Final Project

Christian Munger

4/30/2021

0: Company Descriptions
The companies chosen for this project are Microsoft and Wells Fargo & Company. Microsoft belongs to the technology sector of the economy. Microsoft develops and sells software and electronics for consumer use. Wells Fargo & Company is a United States financial services company that offers consumer banking, investment, and mortgage products. Wells Fargo & Company falls under the financial services sector of the economy. I chose these two companies to obtain a comparison for the risk involved when investing in a company based on technology versus one based on financial services.

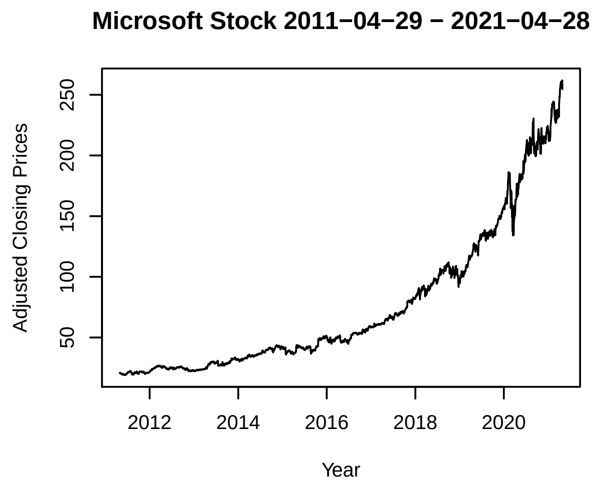
```
MSFT <- read.csv("C:\\Users\\cmunger\\OneDrive\\Risk Theory\\MSFT.csv")


WFC <- read.csv("C:\\Users\\cmunger\\OneDrive\\Risk Theory\\WFC.csv")


MSFT.prices = MSFT[,6]    #Gets adjusted closing prices
WFC.prices = WFC[,6]


MSFT.date=as.Date(MSFT[,1])
WFC.date=as.Date(WFC[,1])
```

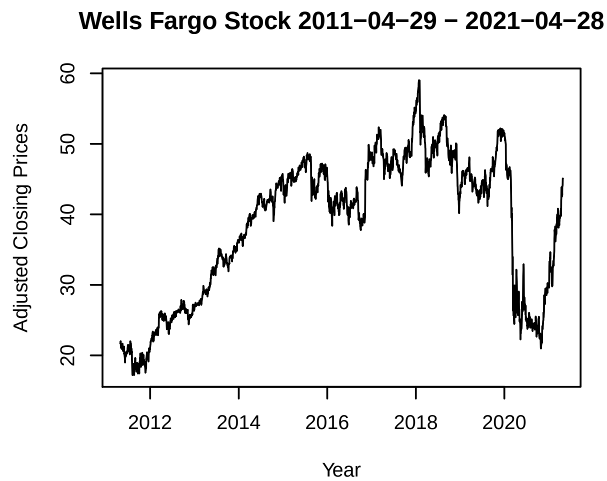
```
#Arithmetic Returns
arith.returns.MSFT <- -diff(MSFT.prices)/MSFT.prices[-length(MSFT.prices)]
arith.returns.WFC <- -diff(WFC.prices)/WFC.prices[-length(WFC.prices)]


#Log Returns
log.returns.MSFT <- diff(log(MSFT.prices))
log.returns.WFC <- diff(log(WFC.prices))
```

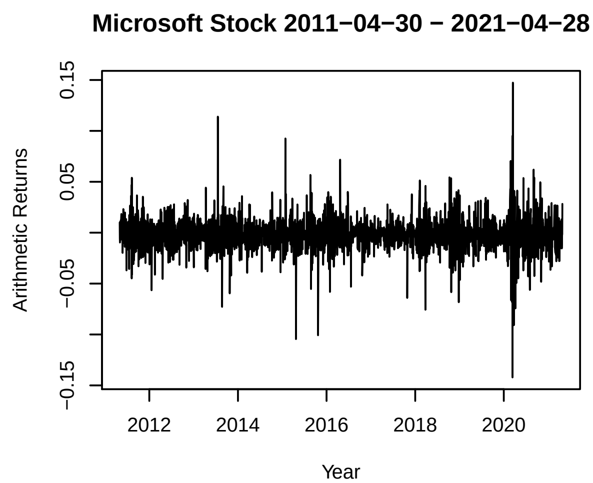
1: a)

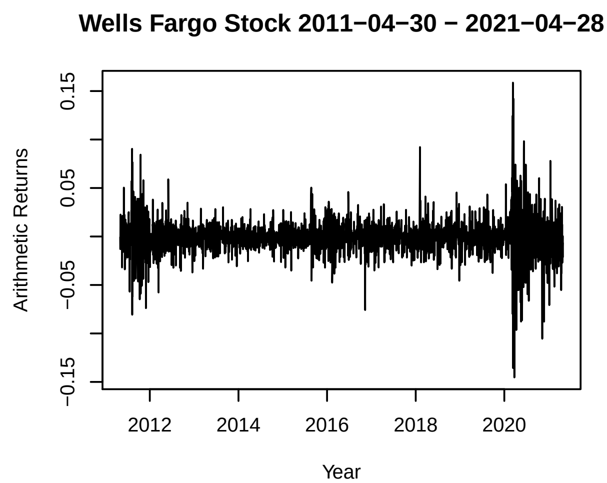
```
par(mfrow = c(3,2))
#Plots of Adj. prices
plot(MSFT.date,MSFT.prices, type = 'l',main="Microsoft Stock 2011-04-29 - 2021-04-28",
     xlab = 'Year', ylab = 'Adjusted Closing Prices')
plot(WFC.date, WFC.prices,  type = 'l',main="Wells Fargo Stock 2011-04-29 - 2021-04-28",
     xlab = 'Year', ylab = 'Adjusted Closing Prices')


#Plots of Arithmetic Returns
plot(MSFT.date[-1],arith.returns.MSFT, type = 'l',main="Microsoft Stock 2011-04-30 - 2021-04-28",
     xlab = 'Year', ylab = 'Arithmetic Returns')
plot(WFC.date[-1], arith.returns.WFC,  type = 'l',main="Wells Fargo Stock 2011-04-30 - 2021-04-28",
```

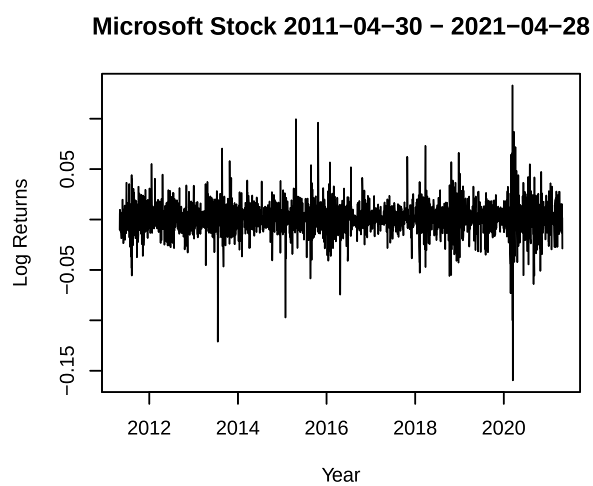
```
     xlab = 'Year', ylab = 'Arithmetic Returns')



#Plots of Log Returns
plot(MSFT.date[-1],log.returns.MSFT, type = 'l',main="Microsoft Stock 2011-04-30 - 2021-04-28",
     xlab = 'Year', ylab = 'Log Returns')
plot(WFC.date[-1], log.returns.WFC,  type = 'l',main="Wells Fargo Stock 2011-04-30 - 2021-04-28",
     xlab = 'Year', ylab = 'Log Returns')
```

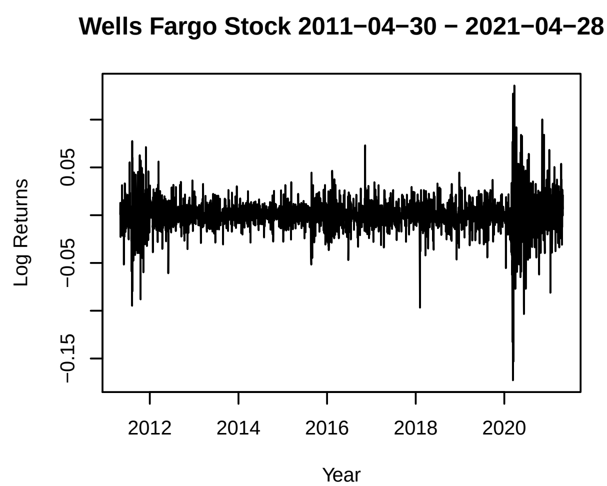








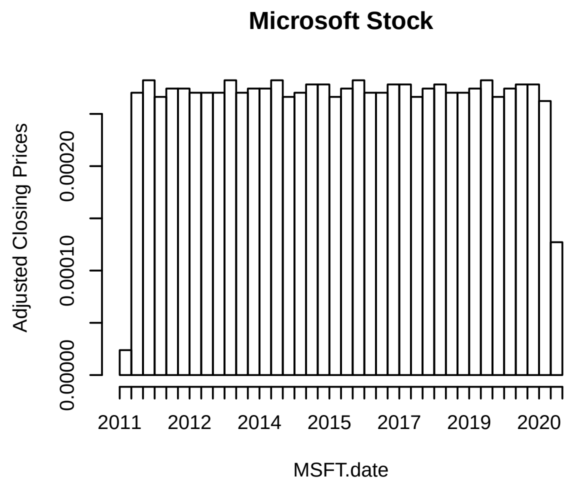


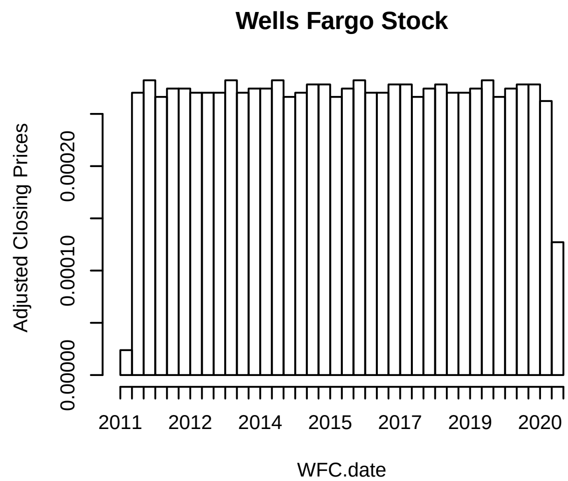


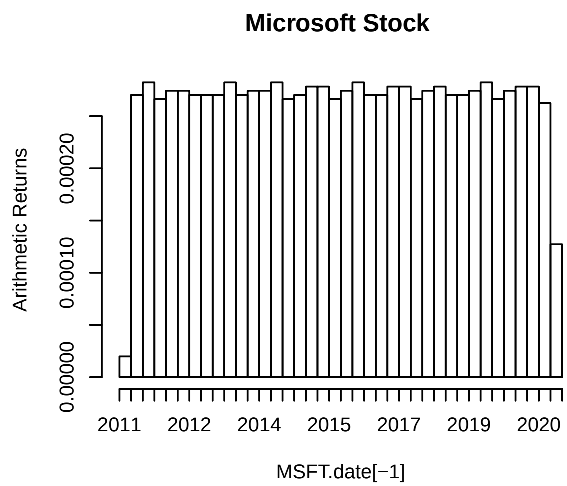
Since 2016, Microsoft has experienced tremendous growth in its daily closing prices. Through these plots, we can see several instances of volatility clustering. The most concerning amount of volatility clustering appears to occur in early 2020 when the covid pandemic began. However, it appears that Microsoft stock has responded better since the pandemic than Wells Fargo. Historically, there appear to be more occurrences of spiking volatility in Microsoft than Wells Fargo. This may be due to the large growth of closing prices Microsoft has experienced over the years.

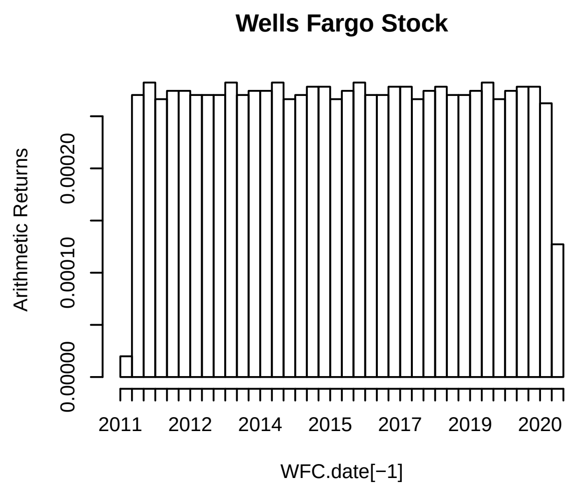
b)

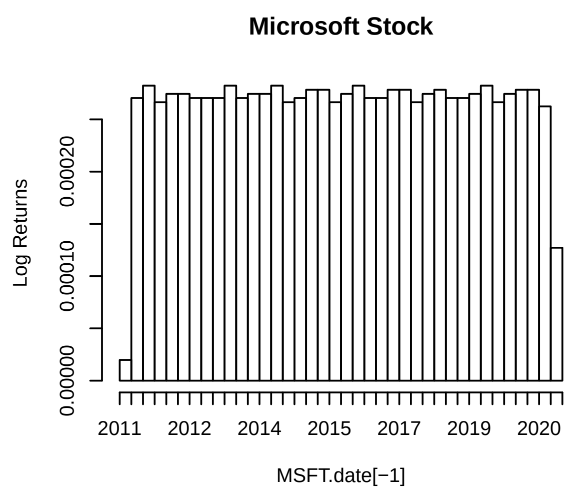
```r
par(mfrow = c(3,2))
#Histograms of Adj. prices
hist(MSFT.date,MSFT.prices, breaks = 50,main="Microsoft Stock", ylab = 'Adjusted Closing Prices')
hist(WFC.date, WFC.prices, breaks = 50,main="Wells Fargo Stock" ,ylab = 'Adjusted Closing Prices')


#Histograms of Arithmetic Returns
hist(MSFT.date[-1],arith.returns.MSFT, breaks = 50,main="Microsoft Stock ",
    , ylab = 'Arithmetic Returns')
hist(WFC.date[-1], arith.returns.WFC, breaks = 50,main="Wells Fargo Stock",
ylab = 'Arithmetic Returns')


#Histograms of Log Returns
hist(MSFT.date[-1],log.returns.MSFT, breaks = 50,main="Microsoft Stock",
 ylab = 'Log Returns')
hist(WFC.date[-1], log.returns.WFC, breaks = 50,main="Wells Fargo Stock" ,  ylab = 'Log Returns')
```

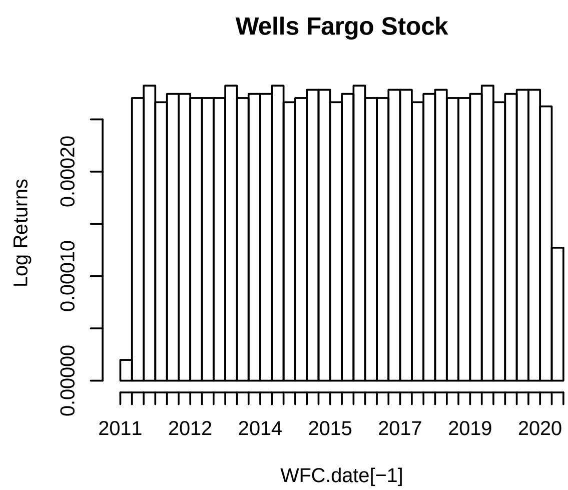








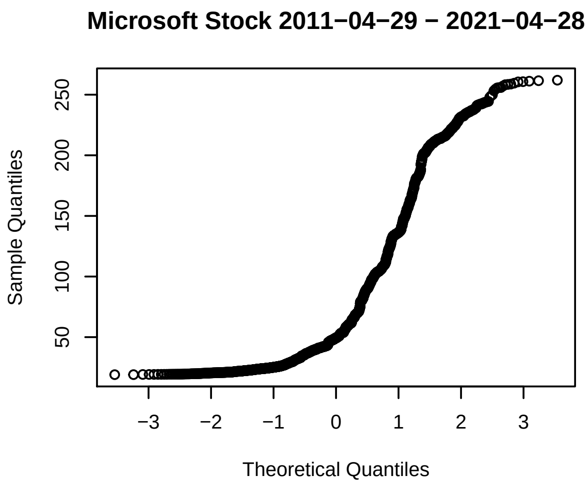


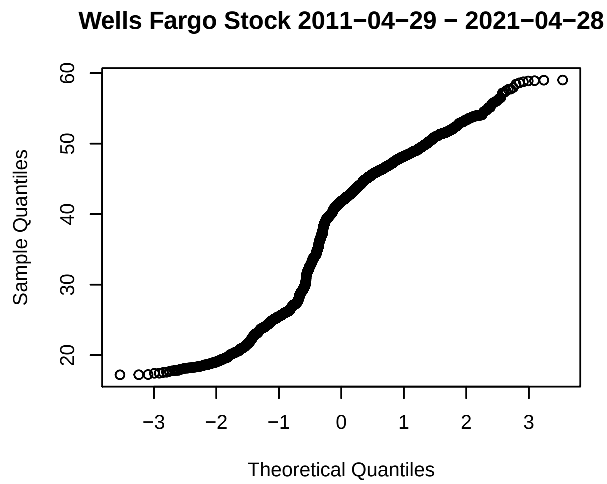


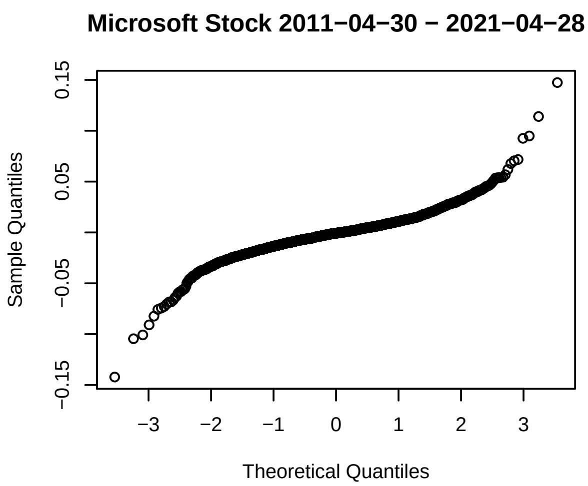
Among these histograms, there appears to be symmetry for both companies. The histograms do not appear to be normally distributed as there is no bell curve shape to the histograms. However, the histograms appear to be uniformly distributed with a flat shape throughout each plot.

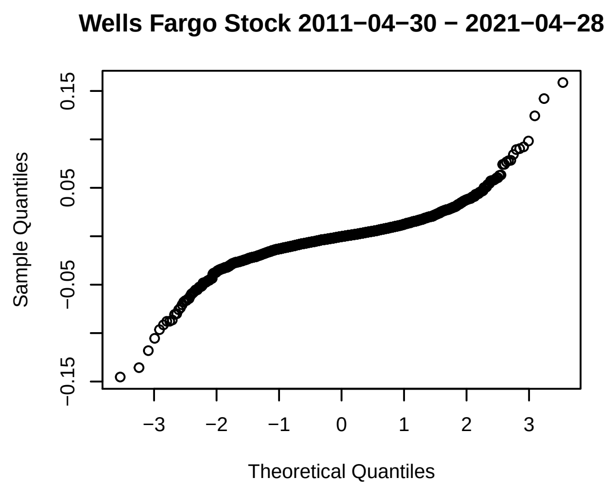
c)

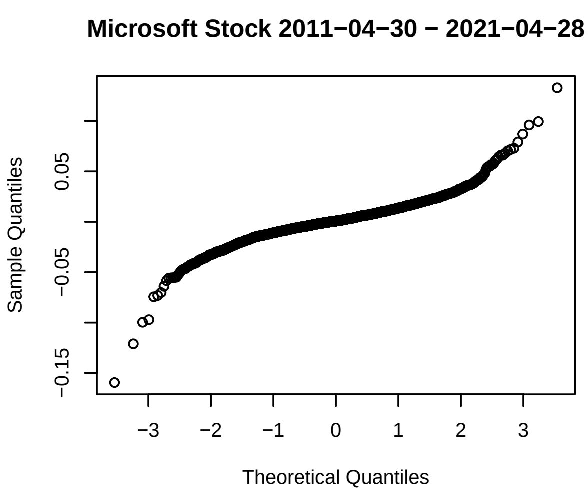
```r
par(mfrow = c(3,2))

#QQplot for Adjusted Closing Prices
qqnorm(MSFT.prices,main="Microsoft Stock 2011-04-29 - 2021-04-28")
qqnorm(WFC.prices,main="Wells Fargo Stock 2011-04-29 - 2021-04-28")

#QQplot for Arithmetic Returns
qqnorm(arith.returns.MSFT,main="Microsoft Stock 2011-04-30 - 2021-04-28")
qqnorm(arith.returns.WFC,main="Wells Fargo Stock 2011-04-30 - 2021-04-28")

#QQplot for Log Returns
qqnorm(log.returns.MSFT,main="Microsoft Stock 2011-04-30 - 2021-04-28")
qqnorm(log.returns.WFC,main="Wells Fargo Stock 2011-04-30 - 2021-04-28")
```

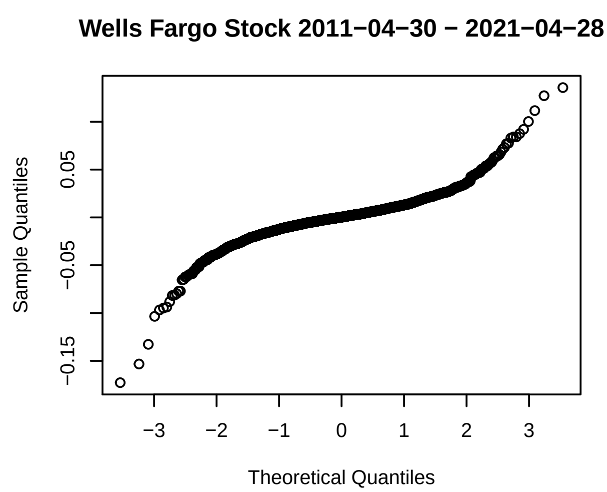












Again, these distributions do not appear to be normally distributed. Through each of the qqplots, there is heavy curving, especially in the tails. This is indicative of a non-normal distribution.

2: a)

```r
#Mean of negative arithmetic returns
mean.MSFT <- mean(-arith.returns.MSFT)
mean.MSFT
```

```
## [1] 0.00113131
```

```r
mean.WFC <- mean(-arith.returns.WFC)
mean.WFC
```

```
## [1] 0.0004632717
```

```r
#Variance of negative arithmetic returns
variance.MSFT <- var(-arith.returns.MSFT)
variance.MSFT
```

```
## [1] 0.0002654581
```

```r
variance.WFC <- var(-arith.returns.WFC)
variance.WFC
```

```
## [1] 0.0003448898
```

b)

```r
#Parametric approach to estimate VaR.95 and ES.95
VaR.MSFT_1 <- 100*(sqrt(variance.MSFT)*1.645 + mean.MSFT)
VaR.MSFT_1
```

```
## [1] 2.793311
```

```r
VaR.WFC_1 <- 100*(sqrt(variance.WFC)*1.645 + mean.WFC)
VaR.WFC_1
```

```
## [1] 3.101291
```

```r
ES.MSFT_1 <- 100*(sqrt(variance.MSFT)/(.05*sqrt(2*pi))*exp(-(1.645^2)/2) + mean.MSFT)
ES.MSFT_1
```

```
## [1] 3.473076
```

```r
ES.WFC_1 <- 100*(sqrt(variance.WFC)/(.05*sqrt(2*pi))*exp(-(1.645^2)/2) + mean.WFC)
ES.WFC_1
```

```
## [1] 3.876112
```

c)

```r
#Nonparametric(Historical) approach to calculate VaR.95 and ES.95
w <- .05*2515
w    #Round down to 125
```

```
## [1] 125.75
```

```r
#Sorting the returns from smallest to largest
MSFT.sorted <- sort(arith.returns.MSFT)
WFC.sorted <- sort(arith.returns.WFC)

#Multiply by -1
negMSFT.sorted <- -1*MSFT.sorted
negWFC.sorted <- -1*WFC.sorted
```

```r
#Using w to get the return at position 125
negMSFT.sorted[125]
```

```
## [1] 0.02450515
```

```r
negWFC.sorted[125]
```

```
## [1] 0.02648294
```

```r
hist.VaR.MSFT <- 100*negMSFT.sorted[125]
hist.VaR.MSFT
```

```
## [1] 2.450515
```

```r
hist.VaR.WFC <- 100*negWFC.sorted[125]
hist.VaR.WFC
```

```
## [1] 2.648294
```

```r
hist.ES.MSFT <- 100*(sum(negMSFT.sorted[1:125])/125)
hist.ES.MSFT
```

```
## [1] 3.889054
```

```r
hist.ES.WFC <- 100*(sum(negWFC.sorted[1:125])/125)
hist.ES.WFC
```

```
## [1] 4.445121
```

d)

```r
upperbound.MSFT <- 100*sqrt(variance.MSFT)*sqrt(1/(2*.05))
upperbound.MSFT
```

```
## [1] 5.152263
```

```r
upperbound.WFC <- 100*sqrt(variance.WFC)*sqrt(1/(2*.05))
upperbound.WFC
```

```
## [1] 5.872732
```

3: a)

```r
#Mean Vector
mean.vector_1 <-matrix(c(mean(arith.returns.MSFT),mean(arith.returns.WFC)),nrow =2,ncol =1)
mean.vector_1
```

```
##                [,1]
## [1,] -0.0011313101
## [2,] -0.0004632717
```

```r
#Covariance Matrix
cov.matrix <- matrix(data = c(cov(cbind(arith.returns.MSFT,arith.returns.WFC))), nrow = 2, ncol = 2)
cov.matrix
```

```
##              [,1]         [,2]
## [1,] 0.0002654581 0.0001359078
## [2,] 0.0001359078 0.0003448898
```

```r
cor(arith.returns.MSFT, arith.returns.WFC)
```

```
## [1] 0.4491655
```

The correlation of 0.4491655 is rather large for the two companies. This means that over time there is a positive linear trend for the arithmetic returns for Microsoft and Wells Fargo.

b)

```
#Weight vector for portfolio investing half in each company
weighted_1 <- matrix(data =c(.5,.5),nrow =2,ncol =1)
weighted_1
```

```
##      [,1]
## [1,]  0.5
## [2,]  0.5
```

```
#Expected Return
portfolio.mean_1 <-t(weighted_1)%*%mean.vector_1
portfolio.mean_1
```

```
##               [,1]
## [1,] -0.0007972909
```

```
#Portfolio Variance
portfolio.variance_1 <-t(weighted_1)%*%cov.matrix%*%weighted_1
portfolio.variance_1
```

```
##              [,1]
## [1,] 0.0002205409
```

c)

```
portfolio.VaR <- 1000*(sqrt(portfolio.variance_1)*2.326 - portfolio.mean_1)
portfolio.VaR
```

```
##          [,1]
## [1,] 35.33983
```

4:We are now assuming the mean vector of the data is the zero vector. This is a reasonable assumption as the previously calculated mean vector is very small(<.005).

a)

```
#Weight vector with .75 in MSFT and .25 in WFC
weighted_2 <- matrix(data =c(.75,.25),nrow =2,ncol =1)
weighted_2
```

```
##      [,1]
## [1,] 0.75
## [2,] 0.25
```

```
#Variance of portfolio
portfolio.variance_2 <-t(weighted_2)%*%cov.matrix%*%weighted_2
portfolio.variance_2
```

```
##              [,1]
## [1,] 0.0002218413
```

```
#VaR.95 for this portfolio
VaR.old <- 1000*sqrt(portfolio.variance_2)*1.645
VaR.old
```

```
##          [,1]
## [1,] 24.50118
```

b)

```r
#Individual VaR's using variances from covariance matrix
ind.VaR.1 <- 750*1.645*sqrt(0.0002667026)
ind.VaR.1
```

```
## [1] 20.14841
```

```r
ind.VaR.2 <- 250*1.645*sqrt(0.0003497544)
ind.VaR.2
```

```
## [1] 7.691083
```

c)

```r
#Undiversified VaR
undiversified.VaR <- ind.VaR.1 + ind.VaR.2
undiversified.VaR
```

```
## [1] 27.83949
```

We see that Undiversified VaR is greater than or equal to Diversified VaR. This is reasonable as we would expect a diversified portfolio to have less risk than an undiversified portfolio.

d)

```r
#Beta vector
beta <- (1/0.0002218413)*(cov.matrix%*%weighted_2)     #0.0002218413 is the portfolio variance
beta
```

```
##           [,1]
## [1,] 1.0506185
## [2,] 0.8481438
```

So, for each asset we have $\beta_1 = 1.0506185$ and $\beta_2 = 0.8481438$

e)

```r
delta.VaR <- 1.645*sqrt(0.0002218413)*beta
delta.VaR
```

```
##            [,1]
## [1,] 0.02574140
## [2,] 0.02078053
```

For each asset we have $\Delta VaR_1 = 0.02574140$ and $\Delta VaR_2 = 0.02078053$

f)

```r
#Approximate incremental VaR
approx.incremental.VaR <- -10*0.02574140 + 5*0.02078053
approx.incremental.VaR
```

```
## [1] -0.1535114
```

```r
W.new <- matrix(c(740, 255), byrow = T)
W.new
```

```
##      [,1]
## [1,]  740
## [2,]  255
```

```r
#New weight vector with parameters selling $10 of first stock and investing $5 in second stock.
weighted.new <- matrix(c(740/955, 255/955), byrow = T)
weighted.new
```

```
##           [,1]
## [1,] 0.7748691
## [2,] 0.2670157
```

```r
variance.new <- t(weighted.new)%*%cov.matrix%*%weighted.new
variance.new
```

```
##             [,1]
## [1,] 0.000240216
```

```r
VaR.new <- 955*sqrt(variance.new)*1.645
VaR.new
```

```
##          [,1]
## [1,] 24.34839
```

```r
#Exact incremental VaR
exact.incremental.VaR <- VaR.new - VaR.old
exact.incremental.VaR
```

```
##            [,1]
## [1,] -0.1527962
```

After approximating incremental VaR and using the exact calculation for incremental VaR we can see that the numbers are very similar. However, the approximation takes much less time to calculate. Thus, one would use the approximation method if they needed to calculate incremental VaR quickly.

5: a)

```r
hedge <- -(0.0001359078*750)/0.0003448898
hedge
```

```
## [1] -295.5461
```

```r
#The cost of the resulting portfolio is (750, -295.5461). So, we are now investing $454.4539 total.
```

This hedge appears to be a reasonable investment that lowers the initial investment from the previous problem. This is done to minimize the VaR and offset the chance that our investment will lose value.

b)

```r
#Weight vector using our hedge
weighted.hedge <- matrix(c(750/454.4539, -295.5461/454.4539), byrow = T)
weighted.hedge
```

```
##            [,1]
## [1,]  1.6503324
## [2,] -0.6503324
```

```r
#Variance using our hedge
variance.hedge <- t(weighted.hedge)%*%cov.matrix%*%weighted.hedge
variance.hedge
```

```
##              [,1]
## [1,] 0.0005771359
```

```r
#VaR.95 with hedge investment
VaR.hedge <- 454.4539*sqrt(variance.hedge)*1.645
VaR.hedge
```

```
##          [,1]
## [1,] 17.95952
```

Compared to the previously calculated VaR in part 4a, this VaR using the hedge is much less.