

MINI Project 2

CAPM Validation



STAT 686 Group 4

Christina Wang, Oliver (Ran) Jin, Rui Qin, Xingyue Zhang

16 Feb, 2018

Contents

Introduction.....	2
Part I Data	2
Download	2
Cleaning data	2
Methodologies	3
Part II Summary	4
Graphics and Table	4
Analysis.....	10
Conclusion.....	10
Reference	11
Appendix (Written in R)	12

Introduction

The efficient market hypothesis (EMH) states because of the market efficiency, the market prices always reflect all relevant information to invest; Therefore, it's impossible to outperform the overall market through stock selection or market timing, and the only way for an investor to get potential higher returns is to take higher risks. In this project, we tested this hypothesis by replicating and extending the results of Wojciechowski and Thompson (2006). Our result plots show that for the 7 years, 1972, 1973, 1987, 1998, 1999, 2008, and 2009, most random portfolios lie above the capital market line (CML), which show strong evidence against the EMH.

This reports consists of two main parts. The first part includes the description of our datasets and the methodologies that we used for calculations and simulations. In the second part, we show the results from our analysis in graphical and tabular forms for each year along with comments. The R code is attached in the appendix.



Part I Data

Download

There are three sets of data that we downloaded from Wharton Research Data Services for all 7 years. The first dataset has the daily stock return for the 7 years of interest, with variables of company name, ticker, exchange code, holding period return, number of shares outstanding, and prices. The second dataset is the daily returns for the index NYSE/AMEX/NASDAQ/ARCA for each year. The third dataset includes the Fama French daily risk free rate for each year. And all the above datasets were saved in csv format.

Cleaning data

The first dataset is the only one that needed to be cleaned. We filtered out the missing entries in PRC (price), RET (return), and SHROUT (number of shares outstanding).

Methodologies

We chose the median of all trading days within the year to be the representative number of trading days for each year. However, for 1972, the median is 9 which is a very unreasonable number. Thus, we have selected the maximum number of trading days in 1972 which is 251. On the other hand, after picking out the 1000 Market Cap from all stocks for each year, we have to produce portfolios with size = 30, 100, 500, 1000 with randomized weights. The following three formulas (obtained from the project instructions) are we have used for calculations:

1. Market Cap = price * number of shares outstanding

This is used when we are picking out the 1000 Market Cap from all stocks.

2. Market Reference Point (μ , σ) = (annualized returns from daily returns, annualized volatility from daily volatility)

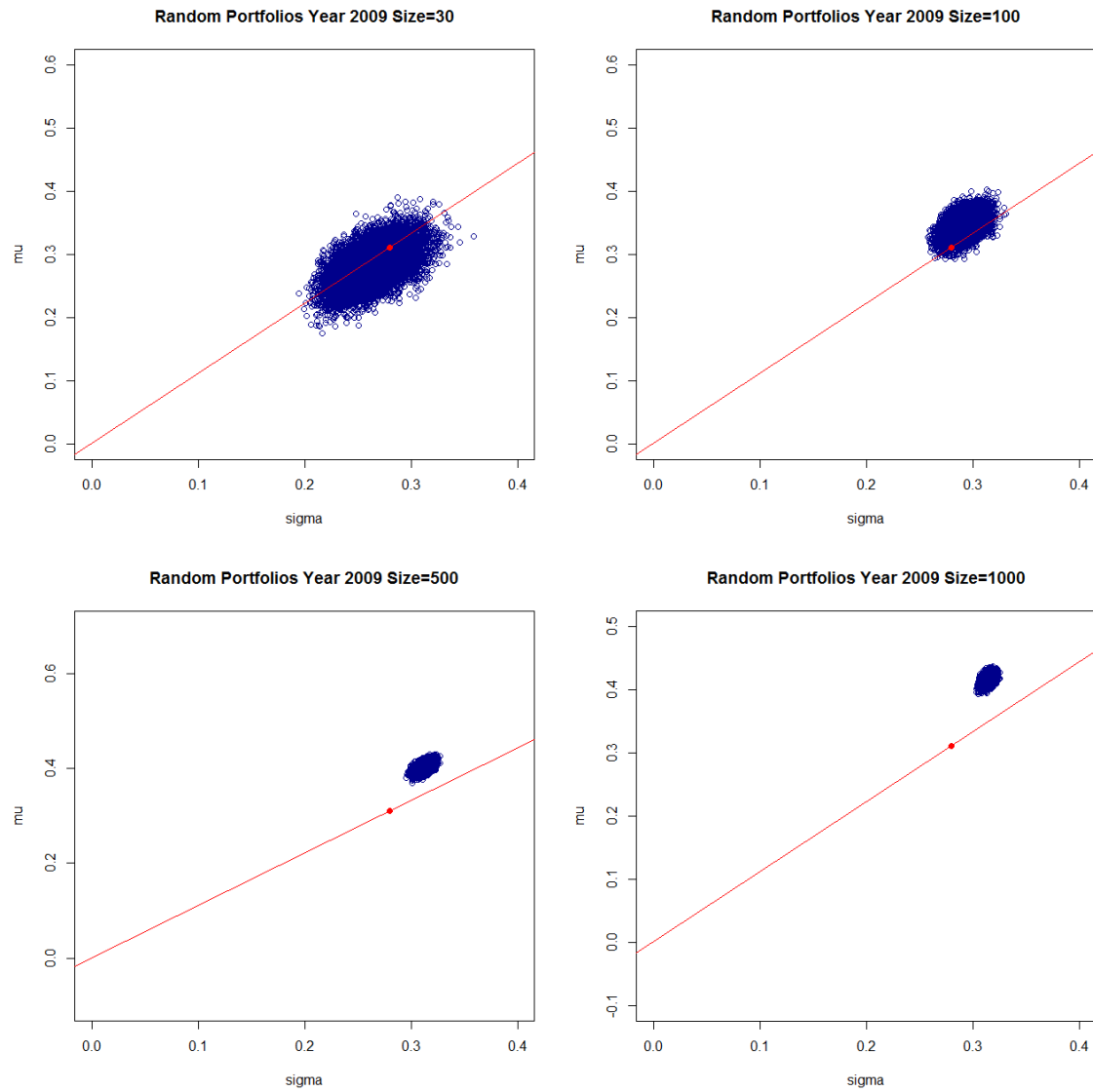
We did not explicitly write out this formula and we have used the built in function (return.annualized) from the package of PerformanceAnalytics in R for this particular calculation. And this is used to plot a market reference point when plotting μ vs. σ .

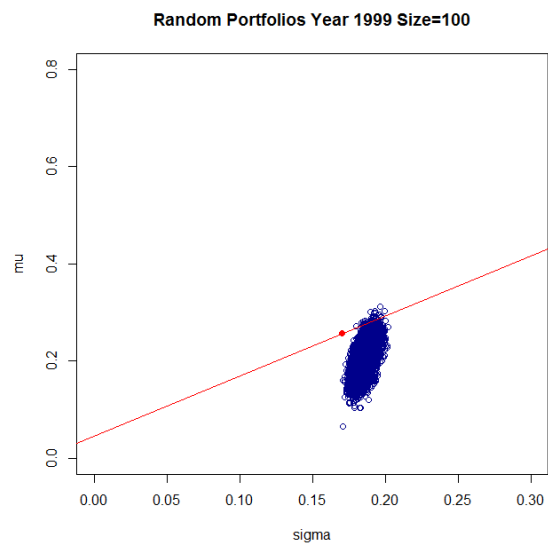
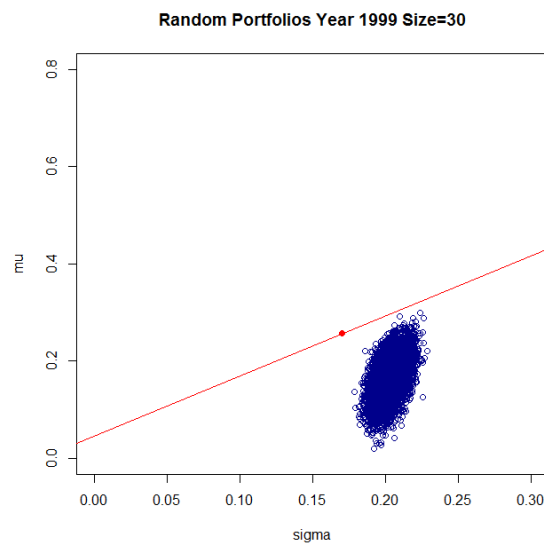
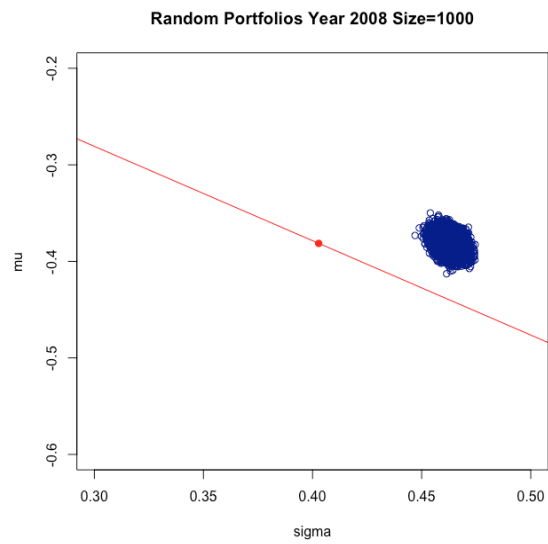
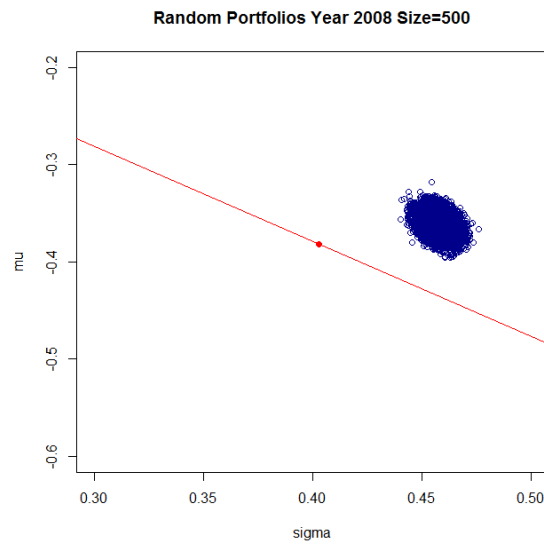
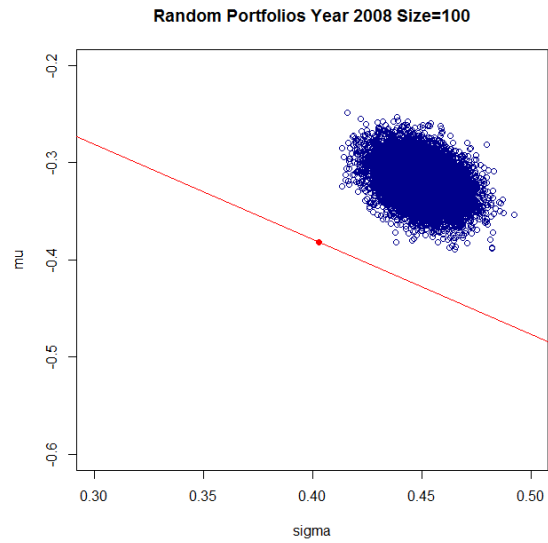
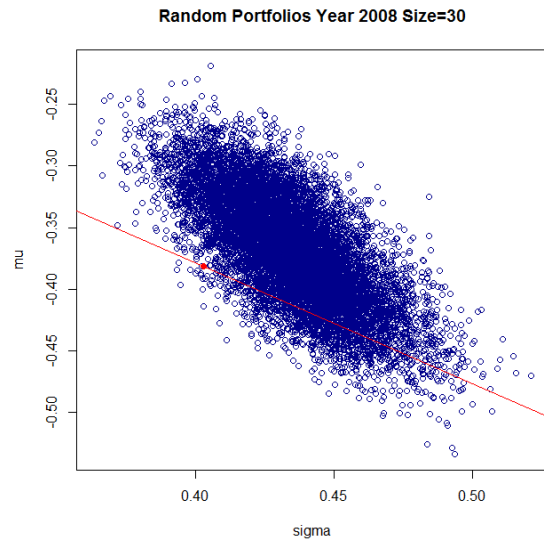
3. Fama French risk free rate = (highest rate of the year + lowest rate of the year)/2 * number of trading days

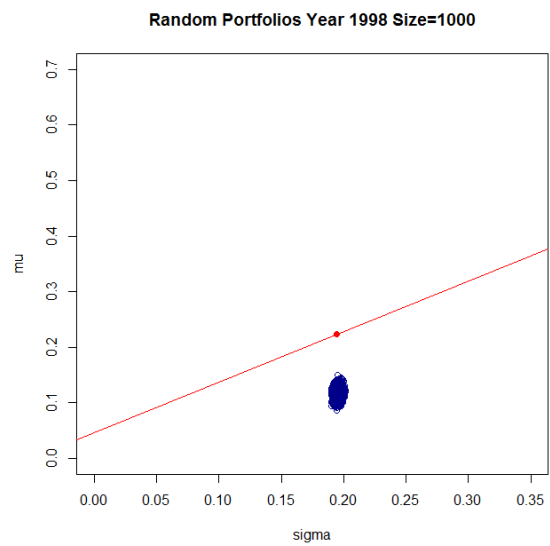
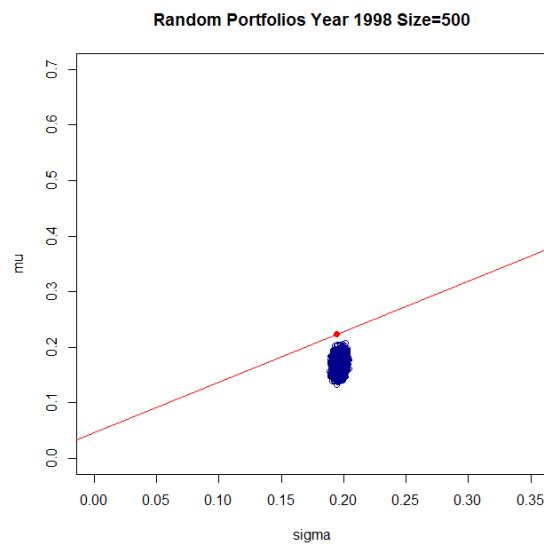
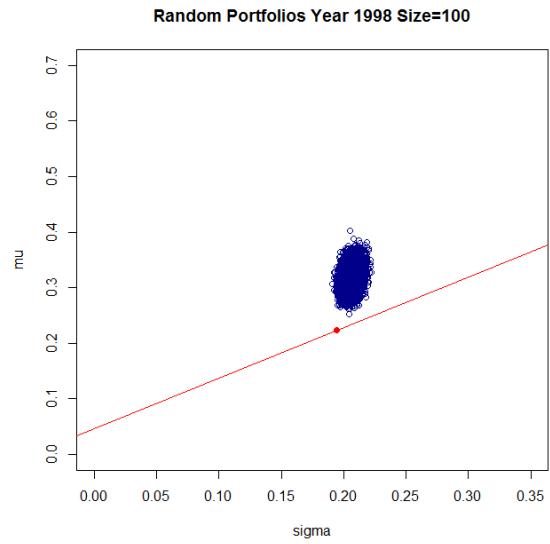
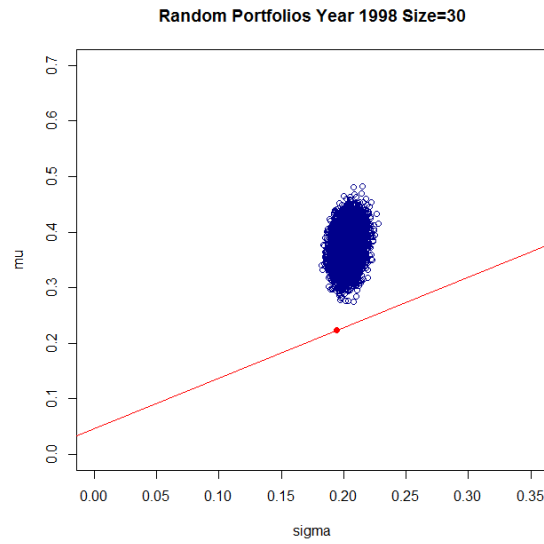
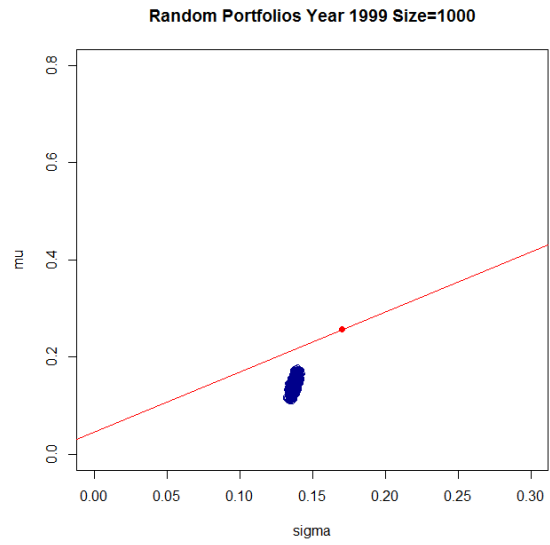
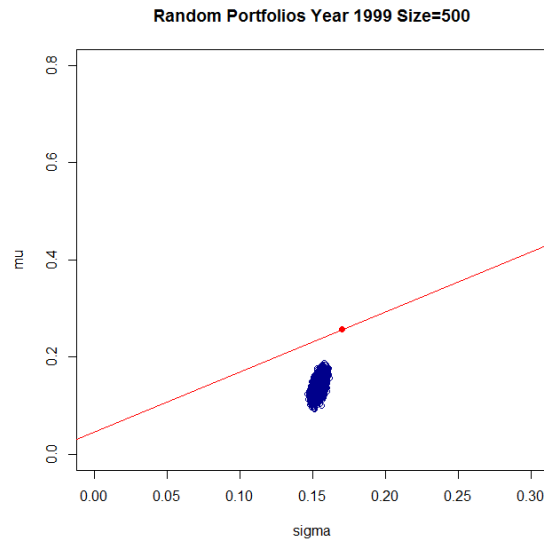
This is used as the intercept term of y-axis for each year when plotting μ vs. σ .

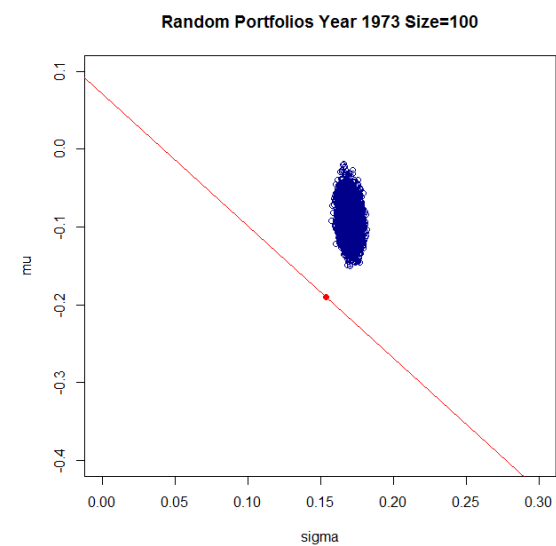
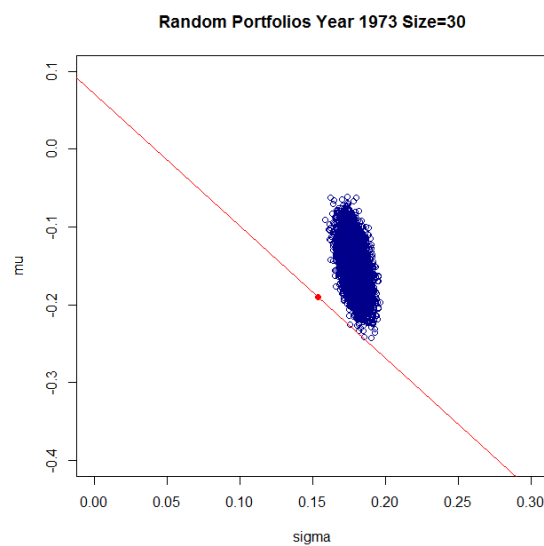
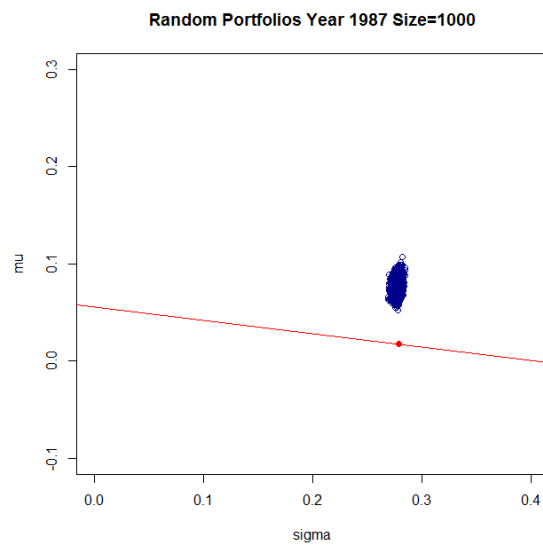
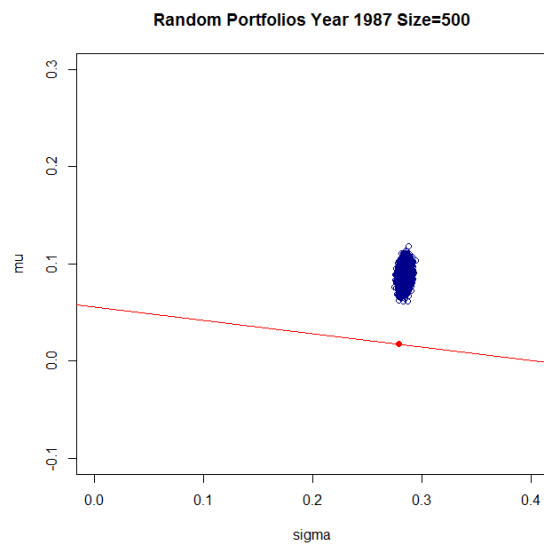
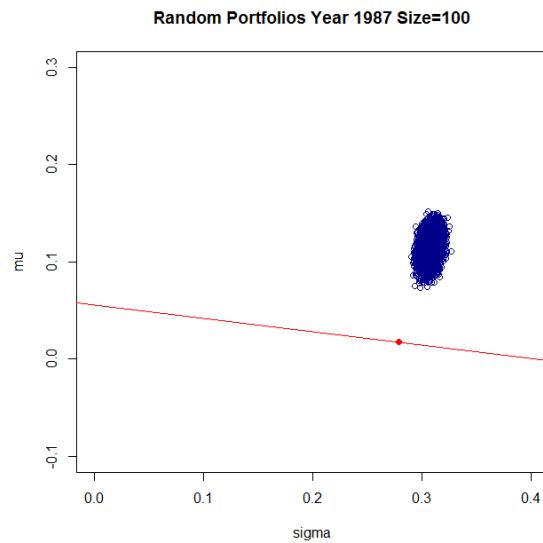
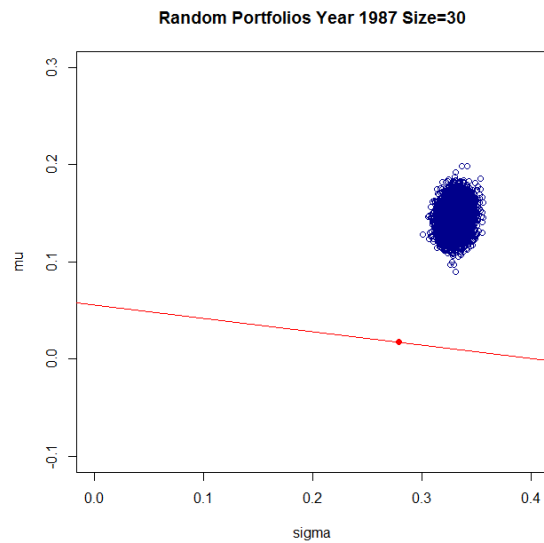
Part II Summary

Graphics and Table









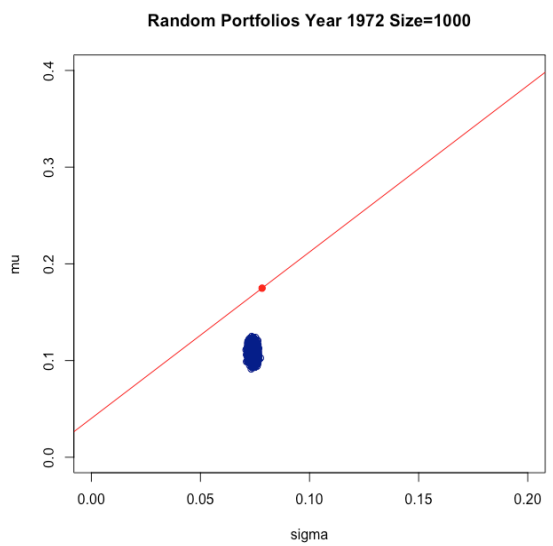
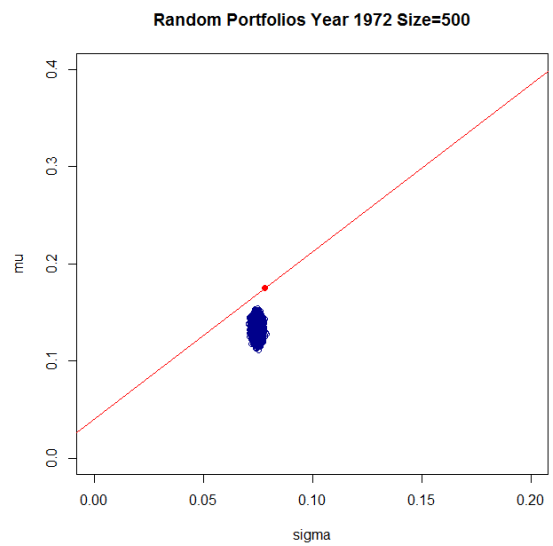
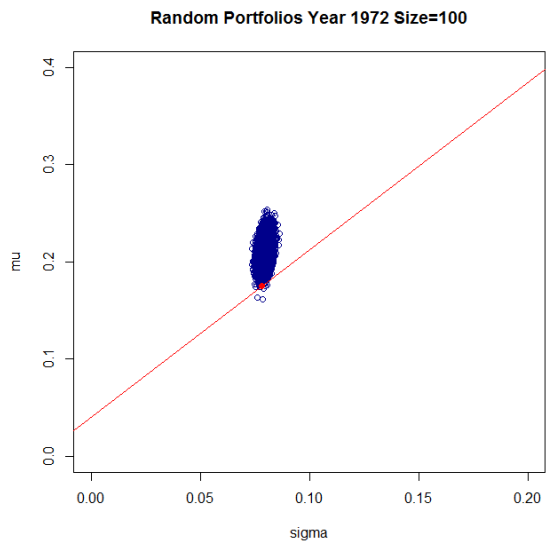
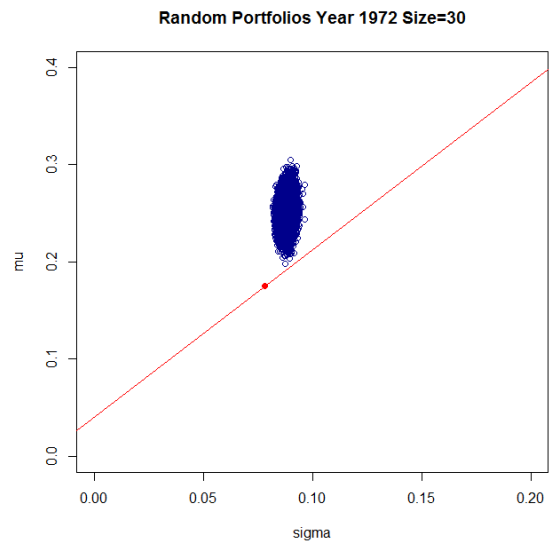
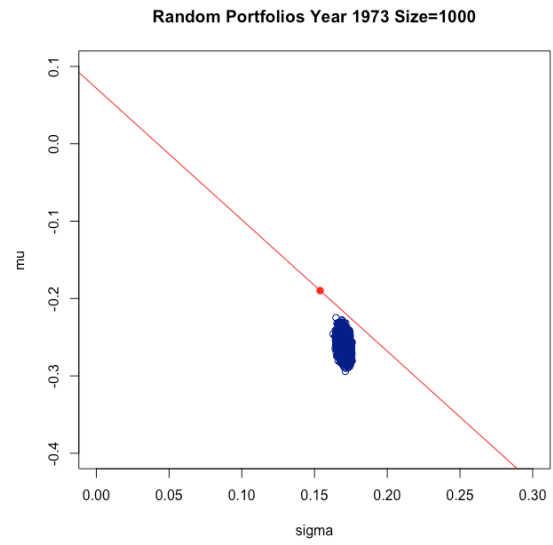
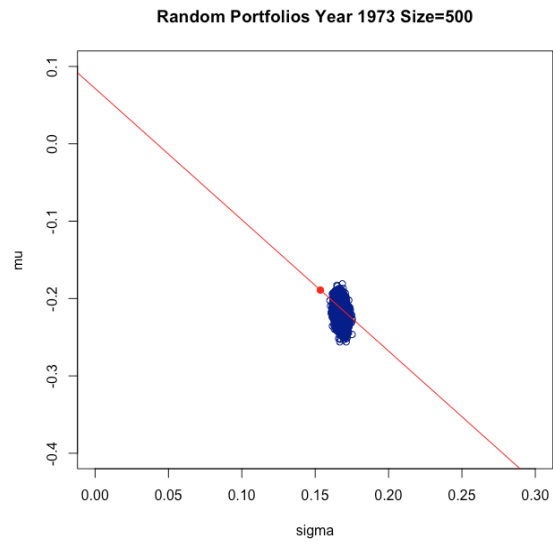
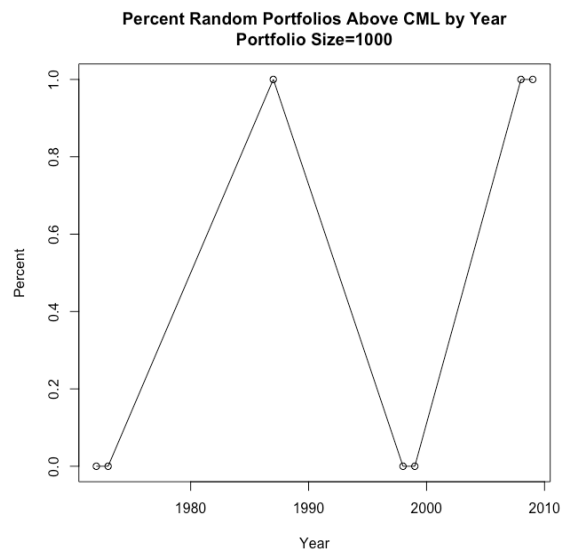
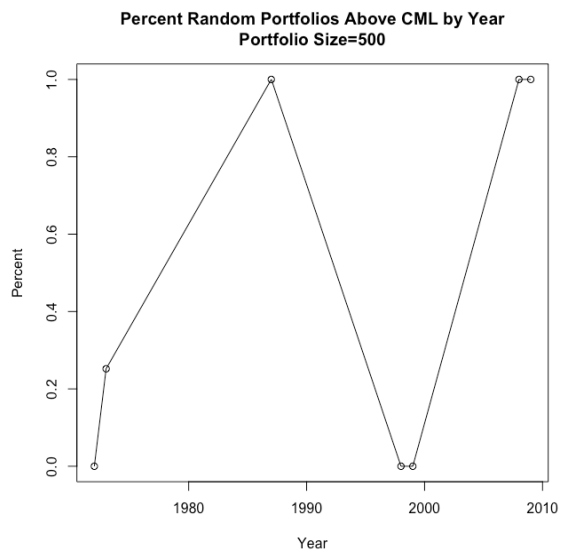
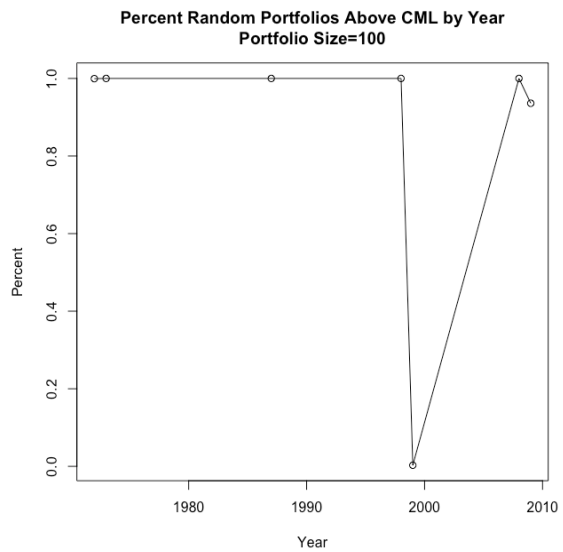
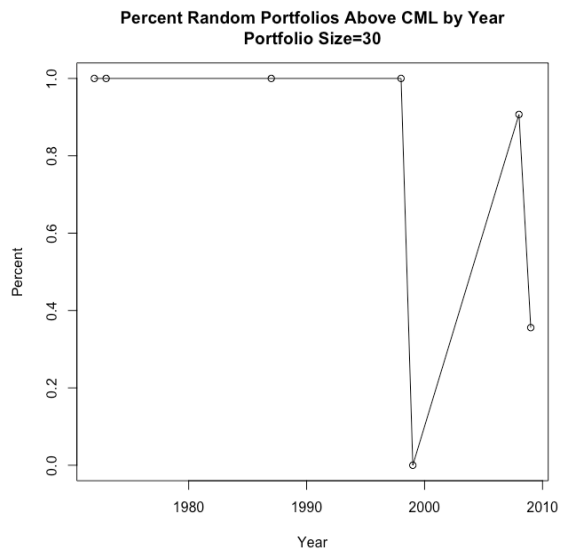


Table: Summary of the Percent of Random Portfolios which Lie Above the CML for Each Year

	1972	1973	1987	1998	1999	2008	2009
30	1	1	1	1	0	0.9069	0.3558
100	0.9995	1	1	1	0.0028	1	0.9360
500	0	0.2520	1	0	0	1	1
1000	0	0	1	0	0	1	1



Analysis

As the size of the portfolio increases, we see that for each year, the variation decreases (the data looks more clogged), which is well-explained by the implication of diversifying portfolios could ultimately decrease diversifiable risk. On the other hand, μ and σ have a positive correlation in 1972, 1998, 1999, 2009. However, μ and σ sometimes have an unexpected correlation such that as σ (risk) increases, μ (return) decreases in year 1973 (oil crash), 1987 (stock markets around the world crashed), 2008 (financial crisis).

For most of the plots above, no relationship between volatility and returns is present; however, for 2008, a negative relationship could be observed, and for 2009, a positive relationship could be observed. We believe this could be largely explained by the financial crisis in 2008, and the recovery in 2009. From the above plots, we could see that most of the random portfolios lie above the CML, and we also noticed that almost all the random portfolios lie under the line in 1999, which is unusual. Also for 1972, 1973, and 1998, as the portfolio size increases, more random portfolios fall under the CML.

The plots of Percent Random Portfolios Above CML By Year below also indicate for all sizes that 1973 and 1987 have close to 0 percent of portfolios above CML which is also an indication that the stock market at the time were in bad financial conditions. The table and plots below indicate that when the portfolio size is small (30, 100), more years have 100% random portfolios above CML.




Conclusion

In this project, we simulated random portfolios with different sizes for 7 years by using the stocks with the highest 30, 50, 500, 1000 market capitalization on the first trading day for each year. The results show that volatility and returns are not positively correlated. Our plots on an yearly basis show that returns and volatility could have positive, negative, or zero correlation with returns. This indicates that there's no risk measure is appropriate for all situations, which also provide us evidence against the efficient market hypothesis.



Reference

- [1] Wharton Research Data Services. <https://wrds-web.wharton.upenn.edu/wrds/>. Accessed Feb 16, 2018
- [2] Cover picture: Investopedia. <https://www.investopedia.com/articles/investing/022615/valuation-models-apples-stock-analysis-capm.asp>. Accessed Feb 16, 2018
- [3] WILLIAM C. WOJCIECHOWSKI and JAMES R. THOMPSON, *Market truths_ theory versus empirical simulations*, Accessed Feb 16, 2018
- 

Appendix (Written in R)

```
setwd("/Users/alisonzhang/Desktop/2018 Spring/STAT 686/Mini Project 2")
dat = read.csv("1617 Data.csv")
fff = read.csv('FamaFrench.csv')
# Keep the year for dat and fff the same!!!!
```

```
View(fff)
```

```
require(dplyr)
```

```
##### Data Preprocessing #####
```

```
class(dat$RET)
dat1 = subset(dat, RET != 'B' & RET != 'C' & RET != "" & (!is.na(dat[,3])) &
              (!is.na(dat[,4])) & (!is.na(dat[,5])))
# Filter out missing PRC, RET, and SHROUT
class(dat1$date)
dat1 = transform(dat1, date = as.Date(as.character(date), "%Y%m%d"))
# Transform the format of date
```

```
dat1$year = format(dat1$date, '%Y')
count_day = dat1 %>% group_by(year, PERMNO) %>% summarize(n = n())
med = count_day %>% group_by(year) %>% summarize(median = median(n))
# Get the median number of trading days for stocks by year
```

```
countt = 0
med$year = as.numeric(med$year)
count_day$year = as.numeric(count_day$year)
traded_comp = vector("list", length(med$year))
names(traded_comp) = med$year
for(i in med$year){
  countt = countt + 1
  subset_year = subset(count_day, year == i)
  med_year = as.numeric(med[countt, 2])
  for(j in 1:length(subset_year$PERMNO)){
    if(as.numeric(subset_year[j, 3]) == med_year){
      traded_comp[[countt]] = c(traded_comp[[countt]], as.integer(subset_year[j, 2]))
    }
  }
}
# Create a dictionary (list) to store the year as key, and the identity of
# company with median number of trading days as value
```

```
dat1$year = as.numeric(dat1$year)
dat2 = data.frame()
countt = 0
for(i in med$year){
  countt = countt + 1
  subset_year = subset(dat1, year == i)
  subset_year = subset_year[(subset_year$PERMNO %in% traded_comp[[countt]]), ]
  dat2 = rbind(dat2, subset_year)
}
# Filter out companies with non-median trading days from dat1 by year
# The new dataset is called dat2
```

```

length(dat2$PERMNO)/length(dat1$PERMNO)
# Coverage: 0.9294203

dat2$RET = as.numeric(as.character(dat2$RET))
dat2$market_cap = dat2$SHROUT * dat2$PRC
# Calculate market_cap
dat2$month = as.numeric(format(dat2$date, '%m'))
dat2$day = as.numeric(format(dat2$date, '%d'))
jan = subset(dat2, month == 1)
ini_traded = jan %>% group_by(year) %>% summarise(min = min(day))
# Get the initial trading day for each year
init_traded_dat = data.frame()
countt = 0
for(i in ini_traded$year){
  countt = countt + 1
  subset_year = subset(jan, year == i)
  init_day_year = as.numeric(ini_traded[countt, 2])
  subset_year = subset(subset_year, day == init_day_year)
  init_traded_dat = rbind(init_traded_dat, subset_year)
}
# Get all data for on the initial trading day for each year
rank_comp = init_traded_dat %>%
  group_by(year) %>%
  mutate(my_ranks = dense_rank(desc(market_cap)))
# Rank the market cap by company by year (dense rank)

##### Input N HERE #####
select_N_comp = function(n){
  countt = 0
  comp_lst = vector("list", length(med$year))
  names(comp_lst) = med$year
  for(i in med$year){
    countt = countt + 1
    subset_year = subset(rank_comp, year == i)
    gett = subset_year[(subset_year$my_ranks %in% 1:n), ]
    comp_lst[[countt]] = gett$PERMNO
  }
  return(comp_lst)
}

#####
# Every time, re-run from here, if changing N #
#####

select_comp = select_N_comp(500)

# Generate year & company list for top N market-cap company
# in dictionary format (key:value)
#####

dat3 = data.frame()
countt = 0
for(i in med$year){

```

```

countt = countt + 1
subset_year = subset(dat2, year == i)
subset_year = subset_year[subset_year$PERMNO %in% select_comp[[countt]], ]
dat3 = rbind(dat3, subset_year)
}
# Get all data for top N market-cap company
dat3$month = NULL
dat3$day = NULL

require(tidyr)
##### Run the model #####
paper_entry = function(n, yearr){
  w = runif(n, min = 0, max = 1)
  W = w/sum(w)
  # Generate random weights w

  dat4 = subset(dat3, year == yearr)
  dat4$date_ = as.integer(format(as.Date(dat4$date), "%Y%m%d"))
  get_lst = cbind(dat4$PERMNO, dat4$RET)
  get_lst = cbind(get_lst, dat4$date_)
  colnames(get_lst) = c("PERMNO", 'return', 'date')
  get_lst = as.data.frame(get_lst)

  R = get_lst %>% spread(PERMNO, return)
  R$date = NULL
  R = as.matrix(R)
  # Generate r the return of company j on day i: company: row; day: column
  N = nrow(R)
  # Let N be the number of trading days in the year

  P = R %*% W
  # Return of the random stock portfolio on a day
  mu = sum(P)/N
  # Average daily return of the random stock portfolio. Not returned
  mu_year = sum(P)
  # Annualized random stock portfolio return
  sigma = sqrt(N/(N-1) * sum((P-mu)^2))
  # Annualized standard deviation.

  result = c(mu_year, sigma)
  return(result)
}

##### Input N, Year HERE #####

##### Keo the N you enter here the same as the N you entered above
simulation = function(n1, year1){
  new_df = data.frame(matrix(ncol = 2, nrow = 0))
  colnames(new_df) = c('mu', 'sigma')
  for(i in 1:10000){
    result = paper_entry(n1, year1)
    new_df[i, ] = result
  }
}

```

```

# Normalize:
# new_df$mu = new_df$mu/mean(new_df$mu)
# new_df$sigma = new_df$sigma/mean(new_df$sigma)

return(new_df)
}
result = simulation(500, 2016)
# plot(x = result$sigma, y = result$mu)
#####

## The larger N is, the larger the range. Check the problem !!!!!

##### Fama French risk free rate
fff = transform(fff, date = as.Date(as.character(date), "%Y%m%d"))
fff$year = format(fff$date, "%Y")
min_max = fff %>% group_by(year) %>% summarise(low = min(rf), high = max(rf))
min_max$average = (min_max$low + min_max$high)/2 * med$median

require(ggplot2)
##### Data Visualization

```