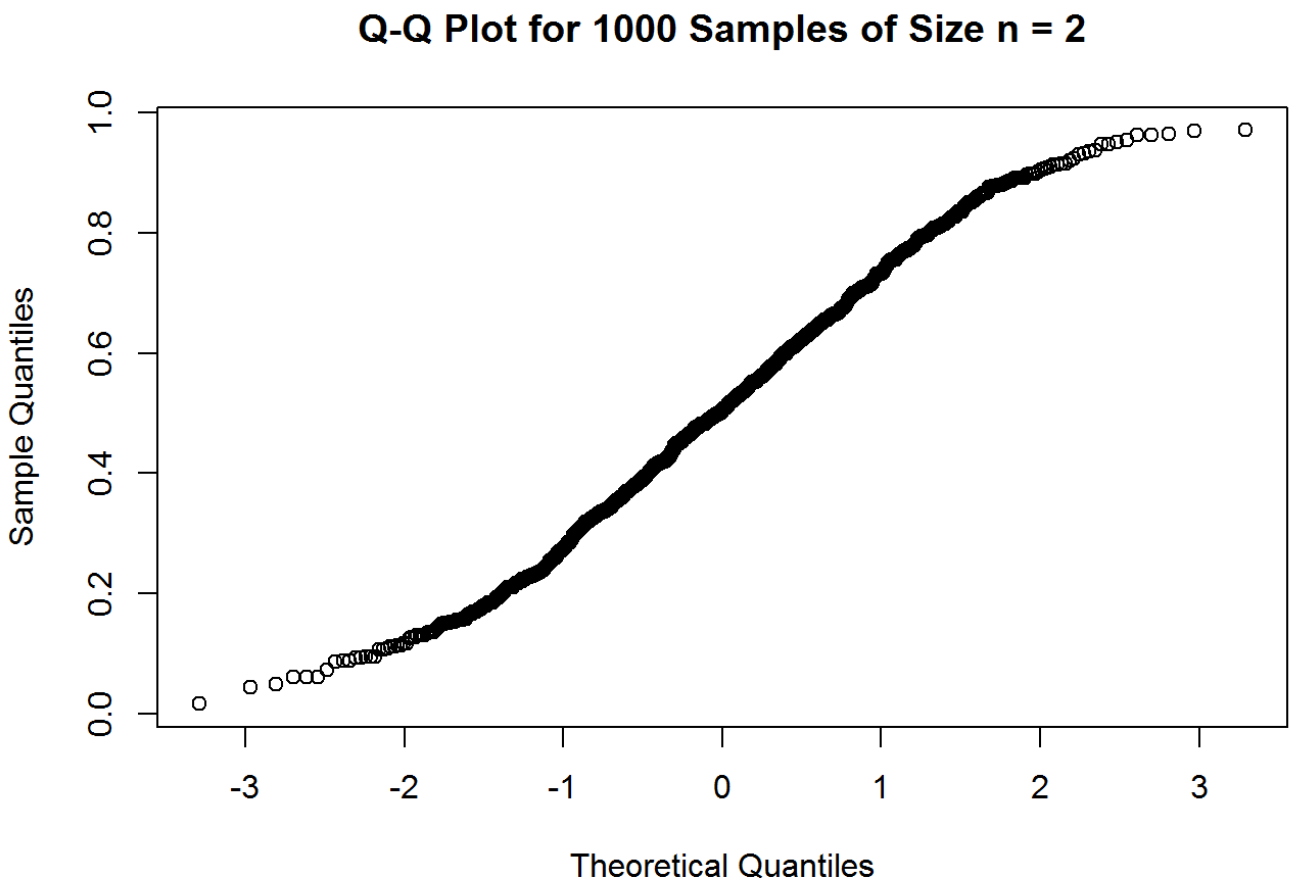
```
#Generate 2000 random samples from U(0,1) and fill in a 1000 x 2 matrix
m <- matrix(runif(2000), nrow = 1000, ncol = 2)
```

b)

```
#generating sample mean for each sample
xbar <- .5*(m[,1] + m[,2])
```

c)

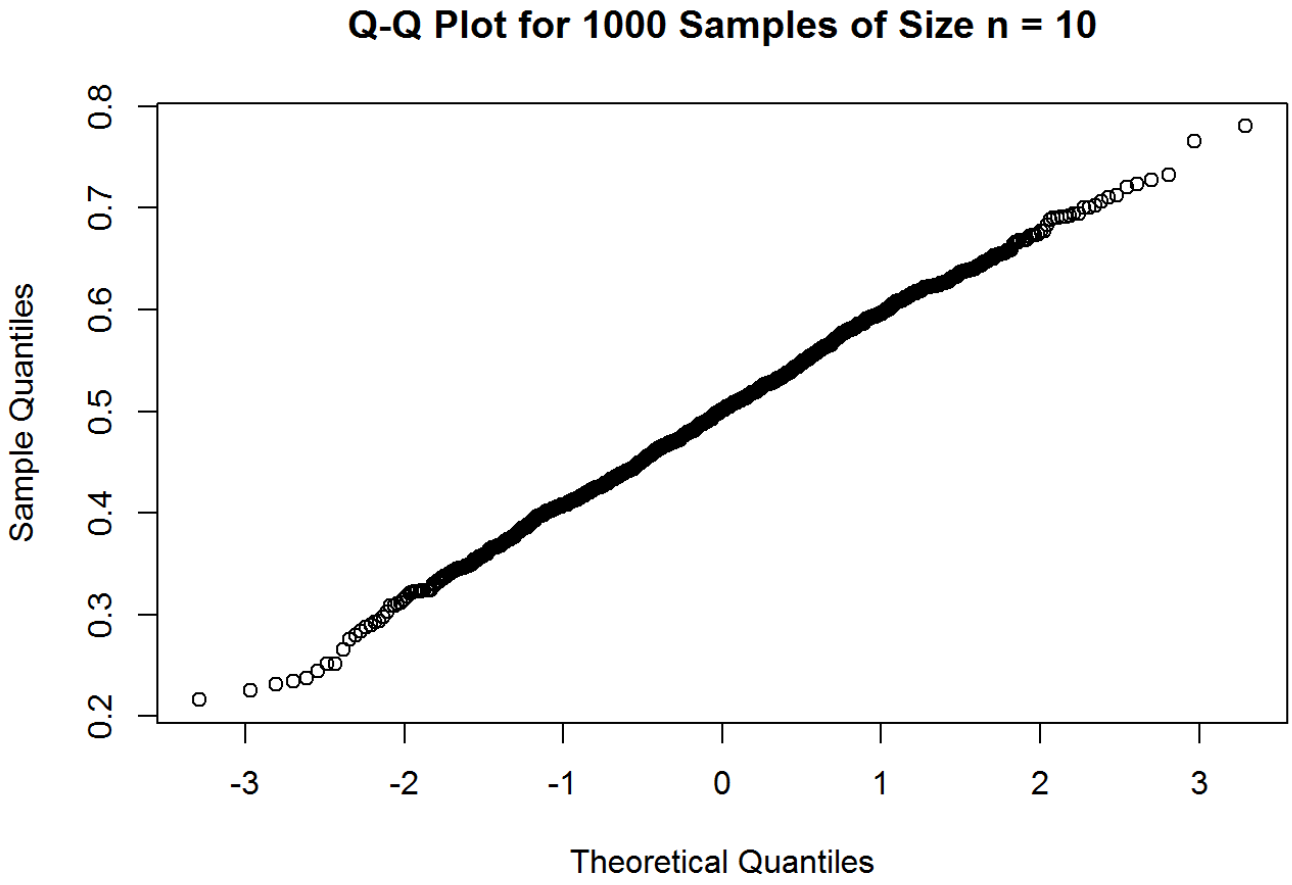
```
#A qqplot for our sample means
qqnorm(xbar, main = "Q-Q Plot for 1000 Samples of Size n = 2")
```



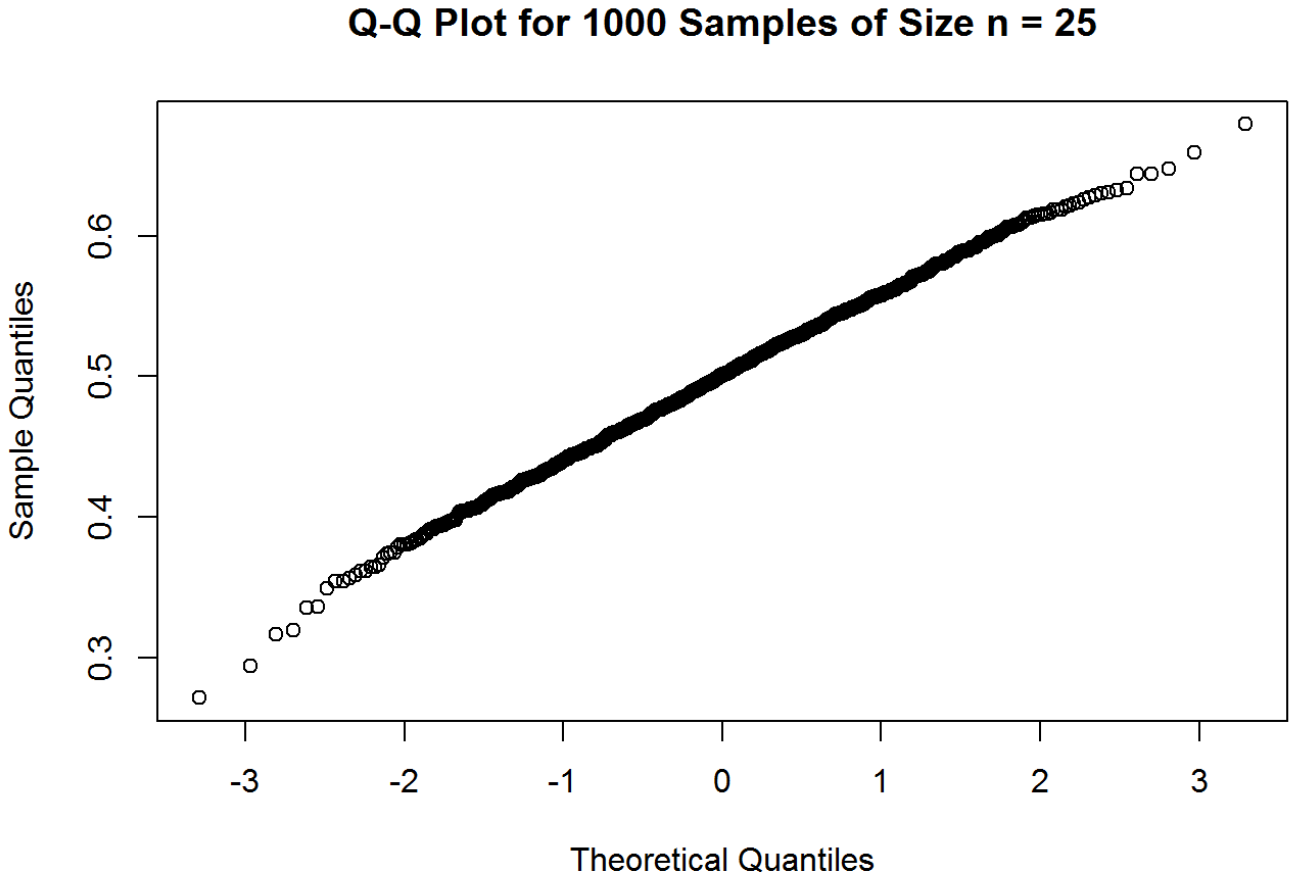
The Q-Q plot looks approximately normal, with most of the points lying on or very close to a line with slope m = 1 (the standard deviation. Thus we can conclude that the sample mean is converging to a normal (0,1) random variable

d)

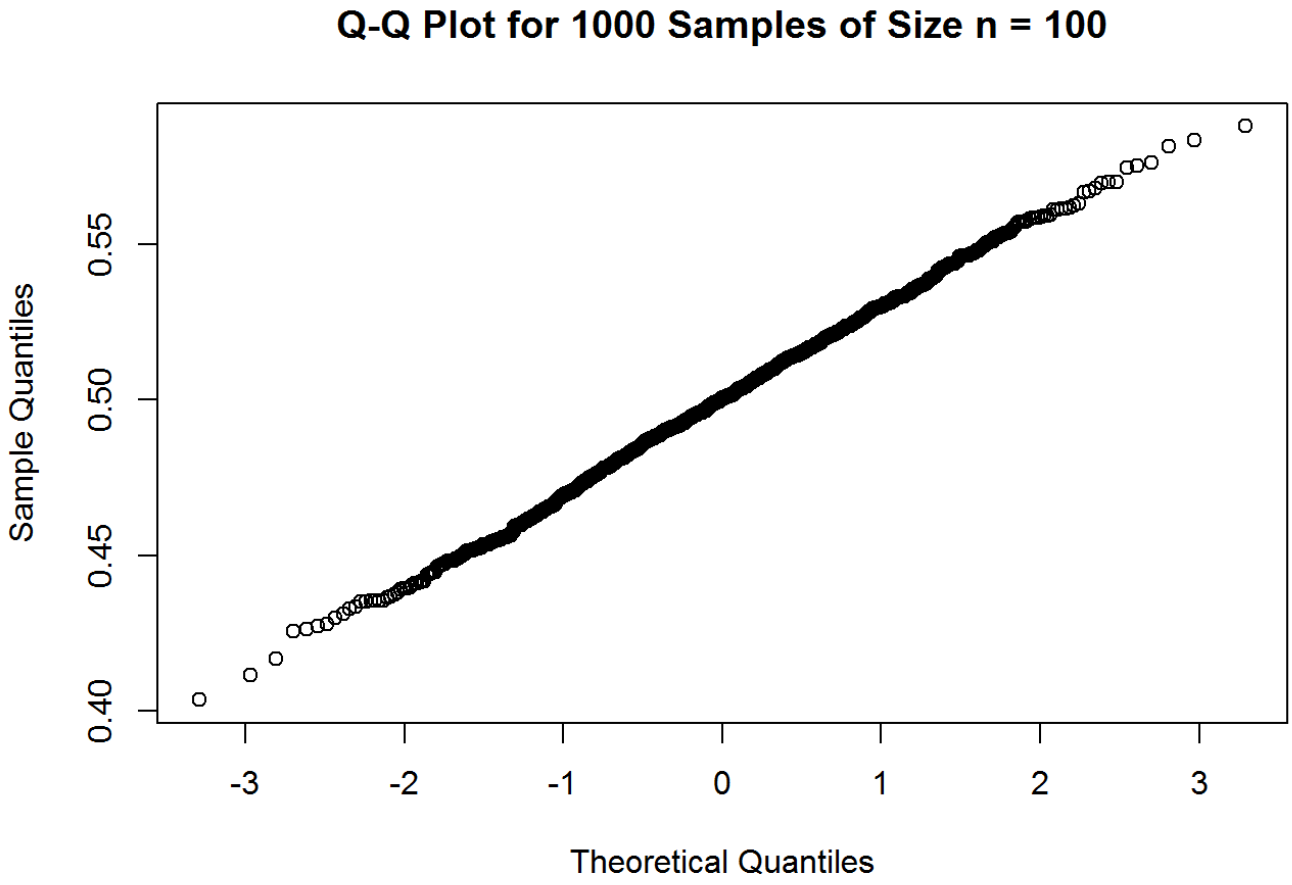
```
#n= 10, using the same functions as above. rowSums() gives me the sum of each row i
n the matrix
ten <- matrix(runif(n= 1000*10), nrow = 1000, ncol = 10)
xbar10 <- (1/(ncol(ten)))*(rowSums(ten))
qqnorm(xbar10, main = "Q-Q Plot for 1000 Samples of Size n = 10")
```




```r
#n = 25
twofive <- matrix(runif(n= 1000*25), nrow = 1000, ncol = 25)
xbar25 <- (1/(ncol(twofive)))*(rowSums(twofive))
qqnorm(xbar25, main = "Q-Q Plot for 1000 Samples of Size n = 25")
```




```r
#n = 100
hund <- matrix(runif(n= 1000*100), nrow = 1000, ncol = 100)
xbar100 <- (1/(ncol(hund)))*(rowSums(hund))
qqnorm(xbar100, main = "Q-Q Plot for 1000 Samples of Size n = 100")
```



e)

We see that as the sample size increases, the Q-Q plots show increased clustering around the 50th quantile, which indicates that it is converging even closer to a normal random variable as we increase n. Note that all of the Q-Q plots indicate approximate normality for the sample mean.

Exercise 3)

a)

```
#Read in the data
titanic <- read.table(file = "http://maitra.public.iastate.edu/stat579/datasets/titanic.txt", header = TRUE, sep = ",")
```

b)

```
#Cross tabulation, making sure that we're counting NAs as well
crosstab <- table(titanic$Sex, titanic$PClass, useNA = "ifany")
crosstab
```

```
##
##           1st 2nd 3rd
##    female 143 107 212
##    male   179 173 499
```

```
#Now a new table with further stratification. This gives us a 3-Dimensional table
additional <- table(titanic$Sex, titanic$PClass, titanic$Survived, useNA = "ifany"
)
additional
```

```
## , ,  = 0
##
##
##           1st 2nd 3rd
##    female   9  13 132
##    male   120 148 441
##
## , ,  = 1
##
##
##           1st 2nd 3rd
##    female 134  94  80
##    male    59  25  58
```

First, note that there are far more men than women in 3rd class (lowest class), so we would expect fewer men to survive. Next, it's easy to see that most of the females in 1st and 2nd class survived (3rd class not as lucky). In all 3 classes, a much lower proportion of the men survived (especially in 2nd and 3rd classes). So from the table, it's clear that women were helped first, then the men who were rescued most likely in order of their class.

c)

```r
#C
#First, subset into men & women
men <- subset(titanic, titanic$Sex == "male")
mensurvive <- subset(men, men$Survived == 1 & !is.na(men$Age))
mendead <- subset(men, men$Survived == 0 & !is.na(men$Age))
women <- subset(titanic, titanic$Sex == "female")
womensurvive <- subset(women, women$Survived == 1 & !is.na(women$Age))
womendead <- subset(women, women$Survived == 0 & !is.na(women$Age))

#Then, calculate mean age of those who survived and those who didn't separately, ne
ed to exclude NAs here
survmean.men <- mean(mensurvive$Age)
survmean.women <- mean(womensurvive$Age)
deadmean.women <- mean(womendead$Age)
deadmean.men <- mean(mendead$Age)


#Next, calculate standard errors using a function. Normally would need to include n
a.rm to remove NAs,
#but I'm already removing them so it doesn't matter for this exercise

stde <- function(x){
  sqrt(var(x)/ length(x))
}

#Calculate standard errors for each group
survstderr.women <- stde(womensurvive$Age)
survstderr.men <- stde(mensurvive$Age)
deadstderr.women <- stde(womendead$Age)
deadstderr.men <- stde(mendead$Age)

#View all of the values we calculated above
survmean.men
```

```
## [1] 25.95188
```

```r
deadmean.men
```

```
## [1] 32.32078
```

```r
survmean.women
```

```
## [1] 30.86714
```

```r
deadmean.women
```

```
## [1] 24.90141
```

```r
survstderr.men
```

```
## [1] 1.578879
```

```
deadstderr.men
```

```
## [1] 0.684307
```

```
survstderr.women
```

```
## [1] 1.01999
```

```
deadstderr.women
```

```
## [1] 1.548272
```

```
#We can just use the t-test function in R to test difference in mean age
t.test(mensurvive$Age, mendead$Age)
```

```
##
##  Welch Two Sample t-test
##
## data:  mensurvive$Age and mendead$Age
## t = -3.7011, df = 132.84, p-value = 0.0003134
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  -9.772608 -2.965201
## sample estimates:
## mean of x mean of y
##  25.95188  32.32078
```

```
t.test(womensurvive$Age, womendead$Age)
```

```
##
##  Welch Two Sample t-test
##
## data:  womensurvive$Age and womendead$Age
## t = 3.2177, df = 135.67, p-value = 0.001617
## alternative hypothesis: true difference in means is not equal to 0
## 95 percent confidence interval:
##  2.299144 9.632324
## sample estimates:
## mean of x mean of y
##  30.86714  24.90141
```

We can just use the t-test to test difference between means, however there are a lot of assumptions to make here. Note the p-values are small enough to reject null hypothesis, so we can conclude that there is a difference in mean age for those who survived and those who died. First, we know from the previous part that class had a large effect on survival chances, so by not controlling for possible age differences in class, we are assuming that age is the same across class (probably not true). We

are controlling for differences in gender, but this may have varied among classes as well. For both men and women, the results of the t-test indicate that there was a difference in age of those who survived and died, where younger men and older women were more likely to survive. However, this was most likely because the older female passengers were more likely to be in 1st class. Also note that we are excluding NAs for 3rd class passengers' Age, which could make a large difference in the outcome (can't make any assumptions on the ages of those who were NAs). Finally, we have not accounted for possible differences in population variances, although the samples are large enough that it may only be a small factor.

Exercise 4

a)

```
#Read in the data, row by row, to a 83 x 108 matrix
anat <- matrix(scan("anat.dat.txt"), nrow = 83, ncol = 108, byrow = TRUE)
activ <- matrix(scan("activ.dat.txt"), nrow = 83, ncol = 108, byrow = TRUE)
```

b and c)

```
#Create a grayscale color map of activation sites
require(grDevices); require(graphics);
filled.contour(activ, col = rev(gray(1:20/20)), axes=FALSE, main = "Brain Activati
on at Tapping of Finger",
               key.axes = axis(4, seq(0, max(activ, na.rm = TRUE), by = .05)))
#overlay the previous plot with the contour of the Brain's anatomy
#First need to resize the plot
par(oma=c(0,0,0,6))

#Then add the contour plot, note we need to add = TRUE to overlay
contour(anat, levels = c(100, seq(from = 250, to = 950, by = 100)), axes = FALSE,
drawlabels= FALSE, add = TRUE)
```

Brain Activation at Tapping of Finger