Stat 107: Quantitative Methods for Economics Take Home Exam

Xiner Zhou

10/20/2016

Note: My answer to Question 21is 0, my version doesn't allow to choose the right answer, I've sent email to Prof. Parzen to confirm this issue.

1) A salesperson in a large bicycle shop is paid a bonus if he sells more than 4 bicycles a day. The probability of selling more than 4 bicycles a day is only .40. If the number of bicycles sold is greater than 4, the distribution of sales is as shown in the table below. The shop has four different models of bicycles. The amount of the bonus paid out varies by type. The bonus for model A is $10; 40% of the bicycles sold are of this type. Model B accounts for 35% of the sales and pays a bonus of $15. Model C has a bonus rating of $20 and makes up 20% of the sales. Finally, model D pays a bonus of $25 for each sale but accounts for only 5% of the sales. Develop a simulation model to calculate the bonus a salesperson can expect in a day. Also report a 95% confidence interval for the mean bonus amount.

No. of Bicycles Sold	Probability
5	.35
6	.45
7	.15
8	.05

.

```
set.seed(02138)
# set the number of simulation
n.sim<-1000000
# initialize place-holder for simulated bonus
bonus<-rep(0,n.sim)

for(i in 1:n.sim){
  # step 1: sell more than 4 bicycles ?
  sell.more.4<-rbinom(n=1,size=1,prob = 0.4)

  if(sell.more.4==1){
  # step 2: if sell more than 4, decide how many sold
    n.sell<-sample(x=c(5,6,7,8), size=1, replace = FALSE, prob
=c(0.35,0.45,0.15,0.05))

  # step 3: for each sold bicycle, decide which model it is and its
associated bonus
    bonus.each<-sample(x=c(10,15,20,25), size=n.sell, replace = TRUE, prob =
c(0.4,0.35,0.2,0.05))

    # step 4: add up bonus
    bonus[i]<-sum(bonus.each)
```

```
  }
}

cat("The expected bonus in a day is", mean(bonus), '\n')

## The expected bonus in a day is 34.17213

cat("95% confident interval for the mean bonus is", "(", quantile(bonus,
probs = c(0.025)),",",quantile(bonus, probs = c(0.975)),")", '\n')

## 95% confident interval for the mean bonus is ( 0 , 110 )
```

---

2) A fund manager is considered three different investments. The first is a stock fund, the second is a bond fund, and the third is a money-market fund that yields a rate of .05. The yearly statistics of all the funds are as follows:

	Expected Return	Standard Deviation
Stock Fund	0.24	0.32
Bond Fund	0.15	0.21
Money-Market	0.05	0

The correlation between the stock and bond fund is 0.10

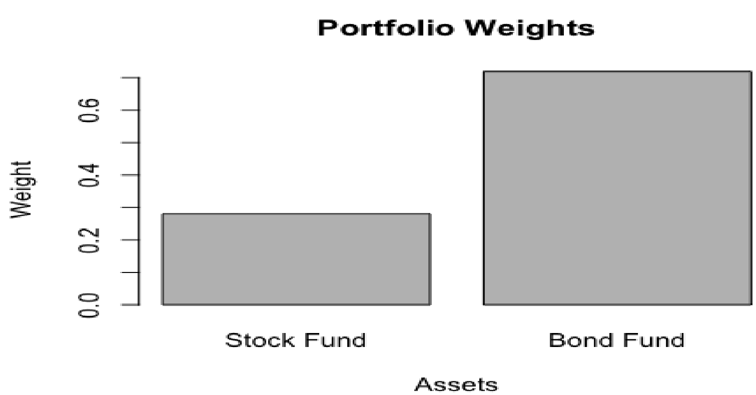
a) Find the risk and return of the minimum variance portfolio consisting of the stock and bond funds.
b) Create a plot of the efficient frontier, allowing short sales.
c) Find the tangent portfolio consisting of the stock and bond funds; produce the weights for the tangent portfolio, as well as the risk and return of the tangent portfolio.
d) Suppose you with to construct a portfolio with an expected return of .28 but all you can do is go long or short the two risky funds. What are the appropriate portfolio weights and the resulting portfolio standard deviation? What reduction in standard deviation could you attain if instead you used the money market fund and the tangent portfolio to construct a portfolio that returned .28?

.

(a)

```
# Loading required libraries
library(quantmod)

# include source code
source("/Volumes/NO NAME/Course Work at Harvard/Introduction to Financial
Statisitcs/Eric Zivot.R")

# expected return vector
er<-c(0.24,0.15)

# covariance matrix
cov.mat<-matrix(c(0.32^2, 0.1*0.32*0.21,
                  0.1*0.32*0.21, 0.21^2),nrow=2,ncol=2,byrow = T)
names(er)<-c("Stock Fund","Bond Fund")
colnames(cov.mat)<-c("Stock Fund","Bond Fund")
rownames(cov.mat)<-c("Stock Fund","Bond Fund")
```

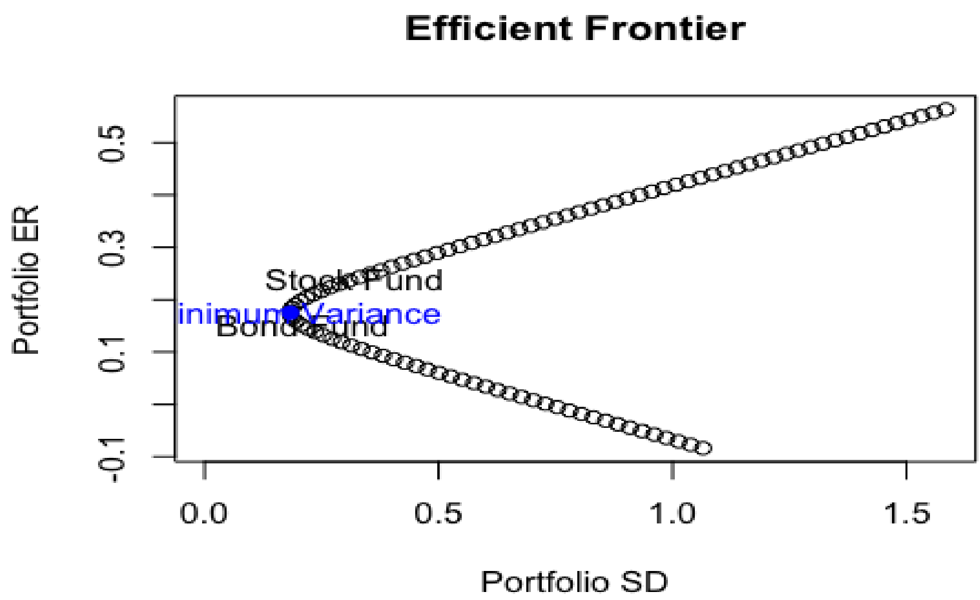
```r
#############################################
## Global Minimum Variance Portfolio
############################################
gmin.port<-globalMin.portfolio(er,cov.mat)
print(gmin.port)

## Call:
## globalMin.portfolio(er = er, cov.mat = cov.mat)
##
## Portfolio expected return:     0.1752833
## Portfolio standard deviation:  0.1833003
## Portfolio weights:
## Stock Fund  Bond Fund
##     0.2809     0.7191

plot(gmin.port)
```



The Global Minimum Variance porfolio puts 28.09% wealth in Stock Fund and 71.91% in Bond Fund, and it has expected return of 17.53% and risk(standard deviation) of 18.33%.

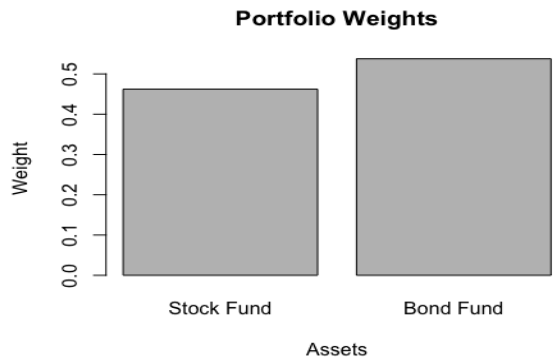
(b)

```r
############################################
## Efficient Frontier Curve
############################################
ef<-efficient.frontier(er, cov.mat, nport=100, alpha.min=-5, alpha.max=5)
plot(ef, plot.assets=T)

# More interestingly, add a point for the Global Minimum Variance Portfolio
points(gmin.port$sd, gmin.port$er, col="blue", pch=21, bg="blue")
text(gmin.port$sd+0.01, gmin.port$er, "Minimum Variance", col="blue")
```



(c)

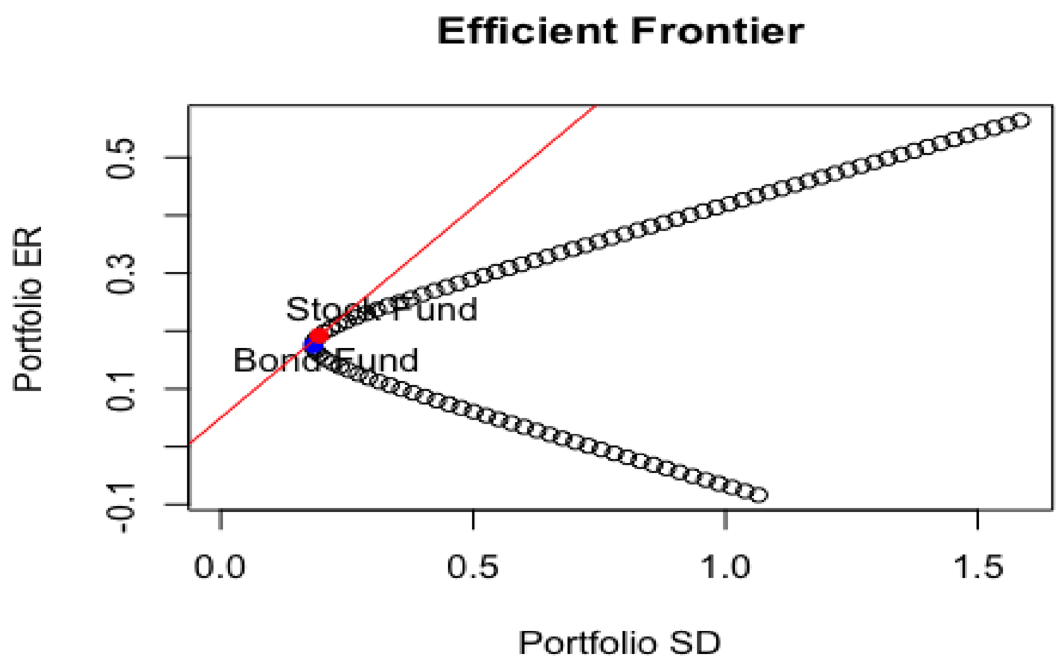
```
#############################################
## Tangency Portfolio and Capital Market Line
#############################################

# Compute tangency portfolio
rk.free<-0.05 # Money Market
tan.port<-tangency.portfolio(er, cov.mat, rk.free)
print(tan.port)

## Call:
## tangency.portfolio(er = er, cov.mat = cov.mat, risk.free = rk.free)
##
## Portfolio expected return:     0.191609
## Portfolio standard deviation:  0.1948776
## Portfolio weights:
## Stock Fund  Bond Fund
##     0.4623     0.5377

plot(tan.port)
```



The tangent portfolio puts 46.23% wealth in Stock Fund and 53.77% in Bond Fund, and it has an expected return of 19.16% and risk(standard deviation) of 19.49%.

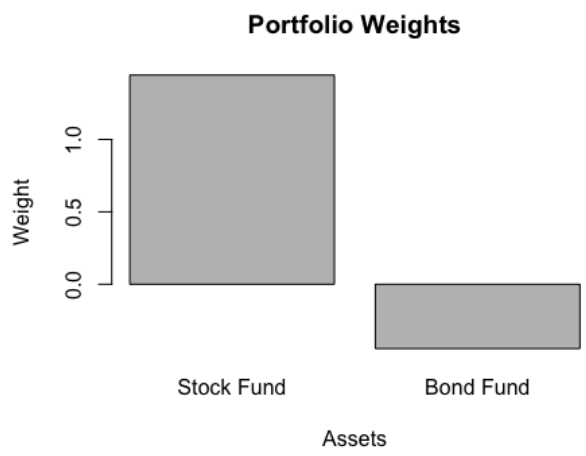
Let's visualize the Efficient Frontier with GMV and Tangent Portfolio.

```
# compute slope of tangent line (aka capital market line)
sr.tan<-(tan.port$er-rk.free)/tan.port$sd

plot(ef, plot.assets=T)

# Adds points to the plot representing GMV and tangent portfolios
points(gmin.port$sd, gmin.port$er, col="blue", pch=21, bg="blue")
points(tan.port$sd, tan.port$er, col="red", pch=21, bg="red")

# Adds a line to the plot representing the CML
abline(a=0.05, b=sr.tan,col="red")
```



(d)

```
eff.port<-efficient.portfolio(er, cov.mat, target.return = 0.28)
print(eff.port)

## Call:
## efficient.portfolio(er = er, cov.mat = cov.mat, target.return = 0.28)
##
## Portfolio expected return:     0.28
## Portfolio standard deviation:  0.462312
## Portfolio weights:
## Stock Fund  Bond Fund
##     1.4444    -0.4444

plot(eff.port)
```



The weight for Stock Fund is 144.44% and weight for Bond Fund is -44.44%, it means you would long Stock Fund and short Bond fund to ahieve the expected return of 28%, the risk(standard deviation) is 46.23%.

If instead you used the money market fund and the tangent portfolio to construct a portfolio that returns 28%:

```
# weight for tangent portfolio
w.tangent<-(28-5)/(19.16-5)
cat("The weight for tangent portflio is", round(w.tangent,2), "%\n")

## The weight for tangent portflio is 1.62 %

cat("The weight for money market is", round(1-w.tangent,2), "%\n")

## The weight for money market is -0.62 %

# risk/standard deviation
cat("The resulting portfolio standard deviation is",
round(w.tangent*19.49,2), "%\n")

## The resulting portfolio standard deviation is 31.66 %

cat("The reduction in standard devation you could attain is", round(46.23-
w.tangent*19.49,2), "%\n")

## The reduction in standard devation you could attain is 14.57 %
```

The portfolio with an expected return of 28% based on two risky assets has standard deviation 46.23%, the associated weights for Stock Fund and Bond Fund is 144.44% and -44.44%, respectively, if willing to go long and short. However, this risk can be reduced to 31.66%, almost reduced one third (1/3) of risk, if instead you used the money market fund and the tangent portfolio to construct a portfoio that has the same expected return.

---

3) Recall Mike's stupid algorithm from the regression slides that fit a line to data using a two point fitting method (we split the X axis using the median of the x's, then found the average x and average y in each subgroup and fit a line to those two points).

   a) Modify the algorithm so it does the same split, but then uses as the two points the median of x and the median of y in each subgroup.
   b) Use the new algorithm, fit the market model for IBM using three years of data starting from 2013-09-01. How does the resulting Beta compare to the usual one found using least squares?
   c) Use the bootstrap to find a 95% confidence interval for the slope of the line found using the new method.

.

(a)

```
# Mike's moidified algorithm
mikeline<-function(x,y){
  xmed<-median(x)
  xbar1<-median(x[x<=xmed])
  ybar1<-median(y[x<=xmed])
  xbar2<-median(x[x>xmed])
  ybar2<-median(y[x>xmed])

  slope<-(ybar2-ybar1)/(xbar2-xbar1)
  inter<-ybar1-slope*xbar1
  cat("Intercept = ",inter, " Slope = ", slope, "\n")
  #return(slope)
}
```

(b)

```
# get IBM monthly adjusted return
getSymbols("IBM",from="2013-09-01")

## [1] "IBM"

ibm.ret<-monthlyReturn(Ad(IBM))

# get S&P 500 index monthly adjusted return
getSymbols("SPY", from="2013-09-01")

## [1] "SPY"

spy.ret<-monthlyReturn(Ad(SPY))

# Use Mike's algorithm, fit the Market Model
mikeline(spy.ret, ibm.ret)

## Intercept =  -0.005915919  Slope =  0.5156135

# Use Least Square Regression, fit the Market Model
lm.fit<-lm(ibm.ret~spy.ret)
coef(lm.fit)
```

```
##  (Intercept)      spy.ret
## -0.008683103  0.800288160
```

Mike's modified model produces a smaller Beta than the usual least square regression produces, 0.5156 versus 0.8, therefore implies less volatility or systematic risk in comparison to the market as a whole.

(c)

```r
# confident interval via Bootstrap
library(boot)

mikeline<-function(x,y){
  xmed<-median(x)
  xbar1<-median(x[x<=xmed])
  ybar1<-median(y[x<=xmed])
  xbar2<-median(x[x>xmed])
  ybar2<-median(y[x>xmed])

  slope<-(ybar2-ybar1)/(xbar2-xbar1)
  inter<-ybar1-slope*xbar1
  #cat("Intercept = ",inter, " Slope = ", slope, "\n")
  return(slope)
}

myfunc<-function(data,i){
  x<-data[i,1]
  y<-data[i,2]
  fit<-mikeline(x, y)
  return(fit)
}
mydata<-cbind(as.numeric(spy.ret),as.numeric(ibm.ret))
boots.fit<-boot(mydata,myfunc,R=100000)
boot.ci(boots.fit)

## BOOTSTRAP CONFIDENCE INTERVAL CALCULATIONS
## Based on 100000 bootstrap replicates
##
## CALL :
## boot.ci(boot.out = boots.fit)
##
## Intervals :
## Level      Normal              Basic
## 95%   (-0.6016,  1.3868 )   (-0.8569,  1.1806 )
##
## Level     Percentile            BCa
## 95%   (-0.1493,  1.8881 )   (-0.1650,  1.8552 )
## Calculations and Intervals on Original Scale
```

4) For this problem we will use the list of stocks in file dow30.csv (the current Dow 30 stock components). The following steps are the work required. The deliverable is parts (c) through (e). We are going to use monthly data from the past three years (starting from 2013-09-01 to 2016-09-01) for all the following calculations. Read in the stock names using the command

```
mystocks=read.csv("http://people.fas.harvard.edu/~mparzen/stat107/dow30.c
sv",header=FALSE,colClasses="character")
```

Note mystocks[i,1] would be the name of the i'th stock.

a) Compute Beta for each stock in the data file, using SPY as the market proxy and adjusted monthly closing prices for all stocks. Use monthly data from 2013-09-01 to 2016-09-01.
b) Compute the standard deviation of monthly returns for each stock in the data file.
c) Produce a scatter plot of beta on the x-axis and standard deviation of monthly returns on the y-axis.
d) Comment on the plot-does there appear to be a good relationship between the x and y variables? Explain.
e) Produce a regression line equation regression for the data in part (c). Are the slope and intercept statistically significant? Interpret what this model implies.
f) In your sample, do large estimates of β correspond with higher R^2 values? Would you expect this always to be the case? Why or why not?

.

(a)

```
# Download stock names
mystocks<-
read.csv("http://people.fas.harvard.edu/~mparzen/stat107/dow30.csv",header =
FALSE, colClasses = "character")

# how many stocks
n.stock<-nrow(mystocks)

# get SPY as market proxy
getSymbols("SPY",from="2013-09-01",to="2016-09-01")

## [1] "SPY"

spy.ret<-monthlyReturn(Ad(SPY))

# Initialize a placeholder for Beta's
beta<-rep(NA,n.stock)
names(beta)<-mystocks[,1]
# Initialize a placeholder for R-squared
R2<-rep(NA,n.stock)
names(R2)<-mystocks[,1]

# Compute Beta and R-squared for each stock
for(i in 1:n.stock){
  ticker<-mystocks[i,1]
  stockdata<-getSymbols(ticker,from="2013-09-01",to="2016-09-01",auto.assign
= FALSE)
  y.ret<-monthlyReturn(Ad(stockdata))
  beta[i]<-coef(lm(y.ret~spy.ret))[2]
  R2[i]<-summary(lm(y.ret~spy.ret))$r.squared
```

```
}
beta
```

```
##        AXP       AAPL         BA        CAT       CSCO        CVX
## 1.28780006 1.52005585 1.31298263 1.35482519 1.14499834 1.21474883
##         DD        XOM         GE         GS         HD        IBM
## 1.85453944 0.80797719 1.23246595 1.23868401 1.01629234 0.78188226
##       INTC        JNJ         KO        JPM        MCD        MMM
## 1.15387437 0.70391472 0.81062976 1.08763873 0.64278186 1.00592520
##        MRK       MSFT        NKE        PFE         PG        TRV
## 0.73581486 1.37373461 0.53965005 0.96611797 0.60482902 1.17625507
##        UNH        UTX         VZ          V        WMT        DIS
## 0.54828803 0.94344919 0.53294059 1.16055308 0.09761214 1.44669225
```

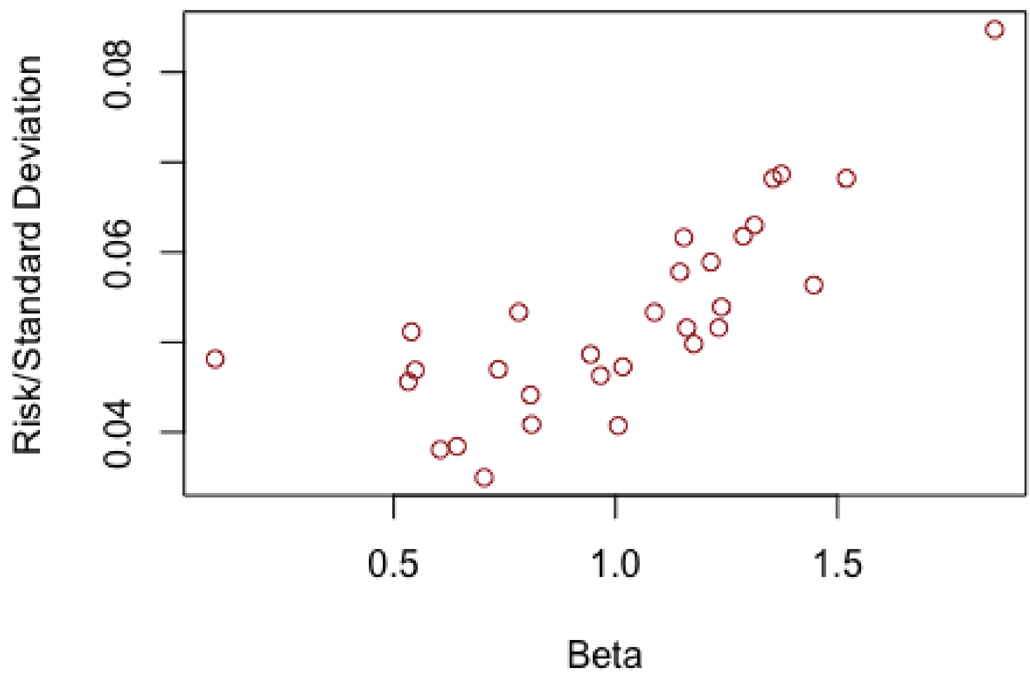
(b)

```
# Compute the standard deviation of monthly returns for each stock in the data file
# Initialize a placeholder for sd
sd.vec<-rep(NA,n.stock)
names(sd.vec)<-mystocks[,1]
# Compute sd for each stock
for(i in 1:n.stock){
  ticker<-mystocks[i,1]
  stockdata<-getSymbols(ticker,from="2013-09-01",to="2016-09-01",auto.assign = FALSE)
  y.ret<-monthlyReturn(Ad(stockdata))
  sd.vec[i]<-sd(y.ret)
}

sd.vec
```

```
##        AXP       AAPL         BA        CAT       CSCO        CVX
## 0.06178961 0.06819514 0.06300284 0.06817515 0.05779864 0.05890890
##         DD        XOM         GE         GS         HD        IBM
## 0.08474073 0.04413338 0.05160430 0.05390488 0.04726693 0.05333572
##       INTC        JNJ         KO        JPM        MCD        MMM
## 0.06163815 0.03495707 0.04085390 0.05332388 0.03843067 0.04071944
##        MRK       MSFT        NKE        PFE         PG        TRV
## 0.04698572 0.06868396 0.05115607 0.04629331 0.03806209 0.04979770
##        UNH        UTX         VZ          V        WMT        DIS
## 0.04689247 0.04865234 0.04562569 0.05158382 0.04814606 0.05635154
```

(c)

```
# Produce a scatter plot of beta on the x-axis and standard deviation of monthly returns on the y-axis
plot(beta, sd.vec,
     type="p",
     col="brown",
     ylab="Risk/Standard Deviation",
     xlab="Beta")
```



(d)

The plot-Beta versus standard deviation-appear to suggest that there is a positive linear relationship between Beta and risk (as measured by standard deviation). It is reasonable, because individual stock's standard deviation can be decomposed into two components under the Single Index Model -- systematic risk in proportional (Beta) to market risk (same for every stock), and non-systematic/residual risk independent of the market and all other stocks. Therefore, if we regress or plot individual stock's risk on beta, we would expect to observe they fit a line with a positive slope close to the market risk, and the noise represents firm/stock-specific risk independent of the market as a whole.

(e)

```
fit<-lm(sd.vec~beta)
summary(fit)
```

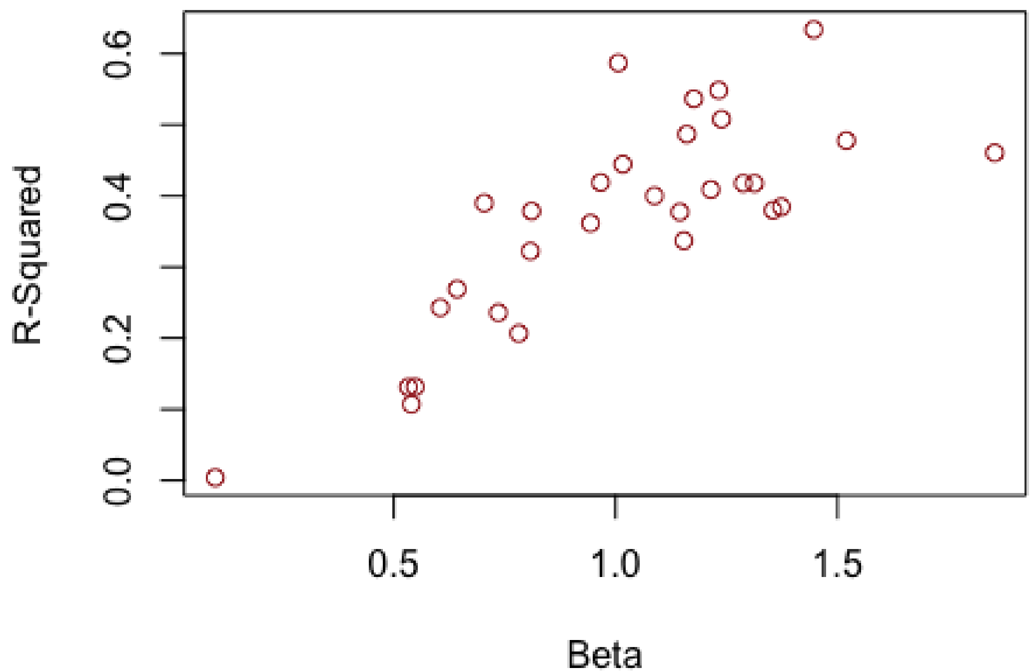
```
##
## Call:
## lm(formula = sd.vec ~ beta)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0118894 -0.0055356 -0.0003023  0.0045135  0.0162856
##
## Coefficients:
```

```
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.029631   0.003815   7.766 1.85e-08 ***
## beta        0.022843   0.003557   6.423 5.91e-07 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.007046 on 28 degrees of freedom
## Multiple R-squared:  0.5957, Adjusted R-squared:  0.5812
## F-statistic: 41.25 on 1 and 28 DF,  p-value: 5.913e-07
```

The intercept = 0.029631 and the slope = 0.022843 are both statistically significant.The model implies that, when individual stock's beta =0, that is, it doesn't move with the market, it still has a standard deviation of 2.96%, this can be viewed as average firm-specific or unsystematic risk that is independent of the overall market; and as the stock responds more sensitively to the market, with each 1-unit increase in beta, the standard deviation increases 2.28%, due to response to market swing, the 2.28% can be viewed as the market risk/standard deviation.

(f)

```
# plot R-squared vs Beta
plot(beta, R2,
     type="p",
     col="brown",
     ylab="R-Squared",
     xlab="Beta")
```



In my sample, large estimate of beta correspond with higher R-squared values, this agrees with my expectation. R-squared measures the percentage of a stock's historical price movements that could be explained by movements in a benchmark index. A larger estimate of beta indicates higher sensitivity of a stock's respond to swings in the market as whole, it's equivalent to say that, more variation in individual stock's movement can be explained by the market, therefore, has higher R-squared. So I expect Beta and R-squared should have a significant positive relationship.

5) In this exercise, you will show that β for a portfolio is a linear combination of the β 's for the stocks in the portfolio.

Consider the Market Model equation for two stocks A and B:

$R_A = \alpha_A + \beta_A R_M + \varepsilon_A$

$R_B = \alpha_B + \beta_B R_M + \varepsilon_B$

For each regression, it is assumed ε_A and ε_A are independent of R_M . Create a portfolio of stocks A and B with share of wealth x_A invested in stock A and share x_B invested in B such that $x_A + x_B = 1$. Show that the beta of the portfolio, denoted βp , satisfies $\beta_p = x_A\beta_A + x_B\beta_B$.

Hint: one way of writing β_p is $\beta_p = \dfrac{Cov(R_p, R_M)}{Var(R_M)}$.

$$\beta_p = \frac{Cov(R_p, R_M)}{Var(R_M)}$$

$$= \frac{Cov(x_A R_A + x_B R_B, R_M)}{Var(R_M)}$$

$$= \frac{x_A Cov(R_A, R_M) + x_B Cov(R_B, R_M)}{Var(R_M)}$$

$$= \frac{x_A Cov(\alpha_A + \beta_A R_A + \epsilon_A, R_M) + x_B Cov(\alpha_B + \beta_B R_B + \epsilon_B, R_M)}{Var(R_M)}$$

Since α_A and α_B are constant, and ϵ_A and ϵ_B are independent of R_M, therefore:

$$= \frac{x_A Cov(\beta_A R_M, R_M) + x_B Cov(\beta_B R_M, R_M)}{Var(R_M)}$$

$$= \frac{x_A \beta_A Cov(R_M, R_M) + x_B \beta_B Cov(R_M, R_M)}{Var(R_M)}$$

$$= \frac{x_A \beta_A Var(R_M) + x_B \beta_B Var(R_M)}{Var(R_M)}$$

$$= x_A\beta_A + x_B\beta_B$$