# Week 2 Assignment

*Matthew Dunne*

*January 17, 2019*

## Week 2: Homework Assignment

### This assignment helps understanding Markowitz efficient frontier, CAPM and APT model

### This assignment is individual

Create efficient frontier, CAPM model and APT model for a group of stocks representing health care sector and industrial sector.
The names of the selected companies are in the file `Industrials_Health_Names.csv`.
The period of observation is `from="2014-7-1",to="2015-7-1"`.

For the sector indices use SPDR XLV (health care sector) and XLI (industrial sector).
For the broad market index use SPY.
For the risk-free rate use Fed Funds effective rate.

Note that it may not be possible to find interpretation of PCA factors in terms of real assets or indices.
In such cases it is possible to use PCA factors without interpretation.

```
suppressWarnings(library(quantmod))
datapath<-"C:/Users/mjdun/Desktop/Financial Analytics/Week 2"
#you added column names in .csv because header=F wasn't working
SP500.Industrials.Health<-read.csv(file=paste(datapath,"Industrials_Health_Names.csv",sep="/"),header=F)
SP500.Industrials.Health.names<-as.character(SP500.Industrials.Health[,1])
SP500.Industrials.Health
```

```
##        V1                     V2          V3
## 1  Ticker                   Name      Sector
## 2     CAT            Caterpillar Industrials
## 3     FDX             FedEx Corp Industrials
## 4      GE          Genl Electric Industrials
## 5     HON Honeywell International Industrials
## 6     LMT        Lockheed Martin Industrials
## 7     NOC       Northrop Grumman Industrials
## 8     UNP          Union Pacific Industrials
## 9     UPS          United Parcel Industrials
## 10    UTX     United Technologies Industrials
## 11     WM       Waste Management Industrials
## 12    ABT     Abbott Laboratories Health Care
## 13    AET                   Aetna Health Care
## 14    HUM                  Humana Health Care
## 15    JNJ       Johnson & Johnson Health Care
## 16    MDT               Medtronic Health Care
## 17    PFE                  Pfizer Health Care
```

```
#suppressWarnings(getSymbols(SP500.Industrials.Health.names,from="2014-7-1",to="2015-7-1"))
suppressWarnings(getSymbols("CAT",from="2014-7-1",to="2015-7-1"))
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```
##
## WARNING: There have been significant changes to Yahoo Finance data.
## Please see the Warning section of '?getSymbols.yahoo' for details.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.yahoo.warning"=FALSE).
```

```
## [1] "CAT"
```

```r
suppressWarnings(getSymbols("FDX",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "FDX"
```

```r
suppressWarnings(getSymbols("GE",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "GE"
```

```r
suppressWarnings(getSymbols("HON",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "HON"
```

```r
suppressWarnings(getSymbols("LMT",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "LMT"
```

```r
suppressWarnings(getSymbols("NOC",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "NOC"
```

```r
suppressWarnings(getSymbols("UNP",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "UNP"
```

```r
suppressWarnings(getSymbols("UPS",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "UPS"
```

```r
suppressWarnings(getSymbols("UTX",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "UTX"
```

```r
suppressWarnings(getSymbols("WM",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "WM"
```

```r
suppressWarnings(getSymbols("ABT",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "ABT"
```

```r
suppressWarnings(getSymbols("AET",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "AET"
```

```r
suppressWarnings(getSymbols("HUM",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "HUM"
```

```r
suppressWarnings(getSymbols("JNJ",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "JNJ"
```

```r
suppressWarnings(getSymbols("MDT",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "MDT"
```

```r
suppressWarnings(getSymbols("PFE",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "PFE"
```

```r
suppressWarnings(getSymbols("XLV",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "XLV"
```

```r
suppressWarnings(getSymbols("XLI",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "XLI"
```

```r
suppressWarnings(getSymbols("SPY",from="2014-7-1",to="2015-7-1"))
```

```
## [1] "SPY"
```

Combine these all into one dataframe. Use the Adjusted prices.

```r
Stock.Returns<-cbind(CAT=CAT$CAT.Adjusted, FDX=FDX$FDX.Adjusted, GE=GE$GE.Adjusted, HON=HON$HON.Adjustec
#Had to do it one by one for markdown
colnames(Stock.Returns)[1]="CAT"
colnames(Stock.Returns)[2]="FDX"
colnames(Stock.Returns)[3]="GE"
colnames(Stock.Returns)[4]="HON"
colnames(Stock.Returns)[5]="LMT"
colnames(Stock.Returns)[6]="NOC"
colnames(Stock.Returns)[7]="UNP"
colnames(Stock.Returns)[8]="UPS"
colnames(Stock.Returns)[9]="UTX"
colnames(Stock.Returns)[10]="WM"
colnames(Stock.Returns)[11]="ABT"
colnames(Stock.Returns)[12]="AET"
colnames(Stock.Returns)[13]="HUM"
colnames(Stock.Returns)[14]="JNJ"
colnames(Stock.Returns)[15]="MDT"
colnames(Stock.Returns)[16]="PFE"
colnames(Stock.Returns)[17]="XLV"
colnames(Stock.Returns)[18]="XLI"
colnames(Stock.Returns)[19]="SPY"
```

Also load the Fed Funds overnight rates for the relevant dates.

```r
datapath<-"C:/Users/mjdun/Desktop/Financial Analytics/Week 2/"
FedFunds.BD<-read.csv(file=paste(datapath,"RIFSPFF_NB.csv",sep="/"))
#convert to date in order to subset
FedFunds.BD$Time.Period<-as.Date(FedFunds.BD$Time.Period,"%m/%d/%Y")
FedFunds.BD <- subset(FedFunds.BD, Time.Period >= "2014-7-1" & Time.Period < "2015-7-1")
```

## Efficient Frontier

Calculate means and standard deviations of daily log-returns for each company.
Calculate mean and standard deviation of daily log-returns for XLV, XLI, and SPY.

```
#get mean and standard deviation of all stocks, indices
Mean.Sd.Stock.Returns<-cbind(sd=apply(Stock.Returns,2,function(z) sd(diff(log(z)))),
                              mean=apply(Stock.Returns,2,function(z) mean(diff(log(z)))))
(Mean.Sd.Stock.Returns)
```

```
##              sd          mean
## CAT 0.013834356 -8.847112e-04
## FDX 0.011283669  4.650245e-04
## GE  0.011564754  1.666120e-04
## HON 0.010592588  3.826395e-04
## LMT 0.010022342  7.349640e-04
## NOC 0.011592605  1.175922e-03
## UNP 0.013983674 -1.184947e-04
## UPS 0.011226796 -1.268324e-04
## UTX 0.010092461 -8.820738e-05
## WM  0.009021509  2.543312e-04
## ABT 0.010626694  7.829841e-04
## AET 0.014065563  1.797412e-03
## HUM 0.020449395  1.588896e-03
## JNJ 0.009741115 -2.203017e-04
## MDT 0.012068492  6.614433e-04
## PFE 0.009010951  5.697634e-04
## XLV 0.009268357  8.069821e-04
## XLI 0.008708534  5.697652e-05
## SPY 0.007552922  2.528348e-04
```

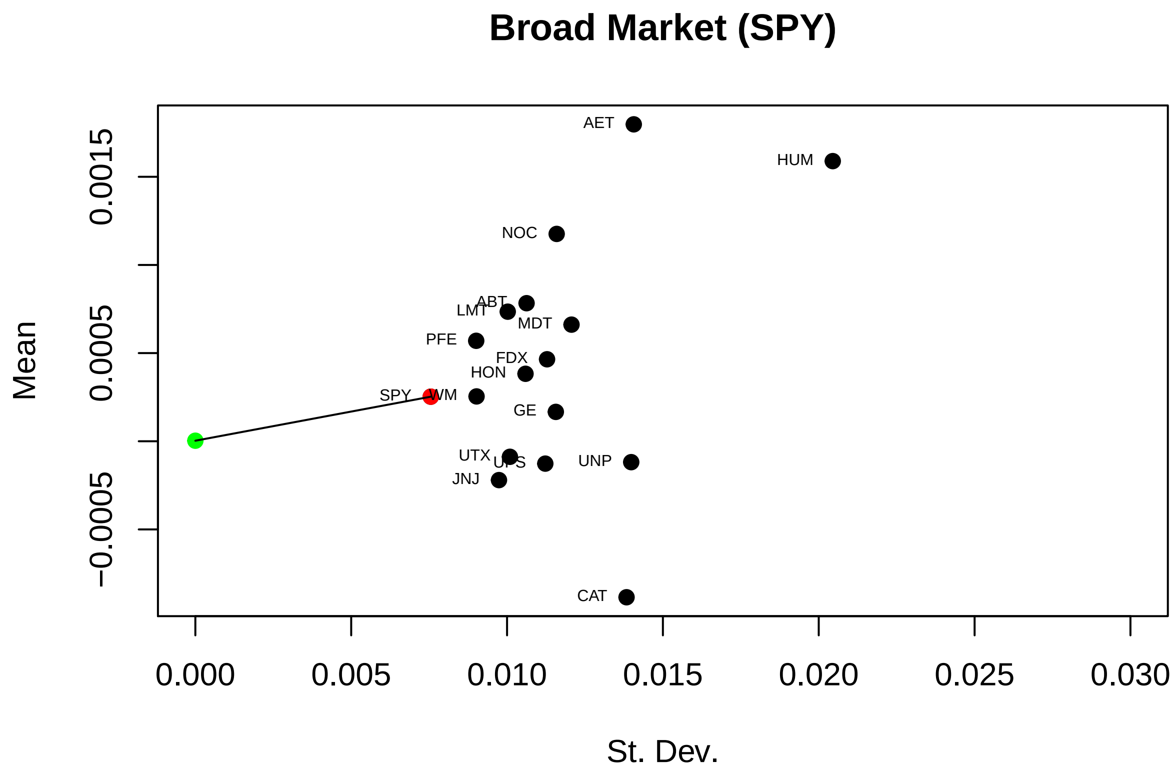
Calculate mean Fed Funds rate for the period 7/1/2014 to 6/30/2015.

```
#expected rate of return of Risk Free Rate. Used 360 instead of 365
Mean.FedFunds<-mean(FedFunds.BD[,2])/100/360
```

### The Broad Market Index (SPY)

Plot the companies and indices on standard deviation-mean diagram.
Add the points for the indices and risk-free rate.
Add the line connecting the points for risk-free rate and SPY.
Put labels with the company names on the graph.

```
#subset to just the stocks and SPY (not XLV, XLI)
Mean.Sd.Broad.Market.Returns<-Mean.Sd.Stock.Returns[c(1:16,19),]
#plot the dots for all the companies (not the indices)
plot(Mean.Sd.Broad.Market.Returns,ylab="Mean",xlab="St. Dev.",pch=19,xlim=c(0,.03), main = "Broad Marke
#plot the dot for the SPY
points(Mean.Sd.Broad.Market.Returns['SPY',1],Mean.Sd.Broad.Market.Returns['SPY',2],col="red",pch=19)
#Plot the Risk Free Return
points(0,Mean.FedFunds,col="green",pch=19)
#Plot the line connecting RFR to SPY
lines(c(0,Mean.Sd.Broad.Market.Returns['SPY',1]),
      c(mean(FedFunds.BD[,2])/100/360,Mean.Sd.Broad.Market.Returns['SPY',2]))
text(Mean.Sd.Broad.Market.Returns,labels=rownames(Mean.Sd.Broad.Market.Returns),cex=.5,pos=2)
```




**Which names would you include in the portfolio?**

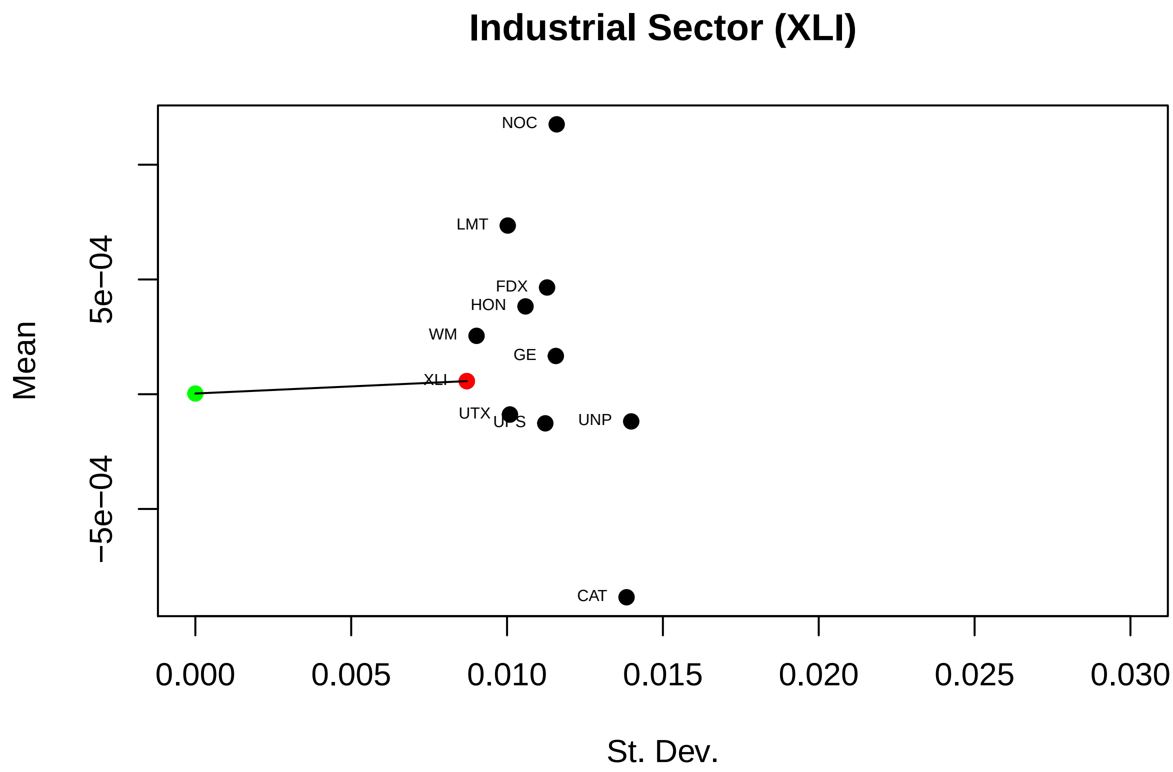
AET, HUM, ABT, MDT, PFE, NOC, LMT, FDX, and HON

**Which sectors do your choices belong to?**

Four are Industrial (NOC, LMT, FDX, HON) and five are in Healthcare (AET, HUM, ABT, MDT, PFE).

### The Industrial Sector Index (XLI)

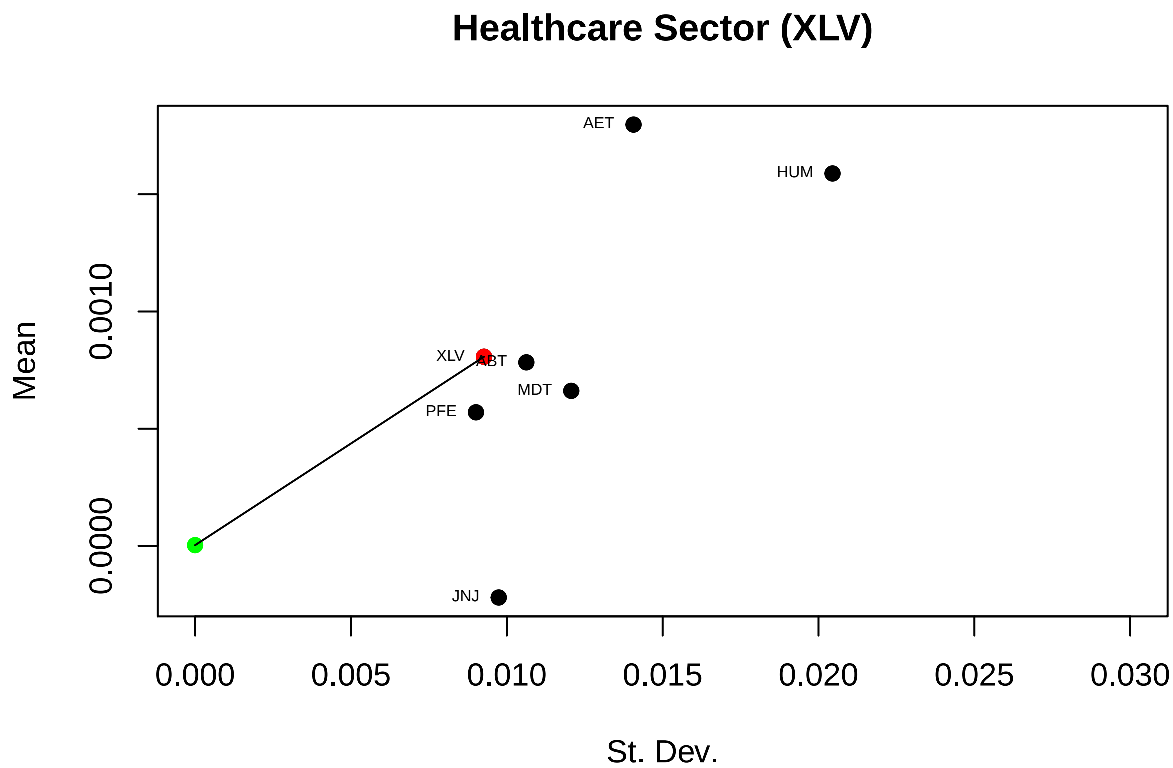
```
#subset to industrial companies and XLI
Mean.Sd.Industrial.Returns<-Mean.Sd.Stock.Returns[c(1:10,18),]
#plot the dots for all the companies (not the indices)
plot(Mean.Sd.Industrial.Returns,ylab="Mean",xlab="St. Dev.",pch=19,xlim=c(0,.03), main = "Industrial Se
#plot the dot for the SPY
points(Mean.Sd.Industrial.Returns['XLI',1],Mean.Sd.Industrial.Returns['XLI',2],col="red",pch=19)
#Plot the Risk Free Return
points(0,Mean.FedFunds,col="green",pch=19)
#Plot the line connecting RFR to SPY
lines(c(0,Mean.Sd.Industrial.Returns['XLI',1]),
       c(mean(FedFunds.BD[,2])/100/360,Mean.Sd.Industrial.Returns['XLI',2]))
text(Mean.Sd.Industrial.Returns,labels=rownames(Mean.Sd.Industrial.Returns),cex=.5,pos=2)
```




We see that several stocks outperform the index (in terms of return/risk tradeoff): GE, WM, etc. as these are above the line.

### The Health Care Sector Index (XLV)

```
#subset to healthcare companies and XLV
Mean.Sd.Healthcare.Returns<-Mean.Sd.Stock.Returns[11:17,]
#plot the dots for all the companies (not the indices)
plot(Mean.Sd.Healthcare.Returns,ylab="Mean",xlab="St. Dev.",pch=19,xlim=c(0,.03), main = "Healthcare Se
#plot the dot for the SPY
points(Mean.Sd.Healthcare.Returns['XLV',1],Mean.Sd.Healthcare.Returns['XLV',2],col="red",pch=19)
#Plot the Risk Free Return
points(0,Mean.FedFunds,col="green",pch=19)
#Plot the line connecting RFR to SPY
lines(c(0,Mean.Sd.Healthcare.Returns['XLV',1]),
       c(mean(FedFunds.BD[,2])/100/360,Mean.Sd.Healthcare.Returns['XLV',2]))
text(Mean.Sd.Healthcare.Returns,labels=rownames(Mean.Sd.Healthcare.Returns),cex=.5,pos=2)
```




We see that only one stock, AET, outperforms the index in terms of return/risk tradeoff.

## CAPM Model

Calculate betas of all the companies on the list to SPY.

**NOTE**: I had to delete 10/13/2014 from the stock data. This was Columbus Day and apparently the stock market was open but the Fed Funds rate was not quoted, or is at least not included in the file. If I do not take it out then there are 252 days in the stock quotes and 251 in the Fed Funds rates.

```
#2014-10-13 is in row 73. Take it out
Stock.Returns<-Stock.Returns[-73,]
```

*First for the Broad Market (SPY)*

```
#get the daily Fed Funds Rate
FedFunds.BD.daily<-FedFunds.BD[,2]/100/360
#get the excess returns
SP500.companies_Excess<-apply(Stock.Returns[,c(1:16,19)],2,function(z) z-FedFunds.BD.daily)
#get the betas. Take out column 17 (SPY). Was 19th now 17th.
SP500.companies.betas<-as.matrix(apply(SP500.companies_Excess[,-17],2,function(z)
+   lm(diff(log(z))~-1+diff(log(SP500.companies_Excess[,'SPY'])))$coefficients))
rownames(SP500.companies.betas)<-rownames(Mean.Sd.Stock.Returns)[-c(17,18,19)]
head(SP500.companies.betas)
```

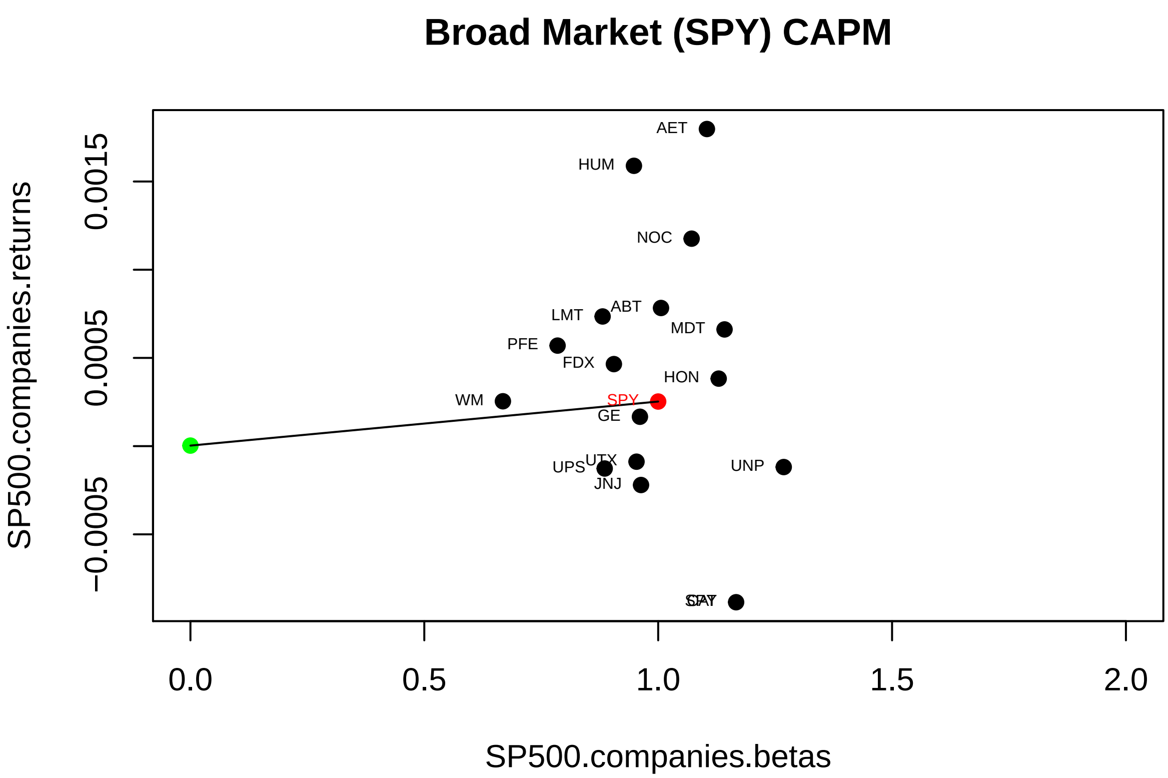
```
##          [,1]
## CAT 1.1666129
```

```
## FDX 0.9052951
## GE  0.9609694
## HON 1.1293767
## LMT 0.8811733
## NOC 1.0714641
```

Note: `Names.List` contains the names (characters), `get(z)` turns the name z into an R variable, like "AMZN" into variable `AMZN`.

Create the CAPM diagram

```
#plot all the companies
plot(SP500.companies.betas, Mean.Sd.Broad.Market.Returns[-c(17,18,19),2], ylab="SP500.companies.returns
#add point for SPY
points(1,Mean.Sd.Broad.Market.Returns['SPY',2],col="red",pch=19)
#add point for Risk Free Rate
points(0,Mean.FedFunds,col="green",pch=19)
#connect points
lines(c(0,1),c(Mean.FedFunds,Mean.Sd.Broad.Market.Returns['SPY',2]))
#add labels for all stocks
text(SP500.companies.betas,Mean.Sd.Broad.Market.Returns[1:16,2],labels=rownames(Mean.Sd.Broad.Market.Re
#add label for SPY
text(1,Mean.Sd.Broad.Market.Returns[17,2],labels="SPY",cex=.5,col="red",pos=2)
```



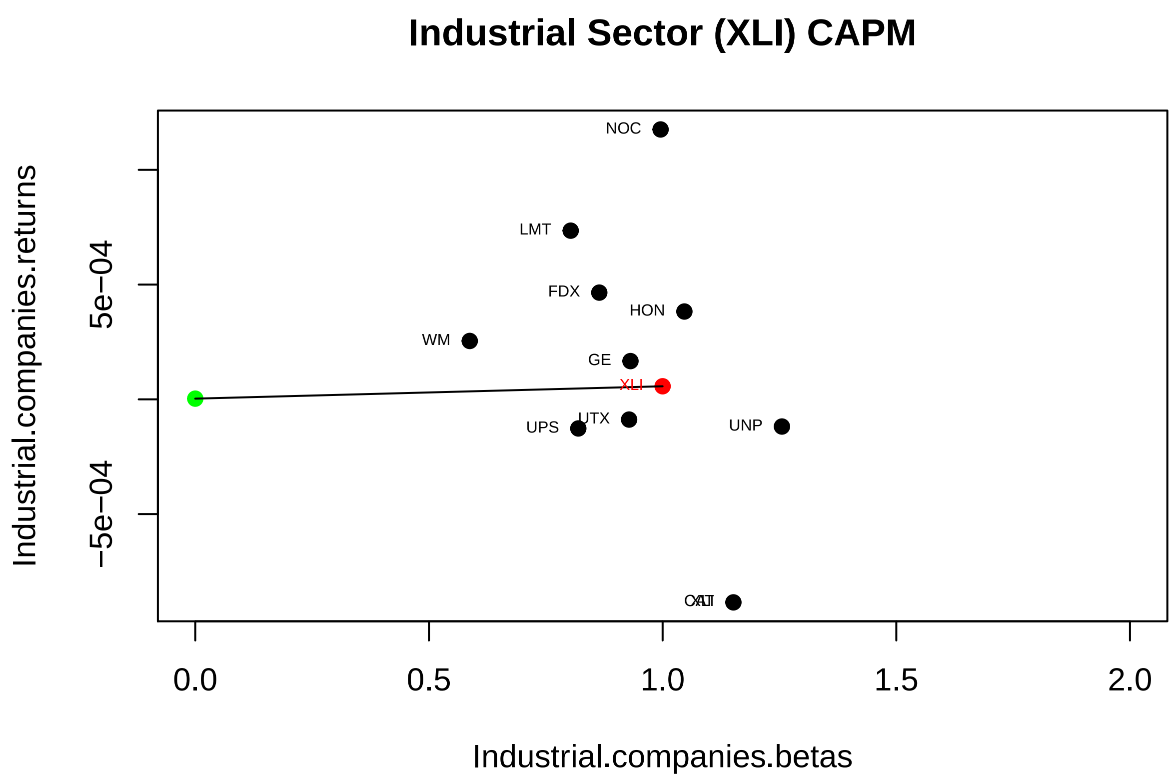

We would add stocks that are above the line because their return given their sensitivity to the market (Beta) is above what we would expect.

*Now for the Industrial Sector (XLI)*

```r
#just for industrial companies (1-10) and XLI
Industrial.companies_Excess<-apply(Stock.Returns[,c(1:10,18)],2,function(z) z-FedFunds.BD.daily)
#get the betas. Take out column 11 (XLI).
Industrial.companies.betas<-as.matrix(apply(Industrial.companies_Excess[,-11],2,function(z)
+   lm(diff(log(z))~-1+diff(log(Industrial.companies_Excess[,'XLI'])))$coefficients))
rownames(Industrial.companies.betas)<-rownames(Mean.Sd.Stock.Returns)[c(1:10)]
```

The Industrial Sector CAPM diagram

```r
#plot all the companies, take out XLI
plot(Industrial.companies.betas, Mean.Sd.Industrial.Returns[-c(11),2], ylab="Industrial.companies.retur
#add point for XLI
points(1,Mean.Sd.Industrial.Returns['XLI',2],col="red",pch=19)
#add point for Risk Free Rate
points(0,Mean.FedFunds,col="green",pch=19)
#connect points
lines(c(0,1),c(Mean.FedFunds,Mean.Sd.Industrial.Returns['XLI',2]))
#add labels for all stocks
text(Industrial.companies.betas,Mean.Sd.Industrial.Returns[1:10,2],labels=rownames(Mean.Sd.Industrial.R
#add label for XLI
text(1,Mean.Sd.Industrial.Returns[11,2],labels="XLI",cex=.5,col="red",pos=2)
```



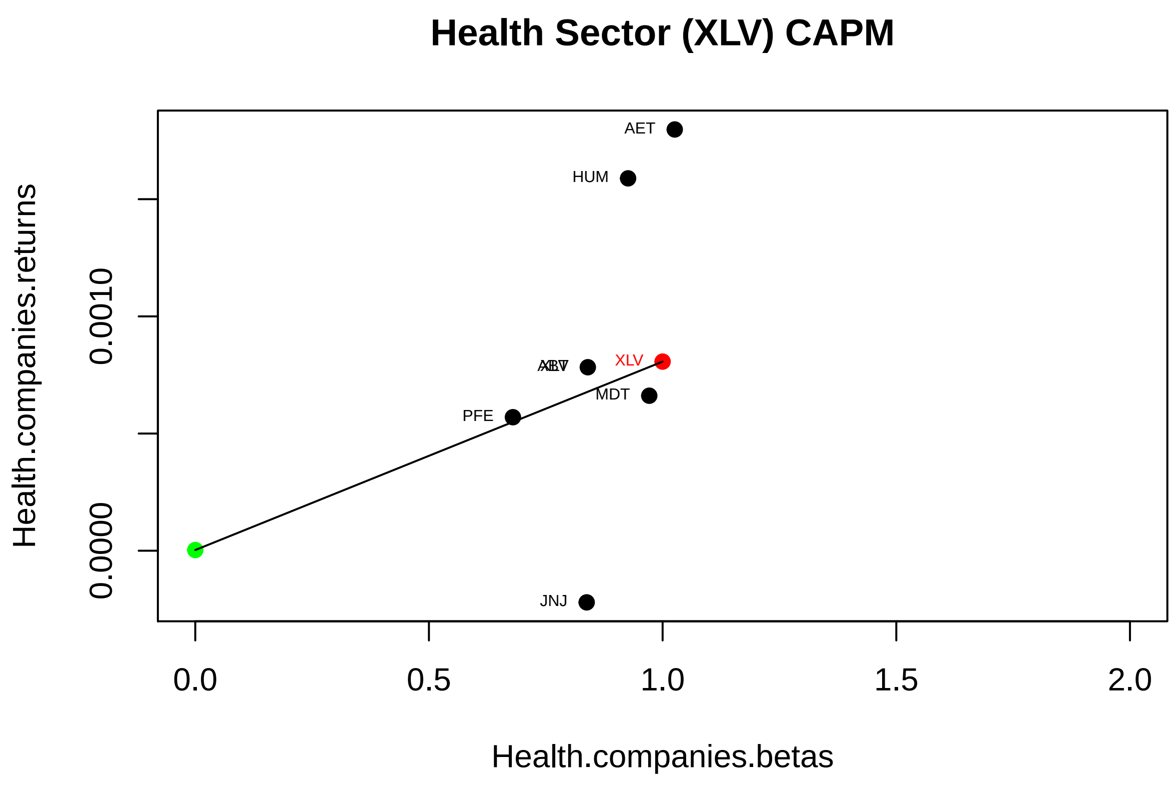

We would add stocks that are above the line because their return given their sensitivity to the Sector (Beta) is above what we would expect, i.e. add WM not UPS.

*And the Healthcare Sector (XLV)*

```
#just for industrial companies (11-16) and XLV
Health.companies_Excess<-apply(Stock.Returns[,c(11:17)],2,function(z) z-FedFunds.BD.daily)
#get the betas. Take out column 7 (XLV).
Health.companies.betas<-as.matrix(apply(Health.companies_Excess[,-7],2,function(z)
+   lm(diff(log(z))~-1+diff(log(Health.companies_Excess[,'XLV'])))$coefficients))
rownames(Health.companies.betas)<-rownames(Mean.Sd.Stock.Returns)[c(11:16)]
```

The Healthcare Sector CAPM diagram

```
#plot all the companies, take out XLV
plot(Health.companies.betas, Mean.Sd.Healthcare.Returns[-7,2], ylab="Health.companies.returns",pch=19,xl
#add point for XLV
points(1,Mean.Sd.Healthcare.Returns['XLV',2],col="red",pch=19)
#add point for Risk Free Rate
points(0,Mean.FedFunds,col="green",pch=19)
#connect points
lines(c(0,1),c(Mean.FedFunds,Mean.Sd.Healthcare.Returns['XLV',2]))
#add labels for all stocks
text(Health.companies.betas,Mean.Sd.Healthcare.Returns[1:6,2],labels=rownames(Mean.Sd.Healthcare.Return
#add label for XLI
text(1,Mean.Sd.Healthcare.Returns[7,2],labels="XLV",cex=.5,col="red",pos=2)
```




We would add stocks that are above the line because their return given their sensitivity to the Sector (Beta) is above what we would expect, i.e. add HUM not JNJ.

## APT model

### Selecting Factors

Start the process of factors selection by doing PCA on the stock portfolio.

Put the raw stock prices into daily log returns.

```
Stock.daily.log.Returns<-diff(log(cbind(Stock.Returns$CAT, Stock.Returns$FDX, Stock.Returns$GE, Stock.Re
```

```
#do PCA on the DAILY LOG RETURNS of the stocks (not the raw numbers)
Stock.Returns.PCA<-princomp(Stock.daily.log.Returns)
#compute the cumulative explantory power as you add components
cumsum(Stock.Returns.PCA$sdev/sum(Stock.Returns.PCA$sdev))
```

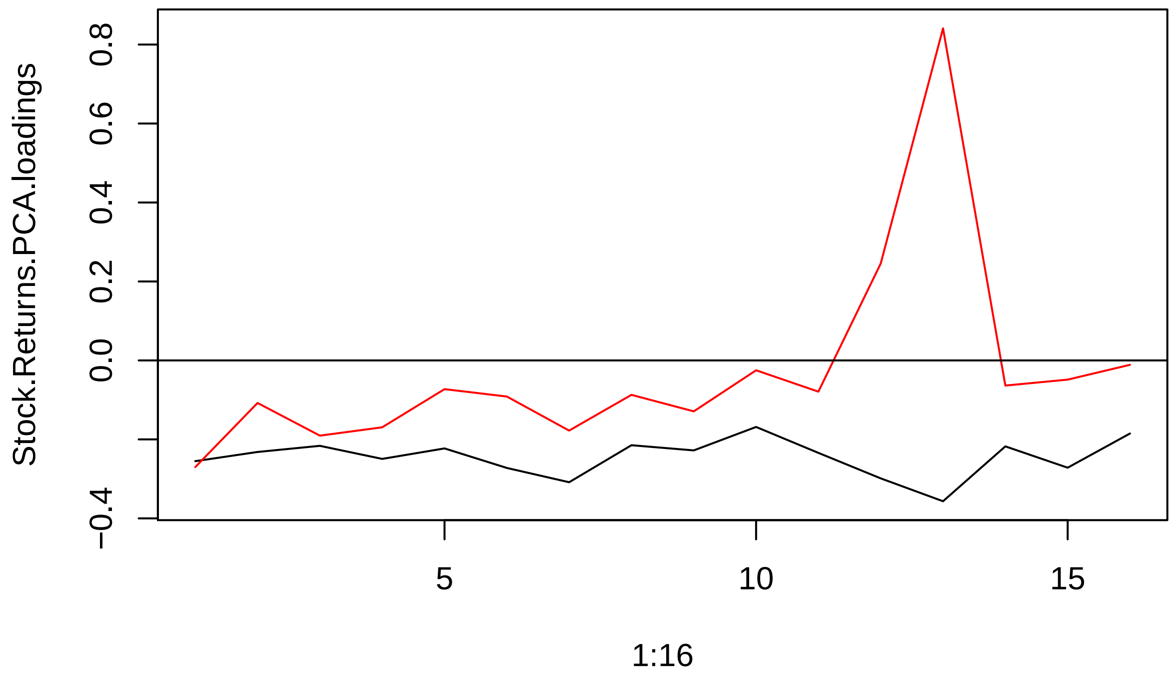
```
##    Comp.1    Comp.2    Comp.3    Comp.4    Comp.5    Comp.6    Comp.7
## 0.2063715 0.3264408 0.4045493 0.4709696 0.5335554 0.5910583 0.6462575
##    Comp.8    Comp.9   Comp.10   Comp.11   Comp.12   Comp.13   Comp.14
## 0.6953620 0.7435447 0.7854264 0.8264275 0.8650322 0.9033991 0.9381446
##   Comp.15   Comp.16
## 0.9704374 1.0000000
```

In order to fit substantially all the stock returns we would need at least eight factors. The first two account for only 33% of the variability. However, they might not be interpretable the further down the list we go.

```
#take first two factors
Stock.Returns.PCA.factors<-as.matrix(Stock.Returns.PCA$scores[,1:2])
#take first two loadings
Stock.Returns.PCA.loadings<-as.matrix(Stock.Returns.PCA$loadings[,1:2])
#get expected return of each asset (mean of original columns)
Stock.Returns.PCA.zero.loading<-Stock.Returns.PCA$center
```

How are the main factors related to our candidates?

```
matplot(1:16,Stock.Returns.PCA.loadings,type="l",lty=1)
abline(h=0)
```

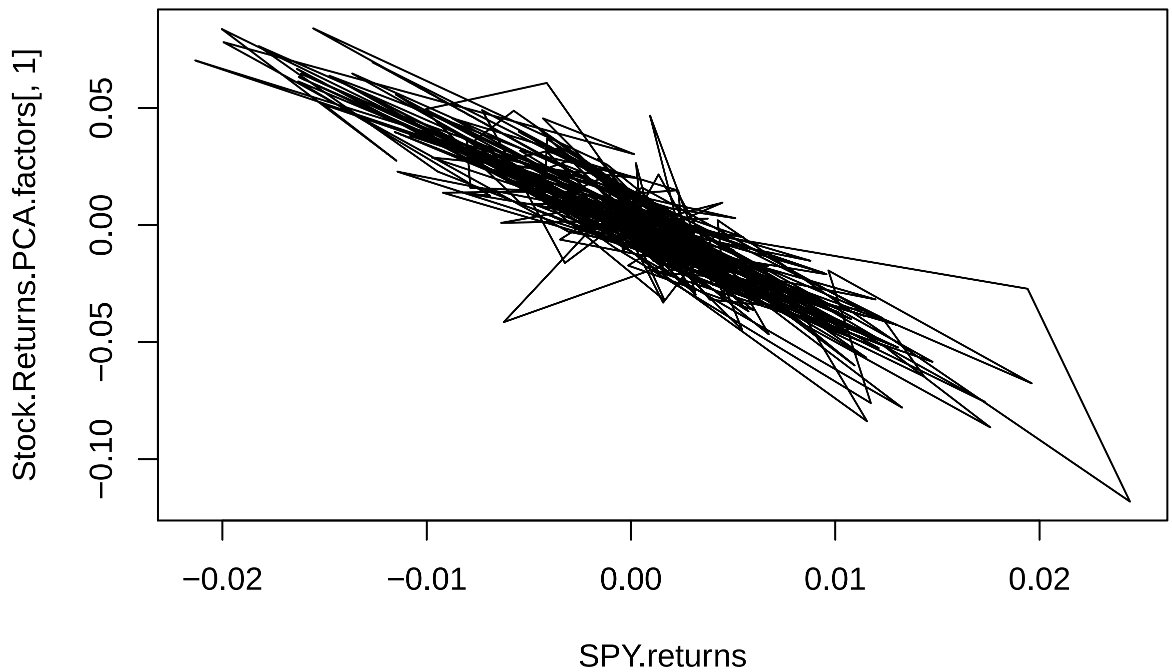



We see the loadings are mostly negative except the second loadings for some companies.

Since all of the stocks are traded as part of the broad stock market we suspect that the first factor may be related to SPY, i.e. that the first factor is mostly negatively correlated with SPY.

*The Correlation of the First Factor with SPY Returns*

```
#get the daily log returns of the SPY, delete first row because it is NA
#this will have length = 250 because you took out a row
SPY.returns<-as.matrix(diff(log(Stock.Returns[, 19]))[-1])
#plot those daily log returns of SPY against the first factor
#this will have length = 251
#now that you have vectors of same length you can do the next line
plot(SPY.returns,Stock.Returns.PCA.factors[,1],type="l")
```




*The Correlation of the Second Factor with SPY Returns*

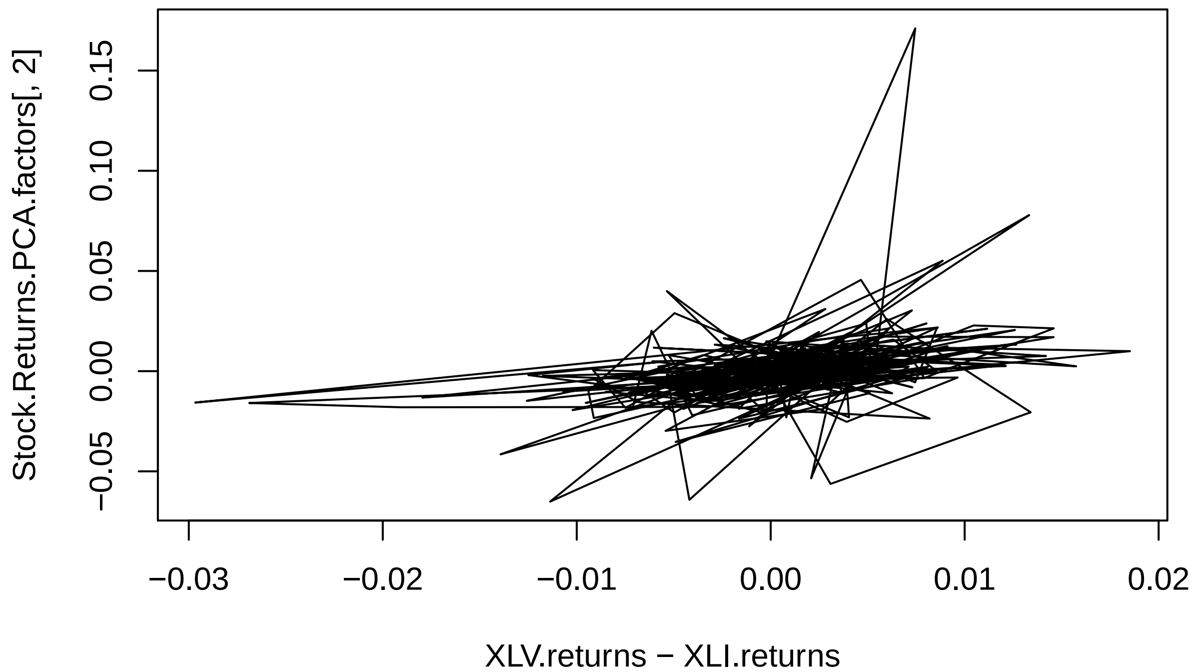
The second loading has opposite signs for some of the healthcare stocks. For reference the Healthcare stocks are stocks 11 through 16 in the loadings graph. *Could this mean the second factor should be interpreted as the spread between XLV and XLI?.*

First we need the daily log returns of both the XLI and XLV.

```
XLI.returns<-as.matrix(diff(log(Stock.Returns[, 18]))[-1])
XLV.returns<-as.matrix(diff(log(Stock.Returns[, 17]))[-1])
```

Then plot the difference (spread) between the sector indices against the second loading.

```
plot(XLV.returns-XLI.returns,Stock.Returns.PCA.factors[,2],type="l")
```




This relationship is not as strong as interpretation of the first factor.
Note that the slope may suddenly change in case of large moves.

*Fit a Linear Model Explaining the Interpretation of Both Factors*

Fit linear models explaining the interpretation of both factors.

```
lm.fit.factor1<-lm(Stock.Returns.PCA.factors[,1]~SPY.returns)
lm.fit.factor2<-lm(Stock.Returns.PCA.factors[,2]~I(XLV.returns-XLI.returns))
summary(lm.fit.factor1)
```

```
##
## Call:
## lm(formula = Stock.Returns.PCA.factors[, 1] ~ SPY.returns)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.067176 -0.008125 -0.000607  0.007879  0.049401
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  0.0010055  0.0008812   1.141    0.255
## SPY.returns -3.9610585  0.1168110 -33.910   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01392 on 248 degrees of freedom
## Multiple R-squared:  0.8226, Adjusted R-squared:  0.8219
```

```
## F-statistic:  1150 on 1 and 248 DF,  p-value: < 2.2e-16
```

```r
summary(lm.fit.factor2)
```

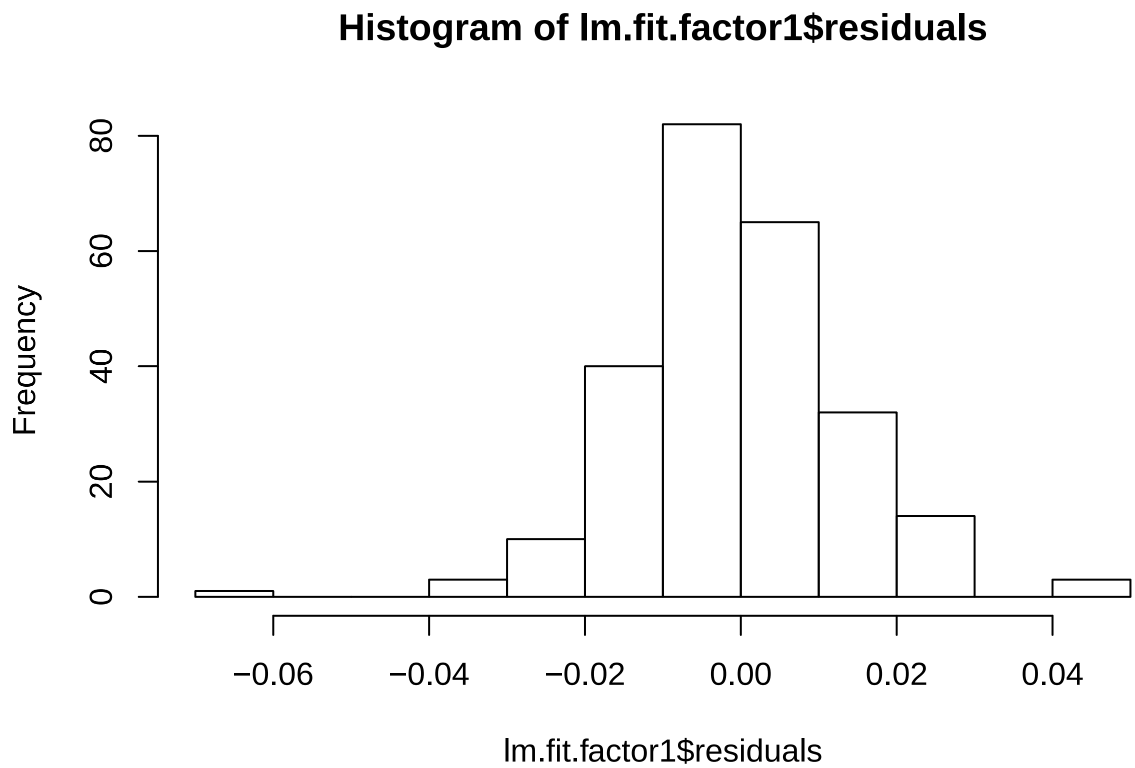
```
##
## Call:
## lm(formula = Stock.Returns.PCA.factors[, 2] ~ I(XLV.returns -
##     XLI.returns))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.058969 -0.007639  0.000216  0.006171  0.163313
##
## Coefficients:
##                                Estimate Std. Error t value Pr(>|t|)
## (Intercept)                  -0.0008716  0.0011236  -0.776    0.439
## I(XLV.returns - XLI.returns)  1.1575516  0.1697586   6.819 6.96e-11 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01765 on 248 degrees of freedom
## Multiple R-squared:  0.1579, Adjusted R-squared:  0.1545
## F-statistic:  46.5 on 1 and 248 DF,  p-value: 6.957e-11
```

In both fits intercepts are practically insignificant, but both slopes are significant. The first factor fit has pretty good $R^2$ (0.82), the second is not strong at all (0.15).

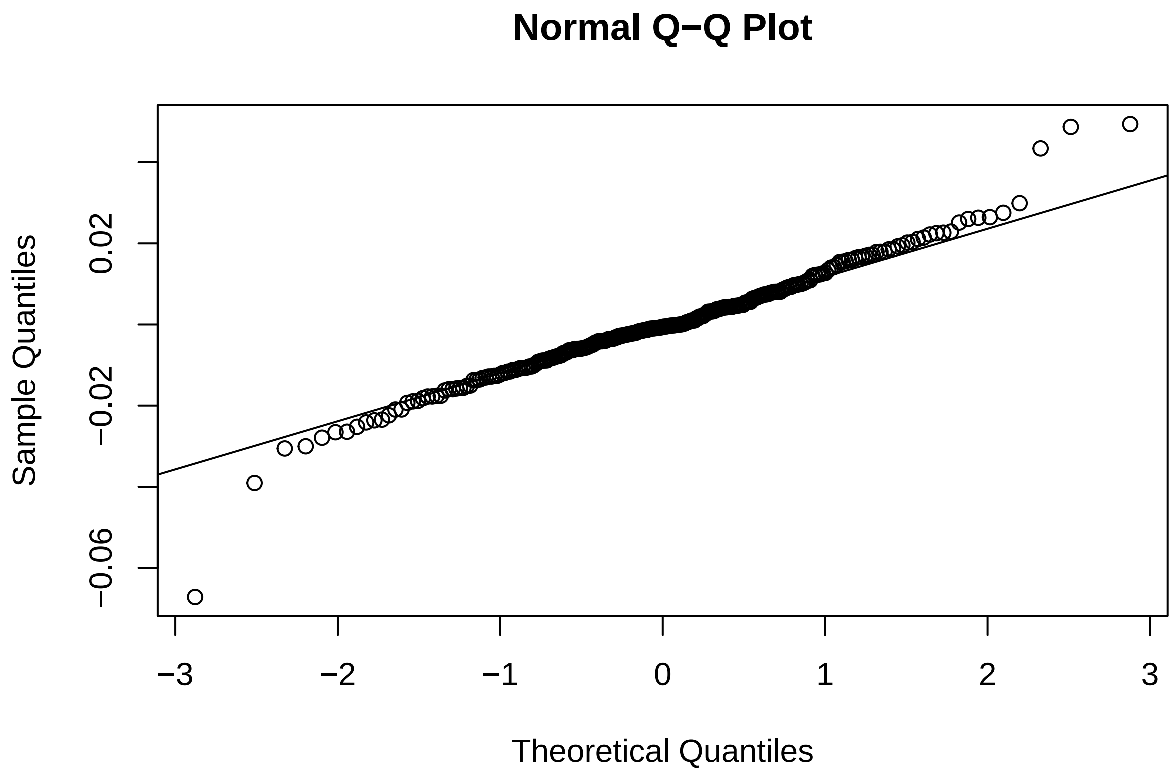
Check the residuals of both fits of factors.

For factor 1 fit:

```r
hist(lm.fit.factor1$residuals)
```

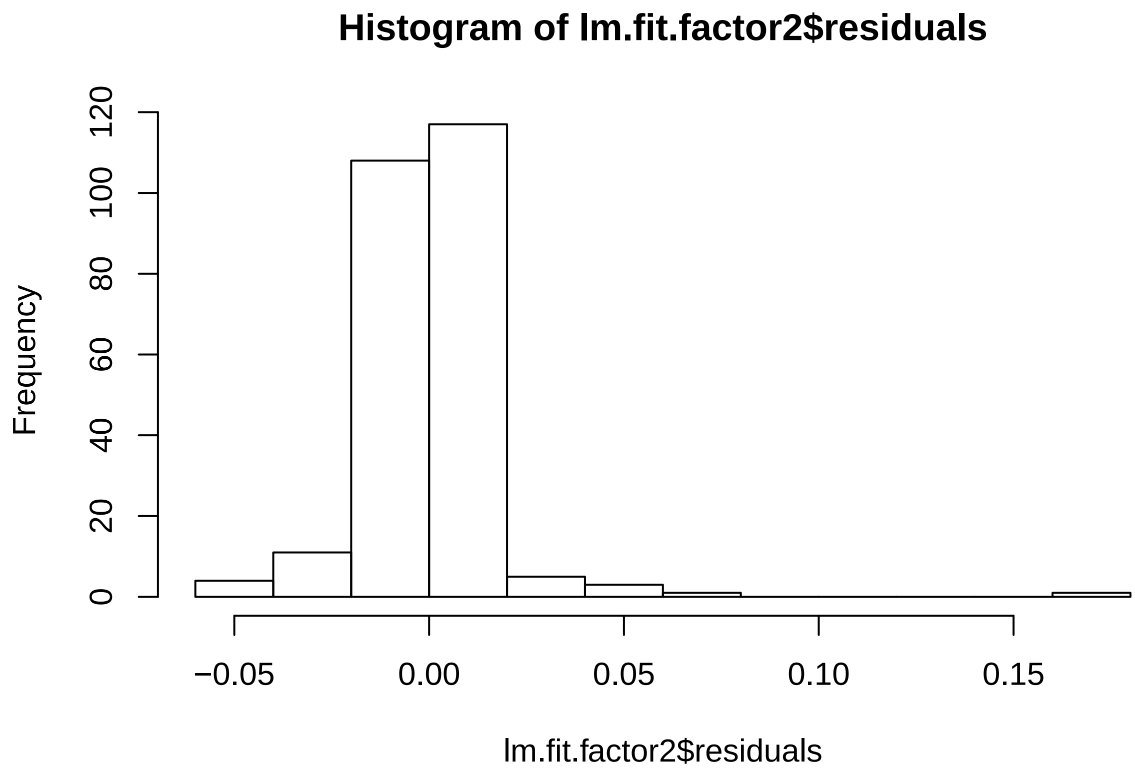



```
qqnorm(lm.fit.factor1$residuals)
qqline(lm.fit.factor1$residuals)
```

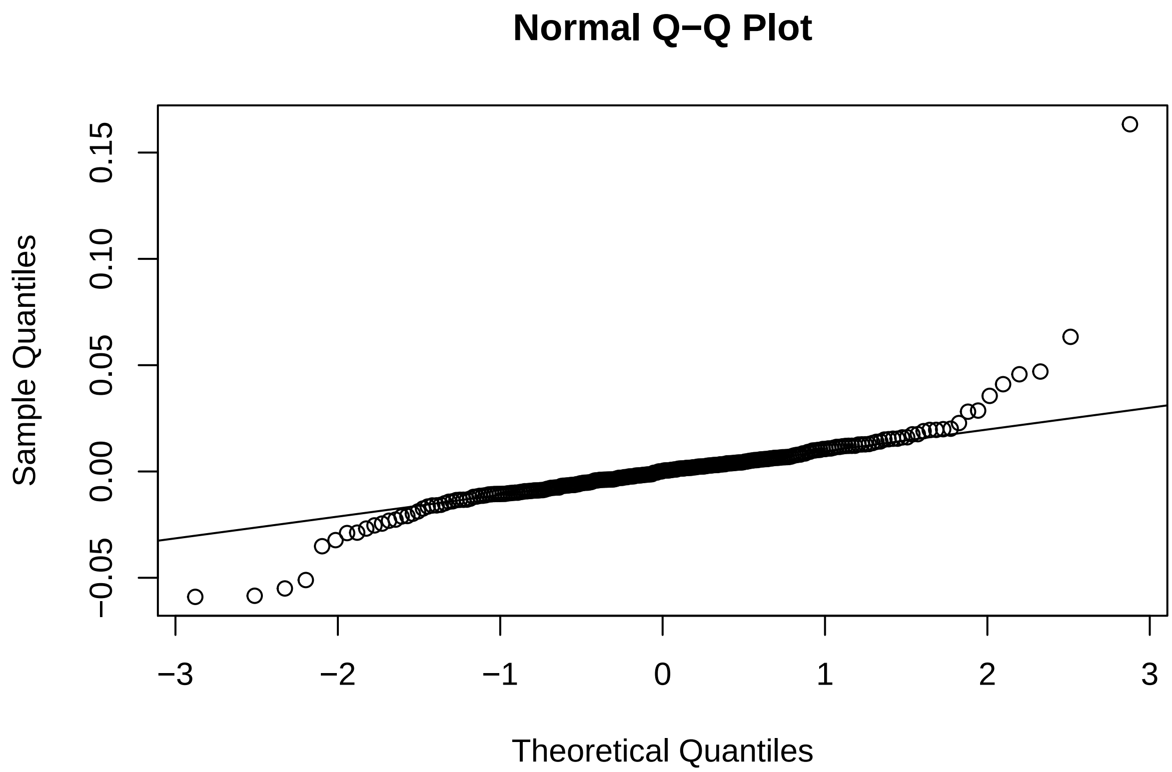



For factor 2 fit:

```
hist(lm.fit.factor2$residuals)
```




```
qqnorm(lm.fit.factor2$residuals)
qqline(lm.fit.factor2$residuals)
```




**Estimation of Betas**

Check that the betas are the same as the PCA factor loadings.

```
#check against daily log returns
Stock.portfolio.betas<-apply(Stock.daily.log.Returns,2,
                             function(z) lm(z~Stock.Returns.PCA.factors[,1]+
                                              Stock.Returns.PCA.factors[,2])$coefficients)
rownames(Stock.portfolio.betas)<-c("Alpha","Factor.1","Factor.2")
#transpose and create as dataframe
Stock.portfolio.betas<-as.data.frame(t(Stock.portfolio.betas))
Stock.portfolio.betas
```

```
##             Alpha   Factor.1    Factor.2
## CAT -8.882500e-04 -0.2551346 -0.27014222
## FDX  4.668846e-04 -0.2318363 -0.10771248
## GE   1.672784e-04 -0.2163137 -0.19050372
## HON  3.841700e-04 -0.2494423 -0.16939292
## LMT  7.379039e-04 -0.2229028 -0.07271017
## NOC  1.180626e-03 -0.2723418 -0.09143880
## UNP -1.189686e-04 -0.3084937 -0.17766800
## UPS -1.273397e-04 -0.2147760 -0.08719503
## UTX -8.856021e-05 -0.2278473 -0.12901112
## WM   2.553485e-04 -0.1685109 -0.02498278
## ABT  7.861161e-04 -0.2341615 -0.07905403
## AET  1.804602e-03 -0.2986871  0.24535107
```

```
## HUM  1.595251e-03 -0.3566456  0.84104959
## JNJ -2.211829e-04 -0.2176526 -0.06367868
## MDT  6.640890e-04 -0.2716818 -0.04858969
## PFE  5.720425e-04 -0.1850857 -0.01112741
```

```r
cbind(zeroLoading=Stock.Returns.PCA.zero.loading,Stock.Returns.PCA.loadings)
```

```
##       zeroLoading     Comp.1      Comp.2
## CAT -8.882500e-04 -0.2551346 -0.27014222
## FDX  4.668846e-04 -0.2318363 -0.10771248
## GE   1.672784e-04 -0.2163137 -0.19050372
## HON  3.841700e-04 -0.2494423 -0.16939292
## LMT  7.379039e-04 -0.2229028 -0.07271017
## NOC  1.180626e-03 -0.2723418 -0.09143880
## UNP -1.189686e-04 -0.3084937 -0.17766800
## UPS -1.273397e-04 -0.2147760 -0.08719503
## UTX -8.856021e-05 -0.2278473 -0.12901112
## WM   2.553485e-04 -0.1685109 -0.02498278
## ABT  7.861161e-04 -0.2341615 -0.07905403
## AET  1.804602e-03 -0.2986871  0.24535107
## HUM  1.595251e-03 -0.3566456  0.84104959
## JNJ -2.211829e-04 -0.2176526 -0.06367868
## MDT  6.640890e-04 -0.2716818 -0.04858969
## PFE  5.720425e-04 -0.1850857 -0.01112741
```

We see that the loadings are equal to our Betas.

### Estimation of Market Price of Risk

In this step we estimate linear model with $\alpha - R_f$ as output and the matrix of $\beta$ as inputs. Here $R_f$ is the average risk-free Fed Funds rate.

```r
Market.Prices.of.risk.fit<-lm(I(Alpha-Mean.FedFunds)~.-1,data=Stock.portfolio.betas)
summary(Market.Prices.of.risk.fit)
```

```
##
## Call:
## lm(formula = I(Alpha - Mean.FedFunds) ~ . - 1, data = Stock.portfolio.betas)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0009452 -0.0004160  0.0001124  0.0002702  0.0008020
##
## Coefficients:
##            Estimate Std. Error t value Pr(>|t|)
## Factor.1 -0.0019768  0.0005161  -3.830  0.00184 **
## Factor.2  0.0016675  0.0005161   3.231  0.00604 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.0005161 on 14 degrees of freedom
## Multiple R-squared:  0.642,  Adjusted R-squared:  0.5908
## F-statistic: 12.55 on 2 and 14 DF,  p-value: 0.0007538
```

These estimates are our $\lambda$'s

The $R^2$ and adjusted $R^2$ are somewhat promising. Interestingly the slope parameters are significant even at the 1% level.

There are only 16 stocks in this data. Not enough to make a distribution of residuals especially meaningful. However, we will continue with the process of APT price evaluation as the theory prescribes.

```
Market.Prices.of.risk<-c(Mean.FedFunds,Market.Prices.of.risk.fit$coefficients)
Market.Prices.of.risk
```

```
##                    Factor.1      Factor.2
##  3.004648e-06 -1.976818e-03  1.667465e-03
```

Now we need to look at the residuals of the equilibrium model to assess the prices of each stock relative to the prediction.

The equation of the equilibrium model is:

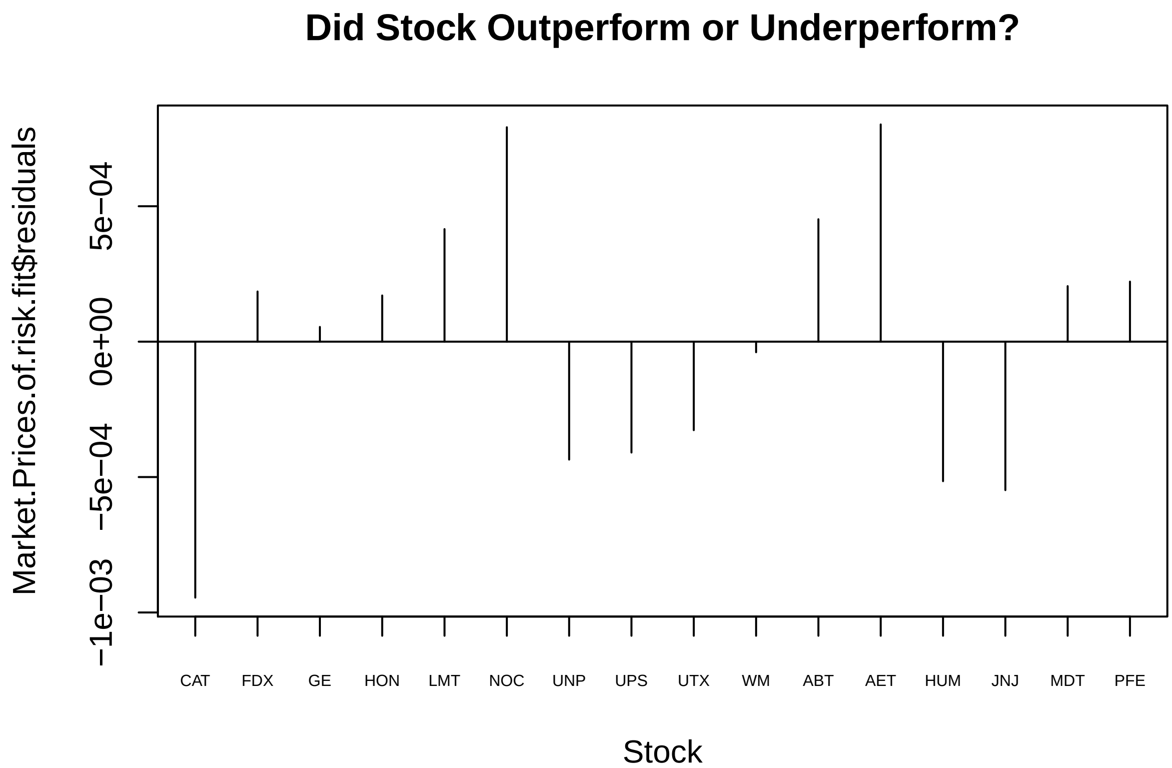
$$\epsilon_i = E(R_i) - (R_f + \lambda_1\beta_1 + \lambda_2\beta_2)$$

where $\lambda_1$ and $\lambda_2$ are the slopes estimated in getting the `Market.Prices.of.risk` variable.

```
Market.Prices.of.risk.fit$residuals
```

```
##           CAT           FDX            GE           HON           LMT
## -9.451568e-04  1.851885e-04  5.431918e-05  1.705200e-04  4.155026e-04
##           NOC           UNP           UPS           UTX            WM
##  7.917223e-04 -4.355541e-04 -4.095228e-04 -3.268560e-04 -3.911365e-05
##           ABT           AET           HUM           JNJ           MDT
##  4.520366e-04  8.020330e-04 -5.151969e-04 -5.482651e-04  2.050405e-04
##           PFE
##  2.217116e-04
```

```
plot(Market.Prices.of.risk.fit$residuals,type="h",xaxt="n",xlab="Stock", main = "Did Stock Outperform o
abline(h=0)
axis(1, at=1:16, cex.axis=0.5, labels=c("CAT","FDX","GE","HON","LMT","NOC", "UNP", "UPS","UTX","WM","AB
```




When residual is positive it means that the mean return of the stock over the sample period $E(R_i)$ is greater than predicted value $R_f + \lambda_1\beta_1 + \lambda_2\beta_2$.

However this only tells us whether the stock *outperformed* the model not whether the stock *will outperform* the model going forward.