HW5-Group7-Cohort2

Cohort 2, Group 7 - Hyeuk Jung, Jiaqi Li, Xichen Luo, Huanyu Liu

March 3, 2019

Principal Component Analysis

```
library(data.table)
library(dplyr)
```

```
## Warning: package 'dplyr' was built under R version 3.5.2
##
## Attaching package: 'dplyr'
## The following objects are masked from 'package:data.table':
##
##     between, first, last
## The following objects are masked from 'package:stats':
##
##     filter, lag
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(sandwich)
library(ggplot2)

industry_48 <- read.csv("48_Industry_Portfolios_vw.csv", header = T) %>% as.data.table; gc()
```

```
##           used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells  620548 33.2    1183282 63.2         NA  1183282 63.2
## Vcells 1165339  8.9    8388608 64.0      16384  2215324 17.0
```

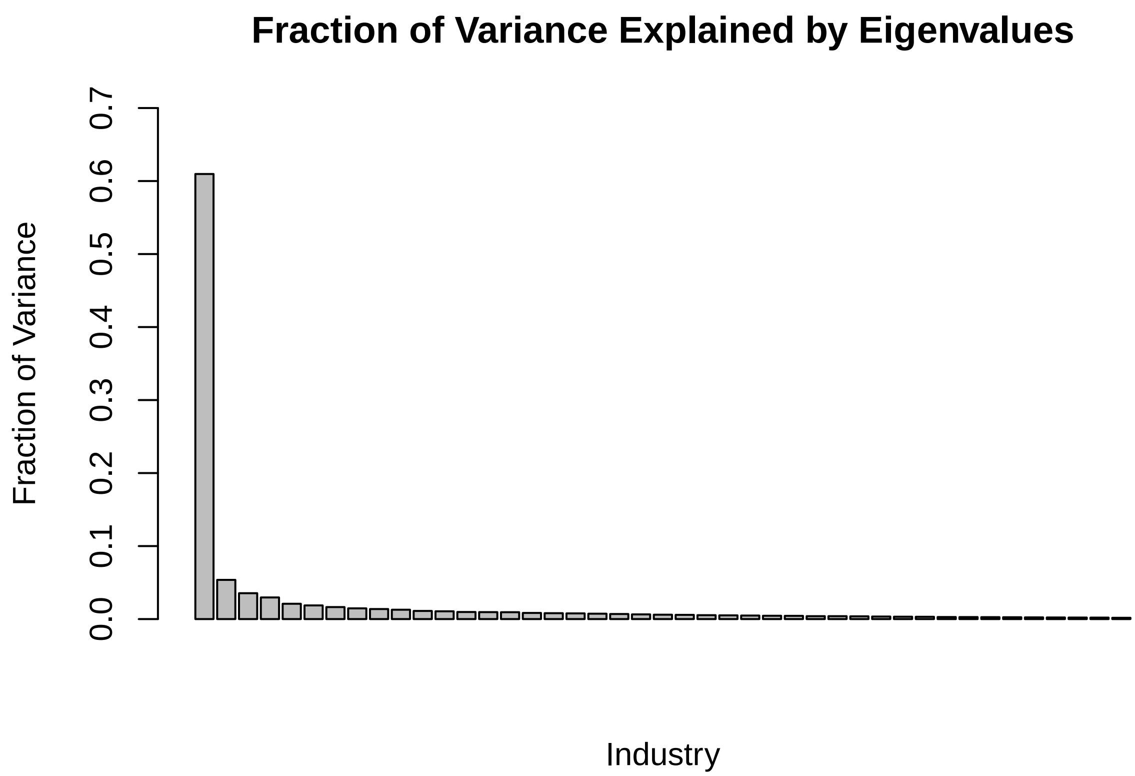
```
famafrench <- read.csv("F-F_Research_Data_Factors.csv", header = T) %>% as.data.table; gc()
```

```
##           used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells  620999 33.2    1183282 63.2         NA  1183282 63.2
## Vcells 1173351  9.0    8388608 64.0      16384  2215324 17.0
```

```
# Data Cleaning and Manipulation
# 1. Subset the data: year 1960~2015
industry_48$year <- as.numeric(substr(industry_48$X, 1, 4))
industry <- industry_48[ (industry_48$year >= 1960) & (industry_48$year < 2016), ]
# 2. Remove columns if it holds -99.99 or missing values
# changing -99.99 values to NA
industry[industry == -99.99] <- NA
industry[industry == -999] <- NA
# removing columns with NAs
industry_filtered <- industry %>% select_if( ~ !any(is.na(.)) )
# 3. Merge industry data and Rf from Fama-French data
data <- merge(industry_filtered, famafrench, by = "X")
```

Problem 1

```r
# 1. Calculate the excess return of each industry
ex.ret <- (data[, 2:44] - data$RF) #/100 # change into percentile dim(ex.ret)
# 2. Get the var-cov matrix
varcov <- cov(ex.ret)
# 3. Get the eigenvalues
eigenvalues <- eigen(varcov) #$values #%>% as.data.table; gc()
# 4. Plot the fraction of variance explained by each eigenvalue
var <- sum(eigenvalues$values)
var_fraction <- eigenvalues$values / var
var_fraction_plot <- barplot(var_fraction,
                             main = "Fraction of Variance Explained by Eigenvalues",
                             ylab = "Fraction of Variance", xlab = "Industry",
                             ylim = c(0, 0.7))
```



Problem 2

(a)

```r
# 2.(a). Explanation power of these three components
sum(var_fraction[1:3])
```

```
## [1] 0.6987876
```

Around 69.88% of the total variance is explained by these 3 factors.

(b)

```r
prcomp_result <- prcomp(ex.ret)
summ_prcomp_result <- summary(prcomp_result) #summ_prcomp_result$rotation
# Using prcomp results (did not find the PC1 ~ PC3 values from princomp result)
largest_pca <- as.data.frame(prcomp_result$rotation[, 1:3])
# monthly returns for each industry*loadings
factor_return <- as.matrix(ex.ret) %*% as.matrix(largest_pca)
facor_return_mean <- apply(factor_return, 2, mean)
```

Mean sample return to these 3 factor portfolios:

```r
facor_return_mean
```

```
##       PC1        PC2        PC3
##  3.7710625  0.2068955 -0.5032282
```

Standard deviation of these 3 factors portfolios

```r
prcomp_result$sdev[1:3]
```

```
## [1] 32.612279  9.679627  7.862954
```

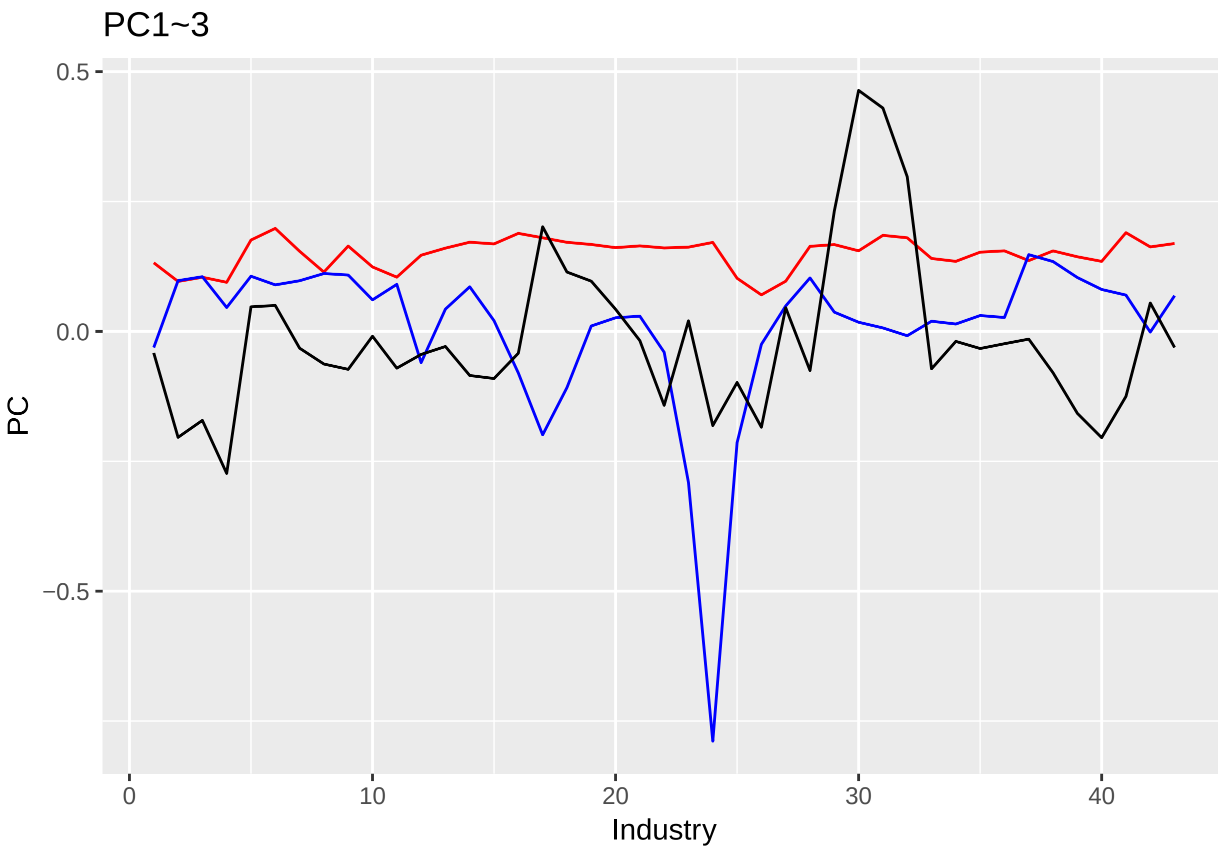
Correlation of these 3 factors portfolios

```r
cor(factor_return)
```

```
##               PC1           PC2           PC3
## PC1  1.000000e+00  7.685248e-15 -6.133312e-16
## PC2  7.685248e-15  1.000000e+00 -3.371765e-17
## PC3 -6.133312e-16 -3.371765e-17  1.000000e+00
```

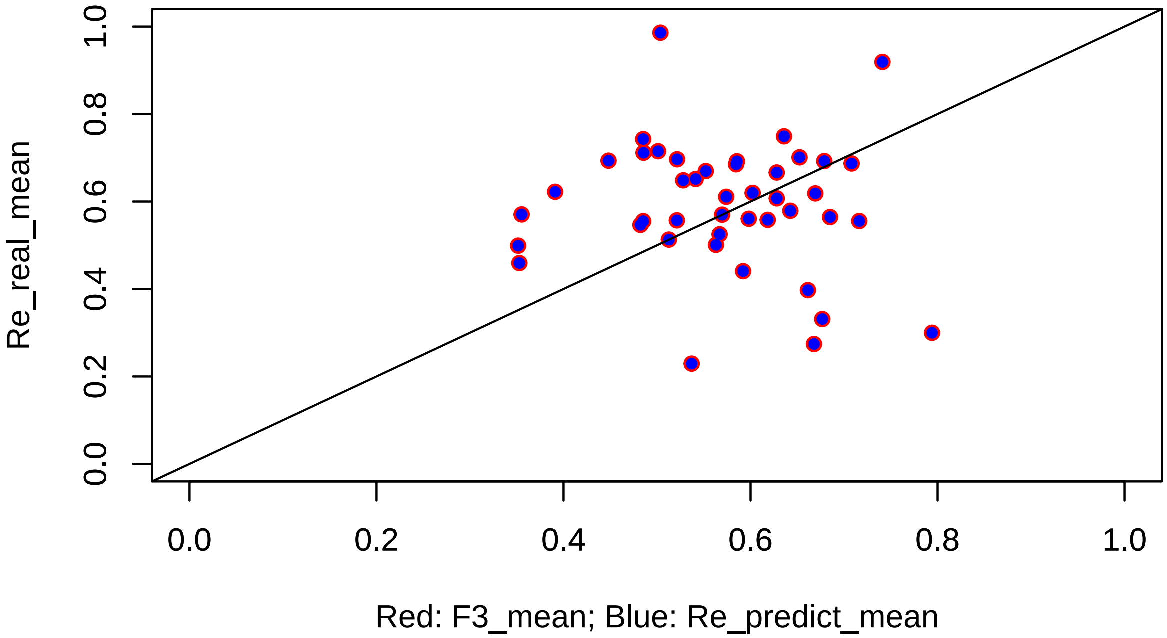
Factor loadings

```r
ggplot(data = largest_pca) +
  geom_line(aes(x = 1:length(largest_pca$PC1), y = PC1), color = "red") +
  geom_line(aes(x = 1:length(largest_pca$PC2), y = PC2), color = "blue") +
  geom_line(aes(x = 1:length(largest_pca$PC3), y = PC3), color = "black") +
  labs(title = "PC1~3", x = "Industry", y = "PC")
```

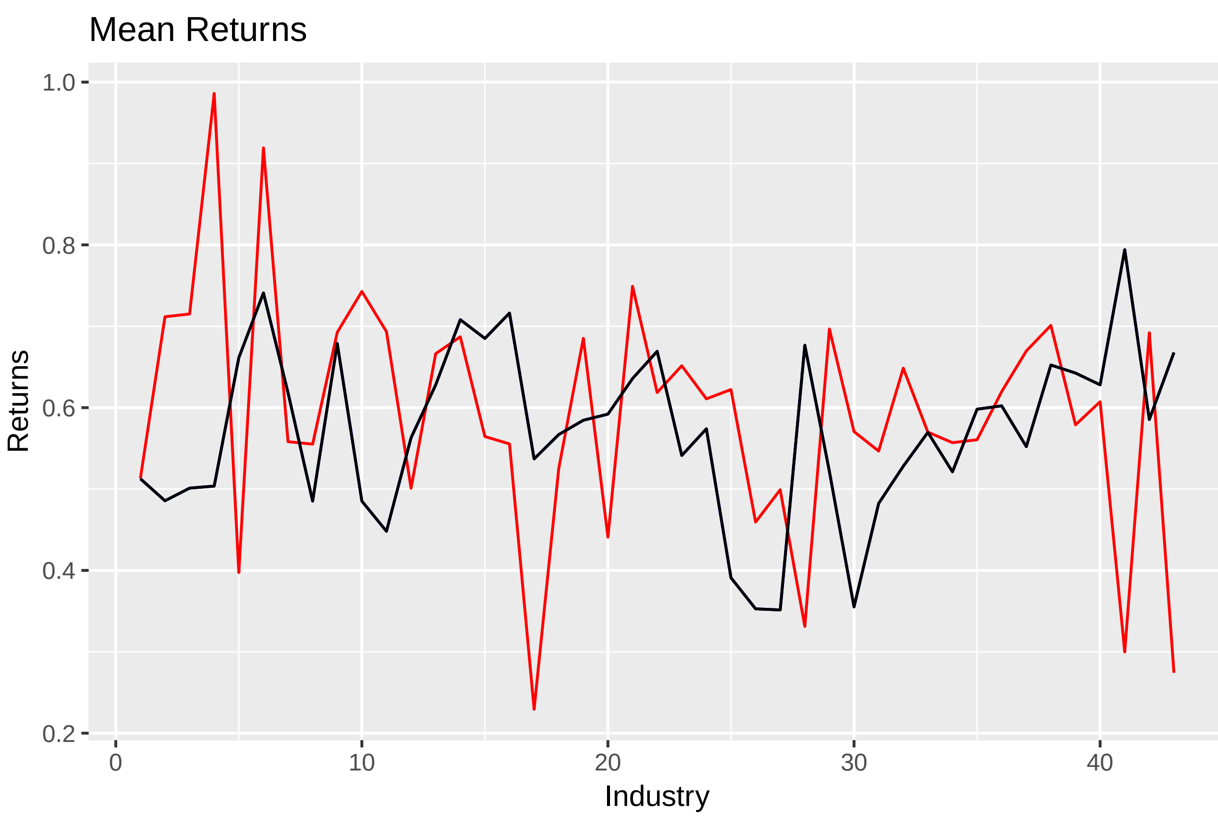


(c)

```
Re_real_mean <- apply(ex.ret, 2, mean) # Realized avg

y <- as.matrix(ex.ret)
X <- solve(prcomp_result$rotation) %*% t(y)
X1 <- as.matrix(X[1, ])
X2 <- as.matrix(X[2, ])
X3 <- as.matrix(X[3, ])
T <- length(y[, 1]) # 672 dim(X1%*%t(as.matrix(largest_pca[, 1])))
F3 <- X1%*%t(as.matrix(largest_pca[, 1])) + X2%*%t(as.matrix(largest_pca[, 2])) +
  X3%*%t(as.matrix(largest_pca[, 3]))
F3_mean <- apply(F3, 2, mean)

betas <- t(largest_pca) # betas for each industry #dim(largest_pca) -> 43by3; dim(betas) -> 3by43
Re_predict <- factor_return %*% betas # dim(factor_return) -> 672by3 x dim(betas) -> 3by43
Re_predict_mean <- apply(Re_predict, 2, mean) #dim(betas) dim(factor_return) -> 672by3

plot(x = F3_mean, y = Re_real_mean, pch = 19, col = "red", ylim = c(0, 1),
     xlim = c(0, 1), xlab = NA, ylab = NA)
abline(0, 1)
par(new=TRUE)
plot(x = Re_predict_mean, y = Re_real_mean, pch = 20, col = "blue",
     ylim = c(0, 1), xlim = c(0, 1), xlab="Red: F3_mean; Blue: Re_predict_mean")
abline(0, 1)
```




```r
final <- cbind(Re_real_mean, Re_predict_mean, F3_mean)
ggplot(data = as.data.frame(final)) +
  geom_line(aes(x = 1:length(final[, 1]), y = Re_real_mean), color = "red") +
  geom_line(aes(x = 1:length(final[, 2]), y = Re_predict_mean), color = "blue") +
  geom_line(aes(x = 1:length(final[, 3]), y = F3_mean), color = "black") +
  labs(title = "Mean Returns", x = "Industry", y = "Returns")
```



(d)

```r
rsquard_1 <- 1 - var(Re_real_mean - F3_mean) / var(Re_real_mean)
rsquard_2 <- 1 - var(Re_real_mean - Re_predict_mean) / var(Re_real_mean)

rsquard_1
```

```
## [1] -0.595244
```

```r
rsquard_2
```

```
## [1] -0.595244
```

Problem 3

```r
industry_25 <- read.csv("25_Portfolios_5x5_vw.csv", header = T) %>% as.data.table; gc()
```

```
##           used (Mb) gc trigger (Mb) limit (Mb) max used (Mb)
## Ncells  829836 44.4    1621690 86.7         NA  1183282 63.2
## Vcells 1925118 14.7    8388608 64.0      16384  3326812 25.4
```

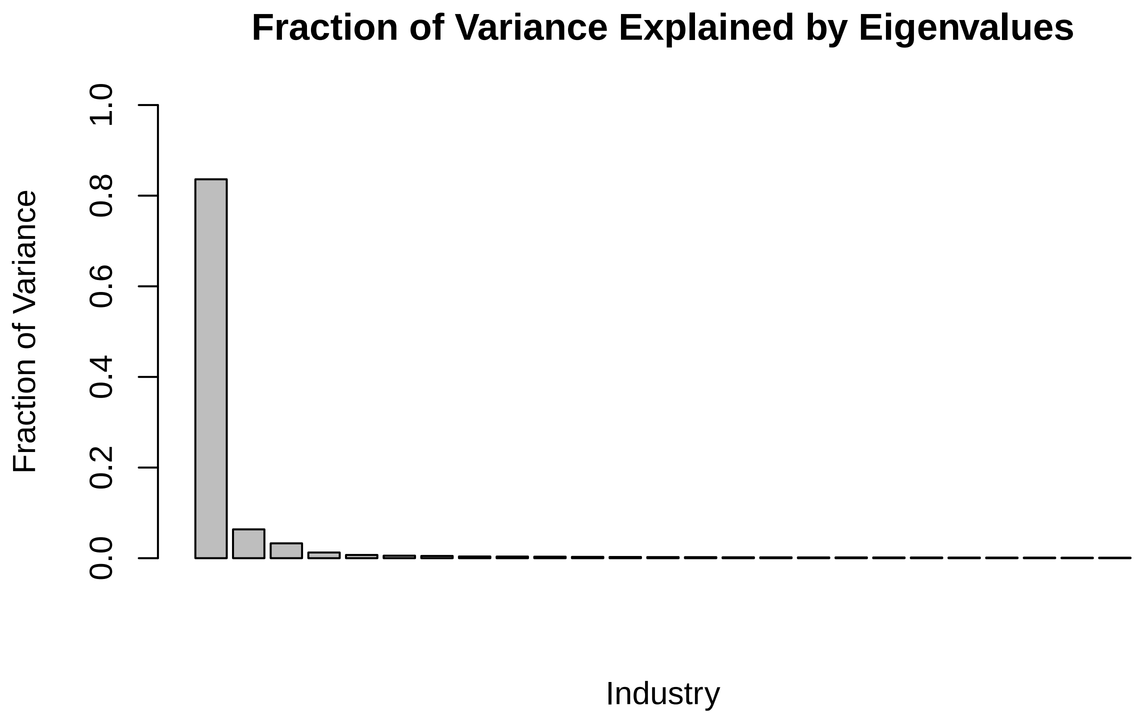
```r
# Data Cleaning and Manipulation
# 1. Subset the data: year 1960~2015
# get year info of each row
industry_25$year <- as.numeric(substr(industry_25$X, 1, 4))
# year condition
industry_2 <- industry_25[ (industry_25$year >= 1960) & (industry_25$year < 2016), ]
# 2. Remove columns if it holds -99.99 or missing values
industry_2[industry_2 == -99.99] <- NA # changing -99.99 values to NA
industry_2[industry_2 == -999] <- NA
# removing columns with NAs
industry_2_filtered <- industry_2 %>% select_if( ~ !any(is.na(.)) )
# 3. Merge industry data and Rf from Fama-French data
data_2 <- merge(industry_2_filtered, famafrench, by = "X")
```

(a)

```r
# 1. Calculate the excess return of each industry
ex.ret_2 <- (data_2[, 2:26] - data$RF) #/100 # change into percentile
dim(ex.ret_2)
```

```
## [1] 672  25
```

```r
# 2. Get the var-cov matrix
varcov_2 <- var(ex.ret_2)
# 3. Get the eigenvalues
eigenvalues_2 <- eigen(varcov_2) #$values #%>% as.data.table; gc()
# 4. Plot the fraction of variance explained by each eigenvalue
var_2 <- sum(eigenvalues_2$values)
var_fraction_2 <- eigenvalues_2$values / var_2
var_fraction_plot_2 <- barplot(var_fraction_2, main = "Fraction of Variance Explained by Eigenvalues",
                               ylab = "Fraction of Variance", xlab = "Industry",
                               ylim = c(0, 1))
```



(b)

```
summary(prcomp(ex.ret_2))
```

```
## Importance of components:
##                            PC1     PC2     PC3     PC4     PC5     PC6
## Standard deviation     25.7406 7.09882 5.09696 3.15026 2.37675 2.09334
## Proportion of Variance  0.8361 0.06359 0.03278 0.01252 0.00713 0.00553
## Cumulative Proportion   0.8361 0.89972 0.93251 0.94503 0.95216 0.95769
##                            PC7     PC8     PC9    PC10   PC11    PC12
## Standard deviation     1.98281 1.72216 1.68682 1.62182 1.5160 1.44661
## Proportion of Variance 0.00496 0.00374 0.00359 0.00332 0.0029 0.00264
## Cumulative Proportion  0.96265 0.96639 0.96998 0.97330 0.9762 0.97884
##                           PC13    PC14    PC15    PC16    PC17   PC18
## Standard deviation     1.37457 1.34548 1.26704 1.22476 1.17035 1.1599
## Proportion of Variance 0.00238 0.00228 0.00203 0.00189 0.00173 0.0017
## Cumulative Proportion  0.98123 0.98351 0.98554 0.98743 0.98916 0.9909
##                           PC19    PC20    PC21   PC22    PC23    PC24
## Standard deviation     1.12064 1.09379 1.04021 1.0145 0.99545 0.92778
## Proportion of Variance 0.00158 0.00151 0.00137 0.0013 0.00125 0.00109
## Cumulative Proportion  0.99244 0.99395 0.99532 0.9966 0.99787 0.99895
##                           PC25
## Standard deviation     0.91140
## Proportion of Variance 0.00105
## Cumulative Proportion  1.00000
```

According to the plot in (a), PC1 covers the largest fraction of variance, following are PC2, PC3, and PC4. In that case, 4 factors needed to explain average returns to the 25 F-F portfolios