Albundy Shoe Sales

Objective: Compute confidence intervals for men shoe sales in the US between 2015 and 2016.

Goal: Determine how many shoes of each size should be in stock each month of the year to ensure customers will have product to buy and avoid overstocking.

```r
getwd() # check working directory
```

```
## [1] "/Users/mine/Downloads"
```

```r
setwd("/Users/mine/Downloads") #change to location of file to be imported
getwd() #check that file index has been changed
```

```
## [1] "/Users/mine/Downloads"
```

```r
library("readxl") #load required library to read excel file
shoe_data <- read_excel("3.17.Practical-example.Confidence-intervals-exercise.xlsx")
#read in the file
```

```r
shoe_data <- shoe_data[-c(1,2),]
#Remove empty rows in beginning of file
names(shoe_data) <- shoe_data[1,]
#Assign column names to the lead row
shoe_data <- shoe_data[-1,]
# Remove lead row as it has now replaced the column names
colnames(shoe_data)[12]<- "Year"
# assign column name to NA column after determining best fit column name
```

```r
library("openxlsx")
# load required library to adjust date column
shoe_data$Date <- convertToDate(shoe_data$Date)
# function to convert excel dates into readable dates in R
```

```r
us_data_2016 <- subset(shoe_data, Country !="United States"
                       & Year == "2016"& Gender == "Male")
#Subset data for US only, year of 2016, and gender is male
us_data_2015 <- subset(shoe_data, Country !="United States"
                       & Year == "2015"& Gender == "Male")
#This is done as our goal is to compare the sales between the two years.
```

```r
us_data_2015$month<-months(us_data_2015$Date)
# Insert month column  to create a table of size sales by month
us_data_2016$month<-months(us_data_2016$Date)
t_2015<- table(as.numeric(us_data_2015$`Size (US)`),
```

```r
                factor(us_data_2015$month,levels=month.name))
# create a table with the sizes and months for each year
t_2016<- table(as.numeric(us_data_2016$`Size (US)`),
                factor(us_data_2016$month,levels=month.name))
```

```r
df_2015<-as.data.frame.matrix(t_2015)
#Convert tables to dataframes for easier data calculations
df_2016<-as.data.frame.matrix(t_2016)
```

```r
df_combined<-cbind(df_2015, df_2016)
# combine two dataframes 2015 first 2016 second
```

```r
library("matrixStats")#load library to compute row operations on dataframes
mean<- rowMeans(df_combined) #Yields means by row
std_error<- rowSds(as.matrix(df_combined))/sqrt(24)
# yields standard deviation by row and then computes standard error

margin_error<- std_error*qt(.025, 23, lower.tail=FALSE)
```

```r
lower_ci<-mean-margin_error
# calculate upper and lower CI for mens shoe sales in the US
upper_ci<-mean+margin_error
```

```r
df_combined$mean<-mean # adding new calculations to existing dataframe
df_combined$std_error<-std_error
df_combined$margin_error<-margin_error
df_combined$lower_ci<-lower_ci
df_combined$upper_ci<-upper_ci
df_combined$pairstocarry<-round(upper_ci)
```

```r
df_combined[,c(25,30)]
```

```
##           mean pairstocarry
## 6     1.708333            2
## 6.5   1.625000            2
## 7     2.625000            4
## 7.5   3.625000            5
## 8     7.083333            8
## 8.5  16.125000           19
## 9    30.000000           35
## 9.5  38.958333           43
## 10   26.916667           31
## 10.5 28.333333           32
## 11   11.208333           13
## 11.5  8.333333           10
## 12    5.250000            7
## 13    1.708333            2
## 14    2.083333            3
## 15    3.041667            4
```

In conclusion the pairstocarry column represents the rough number of shoes to have in stock per size each month of the year. With this information I can be 95% percent sure that the number of shoes purchased will not exceed the value in the pairstocarry column for any given month of the year.