SDGB 7844 HW 5: Portfolio Optimization

Jiayin Hu

2018/12/6

Before the questions and answers, we list the R packages we need in this homework report.

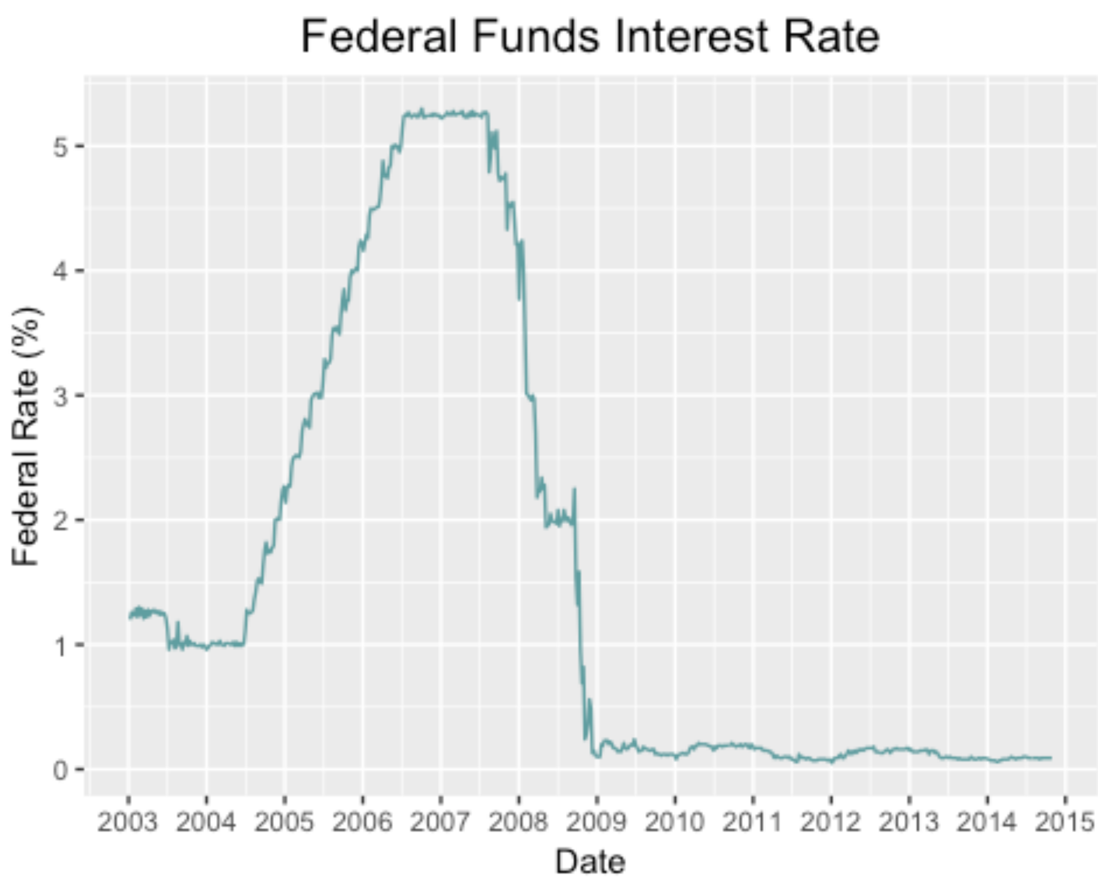
```r
#setting the packages needed
library(tidyverse)
library(lubridate)
library(ggplot2)
library(zoo)
```

Q1

Upload the data in "asset data.txt" into R and call the tibble `data.x`. The columns are defined as follows: `date`, `close.spy` is the closing price for the S&P500 ETF, `close.tlt` is the closing price for the long term treasury bond ETF, and `fed.rate` is the federal funds interest rate in percent form. Look at the data type for the column date; just like the data types numeric, logical, etc., R has one for date/time data. Extract only the observations where the federal funds rate is available (so you are left with weekly data); this is the data you will use for the rest of the analysis. What is the start date and end date of this reduced data set? Graph the federal funds interest rate as a time series. Describe what you see in the plot and relate it briefly to the most recent financial crisis.

```r
data.x <- read_delim("asset_data.txt", col_names = TRUE, delim = ",")

## Parsed with column specification:
## cols(
##   date = col_date(format = ""),
##   close.spy = col_double(),
##   close.tlt = col_double(),
##   fed.rate = col_double()
## )

class(data.x$date) # data type for the column date

## [1] "Date"

data.x <- drop_na(data.x) # weekly data

head(data.x, n = 1)$date # start date

## [1] "2003-01-08"

tail(data.x, n = 1)$date # end date
```

```
## [1] "2014-10-29"

# plot federal funds interest rate
int_plot <- data.x %>% ggplot(aes(x = date, y = fed.rate)) +
  geom_line(col = "cadetblue", size = 0.5) +
#  geom_point(pch = 19, col = "cadetblue", size = 0.3) +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y") +
  scale_y_continuous(breaks = seq(0, 6, by = 1)) +
  ylab("Federal Rate (%)") +
  xlab("Date") +
  ggtitle("Federal Funds Interest Rate") +
  theme(plot.title = element_text(size = 15, hjust = 0.5))
int_plot
```



The data type for the column date is “Date”. The start date is 2003-01-08. The end date is 2014-10-29.

From the plot, we can see

1. The federal funds interest rate from 2003 to the first half of 2004 was around 1%.

2. The federal funds interest rate increased constantly and dramaticly from the second half of 2004 to the first half of 2006.

3. The federal funds interest rate remained stably above 5% from the second half of 2006 to the first half of 2007.

4. The federal funds interest rate experienced sharply decrease between the second half of 2007 and 2008.

5. After 2009, the federal funds interest rate kept in a low level (close to 0) until the given date in 2014.
   Obviously, during the financial crisis in 2017-2018, the federal funds interest rate fell continuously.

Q2

Now we will split the data into training and test sets. The training data will be used to compute our portfolio weights and our test set will be used to evaluate our portfolio. Make two separate tibbles: (a) the training set should contain all observations before 2014 and (b) the test set should contain all observations in 2014. How many observations are in each subset?

```
test <- filter(data.x, year(data.x$date) == "2014")
nrow(test)

## [1] 43

training <- filter(data.x, year(data.x$date) < "2014")
nrow(training)

## [1] 570
```

There are 570 observations in the training set and 43 ovservations in the test set.

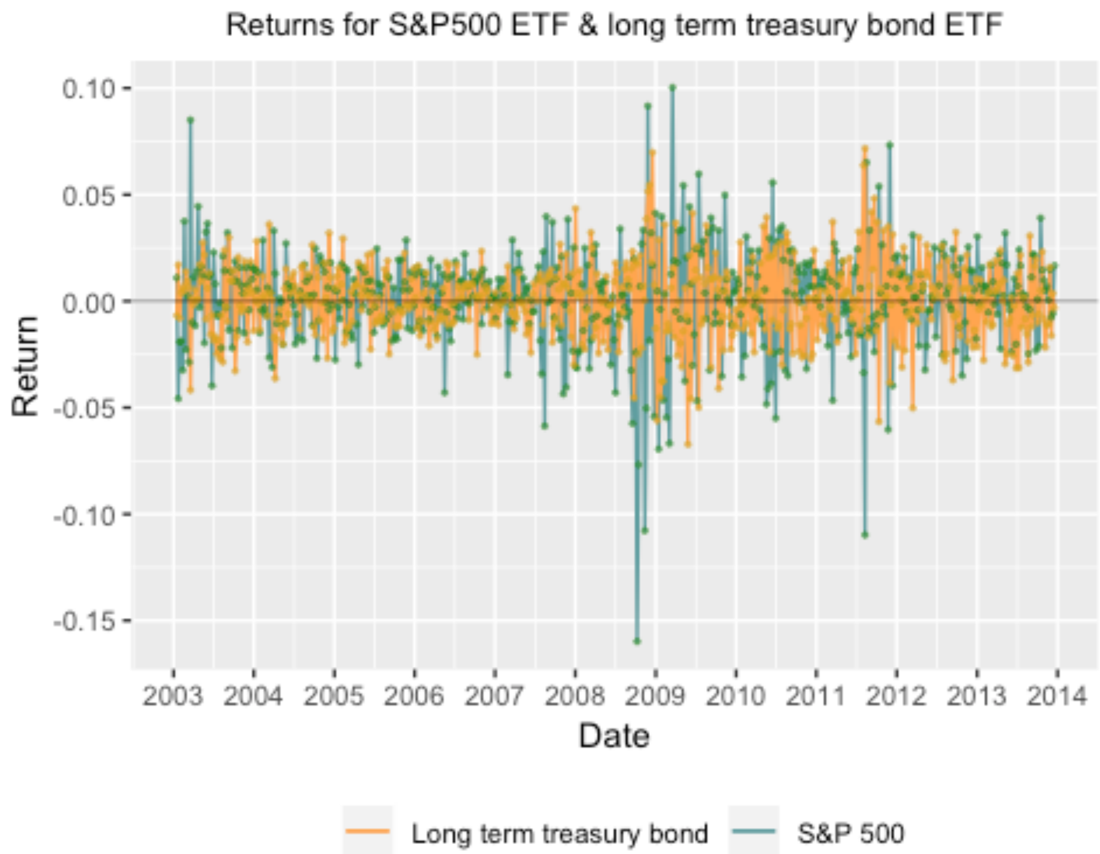
Q3

The federal funds interest rate is in percent form so convert it to decimal (i.e., fractional) form. Then, for the S&P 500 and long term treasury bonds ETF assets, compute the returns using the following formula:

$$r_t = \frac{p_t - p_{t-1}}{p_t}$$

where r_t is the return at time t, p_t is the asset price at time t, and p_{t-1} is the asset price at time $t - 1$ (i.e., the previous period). Add both sets of returns to your training set tibble. These returns are also called total returns. Construct a single time series plot with the returns for both assets plotted. Add a dotted, horizontal line at $y = 0$ to the plot. Compare the two returns series. What do you see?

```r
training$fed.rate <- training$fed.rate/100
spy.return <- (training$close.spy - 
lag(training$close.spy))/lag(training$close.spy)
tlt.return <- (training$close.tlt - 
lag(training$close.tlt))/lag(training$close.tlt)
training$spy.return = spy.return
training$tlt.return = tlt.return

return_plot <- ggplot(data = training, aes(x = date)) +
  geom_line(aes(y = spy.return, colour = "S&P 500"), size = 0.5) +
  geom_line(aes(y = tlt.return, colour = "Long term treasury bond"), size = 
0.5) +
  geom_point(aes(y = spy.return), pch = 19, col = "forestgreen", alpha = 0.5, 
size = 0.5) +
  geom_point(aes(y = tlt.return), pch = 19, col = "goldenrod", alpha = 0.5, 
size = 0.5) +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y") +
  scale_y_continuous(breaks = seq(-0.25, 0.15, by = 0.05)) +
  scale_colour_manual("",
                      breaks = c("Long term treasury bond", "S&P 500"),
                      values = c("tan1", "cadetblue")) +
  ylab("Return") +
  xlab("Date") +
  geom_hline(yintercept = 0, alpha = 0.9, size = 0.2) +
  ggtitle("Returns for S&P500 ETF & long term treasury bond ETF") +
  theme(plot.title = element_text(size = 10, hjust = 0.5),
        legend.position = "bottom")

return_plot
```

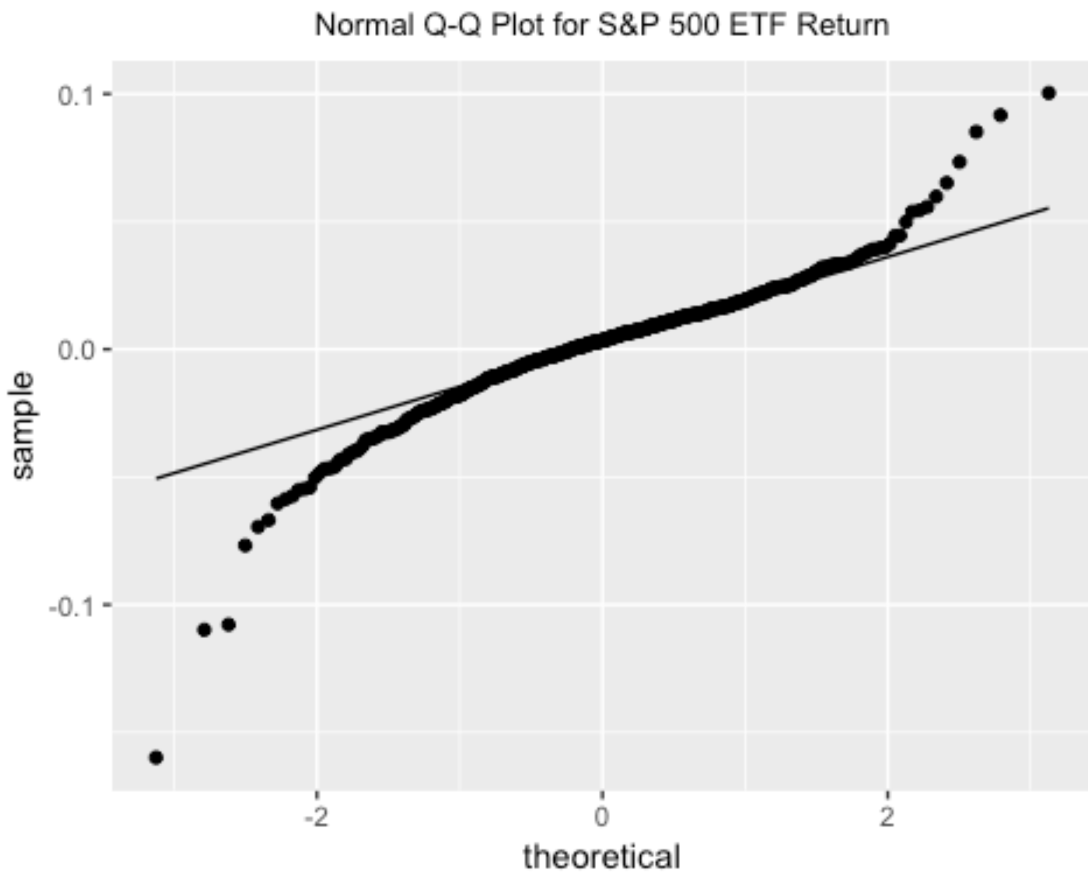


The returns of these two ETF fluactuate around 0. The range of the return of S&P500 ETF is larger than the return of long term treasury bond. S&P 500 EFT is more likely shows larger gain or loss.

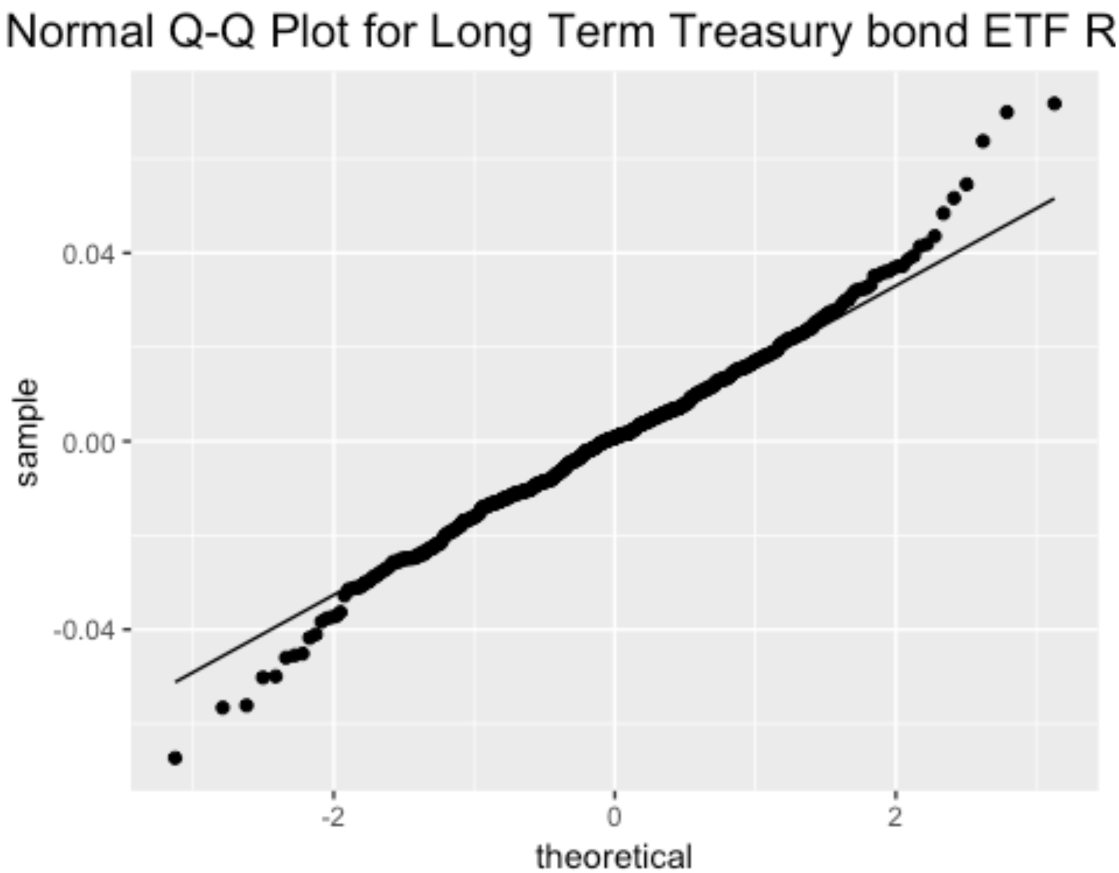
Q4

The Sharpe ratio calculation assumes that returns are normally distributed. Construct two normal quantile plots, one for training set returns of each asset. Is this assumption satisfied? Justify your answer.

```
spy.qq <- training %>% ggplot(aes(sample = spy.return)) + stat_qq() +
stat_qq_line() +
  ggtitle("Normal Q-Q Plot for S&P 500 ETF Return") +
  theme(plot.title = element_text(size = 10, hjust = 0.5))
spy.qq
```




```
tlt.qq <- training %>% ggplot(aes(sample = tlt.return)) + stat_qq() +
stat_qq_line() +
  ggtitle("Normal Q-Q Plot for Long Term Treasury bond ETF Return") +
  theme(plot.title = element_text(size = 15, hjust = 0.5))
tlt.qq
```



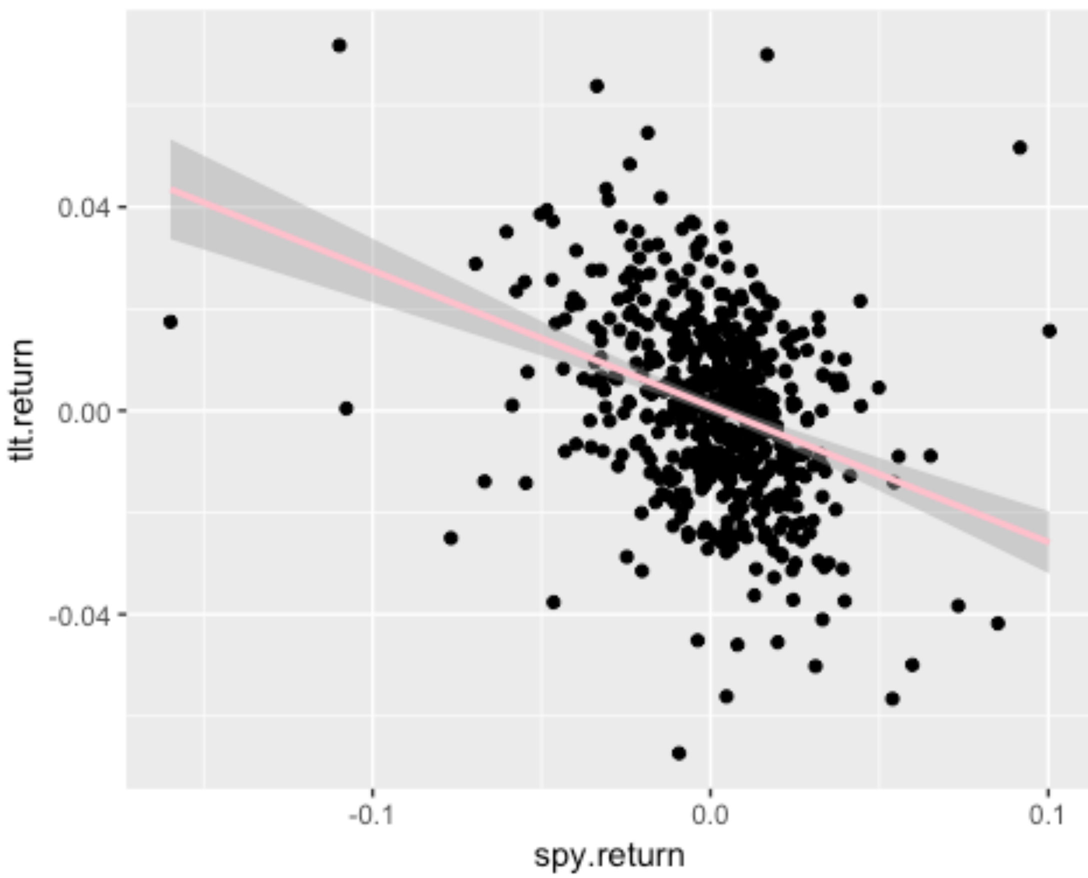
In these two plot, the points do not form a straight line appoximately. Therefore, the returns do not comply with normal distribution.

Q5

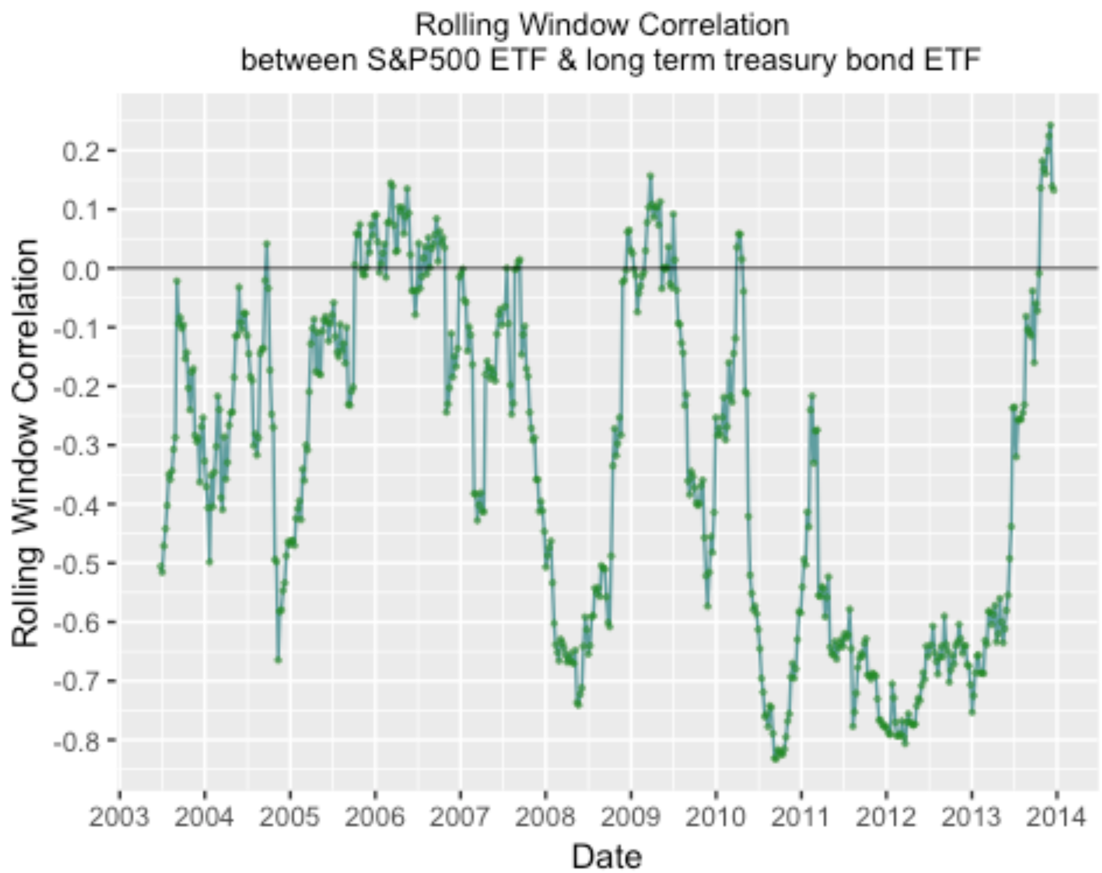
Compute the correlation between the S&P500 and long term treasury bond returns in the training set and interpret it. Now, we will compute a rolling-window correlation as follows: compute the correlation between the two asset returns only using the first 24 weeks of data (i.e., weeks 2 to 25), next compute the correlation between the two asset returns for data from week 3 through 26, then week 4 through 27, and so forth. Once you compute the rolling-window correlations, make a time series plot of the rolling- window correlation with each point plotted on the last day of the window. Add a horizontal, dotted, gray line at 0 to your plot. Is the correlation or rolling-window correlation a better way to describe the relationship between these two assets? Justify your answer.

```
cor(training$spy.return, training$tlt.return, use = "pairwise.complete.obs")

## [1] -0.3439013

ggplot(training, aes(x = spy.return, y = tlt.return)) + geom_point() +
geom_smooth(method = lm, se = TRUE, col = "pink")
```

```
## Warning: Removed 1 rows containing non-finite values (stat_smooth).

## Warning: Removed 1 rows containing missing values (geom_point).
```




```
#Rolling-window
rwcor <- rollapply(data.frame(training$spy.return, training$tlt.return),
                   24, function(x) cor(x[,1],x[,2]), by.column = FALSE)

cordata <- data.frame(training[25:nrow(training), "date"], rwcor[-1])
ggplot(cordata, aes(date, rwcor..1.)) +
  geom_line(col = "cadetblue", size = 0.5) +
  geom_point(pch = 19, col = "forestgreen", alpha = 0.5, size = 0.5) +
  geom_hline(yintercept = 0, alpha = 0.9, size = 0.2) +
  scale_x_date(date_breaks = "1 year", date_labels = "%Y") +
  scale_y_continuous(breaks = seq(-1, 0.3, by = 0.1)) +
  ylab("Rolling Window Correlation") +
  xlab("Date") +
  geom_hline(yintercept = 0, alpha = 0.9, size = 0.2) +
  ggtitle("Rolling Window Correlation \n between S&P500 ETF & long term
treasury bond ETF") +
  theme(plot.title = element_text(size = 10, hjust = 0.5))
```



The correlation between the return of S&P500 and long term treasury bond returns in the training set is -0.3439, which means the returns of these two assets were negatively related. However, because the value -0.3439 is not so much close to 1, the relationship were not very obvious. And through the scatter plot, the slope of the regression line is negative, but only a little points are in the 95% confidence interval.

The rolling-window correlation represents the relationship of these two assets in 24-week period.

From my perspective, rolling-window correlation is better to describe the relationship. Because financial data is influenced by a lot of factors, the return of asset can change significantly in a relatively short period.

The correlation considers all the returns from 2003 to 2014, which is a long period, so the measurement only provides an overall situation that they were negatively related. Through the rolling-window correlation, we can see that though in most periods the assets were negatively related, they showed positive related in some periods. Also, their negatively related relationship were very obvious.

Q6

Compute the Sharpe ratios for each asset on the training set. Which asset is a better investment? Justify your answer.

```
training$spy.exreturn <- training$spy.return - lag(training$fed.rate)/52
training$tlt.exreturn <- training$tlt.return - lag(training$fed.rate)/52

training$spy.exreturn.ind <- 100
for (i in 1:(nrow(training) - 1))
  training[i + 1, "spy.exreturn.ind"] <- training[i, "spy.exreturn.ind"] * (1
+ training[i + 1, "spy.exreturn"] )

training$tlt.exreturn.ind <- 100
for (i in 1:(nrow(training) - 1))
  training[i + 1, "tlt.exreturn.ind"] <- training[i, "tlt.exreturn.ind"] * (1
+ training[i + 1, "tlt.exreturn"] )

n <- (nrow(training) - 1) / 52

spy.cagr <- (training$spy.exreturn.ind[nrow(training)]/100)^(1/n) - 1
spy.vol <- sqrt(52) * sd(training$spy.exreturn, na.rm = TRUE)
spy.sp <- spy.cagr/spy.vol
spy.sp

## [1] 0.2807176

tlt.cagr <- (training$tlt.exreturn.ind[nrow(training)]/100)^(1/n) - 1
tlt.vol <- sqrt(52) * sd(training$tlt.exreturn, na.rm = TRUE)
tlt.sp <- tlt.cagr/tlt.vol
tlt.sp

## [1] -0.01095925
```

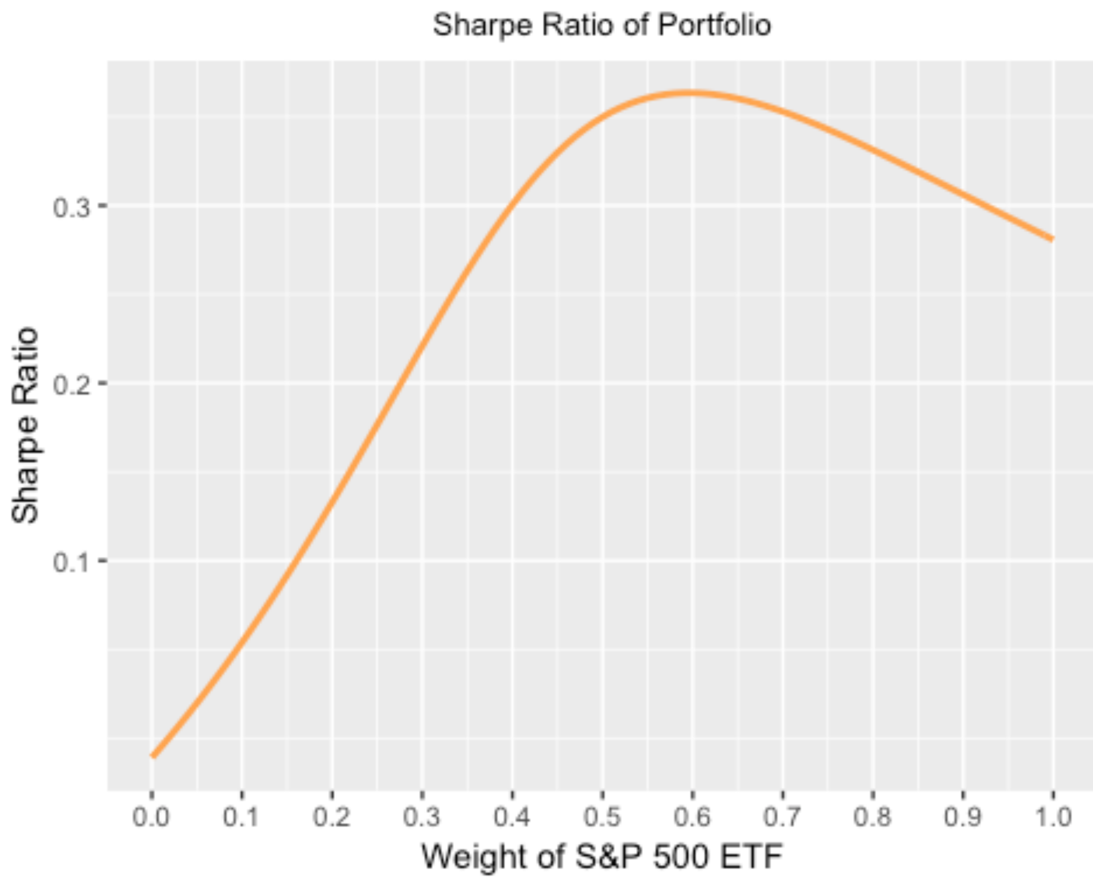
The Sharpe ratio of S&P500 ETF and long term treasury bond ETF are 0.2807176 and -0.01095925 respectively. Sharpe ratio is use to measure the excess return the asset earns in a unit volatility, which means higher Sharpe ratio is better. Therefore, S&P500 ETF is better.

Q7

Write a function which takes the following inputs: (a) a vector of portfolio weights (call this argument x; weights are between 0 and 1), (b) a vector of returns for asset 1, (c) a vector of returns for asset 2, and (d) a vector of the corresponding weekly federal funds interest rates. The function will then do the following: for each weight value in your vector x, you will compute the Sharpe ratio for the corresponding portfolio. Use `stat_function()` to plot the function you just wrote. Weights between 0 and 1 should be on the x-axis and the Sharpe ration should be on the y-axis. The training set data should be used as the input for (b), (c), and (d) above. Do you see a portfolio weight that produces the maximum Sharpe ratio?

```
SpRatio <- function(x, r_spy, r_tlt, r_f){
  sharpe <- c()
```

```r
  for (i in 1:length(x)) {
    r_p <- x[i] * r_spy + (1 - x[i]) * r_tlt
    er_p <- r_p - lag(r_f)/52
    erind_p <- c(100)
    for (j in 1:(length(r_p) - 1))
      erind_p[j + 1] <- erind_p[j] * (1 + er_p[j + 1]) # end loop
    n <- (length(r_p) - 1) / 52
    cagr <- (erind_p[length(r_p)]/100)^(1/n) - 1
    vol <- sqrt(52) * sd(er_p, na.rm = TRUE)
    sp <- cagr/vol
    sharpe[i] <- sp # end loop
  }
  return(sharpe)
} # end function

### stat_function plot
r_spy <- training$spy.return
r_tlt <- training$tlt.return
r_f <- training$fed.rate

ggplot(data.frame(x = c(0, 1)), aes(x)) +
  stat_function(aes(), fun = SpRatio,  args = list(r_spy, r_tlt, r_f), size =
1, col = "tan1") +
  scale_x_continuous(breaks = seq(0, 1, by = 0.1)) +
  scale_y_continuous(breaks = seq(0.1, 0.6, by = 0.1)) +
  ylab("Sharpe Ratio") +
  xlab("Weight of S&P 500 ETF") +
  ggtitle("Sharpe Ratio of Portfolio") +
  theme(plot.title = element_text(size = 10, hjust = 0.5))
```



From the plot, when the weight of the first asset is around 60%, the portfolio produces the maximum Sharpe ratio over 0.35.

Q8

Using the training set, use `optimize()` to determine the optimum weight for each asset using the function you wrote in question 7; how much of the funds should be allocated to each asset? What is the Sharpe ratio of the overall portfolio? According to your analysis, is it better to invest in S&P500 only, long term treasury bonds only, or your combined portfolio? Justify your answer.

```
w1 <- optimize(f = function(x) {return(SpRatio(x = x, r_spy = r_spy, r_tlt =
r_tlt, r_f = r_f))},
         lower = 0, upper = 1, maximum = TRUE)
x1 = w1$maximum
sp.portfolio = w1$objective
x2 <- 1 - x1
x1

## [1] 0.5958502

x2
```

```
## [1] 0.4041498

sp.portfolio

## [1] 0.3634139
```

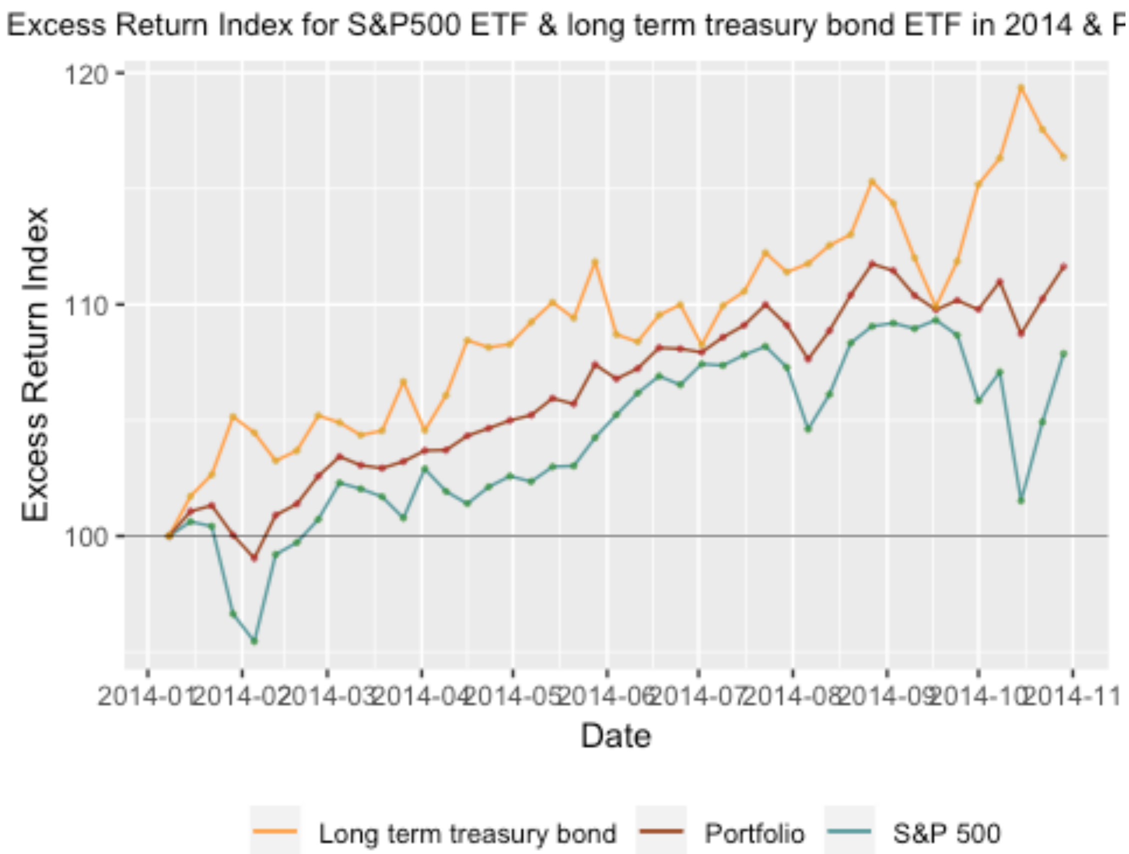
We should allocate 59.59% of the fund to S&P500 ETF and 40.41% of the fund to long term treasury ETF. The Sharpe ratio is 0.3634, which is larger than the Sharpe ratios of the two kinds of single asset. So the combined portfolio is better.

Q9

For the remainder of this assignment, we will be evaluating our portfolio using the test set data. We will be comparing three strategies: investing only in the S&P500, investing only in long term treasury bonds, and investing in the combined portfolio. In your test set, convert the federal funds interest rate from percent to decimal form and compute the returns series for each of the three assets. Next, compute the excess returns index for each asset in the test set (as outlined in question 6). Plot the excess returns index for each asset on the same time series plot. Add a dotted, horizontal line at y = 100. Describe what you see.

```
test$fed.rate <- test$fed.rate/100

spy.return.test <- (test$close.spy - lag(test$close.spy))/lag(test$close.spy)
tlt.return.test <- (test$close.tlt - lag(test$close.tlt))/lag(test$close.tlt)
test$spy.return = spy.return.test
test$tlt.return = tlt.return.test
test$spy.exreturn <- test$spy.return - lag(test$fed.rate)/52
test$tlt.exreturn <- test$tlt.return - lag(test$fed.rate)/52

test$spy.exreturn.ind <- 100
for (i in 1:(nrow(test) - 1))
  test[i + 1, "spy.exreturn.ind"] <- test[i, "spy.exreturn.ind"] * (1 +
test[i + 1, "spy.exreturn"] )

test$tlt.exreturn.ind <- 100
for (i in 1:(nrow(test) - 1))
  test[i + 1, "tlt.exreturn.ind"] <- test[i, "tlt.exreturn.ind"] * (1 +
test[i + 1, "tlt.exreturn"] )

x = x1
test$portfolio.return <- x * test$spy.return + (1 - x) * test$tlt.return
test$portfolio.exreturn <- test$portfolio.return - lag(test$fed.rate)/52
test$portfolio.exreturn.ind <- 100
for (i in 1:(nrow(test) - 1))
  test[i + 1, "portfolio.exreturn.ind"] <- test[i, "portfolio.exreturn.ind"]
* (1 + test[i + 1, "portfolio.exreturn"] )
```

```r
# plot test data excess return index
test_exreturnind_plot <- ggplot(data = test, aes(x = date)) +
  geom_line(aes(y = spy.exreturn.ind, colour = "S&P 500"), size = 0.5) +
  geom_line(aes(y = tlt.exreturn.ind, colour = "Long term treasury bond"),
size = 0.5) +
  geom_line(aes(y = portfolio.exreturn.ind, colour = "Portfolio"), size =
0.5) +
  geom_point(aes(y = spy.exreturn.ind), pch = 19, col = "forestgreen", alpha
= 0.5, size = 0.5) +
  geom_point(aes(y = portfolio.exreturn.ind), pch = 19, col = "firebrick",
alpha = 0.5, size = 0.5) +
  geom_point(aes(y = tlt.exreturn.ind), pch = 19, col = "goldenrod", alpha =
0.5, size = 0.5) +
  scale_x_date(date_breaks = "1 month", date_labels = "%Y-%m") +
  scale_y_continuous(breaks = seq(-90, 120, by = 10)) +
  scale_colour_manual("",
                      breaks = c("Long term treasury bond", "Portfolio", "S&P
500"),
                      values = c("tan1", "sienna", "cadetblue")) +
  ylab("Excess Return Index") +
  xlab("Date") +
  geom_hline(yintercept = 100, alpha = 0.9, size = 0.2) +
  ggtitle("Excess Return Index for S&P500 ETF & long term treasury bond ETF
in 2014 & Portfolio") +
  theme(plot.title = element_text(size = 10, hjust = 0.5),
        legend.position = "bottom")

test_exreturnind_plot
```



The three strategies shows similar increasing trend overall. The excess return of long term treasury bond ETF is highest and excess return of S&P500 ETF is lowest. Because the prtfolio of a combinatio of these two assets, the excess return is between them.

Q10

The excess returns index can be interpreted as follows: if you invested in $100 in at time $t = 1$, the index value at time T represents how much you have earned in addition to (i.e., in excess of) the risk free interest rate. If you invested $100 in each asset (portfolio, all in long term treasury bonds, or all in S&P500) in the first week of January, 2014 , how much would you have at the end of the test set period for each asset in addition to the risk-free interest rate? Did your portfolio perform well in the test set? Justify your answer.

```
test[nrow(test), c("spy.exreturn.ind", "tlt.exreturn.ind",
"portfolio.exreturn.ind")]
```

```
## # A tibble: 1 x 3
##   spy.exreturn.ind tlt.exreturn.ind portfolio.exreturn.ind
##              <dbl>            <dbl>                  <dbl>
## 1             108.             116.                   112.
```

The strategy only investing in S&P500 gets 107.9. The strategy with only long term treasury bond earns 116.4. And the portfolio gets 112.2. The portfolio does not overperform investing only in long term treasury bonds. The weight is derived from optimizing Sharpe ratio in the previous period, the absolute excess return is not the same measurement as the Sharpe ratio. And, in different time period, the optimal weight can be different.