Financial Econometrics - Group Assignment

Group members:

- Baldoni Chiara
- Pinna Paola
- Torella Marta

EDA

```
data <- read.csv("FIN 36182-Industry_Portfolios.CSV")
head(data)
sum(is.na(data))
anyDuplicated(data)

data[, 2:6] <- data[, 2:6] / 100 ## Convert returns to decimal form as asked
head(data)
```

Task 1

1. Report the arithmetic mean of the returns for each of the five industries over the entire sample.

```
monthly_means <- colMeans(data[, c("Cnsmr", "Manuf", "HiTec", "Hlth", "Other")])
annualized_means <- monthly_means * 12
annualized_means <- round(annualized_means, 4)
annualized_means
```

2. Report the standard deviation of the returns for each of the five industries over the entire sample.

```
monthly_std <- apply(data[, c("Cnsmr", "Manuf", "HiTec", "Hlth", "Other")], 2, sd)
annualized_std <- monthly_std * sqrt(12)
annualized_std <- round(annualized_std, 4)
annualized_std
```

3. Report the Sharpe ratio of each industry.

```
sharpe_ratios <- annualized_means / annualized_std
sharpe_ratios <- round(sharpe_ratios, 4)
sharpe_ratios
```

```
results_table1 <- data.frame(
  Mean_Returns = annualized_means,
  Std_Returns = annualized_std,
  Sharpe_Ratio = sharpe_ratios
```

```
 )
results_table1
```

5. Provide a table (5×5) with the sample correlation between the returns of the five industries.

```
correlation_matrix <- cor(data[, c("Cnsmr", "Manuf", "HiTec", "Hlth", "Other")])
correlation_matrix <- round(correlation_matrix, 4)
correlation_df <- as.data.frame(correlation_matrix)
correlation_df
```

```
library(corrplot)
corrplot(correlation_matrix, method = "color", type = "upper",
         tl.col = "black", tl.srt = 45,
         addCoef.col = "black",
         number.cex = 0.8,
         number.digits = 2)
```

6. Construct a time series of the simple, non-cumulative returns of a portfolio where capital is allocated equally across the first four industries (excluding Other). Report the arithmetic mean, standard deviation and Sharpe.

```
portfolio_returns <- rowMeans(data[, c("Cnsmr", "Manuf", "HiTec", "Hlth")])
data$Portfolio_Returns <- portfolio_returns
mean_portfolio <- mean(portfolio_returns)
sd_portfolio <- sd(portfolio_returns)

annualized_mean_portfolio <- mean_portfolio * 12
annualized_sd_portfolio <- sd_portfolio * sqrt(12)
sharpe_ratio_portfolio <- annualized_mean_portfolio / annualized_sd_portfolio

portfolio_results <- data.frame(
  Metric = c("Annualized Mean Return", "Annualized Standard Deviation", "Sharpe Ratio"),
  Value = round(c(annualized_mean_portfolio, annualized_sd_portfolio, sharpe_ratio_portfolio), 4)
)
print(portfolio_results)
```

```
library(ggplot2)
data$Date <- as.character(data$Date)
data$Date <- paste0(data$Date, "01")
data$Date <- as.Date(data$Date, format = "%Y%m%d")

plot_data <- data.frame(Date = data$Date, Portfolio_Returns = data$Portfolio_Returns)
ggplot(plot_data, aes(x = Date, y = Portfolio_Returns)) +
  geom_line(color = "blue") +
  labs(title = "Portfolio Returns Over Time",
       x = "Date",
       y = "Portfolio Returns") +
  theme_minimal()
```

Task 2

In this task you will treat the portfolio you computed in Task 1, point (6), as the market portfolio, denote its returns as Rm, and will estimate and interpret beta and alpha coefficients in the context of the CAPM.

1. Compute the kurtosis and skeweness of Rm

```
library(moments)

skewness_value <- skewness(data$Portfolio_Returns, na.rm = TRUE)
kurtosis_value <- kurtosis(data$Portfolio_Returns, na.rm = TRUE)

kurtosis_value <- round(kurtosis_value, 4)
skewness_value <- round(skewness_value, 4)

portfolio_results2 <- data.frame(
  Metric = c("Kurtosis", "Skewness"),
  Value = c(kurtosis_value, skewness_value)
)
print(portfolio_results2)
```

```
library(ggplot2)
ggplot(data = data.frame(returns = data$Portfolio_Returns), aes(x = returns)) +
  geom_histogram(aes(y = ..density..), bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  geom_density(color = "red", size = 1) +
  stat_function(fun = dnorm,
                args = list(mean = mean(portfolio_returns, na.rm = TRUE),
                            sd = sd(portfolio_returns, na.rm = TRUE)),
                color = "green", linetype = "dashed", size = 1) +
  labs(title = "Distribution of Portfolio Returns with Normal Distribution Overlay",
       x = "Portfolio Returns",
       y = "Density") +
  theme_minimal()
```

3. Repeat point (1), but eliminating the first 70 years of data (i.e. from 199706).

```
data_filtered <- data[data$Date >= as.Date("1997-06-01"), ]
portfolio_returns_filtered <- data_filtered$Portfolio_Returns

skewness_filtered <- skewness(portfolio_returns_filtered, na.rm = TRUE)
kurtosis_filtered <- kurtosis(portfolio_returns_filtered, na.rm = TRUE)

skewness_filtered <- round(skewness_filtered, 4)
kurtosis_filtered <- round(kurtosis_filtered, 4)

portfolio_results3 <- data.frame(
  Metric = c("Kurtosis", "Skewness"),
  Value = c(kurtosis_filtered, skewness_filtered)
)
print(portfolio_results3)
```

```
ggplot(data = data.frame(returns = portfolio_returns_filtered), aes(x = returns)) +
  geom_histogram(aes(y = after_stat(density)), bins = 30, fill = "blue", color = "black", alpha = 0.7) +
  geom_density(color = "red", linewidth = 1) +
  stat_function(fun = dnorm,
                args = list(mean = mean(portfolio_returns_filtered, na.rm = TRUE),
                            sd = sd(portfolio_returns_filtered, na.rm = TRUE)),
                color = "green", linetype = "dashed", linewidth = 1) +
  labs(title = "Distribution of Filtered Portfolio Returns with Normal Distribution Overlay",
```

```
       x = "Portfolio Returns",
       y = "Density") +
  theme_minimal()
```

4. Compute and report the covariance of the first four industries with Rm

```
cov_Cnsmr_Rm <- cov(data$Cnsmr, data$Portfolio_Returns, use = "complete.obs")
cov_Manuf_Rm <- cov(data$Manuf, data$Portfolio_Returns, use = "complete.obs")
cov_HiTec_Rm <- cov(data$HiTec, data$Portfolio_Returns, use = "complete.obs")
cov_Hlth_Rm <- cov(data$Hlth, data$Portfolio_Returns, use = "complete.obs")

cov_Cnsmr_Rm <- round(cov_Cnsmr_Rm, 4)
cov_Manuf_Rm <- round(cov_Manuf_Rm, 4)
cov_HiTec_Rm <- round(cov_HiTec_Rm, 4)
cov_Hlth_Rm <- round(cov_Hlth_Rm, 4)

covariances <- data.frame(
  Industry = c("Cnsmr", "Manuf", "HiTec", "Hlth"),
  Covariance_with_Rm = c(cov_Cnsmr_Rm, cov_Manuf_Rm, cov_HiTec_Rm, cov_Hlth_Rm)
)

print(covariances)
```

5. Use the results obtained so far to compute the beta values for the first four industries for the full sample

```
# calculate the variance of the portfolio returns (Rm)
var_Rm <- var(data$Portfolio_Returns, use = "complete.obs")
var_Rm <- round(var_Rm, 4)

# calculate the beta values for each industry
beta_Cnsmr <- cov_Cnsmr_Rm / var_Rm
beta_Manuf <- cov_Manuf_Rm / var_Rm
beta_HiTec <- cov_HiTec_Rm / var_Rm
beta_Hlth <- cov_Hlth_Rm / var_Rm

beta_Cnsmr <- round(beta_Cnsmr, 4)
beta_Manuf <- round(beta_Manuf, 4)
beta_HiTec <- round(beta_HiTec, 4)
beta_Hlth <- round(beta_Hlth, 4)

beta_values <- data.frame(
  Industry = c("Cnsmr", "Manuf", "HiTec", "Hlth"),
  Beta = c(beta_Cnsmr, beta_Manuf, beta_HiTec, beta_Hlth)
)
print(beta_values)
```

6. Compute the beta values for the first four industries for the sample starting from 199706. Briefly comment on how results compare with those in point (5)

```
var_Rm_filtered <- var(portfolio_returns_filtered, use = "complete.obs")
```

```
beta_Cnsmr_filtered <- cov(data_filtered$Cnsmr, data_filtered$Portfolio_Returns, use = "complete.obs") /
beta_Manuf_filtered <- cov(data_filtered$Manuf, data_filtered$Portfolio_Returns, use = "complete.obs") /
beta_HiTec_filtered <- cov(data_filtered$HiTec, data_filtered$Portfolio_Returns, use = "complete.obs") /
beta_Hlth_filtered <- cov(data_filtered$Hlth, data_filtered$Portfolio_Returns, use = "complete.obs") / v

beta_Cnsmr_filtered <- round(beta_Cnsmr_filtered, 4)
beta_Manuf_filtered <- round(beta_Manuf_filtered, 4)
beta_HiTec_filtered <- round(beta_HiTec_filtered, 4)
beta_Hlth_filtered <- round(beta_Hlth_filtered, 4)

beta_values_filtered <- data.frame(
  Industry = c("Cnsmr", "Manuf", "HiTec", "Hlth"),
  Beta_Filtered = c(beta_Cnsmr_filtered, beta_Manuf_filtered, beta_HiTec_filtered, beta_Hlth_filtered)
)
print(beta_values_filtered)
```

```
comparison <- data.frame(
  Industry = c("Cnsmr", "Manuf", "HiTec", "Hlth"),
  Beta_Full_Sample = c(beta_Cnsmr, beta_Manuf, beta_HiTec, beta_Hlth),  # point 5
  Beta_Filtered = c(beta_Cnsmr_filtered, beta_Manuf_filtered, beta_HiTec_filtered, beta_Hlth_filtered)
)
print(comparison)
```

7. Assuming a risk-free rate of 5%, compute Jensen's alpha for each of the first four industries (on the full sample). Report the alpha in percentage terms.

```
annualized_returns_industries <- annualized_means
risk_free_rate <- 0.05

alpha_Cnsmr <- (annualized_returns_industries["Cnsmr"] - (risk_free_rate + beta_Cnsmr * (annualized_mean
alpha_Manuf <- (annualized_returns_industries["Manuf"] - (risk_free_rate + beta_Manuf * (annualized_mean
alpha_HiTec <- (annualized_returns_industries["HiTec"] - (risk_free_rate + beta_HiTec * (annualized_mean
alpha_Hlth <- (annualized_returns_industries["Hlth"] - (risk_free_rate + beta_Hlth * (annualized_mean_po

jensens_alpha <- data.frame(
  Industry = c("Cnsmr", "Manuf", "HiTec", "Hlth"),
  Alpha = round(c(alpha_Cnsmr, alpha_Manuf, alpha_HiTec, alpha_Hlth), 4)
)
print(jensens_alpha)
```

Task 3

Use the lm( ) function to run a few regressions:

1. Regress Rm(t) on an intercept and on Rm(t−1). Report estimates and t-statistics. Briefly interpret the results

```
library(dplyr)
data <- data %>%
  mutate(Rm_lag = lag(Portfolio_Returns, 1))
head(data)
```

```
model <- lm(Portfolio_Returns ~ Rm_lag, data = data)
summary(model)
```

```
library(ggplot2)
ggplot(data, aes(x = Rm_lag, y = Portfolio_Returns)) +
  geom_point(color = "blue", alpha = 0.6) +  # Scatter plot of returns
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # Regression line without confidence interva
  labs(title = "Regression of Portfolio Returns on Lagged Returns",
       x = "Lagged Portfolio Returns (R_m(t-1))",
       y = "Portfolio Returns (R_m(t))") +
  theme_minimal()
```

2. Regress Rm(t) on an intercept and on (Pm(t−1)/Pm(t−13)−1). Report estimates and t-statistics. Briefly interpret the results

```
data$Pm <- NA  # initializing the price series
data$Pm[1] <- 1  # arbitrary starting price

# calculate Pm(t) iteratively based on Rm(t) = (Pm(t)/Pm(t-1)) - 1
for (i in 2:nrow(data)) {
  data$Pm[i] <- data$Pm[i - 1] * (1 + data$Portfolio_Returns[i])
}

data$Pm_lag1 <- dplyr::lag(data$Pm, 1)
data$Pm_lag13 <- dplyr::lag(data$Pm, 13)
data$Pm_ratio <- (data$Pm_lag1 / data$Pm_lag13) - 1
head(data,20)
```

```
model2 <- lm(Portfolio_Returns ~ Pm_ratio, data = data)
summary(model2)
```

```
ggplot(data, aes(x = Pm_ratio, y = Portfolio_Returns)) +
  geom_point(color = "blue", alpha = 0.6) +  # Scatter plot of Portfolio Returns vs. Pm_ratio
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # Regression line without confidence interva
  labs(title = "Regression of Portfolio Returns on Pm_ratio",
       x = "Pm_ratio (Pm(t-1) / Pm(t-13) - 1)",
       y = "Portfolio Returns (R_m(t))") +
  theme_minimal()
```

4. Regress Rm(t) on an intercept and on abs(Rm(t)). Report estimates and t-statistics. Briefly interpret the results

```
data <- data %>%
  mutate(abs_Rm = abs(Portfolio_Returns))
head(data)
```

```
model_3 <- lm(Portfolio_Returns ~ abs_Rm, data = data)
summary(model_3)
```

```
ggplot(data, aes(x = abs_Rm, y = Portfolio_Returns)) +
  geom_point(color = "blue", alpha = 0.6) +  # Scatter plot of returns
  geom_smooth(method = "lm", color = "red", se = FALSE) +  # Regression line without confidence interva
  labs(title = "Regression of Portfolio Returns on Absolute Returns",
       x = "Absolute Returns (abs(R_m(t)))",
```

```
       y = "Portfolio Returns (R_m(t))") +
  theme_minimal()
```

5. Repeat (1) on data from 199706. (Delete all data prior to 199706, then compute lagged returns). Comment briefly

```
data_filtered <- data %>%
  filter(Date >= as.Date("1997-06-01"))

data_filtered <- data_filtered %>%
  mutate(Rm_lag = lag(Portfolio_Returns, 1))

head(data_filtered)
```

```
model_filtered <- lm(Portfolio_Returns ~ Rm_lag, data = data_filtered)
summary(model_filtered)
```

```
ggplot(data_filtered, aes(x = Rm_lag, y = Portfolio_Returns)) +
  geom_point(color = "blue", alpha = 0.6) +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(title = "Regression of Portfolio Returns on Lagged Returns (Post-1997)",
       x = "Lagged Portfolio Returns (R_m(t-1))",
       y = "Portfolio Returns (R_m(t))") +
  theme_minimal()
```

6.Repeat (2) on data from 199706. (Delete all data prior to 199706, then compute lagged returns). Comment briefly

```
head(data_filtered, 20)
```

```
model_2 <- lm(Portfolio_Returns ~ Pm_ratio, data = data_filtered)
summary(model_2)
```

```
ggplot(data_filtered, aes(x = Pm_ratio, y = Portfolio_Returns)) +
  geom_point(color = "blue", alpha = 0.6) +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(title = "Regression of Portfolio Returns on Pm_ratio",
       x = "Pm_ratio (Pm(t-1) / Pm(t-13) - 1)",
       y = "Portfolio Returns (R_m(t))") +
  theme_minimal()
```

7. Repeat (4) on data from 199706. Comment briefly

```
data_filtered <- data_filtered %>%
  mutate(abs_Rm = abs(Portfolio_Returns))
head(data_filtered)
```

```
model_3 <- lm(Portfolio_Returns ~ abs_Rm, data = data_filtered)
summary(model_3)
```

```
ggplot(data_filtered, aes(x = abs_Rm, y = Portfolio_Returns)) +
  geom_point(color = "blue", alpha = 0.6) +
  geom_smooth(method = "lm", color = "red", se = FALSE) +
  labs(title = "Regression of Portfolio Returns on Absolute Returns (Post-1997)",
```

```
       x = "Absolute Returns (abs(R_m(t)))",
       y = "Portfolio Returns (R_m(t))") +
  theme_minimal()
```