Empirical Methods in Finance Homework 7

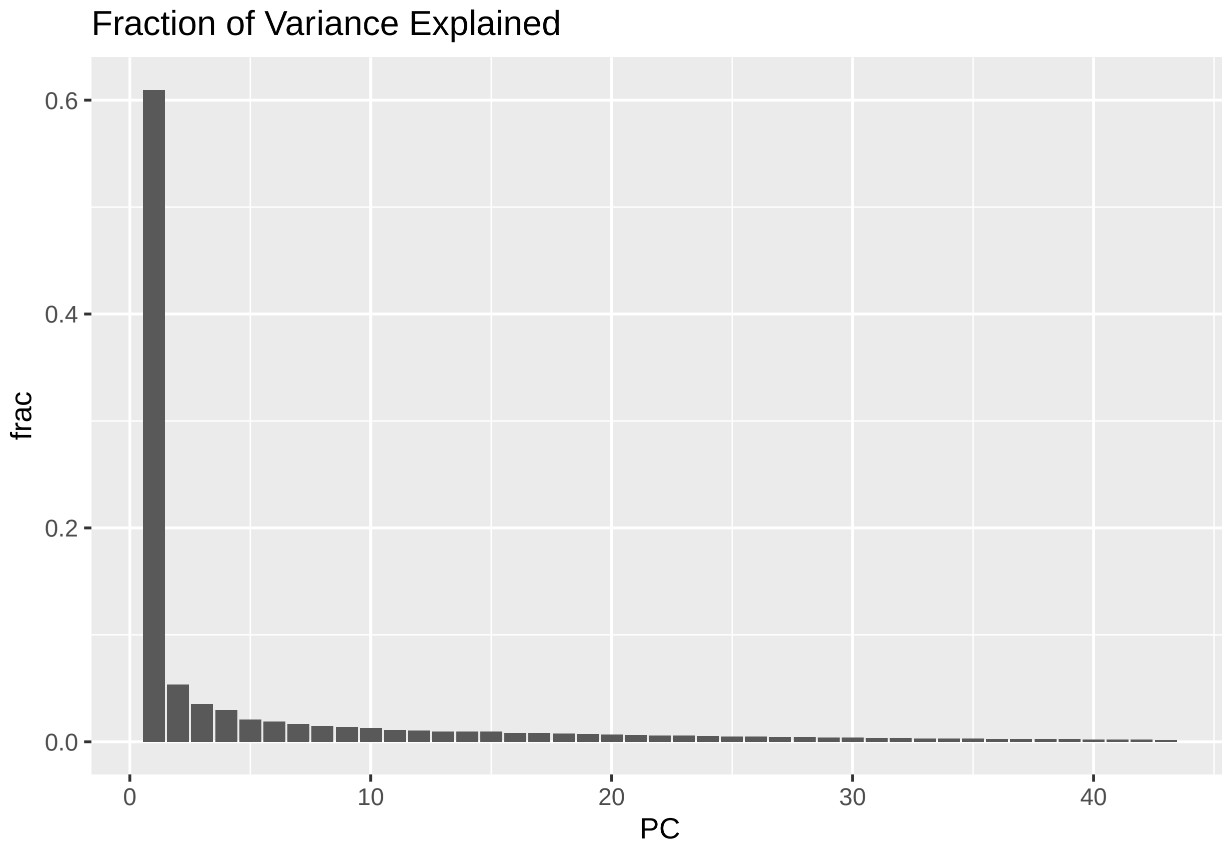
Group 9: Linqi Huang, Abhesh Kumar, Yu Onohara, Maitrayee Patil, Redmond Xia

Problem 1

```
# Importing libraries and organizing data
suppressMessages(library(tidyverse))
suppressMessages(library(readxl))
suppressMessages(library(ggrepel))
suppressMessages(library(stargazer))
suppressMessages(library(xtable))
options(xtable.comment = FALSE)

ff <- read_csv('F-F_Research_Data_Factors.csv', skip=3)%>%
  filter(X1>=196001 & X1<=201512)

ind <- read_csv('48_Industry_Portfolios.csv', skip=11) %>%
  filter(X1>=196001 & X1<=201512) %>%
  slice(1:nrow(ff)) %>%
  mutate_if(is.numeric, list(~na_if(., "-99.99"))) %>%
  select_if(~!any(is.na(.))) %>%
  mutate(RF = ff$RF)
Ind = ind[2:44] - ind$RF

# Problem 1
cov.mat = cov(Ind) # variance-covariance matrix of the excess return
eigen.cov = eigen(cov.mat) # solving eigen problem
eigen.val = eigen.cov$values # eigen value
eigen.vec = eigen.cov$vectors # eigen vector
frac.var = eigen.val / sum(eigen.val) # fraction of variance explained
eigen.val
```

```
##  [1] 1063.588563   93.701238   61.827898   51.712939   36.598735
##  [6]   32.659992   28.721750   25.630900   23.971119   22.374542
## [11]   19.468473   18.634690   16.902934   16.655990   16.462992
## [16]   14.635944   14.081187   13.595840   12.905028   12.132211
## [21]   11.212621   10.432142   10.065567    9.291008    8.862178
## [26]    8.399305    7.913556    7.687662    6.964257    6.926746
## [31]    6.540604    6.043709    5.671311    5.499267    4.895282
## [36]    4.807527    4.615763    4.422294    4.067772    3.820311
## [41]    3.616439    3.478702    3.077869
```

```
# Plotting the fractions
df = data.frame(matrix(ncol = 0, nrow = length(frac.var))) %>%
  mutate(PC = seq(1, length(frac.var), 1)) %>%
  mutate(frac = frac.var)

plot1 = ggplot(data = df, aes(x = PC, y = frac)) + geom_bar(stat = "identity")
plot1 = plot1 + ggtitle("Fraction of Variance Explained")
plot1
```



Problem 2

```
# Problem 2
# a
# accumulative variances from PC1 to PC3
cum.frac.var = sum(frac.var[1:3])
cum.frac.var
```

```
## [1] 0.698805
```

```
# b
# Implementing PCA
pca = prcomp(Ind)

# Picking weights and calculating sample mean, stddev, and correlation
weight = pca$rotation
ind.mean = apply(Ind, 2, mean)
ret = as.matrix(Ind) %*% weight
sample.mean = apply(ret, 2, mean)
sample.stddev = sqrt(apply(ret, 2, var))
cor.mat = cor(ret)
cor.mat = cor.mat[1:3,1:3]
```

```
print(xtable(as.data.frame(sample.mean[1:3])))
```

```
print(xtable(as.data.frame(sample.stddev[1:3])))
```

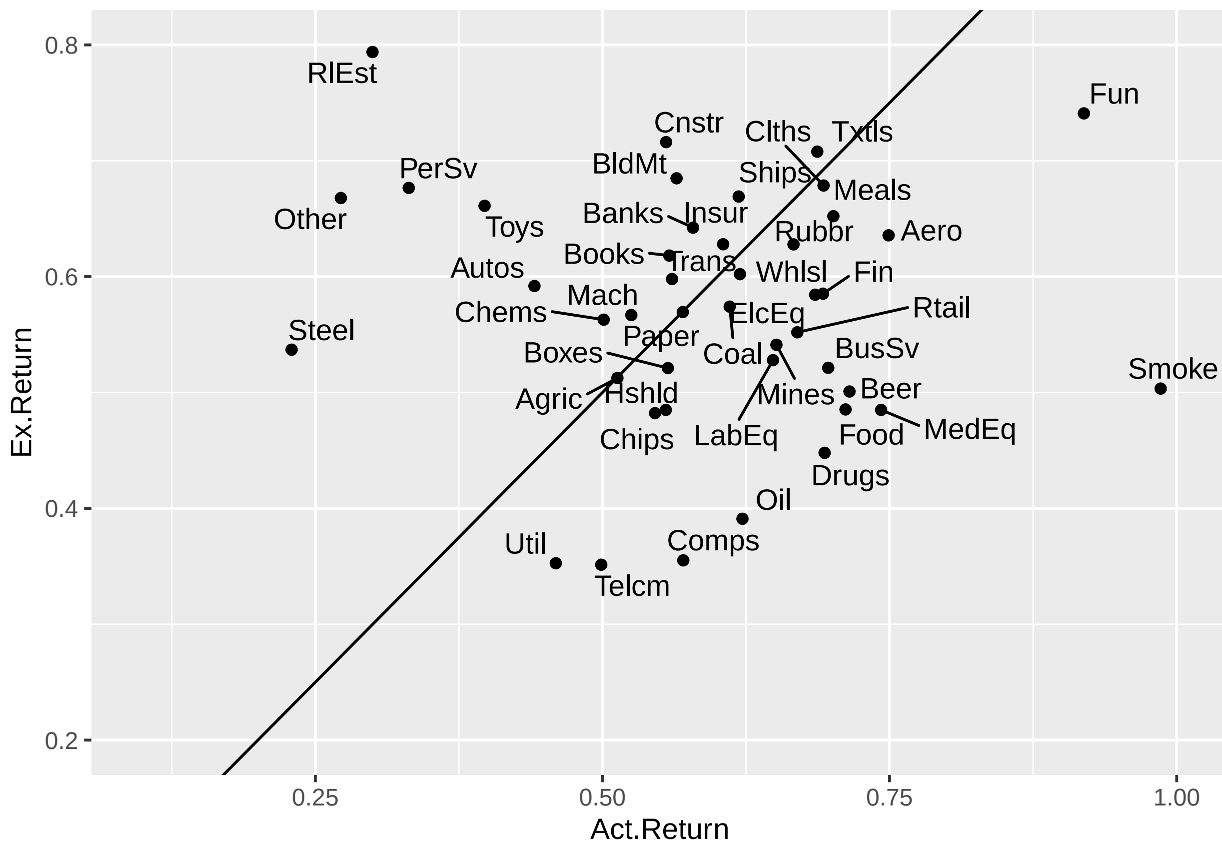
	sample.mean[1:3]
PC1	3.77
PC2	0.21
PC3	-0.50

	sample.stddev[1:3]
PC1	32.61
PC2	9.68
PC3	7.86

```
print(xtable(cor.mat, digit = 6))
```

	PC1	PC2	PC3
PC1	1.000000	0.000000	-0.000000
PC2	0.000000	1.000000	-0.000000
PC3	-0.000000	-0.000000	1.000000

```
# c
# Calculating E[Rt] and E[Ft]. Beta equals to weights
beta = weight[, 1:3]
E.Ft = sample.mean[1:3]
E.Rt = beta %*% E.Ft
act.ret = apply(Ind, 2, mean)

# Plotting the results
df2 = data.frame(matrix(ncol = 0, nrow = length(E.Rt))) %>%
  mutate(Industry = colnames(Ind)) %>%
  mutate(Ex.Return = E.Rt) %>%
  mutate(Act.Return = act.ret)

plot2 = ggplot(data = df2, aes(x = Act.Return, y = Ex.Return, label = Industry)) + geom_point()
plot2 = plot2 + geom_abline(intercept = 0) + xlim(c(0.1,1)) + ylim(c(0.2,0.8))
plot2 = plot2 + geom_text_repel()
plot2
```



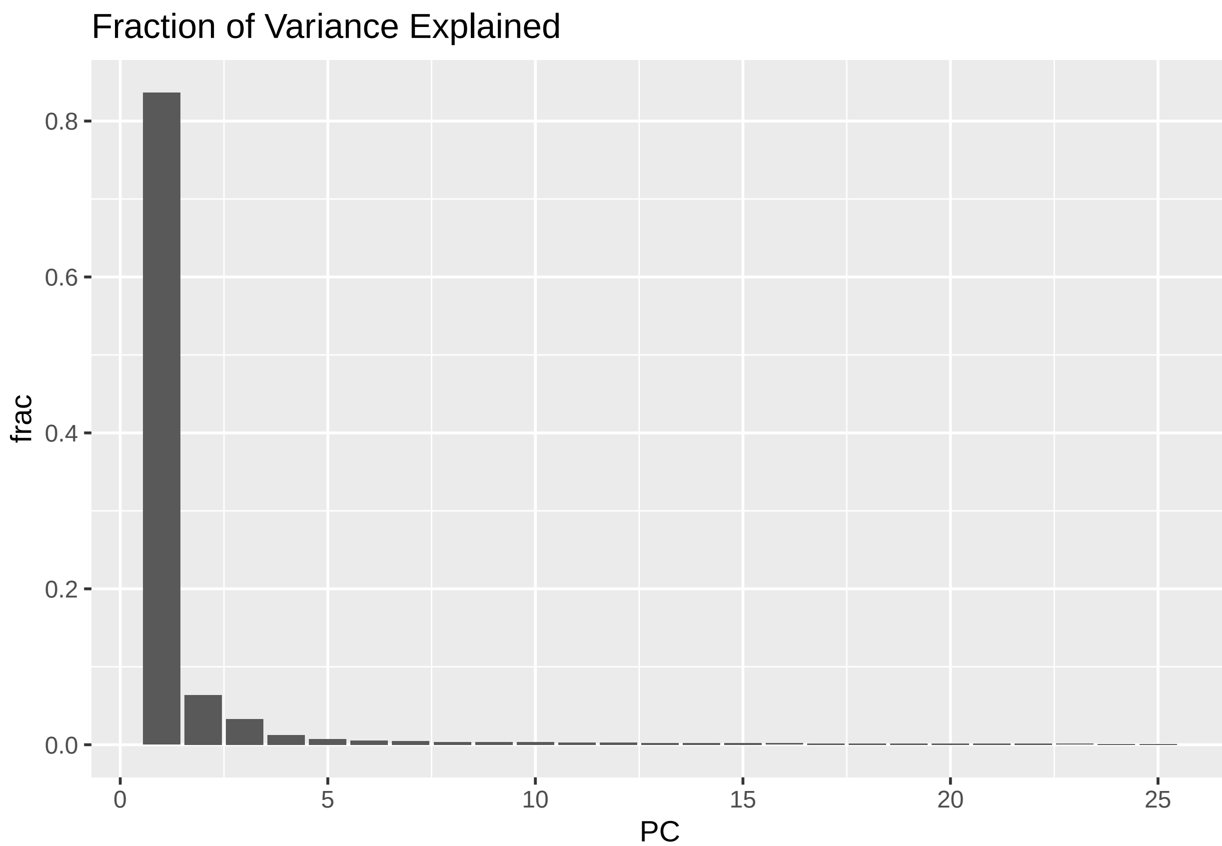

```r
# d
diff = act.ret - E.Rt
diff.va = diff - mean(diff)
test = act.ret - mean(act.ret)
var.d = var(diff)
var.act = var(df2$Act.Return)
Rsquare = 1 - var.d / var.act
Rsquare
```

```
##             [,1]
## [1,] -0.5947519
```

Problem 3

```r
# a
# preparing the data
ff25 <- read_csv('25_Portfolios_5x5.CSV', skip=15)%>%
  filter(X1>=196001 & X1<=201512) %>%
  slice(1:nrow(ff))
FF25 = ff25[,2:26]
FF25 = FF25 - ind$RF

# getting the fraction of variance in the similar way of problem 1
cov.matff = cov(FF25)
eigen.covff = eigen(cov.matff)
eigen.valff = eigen.covff$values
```

```r
eigen.vecff = eigen.covff$vectors
frac.varff = eigen.valff / sum(eigen.valff)

# Organizing the data
df3 = data.frame(matrix(ncol = 0, nrow = length(frac.varff))) %>%
  mutate(PC = seq(1, length(frac.varff), 1)) %>%
  mutate(frac = frac.varff)

# plotting the fraction of variance explained
plot3 = ggplot(data = df3, aes(x = PC, y = frac)) + geom_bar(stat = "identity")
plot3 = plot3 + ggtitle("Fraction of Variance Explained")
plot3
```



By the PC5, the accumulative fraction of variance explained reaches 95%.

```r
# b
# accumulative variances
cum.frac.varff = cumsum(frac.varff[1:25])
cum.frac.varff
```

```
##  [1] 0.8362506 0.8998043 0.9325635 0.9450868 0.9522204 0.9577525 0.9627125
##  [8] 0.9664420 0.9700398 0.9733405 0.9762290 0.9788752 0.9812569 0.9835512
## [15] 0.9855775 0.9874736 0.9892079 0.9908919 0.9924712 0.9939822 0.9953437
## [22] 0.9966451 0.9979014 0.9989749 1.0000000
```