ETF5500 Group Assignment

Girika, Jyovika Aswale, Richa Anghan, Sia Chawla, Siddhi Ajit Jadhav

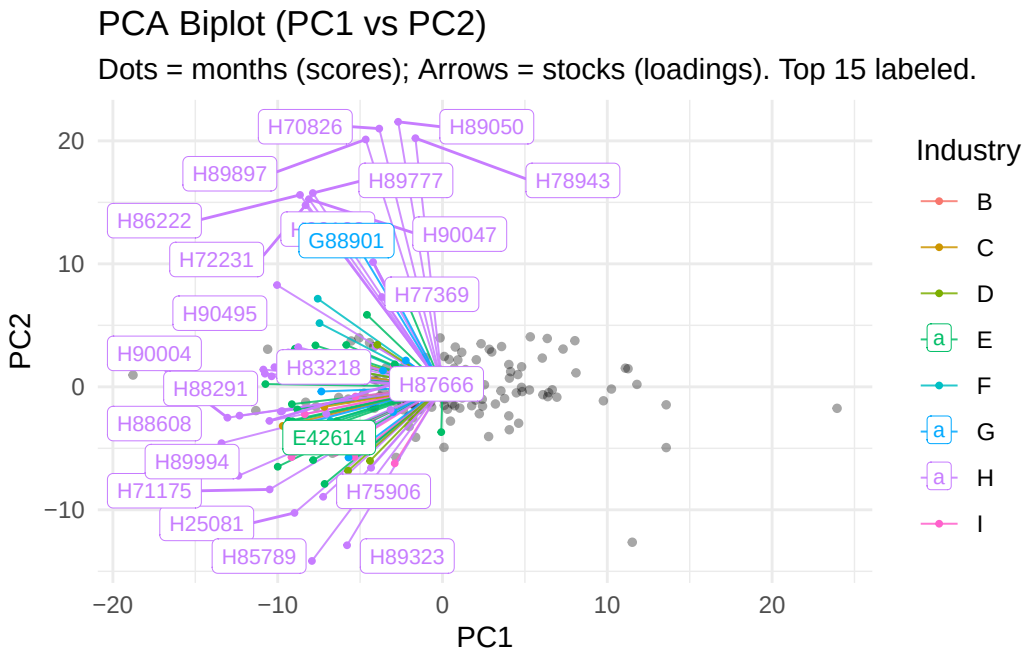
```
library(tidyverse)
library(zoo)
library(ggplot2)
library(ggrepel)
```

```
market_df <- read.csv(here::here("data/Market.csv"))
sample_df <- read.csv(here::here("data/SampleK.csv"))
```

```
# Make a numeric data matrix with all stock columns (drop Date)
stopifnot("Date" %in% names(sample_df))
X <- sample_df[setdiff(names(sample_df), "Date")]

# Make sure everything is numeric (silently convert if needed)
X <- X |> mutate(across(everything(), ~ suppressWarnings(as.numeric(.x))))

# If any column became all-NA, drop it (rare safety)
all_na_cols <- names(X)[sapply(X, function(v) all(is.na(v)))]
if (length(all_na_cols) > 0) X <- X[ , setdiff(names(X), all_na_cols), drop=FALSE]

# Run PCA on scaled data
pca <- prcomp(X, center = TRUE, scale. = TRUE)
```

```
# Build a clean biplot (PC1 vs PC2)
scores <- as_tibble(pca$x[, 1:2]); names(scores) <- c("PC1","PC2")
loads  <- as_tibble(pca$rotation[, 1:2], rownames = "id"); names(loads)[2:3] <- c("PC1","PC2
loads$industry <- substr(loads$id, 1, 1)

# Scale arrows to fit nicely
rng_scores  <- max(abs(c(scores$PC1, scores$PC2)), na.rm = TRUE)
rng_loads   <- max(abs(c(loads$PC1, loads$PC2)),  na.rm = TRUE)
arrow_scale <- ifelse(rng_loads == 0, 1, 0.9 * rng_scores / rng_loads)
```

```r
loads <- loads |>
  mutate(PC1_s = PC1 * arrow_scale,
         PC2_s = PC2 * arrow_scale,
         length = sqrt(PC1^2 + PC2^2))

# Label only the 15 longest arrows so it's not messy
label_ids <- loads |>
  arrange(desc(length)) |>
  slice_head(n = 24) |>
  pull(id)

ggplot() +
  geom_point(data = scores, aes(PC1, PC2), alpha = 0.35, size = 1) +
  geom_segment(data = loads,
               aes(x = 0, y = 0, xend = PC1_s, yend = PC2_s, colour = industry),
               linewidth = 0.35, alpha = 0.85) +
  geom_point(data = loads, aes(PC1_s, PC2_s, colour = industry), size = 0.7) +
  geom_label_repel(data = loads |> filter(id %in% label_ids),
                   aes(PC1_s, PC2_s, label = id, colour = industry),
                   size = 2.7, label.size = 0.15, max.overlaps = Inf, seed = 7) +
  labs(title = "PCA Biplot (PC1 vs PC2)",
       subtitle = "Dots = months (scores); Arrows = stocks (loadings). Top 15 labeled.",
       x = "PC1", y = "PC2", colour = "Industry") +
  theme_minimal()
```




```
colSums(is.na(sample_df))
```

```
  Date E72119 H89777 F85464 H25081 H90004 F24440 G54244 H90047 H75846 B89546
     0      0      0      0      0      0      0      0      0      0      0
H87666 H70121 H75906 I83509 F88857 H70826 E20204 F81043 E11600 E90081 H88608
     0      0      0      0      0      0      0      0      0      0      0
C89731 H89262 E89070 E10382 I65752 H86222 E37381 H75811 I35124 H85414 E78705
     0      0      0      0      0      0      0      0      0      0      0
E83910 H77369 H90108 E89216 H90495 E37402 I87510 I86122 G82171 E42614 H83683
     0      0      0      0      0      0      0      0      0      0      0
I75912 G89866 H85789 G82777 D34367 E92583 E87034 D11308 H50702 E86242 H72231
     0      0      0      0      0      0      0      0      0      0      0
H88290 I79698 E30648 F12758 E87268 I83303 H71175 H41187 H89897 H77120 D84010
     0      0      0      0      0      0      0      0      0      0      0
E79108 C63132 I89927 G81481 D88660 H89050 G88901 H84767 H83218 E25452 E82651
     0      0      0      0      0      0      0      0      0      0      0
H90082 I74500 H89995 E88839 F29647 G76360 H78943 C76279 H89994 G77584 H89323
     0      0      0      0      0      0      0      0      0      0      0
D81084 H80779 H88291 E11825
     0      0      0      0
```

```r
range(sample_df$Date)
```

```
[1] "Y2005M1" "Y2019M9"
```

```r
tibble(stock = names(sample_df)[-1]) %>%
  mutate(industry = substr(stock,1,1)) %>%
  count(industry)
```

```
# A tibble: 8 x 2
  industry     n
  <chr>    <int>
1 B            1
2 C            3
3 D            5
4 E           22
5 F            6
6 G            8
7 H           36
8 I           10
```

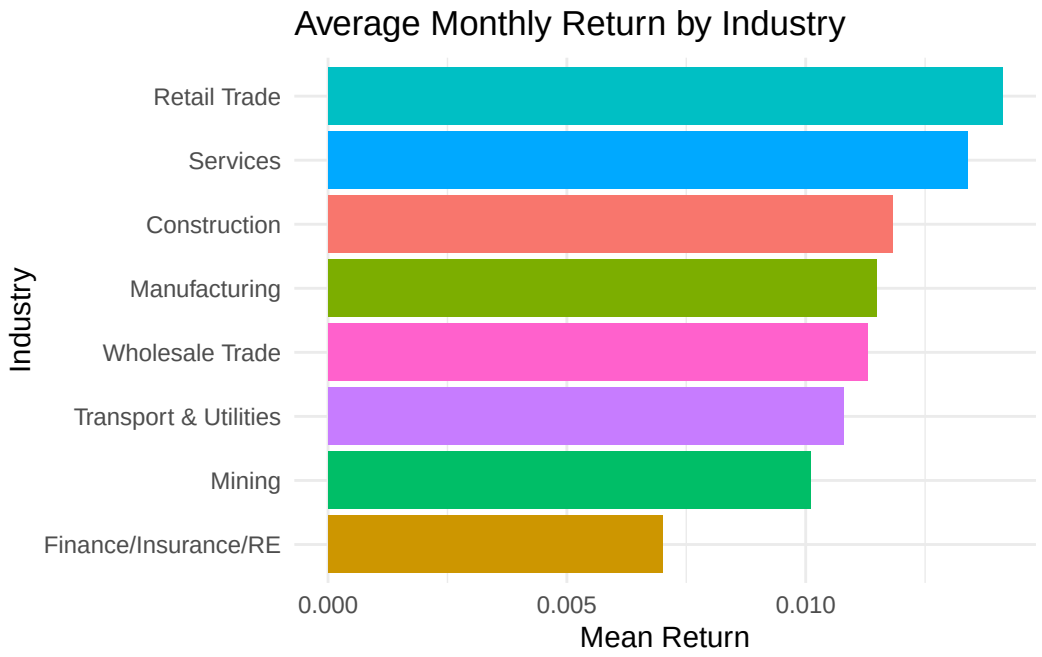
Cleaning

```r
sample_df <- sample_df %>%
  mutate(Date = as.yearmon(str_remove(Date, "Y"), format = "%Y M%m"))

market_df <- market_df %>%
  mutate(Date = as.yearmon(str_remove(Date, "Y"), format = "%Y M%m"))
```

EDA

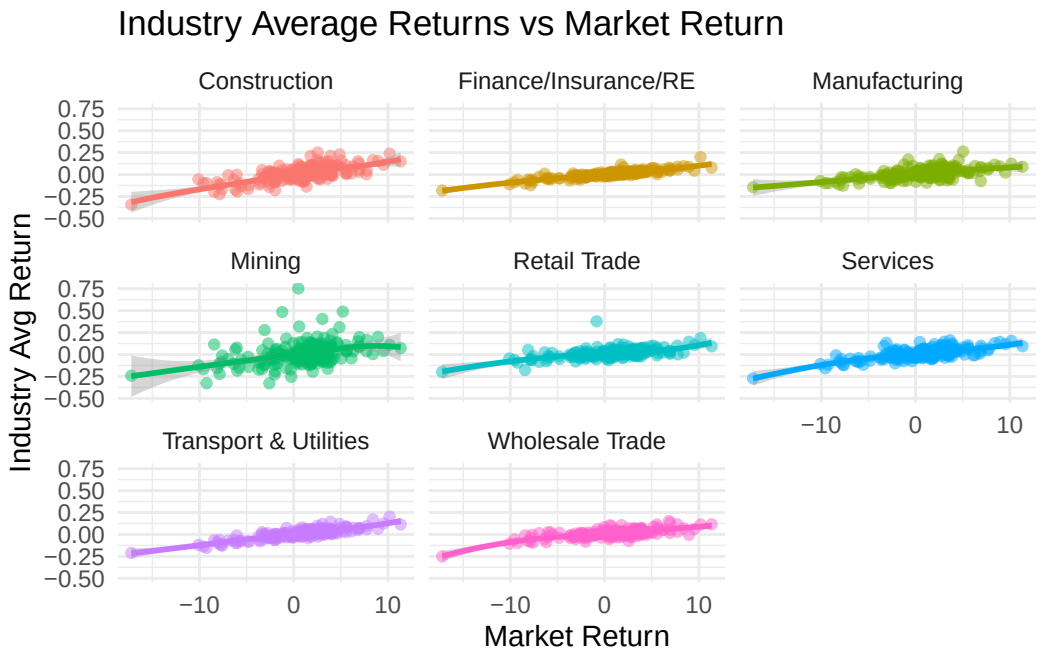
```r
stocks_long <- sample_df %>%
  pivot_longer(-Date, names_to = "stock", values_to = "ret") %>%
  mutate(industry = substr(stock,1,1),
         industry_name = recode(industry,
           B = "Mining",
           C = "Construction",
           D = "Manufacturing",
```

```
          E = "Transport & Utilities",
          F = "Wholesale Trade",
          G = "Retail Trade",
          H = "Finance/Insurance/RE",
          I = "Services"
        ))

industry_summary <- stocks_long %>%
  group_by(industry_name) %>%
  summarise(
    mean_return = mean(ret, na.rm = TRUE),
    sd_return   = sd(ret, na.rm = TRUE),
    .groups = "drop"
  )

ggplot(industry_summary, aes(x = reorder(industry_name, mean_return),
                             y = mean_return, fill = industry_name)) +
  geom_col() +
  coord_flip() +
  labs(title = "Average Monthly Return by Industry",
       x = "Industry", y = "Mean Return") +
  theme_minimal() +
  theme(legend.position = "none")
```



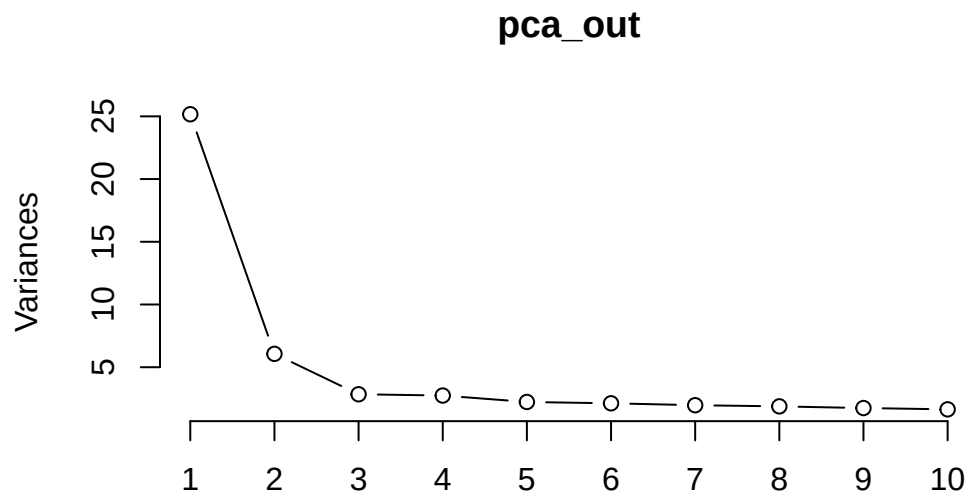

```
sampleXmarket <- stocks_long %>%
  left_join(market_df, by = "Date")

industry_ts <- sampleXmarket %>%
  group_by(Date, industry_name) %>%
  summarise(
    avg_ret = mean(ret, na.rm = TRUE),
    MarketReturn = first(MarketReturn),
    .groups = "drop"
  )

ggplot(industry_ts, aes(x = MarketReturn, y = avg_ret, color = industry_name)) +
  geom_point(alpha = 0.5) +
  geom_smooth() +
  facet_wrap(~ industry_name, scales = "fixed") +
  labs(
    title = "Industry Average Returns vs Market Return",
    x = "Market Return",
    y = "Industry Avg Return"
  ) +
  theme_minimal() +
  theme(legend.position = "none")
```

```
`geom_smooth()` using method = 'loess' and formula = 'y ~ x'
```



##PCA

```
pca_out <- prcomp(sample_df[,-1], scale. = TRUE)
plot(pca_out, type = "line")
```

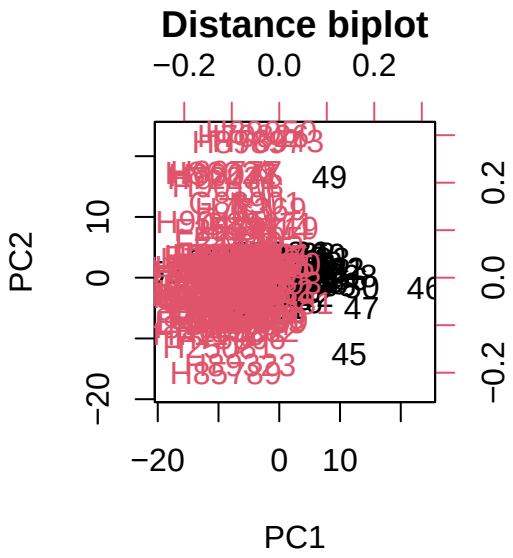



```
summary(pca_out)
```

```
Importance of components:
                          PC1     PC2     PC3     PC4    PC5     PC6     PC7
Standard deviation     5.0167 2.46261 1.68600 1.65579 1.4931 1.45709 1.40371
Proportion of Variance 0.2766 0.06664 0.03124 0.03013 0.0245 0.02333 0.02165
Cumulative Proportion  0.2766 0.34321 0.37445 0.40457 0.4291 0.45240 0.47406
                           PC8     PC9    PC10    PC11   PC12    PC13    PC14
Standard deviation     1.37052 1.32069 1.28059 1.25960 1.2327 1.21921 1.15610
Proportion of Variance 0.02064 0.01917 0.01802 0.01744 0.0167 0.01633 0.01469
Cumulative Proportion  0.49470 0.51386 0.53188 0.54932 0.5660 0.58235 0.59704
                          PC15    PC16    PC17    PC18    PC19    PC20    PC21
Standard deviation     1.13174 1.12637 1.11832 1.10056 1.09625 1.07156 1.06565
Proportion of Variance 0.01408 0.01394 0.01374 0.01331 0.01321 0.01262 0.01248
Cumulative Proportion  0.61112 0.62506 0.63880 0.65211 0.66532 0.67793 0.69041
                          PC22    PC23    PC24    PC25    PC26    PC27    PC28
Standard deviation     1.06357 1.03476 1.00327 0.98757 0.97581 0.96237 0.96181
Proportion of Variance 0.01243 0.01177 0.01106 0.01072 0.01046 0.01018 0.01017
Cumulative Proportion  0.70284 0.71461 0.72567 0.73639 0.74685 0.75703 0.76720
                          PC29    PC30    PC31    PC32    PC33    PC34    PC35
Standard deviation     0.92603 0.92120 0.91286 0.91120 0.89214 0.85622 0.84145
Proportion of Variance 0.00942 0.00933 0.00916 0.00912 0.00875 0.00806 0.00778
```

```
Cumulative Proportion  0.77662 0.78595 0.79510 0.80423 0.81297 0.82103 0.82881
                          PC36    PC37   PC38    PC39    PC40    PC41    PC42
Standard deviation     0.83382 0.81384 0.7926 0.78908 0.76726 0.75024 0.73926
Proportion of Variance 0.00764 0.00728 0.0069 0.00684 0.00647 0.00619 0.00601
Cumulative Proportion  0.83645 0.84373 0.8506 0.85747 0.86394 0.87013 0.87613
                          PC43    PC44    PC45   PC46    PC47    PC48    PC49
Standard deviation     0.73185 0.72430 0.70936 0.7009 0.68718 0.66947 0.66818
Proportion of Variance 0.00589 0.00576 0.00553 0.0054 0.00519 0.00493 0.00491
Cumulative Proportion  0.88202 0.88778 0.89331 0.8987 0.90390 0.90883 0.91373
                          PC50    PC51    PC52    PC53    PC54    PC55    PC56
Standard deviation     0.65844 0.63375 0.62354 0.61361 0.59929 0.58477 0.57658
Proportion of Variance 0.00476 0.00441 0.00427 0.00414 0.00395 0.00376 0.00365
Cumulative Proportion  0.91850 0.92291 0.92718 0.93132 0.93527 0.93902 0.94268
                          PC57    PC58    PC59   PC60    PC61    PC62    PC63
Standard deviation     0.56827 0.56369 0.52958 0.5227 0.51811 0.48441 0.48087
Proportion of Variance 0.00355 0.00349 0.00308 0.0030 0.00295 0.00258 0.00254
Cumulative Proportion  0.94623 0.94972 0.95280 0.9558 0.95875 0.96133 0.96387
                          PC64    PC65    PC66    PC67    PC68    PC69    PC70
Standard deviation     0.47224 0.46386 0.45932 0.45122 0.43903 0.42216 0.41068
Proportion of Variance 0.00245 0.00236 0.00232 0.00224 0.00212 0.00196 0.00185
Cumulative Proportion  0.96632 0.96869 0.97101 0.97324 0.97536 0.97732 0.97917
                          PC71    PC72    PC73   PC74   PC75    PC76    PC77
Standard deviation     0.40240 0.39523 0.39392 0.3819 0.3568 0.35234 0.33959
Proportion of Variance 0.00178 0.00172 0.00171 0.0016 0.0014 0.00136 0.00127
Cumulative Proportion  0.98095 0.98267 0.98437 0.9860 0.9874 0.98874 0.99001
                          PC78    PC79    PC80    PC81   PC82    PC83    PC84
Standard deviation     0.33195 0.31534 0.30869 0.28930 0.2863 0.28109 0.27320
Proportion of Variance 0.00121 0.00109 0.00105 0.00092 0.0009 0.00087 0.00082
Cumulative Proportion  0.99122 0.99231 0.99336 0.99428 0.9952 0.99605 0.99687
                          PC85   PC86    PC87    PC88    PC89    PC90    PC91
Standard deviation     0.24680 0.2345 0.22679 0.22371 0.19455 0.13884 0.10337
Proportion of Variance 0.00067 0.0006 0.00057 0.00055 0.00042 0.00021 0.00012
Cumulative Proportion  0.99754 0.9981 0.99870 0.99925 0.99967 0.99988 1.00000
```

```r
biplot(pca_out, scale = 0, main = "Distance biplot")
```



Question 1

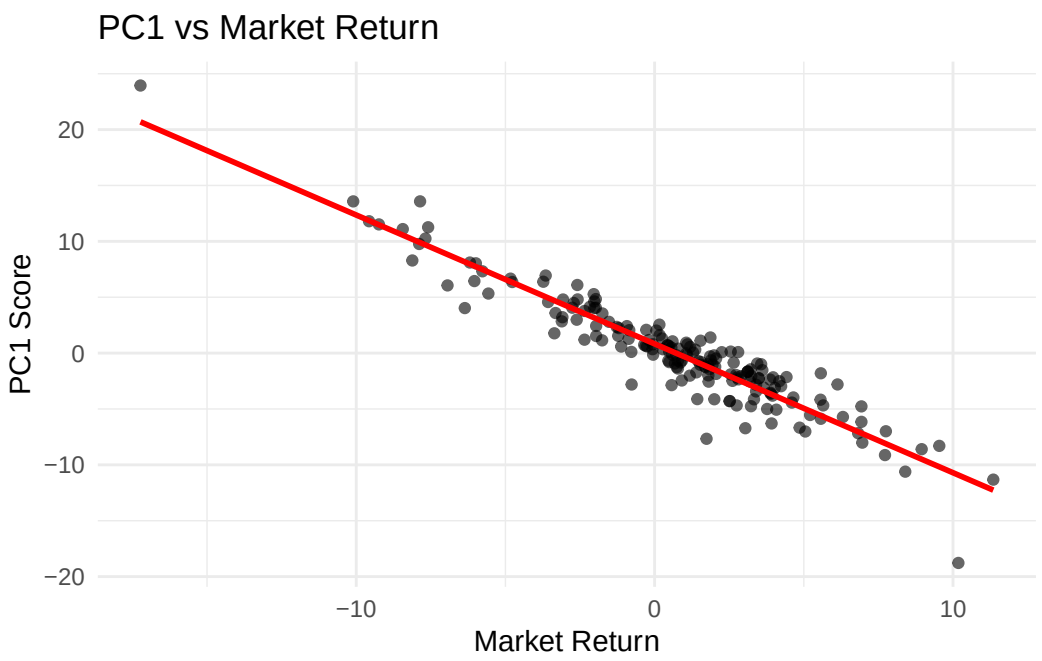
```
pc_scores <- as.data.frame(pca_out$x)
pc1_scores <- pc_scores$PC1

pca_vs_market <- tibble(
  Date = sample_df$Date,
  PC1 = pc1_scores
) %>%
  left_join(market_df, by = "Date")

cor(pca_vs_market$PC1, pca_vs_market$MarketReturn, use = "pairwise.complete.obs")
```

```
[1] -0.9456026
```

```
ggplot(pca_vs_market, aes(x = MarketReturn, y = PC1)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE, color = "red") +
  labs(
    title = "PC1 vs Market Return",
```

```
    x = "Market Return",
    y = "PC1 Score"
  ) +
  theme_minimal()
```

```
`geom_smooth()` using formula = 'y ~ x'
```



References

Date formating using zoo in R. (2017, January). [Online post]. https://stackoverflow.com/questions/41588737/date-formating-using-zoo-in-r?rq=3