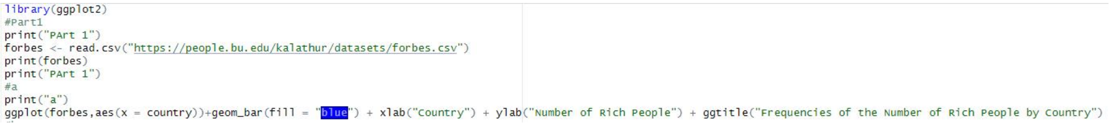
Weilin Lu

CS544

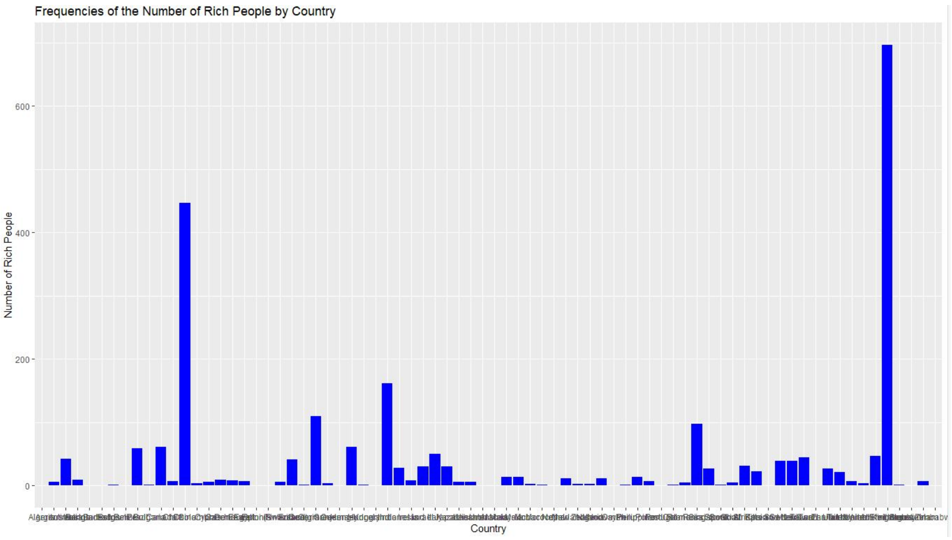
Assignment 3

Part 1

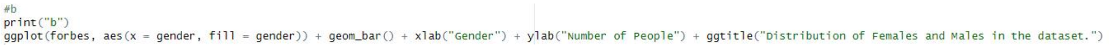
a) :



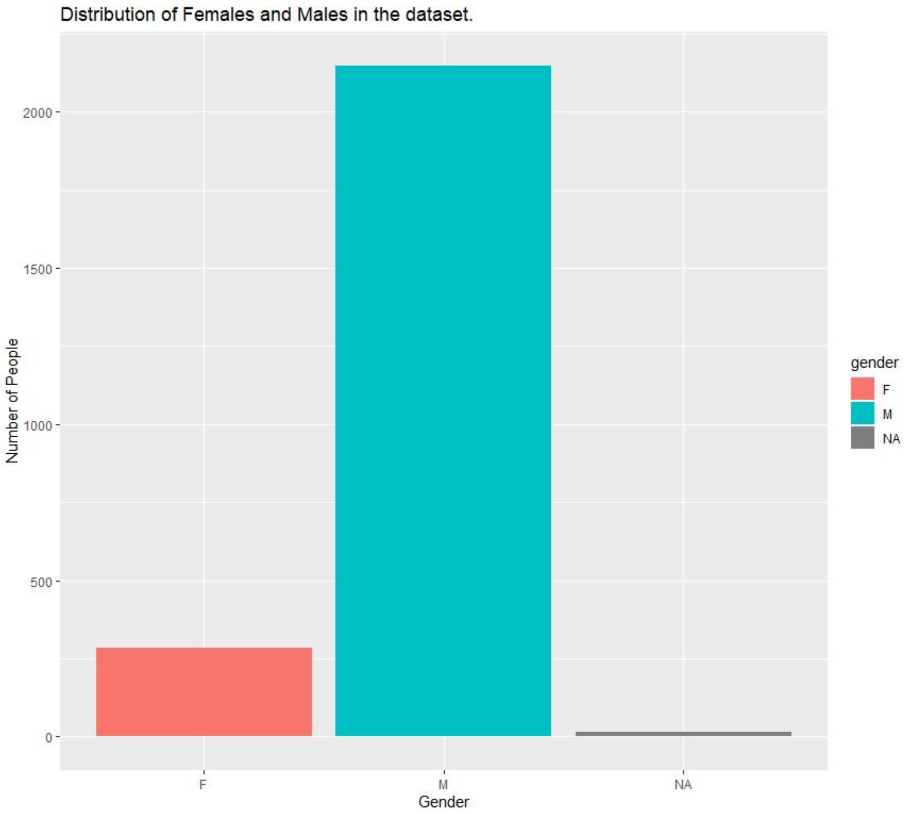
```
library(ggplot2)
#Part1
print("PArt 1")
forbes <- read.csv("https://people.bu.edu/kalathur/datasets/forbes.csv")
print(forbes)
print("PArt 1")
#a
print("a")
ggplot(forbes,aes(x = country))+geom_bar(fill = "blue") + xlab("Country") + ylab("Number of Rich People") + ggtitle("Frequencies of the Number of Rich People by Country")
```



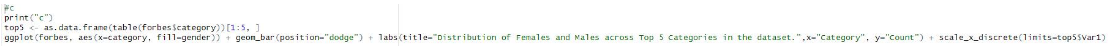
b) :



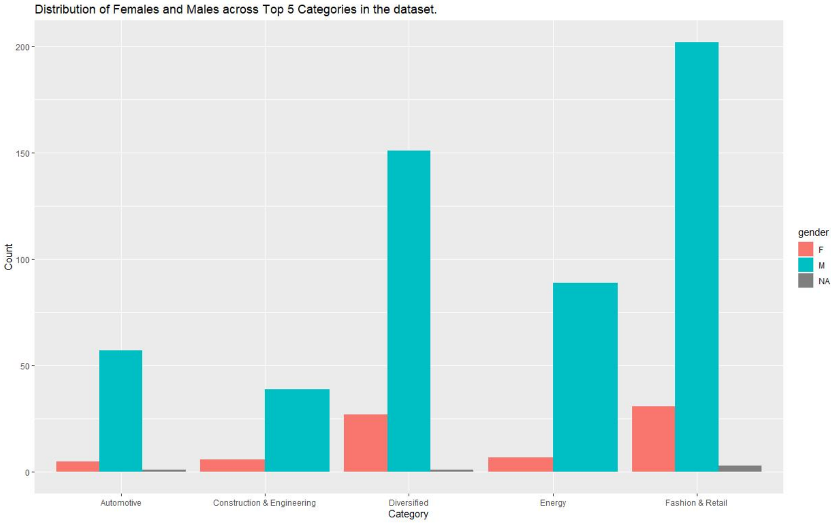
```
#b
print("b")
ggplot(forbes, aes(x = gender, fill = gender)) + geom_bar() + xlab("Gender") + ylab("Number of People") + ggtitle("Distribution of Females and Males in the dataset.")
```



c) :

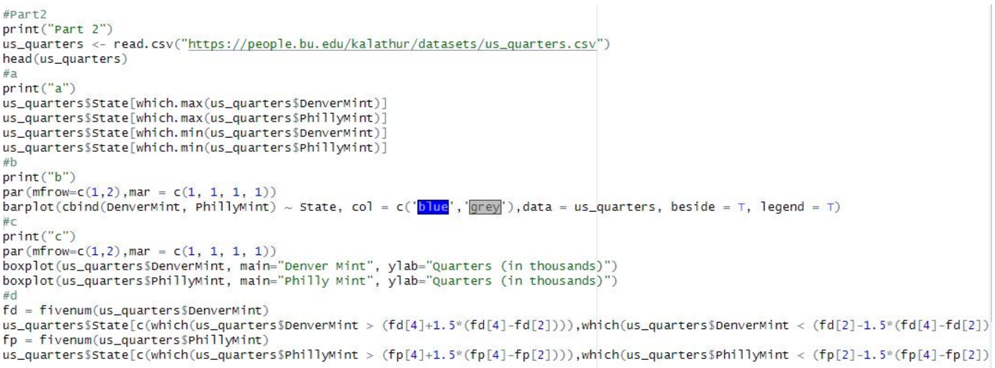


```
#c
print("c")
top5 <- as.data.frame(table(forbes$category))[1:5, ]
ggplot(forbes, aes(x=category, fill=gender)) + geom_bar(position="dodge") + labs(title="Distribution of Females and Males across Top 5 Categories in the dataset.",x="Category", y="Count") + scale_x_discrete(limits=top5$Var1)
```



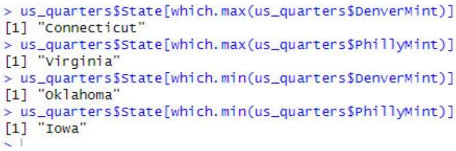
Part2

Code



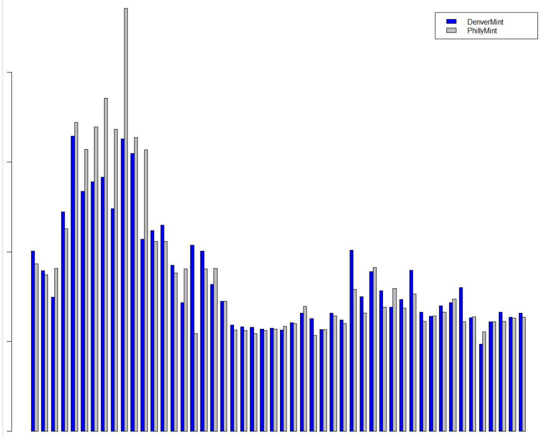
```
#Part2
print("Part 2")
us_quarters <- read.csv("https://people.bu.edu/kalathur/datasets/us_quarters.csv")
head(us_quarters)
#a
print("a")
us_quarters$State[which.max(us_quarters$DenverMint)]
us_quarters$State[which.max(us_quarters$PhillyMint)]
us_quarters$State[which.min(us_quarters$DenverMint)]
us_quarters$State[which.min(us_quarters$PhillyMint)]
#b
print("b")
par(mfrow=c(1,2),mar = c(1, 1, 1, 1))
barplot(cbind(DenverMint, PhillyMint) ~ State, col = c('blue','grey'),data = us_quarters, beside = T, legend = T)
#c
print("c")
par(mfrow=c(1,2),mar = c(1, 1, 1, 1))
boxplot(us_quarters$DenverMint, main="Denver Mint", ylab="Quarters (in thousands)")
boxplot(us_quarters$PhillyMint, main="Philly Mint", ylab="Quarters (in thousands)")
#d
fd = fivenum(us_quarters$DenverMint)
us_quarters$State[c(which(us_quarters$DenverMint > (fd[4]+1.5*(fd[4]-fd[2]))),which(us_quarters$DenverMint < (fd[2]-1.5*(fd[4]-fd[2])
fp = fivenum(us_quarters$PhillyMint)
us_quarters$State[c(which(us_quarters$PhillyMint > (fp[4]+1.5*(fp[4]-fp[2]))),which(us_quarters$PhillyMint < (fp[2]-1.5*(fp[4]-fp[2])
```

a) :



```
> us_quarters$State[which.max(us_quarters$DenverMint)]
[1] "Connecticut"
> us_quarters$State[which.max(us_quarters$PhillyMint)]
[1] "Virginia"
> us_quarters$State[which.min(us_quarters$DenverMint)]
[1] "Oklahoma"
> us_quarters$State[which.min(us_quarters$PhillyMint)]
[1] "Iowa"
```

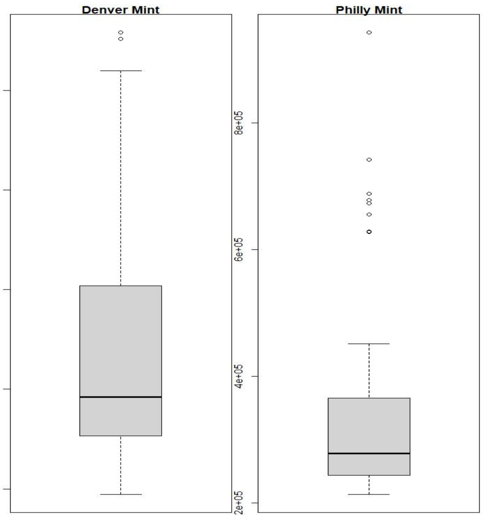
b) :



Two kind of mints have a strong correlation:these two mints cant have same amounts in each state.

From the barplot, i find there are some states have very high number of mints. These states could be outlier.

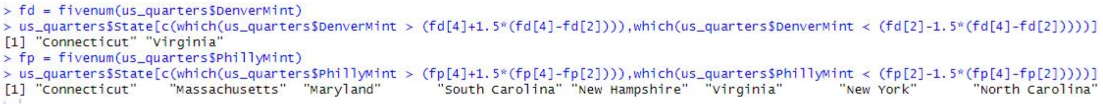
c) :



Philly mint has many outliers

Denver mint has a bigger range amount than Philly mint

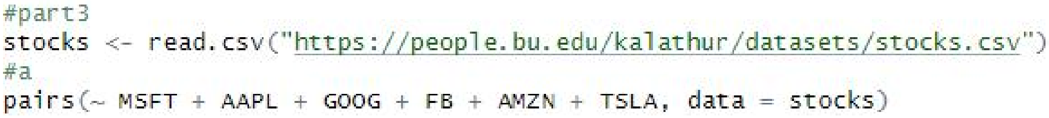
d) :



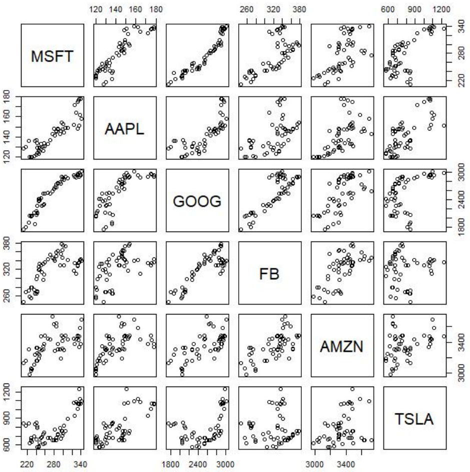
```
> fd = fivenum(us_quarters$DenverMint)
> us_quarters$State[c(which(us_quarters$DenverMint > (fd[4]+1.5*(fd[4]-fd[2]))),which(us_quarters$DenverMint < (fd[2]-1.5*(fd[4]-fd[2]))))]
[1] "Connecticut" "Virginia"
> fp = fivenum(us_quarters$PhillyMint)
> us_quarters$State[c(which(us_quarters$PhillyMint > (fp[4]+1.5*(fp[4]-fp[2]))),which(us_quarters$PhillyMint < (fp[2]-1.5*(fp[4]-fp[2]))))]
[1] "Connecticut"    "Massachusetts"  "Maryland"       "South Carolina" "New Hampshire"  "Virginia"       "New York"       "North Carolina"
```

Part 3

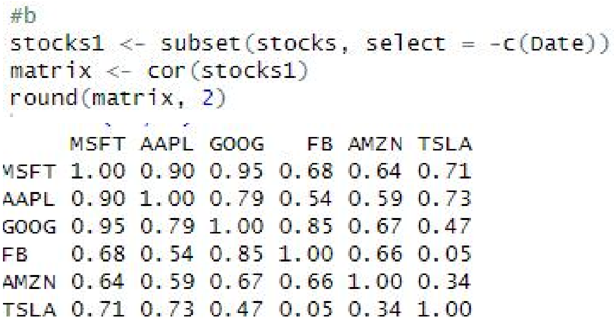
a):



```
#part3
stocks <- read.csv("https://people.bu.edu/kalathur/datasets/stocks.csv")
#a
pairs(~ MSFT + AAPL + GOOG + FB + AMZN + TSLA, data = stocks)
```



b):



```
#b
stocks1 <- subset(stocks, select = -c(Date))
matrix <- cor(stocks1)
round(matrix, 2)

     MSFT AAPL GOOG   FB AMZN TSLA
MSFT 1.00 0.90 0.95 0.68 0.64 0.71
AAPL 0.90 1.00 0.79 0.54 0.59 0.73
GOOG 0.95 0.79 1.00 0.85 0.67 0.47
FB   0.68 0.54 0.85 1.00 0.66 0.05
AMZN 0.64 0.59 0.67 0.66 1.00 0.34
TSLA 0.71 0.73 0.47 0.05 0.34 1.00
```

c):

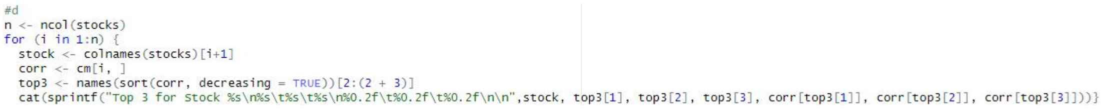
1:Two stocks move in the same direction, which means they have a positive correlation. Otherwise, it is a negative correlation.

2:If two stocks have a value of 1, it represents a strong correlation, and if it is -1, it is a negative correlation. If it is 0, there is no correlation

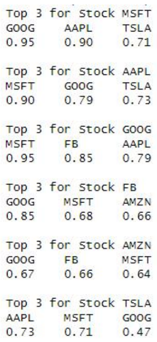
3:The matrix only systemizes the linear correlation between stocks and does not contain other relationships

4:If the diagonal elements of the correlation matrix are all 1, it means that the stock is strongly correlated with itself

d):



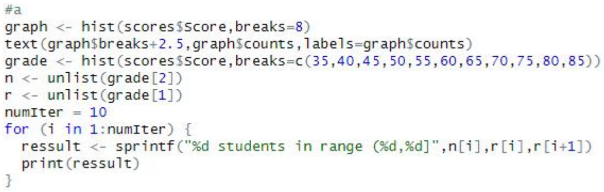
```
#d
n <- ncol(stocks)
for (i in 1:n) {
  stock <- colnames(stocks)[i+1]
  corr <- cm[i, ]
  top3 <- names(sort(corr, decreasing = TRUE))[2:(2 + 3)]
  cat(sprintf("Top 3 for Stock %s\n%s\t%s\t%s\n%0.2f\t%0.2f\t%0.2f\n\n",stock, top3[1], top3[2], top3[3], corr[top3[1]], corr[top3[2]], corr[top3[3]]))}
```



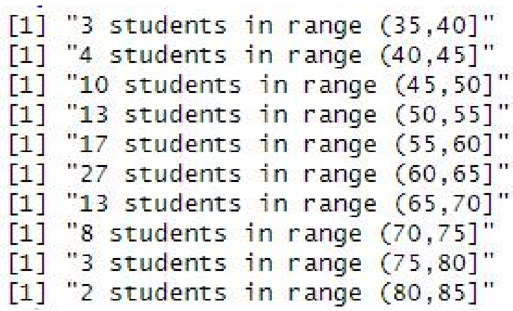
```
Top 3 for Stock MSFT
GOOG    AAPL    TSLA
0.95    0.90    0.71

Top 3 for Stock AAPL
MSFT    GOOG    TSLA
0.90    0.79    0.73

Top 3 for Stock GOOG
MSFT    FB      AAPL
0.95    0.85    0.79

Top 3 for Stock FB
GOOG    MSFT    AMZN
0.85    0.68    0.66

Top 3 for Stock AMZN
GOOG    FB      MSFT
0.67    0.66    0.64

Top 3 for Stock TSLA
AAPL    MSFT    GOOG
0.73    0.71    0.47
```

Part4

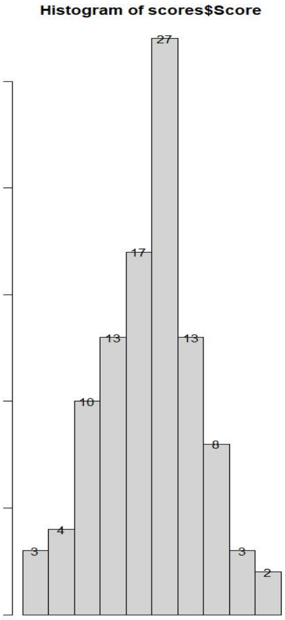
a) :



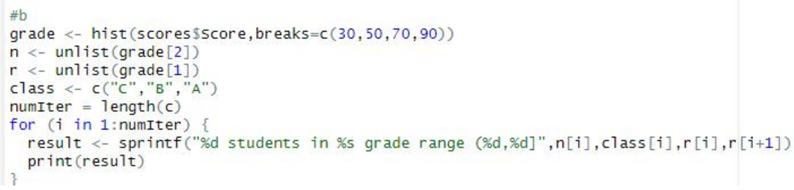
```
#a
graph <- hist(scores$Score,breaks=8)
text(graph$breaks+2.5,graph$counts,labels=graph$counts)
grade <- hist(scores$Score,breaks=c(35,40,45,50,55,60,65,70,75,80,85))
n <- unlist(grade[2])
r <- unlist(grade[1])
numIter = 10
for (i in 1:numIter) {
  ressult <- sprintf("%d students in range (%d,%d]",n[i],r[i],r[i+1])
  print(ressult)
}
```



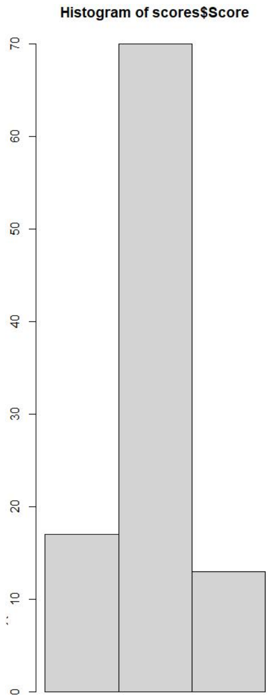
```
[1] "3 students in range (35,40]"
[1] "4 students in range (40,45]"
[1] "10 students in range (45,50]"
[1] "13 students in range (50,55]"
[1] "17 students in range (55,60]"
[1] "27 students in range (60,65]"
[1] "13 students in range (65,70]"
[1] "8 students in range (70,75]"
[1] "3 students in range (75,80]"
[1] "2 students in range (80,85]"
```

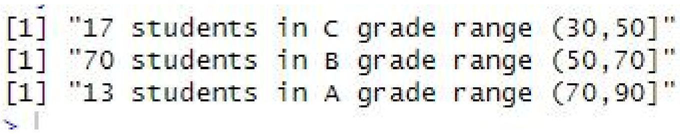


b) :



```
#b
grade <- hist(scores$Score,breaks=c(30,50,70,90))
n <- unlist(grade[2])
r <- unlist(grade[1])
class <- c("C","B","A")
numIter = length(c)
for (i in 1:numIter) {
  result <- sprintf("%d students in %s grade range (%d,%d]",n[i],class[i],r[i],r[i+1])
  print(result)
}
```





```
[1] "17 students in C grade range (30,50]"
[1] "70 students in B grade range (50,70]"
[1] "13 students in A grade range (70,90]"
>
```