Problem Set One

Josh Virene

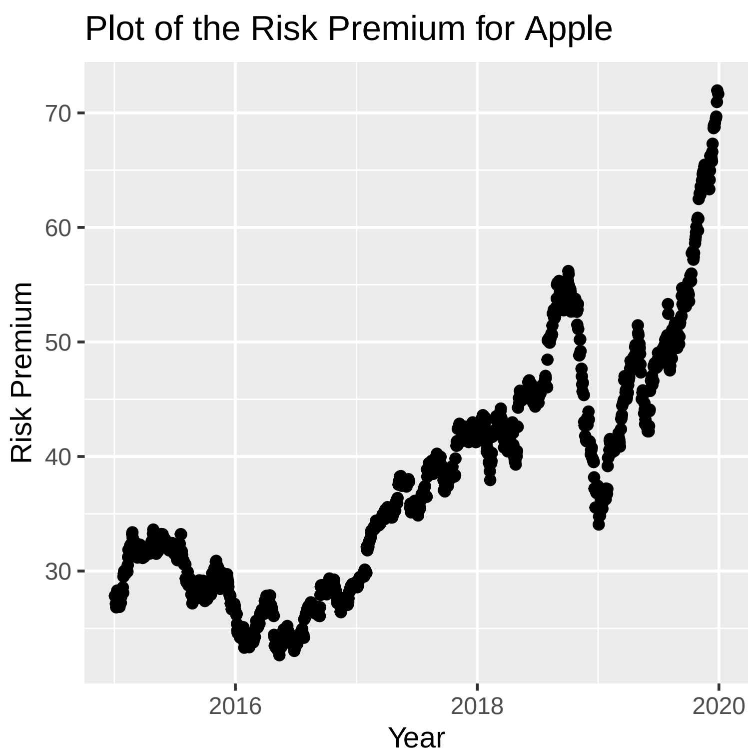
2023-10-12

Problem One:

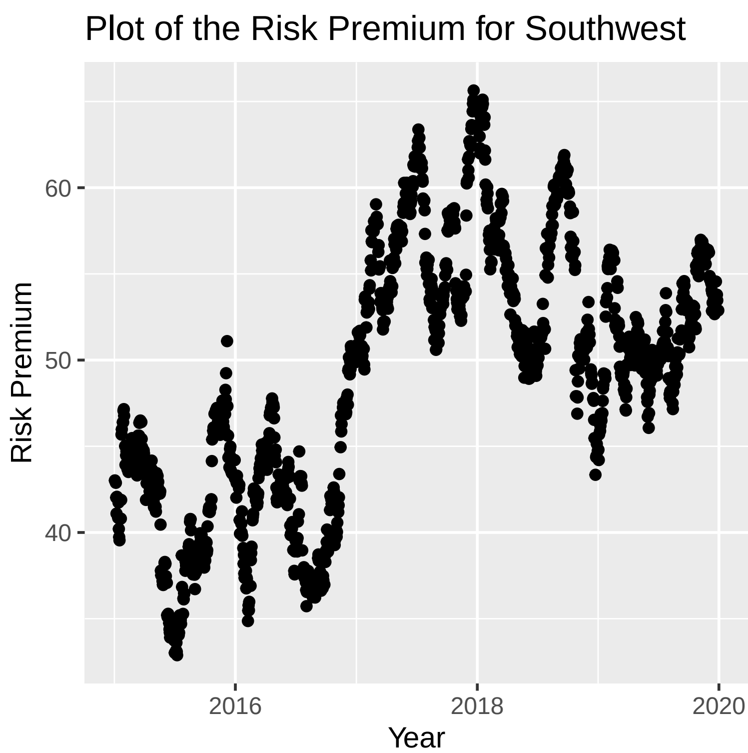
Combine your data into a single file where you have dates, stock prices for your "risky" and "safe" assets, and data on your market index, and risk free rates. What are we using as the risk-free rate and the market return rate? Compute the returns on each of your stocks. Divide your sample in two, the first half spanning January 2010-December 2014, and the second half covering January 2015–December 2019. Choose the half you want to work with.

- The file is available as "p2_dataset.xlsx"
- The risk free rate is given by the 13-week treasury yield (^IRX), and the market return rate is given by the S&P500 index (^GSPC)

```
##
## Apple Risk Premium Summary Statistics
## ===============================================
##                         Dependent variable:
##                     ---------------------------
##                             y_variable
## -----------------------------------------------
## x_variable                   0.03***
##                             (0.0003)
##
## Constant                    -29.60***
##                              (0.69)
##
## -----------------------------------------------
## Observations                  1,255
## R2                            0.89
## Adjusted R2                   0.89
## Residual Std. Error      3.49 (df = 1253)
## F Statistic         9,827.61*** (df = 1; 1253)
## ===============================================
## Note:               *p<0.1; **p<0.05; ***p<0.01
```




```
##
## Southwest Risk Premium Summary Statistics
## ==================================================
##                          Dependent variable:
##                      ---------------------------
##                               y_variable
## --------------------------------------------------
## x_variable                      0.01***
##                                (0.0004)
##
## Constant                       13.55***
##                                 (1.07)
##
## --------------------------------------------------
## Observations                     1,255
## R2                               0.48
## Adjusted R2                      0.48
## Residual Std. Error        5.42 (df = 1253)
## F Statistic           1,140.20*** (df = 1; 1253)
## ==================================================
## Note:                 *p<0.1; **p<0.05; ***p<0.01
```



Problem Two

Problem Three

DO NOT EDIT: THIS IS THE TEMPLATE FOR ADDITIONAL CODE CHUNKS

Appendix

```r
library(knitr)
knitr::opts_chunk$set(echo = FALSE, message = FALSE, warning = FALSE, fig.width = 4,
    fig.height = 4, tidy = TRUE)

# obtaining all of the needed packages
library(tidyquant)
library(tidyverse)
library(data.table)
library(dplyr)
library(writexl)
library(stargazer)
# The more efficient means of getting packages (though I cannot get R to
# compile due to CRAN mirroring issues) install.packages('pacman')
# library(pacman) p_load(tidyquant, tidyverse, data.table, dplyr, writexl)

# establish working directory:
mydir <- getwd()


# code for problem 1a

# download the daily S&P 500 indices, and the daily yields of 13-week treasury
# yield:

# establish the two periods over which we are running analysis period one:
p1_start <- "2010-01-01"
p1_end <- "2014-12-31"

p2_start <- "2015-01-01"
p2_end <- "2019-12-31"

tech_industry <- c("GOOGL", "AAPL")
airline_industry <- c("LUV", "DAL")
combined <- c("GOOGL", "AAPL", "LUV", "DAL", "^GSPC", "^IRX")

full_dataset <- tq_get(combined, get = "stock.prices", from = p1_start, to = p2_end)

p2_dataset <- tq_get(combined, get = "stock.prices", from = p2_start, to = p2_end)

# Write the dataset to an excel file for reference and later use:
write_xlsx(p2_dataset, "p2_dataset.xlsx")

# another comment, because the p2_dataset, in its current form has NAs from the
# trx data, we will remove these to ensure that the regression models are
# running
p2_c_dataset <- na.omit(p2_dataset)

# code for problem 1b

# Estimate the CAPM model

# model for apple (technology, the supposed safe industry)
apple_data <- subset(p2_c_dataset, p2_c_dataset$symbol == "AAPL")
```

```
sp_data <- subset(p2_c_dataset, p2_c_dataset$symbol == "^GSPC")
trs_data <- subset(p2_c_dataset, p2_c_dataset$symbol == "^IRX")

a <- inner_join(apple_data, sp_data, by = "date")
a <- inner_join(a, trs_data, by = "date")
a$x_variable = a$high.y - a$high
y_variable <- a$high.x - a$high
a_date <- a$date

testlm <- lm(y_variable ~ a$x_variable, data = a)
stargazer(testlm, type = "text", digits = 2, title = "Apple Risk Premium Summary Statistics")

# plot the result:
p <- ggplot(data = a) + geom_point(aes(x = a_date, y = y_variable)) + ylab("Risk Premium") +
    xlab("Year") + ggtitle("Plot of the Risk Premium for Apple")
p

# model for Southwest (Airlines, the supposed risky industry)
sw_data <- subset(p2_c_dataset, p2_c_dataset$symbol == "LUV")
sp_data <- subset(p2_c_dataset, p2_c_dataset$symbol == "^GSPC")
trs_data <- subset(p2_c_dataset, p2_c_dataset$symbol == "^IRX")

b <- inner_join(sw_data, sp_data, by = "date")
b <- inner_join(b, trs_data, by = "date")
b$x_variable <- b$high.y - b$high
y_variable <- b$high.x - b$high
b_date <- b$date



testlm <- lm(y_variable ~ b$x_variable, data = b)
stargazer(testlm, type = "text", digits = 2, title = "Southwest Risk Premium Summary Statistics")

p <- ggplot(data = b) + geom_point(aes(x = b_date, y = y_variable)) + ylab("Risk Premium") +
    xlab("Year") + ggtitle("Plot of the Risk Premium for Southwest")
p

# code for problem 2a

# code for problem 3 setup

# download the daily S&P 500 indices, and the daily yields of 13-week treasury
# yield:

# establish the two periods over which we are running analysis period one:
p1_start <- "2019-09-19"
p1_end <- "2022-09-18"

combined <- c("DRI", "ZM", "PTON", "COST", "WMT", "^GSPC")

q5_dataset <- tq_get(combined, get = "stock.prices", from = p1_start, to = p1_end)

# Write the dataset to an excel file for reference and later use:
```

```r
write_xlsx(q5_dataset, "q5_dataset.xlsx")

# code for problem 3a

# establish the event windows:
ca_lockdown <- as_date("2019-03-19")
event_window_start <- ca_lockdown - ddays(30)
event_window_end <- ca_lockdown + ddays(30)

estimation_window_start <- ca_lockdown - ddays(120)
estimation_window_end <- ca_lockdown + ddays(120)

# before plotting, need to figure out a way to only pull the results within the
# estimation window obviously this is inclusive of the event window. We can
# also filter so that we only have daily close prices
q5_dataset$date <- as.Date(q5_dataset$date, format = "%d/%m/%Y")
date_range <- q5_dataset$date <= estimation_window_start & q5_dataset$date >= estimation_window_end
filtered <- q5_dataset[date_range, ]


p <- ggplot(data = q5_dataset) + geom_point()

# code for problem 1x
```