Assignment 04, Question 1&2

Jeff Nguyen

March 17, 2021

University of Southern California
Marshall School of Business
FBE 543 Forecasting and Risk Analysis

Student Name: Ngoc Son (Jeff) Nguyen

Question 1

Downloading data:

```
library(quantmod)
```

```
## Loading required package: xts

## Loading required package: zoo

##
## Attaching package: 'zoo'

## The following objects are masked from 'package:base':
##
##     as.Date, as.Date.numeric

## Loading required package: TTR

## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo

## Version 0.4-0 included new data defaults. See ?getSymbols.
```

```
# Set start date and end date of data
start_date <- "2018-01-01"
end_date <- "2021-03-17"

# Get data
getSymbols("AAPL", src = "yahoo", from = start_date, to = end_date)
```

```
## 'getSymbols' currently uses auto.assign=TRUE by default, but will
## use auto.assign=FALSE in 0.5-0. You will still be able to use
## 'loadSymbols' to automatically load data. getOption("getSymbols.env")
## and getOption("getSymbols.auto.assign") will still be checked for
## alternate defaults.
##
## This message is shown once per session and may be disabled by setting
## options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```
## [1] "AAPL"
```

```
getSymbols("^GSPC", src = "yahoo", , from = start_date, to = end_date) # S&P 500
```

```
## [1] "^GSPC"
```

```
# Adjusted Prices
adjAAPL <- AAPL$AAPL.Adjusted
adjGSPC <- GSPC$GSPC.Adjusted
```

a. Graph your AAPL against time (scatter diagram). Comment on the existence of time trend, seasonal trend, cyclical trend, autocorrelation, randomness, structural breaks, and outliers.

```
plot(adjAAPL, main="AAPL", col="Blue")
```

Time Trend: AAPL displays time trend, as price increases over time.
Seasonal Trend: AAPL displays seasonal trend with up and down spikes in price daily.
Cyclical Trend: AAPL is affected by business cycle of peaks and troughs.
Autocorrelation: AAPL rises for some times and they rise and vice versa.
Randomness: Price of AAPL is unpredictable via inspection.
Structural Breaks: AAPL does not experience any structural break during this time frame as price always recover quickly.
Outliers: AAPL price has several outliers during this period (Jan 2019, March 2020) where price fell more than 30% and recovered.

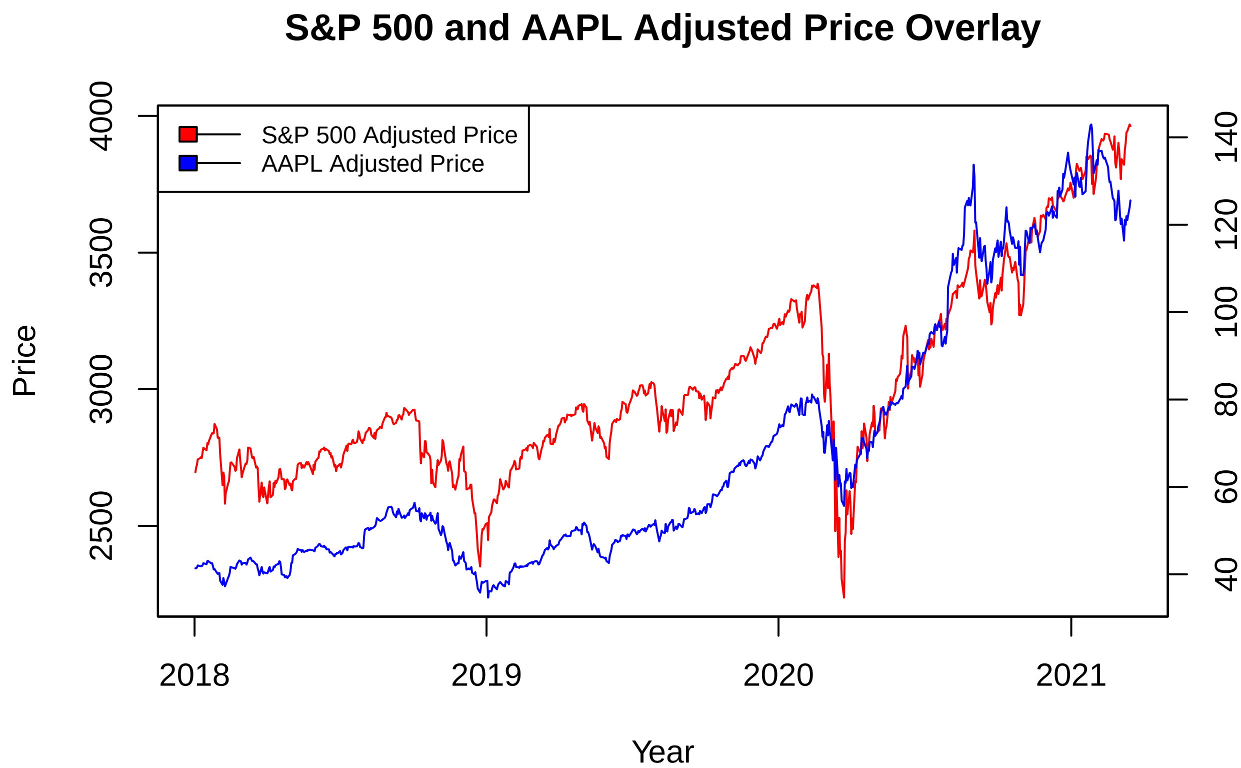
b. Graph S&P 500. Comment on the existence of time trend, seasonal trend, cyclical trend, autocorrelation, randomness, structural breaks, and outliers.

```
plot(adjGSPC, main="S&P500", col="Red")
```

Time Trend: S&P 500 displays time trend, as price increases over time.
Seasonal Trend: S&P 500 displays seasonal trend with up and down spikes in price daily.
Cyclical Trend: S&P 500 is affected by business cycle of peaks and troughs.
Autocorrelation: S&P 500 rises for some times and they rise and vice versa.
Randomness: Price of S&P 500 is unpredictable via inspection.
Structural Breaks: S&P 500 does not experience any structural break during this time frame as price always recover quickly.
Outliers: S&P 500 price has several outliers during this period (Jan 2019, March 2020) where price fell more than 30% and recovered.

c. Graph your variable against the market index S&P 500 on x-y axis. Comment on the behavior and the relationship between the two variables.

```r
# Initialize xts objects contain adjusted price for S&P 500 and AAPL and merge
gspc_xts <- as.xts(GSPC[,"GSPC.Adjusted"])
aapl_xts <- as.xts(AAPL[,"AAPL.Adjusted"])
price_compare <- merge.xts(gspc_xts, aapl_xts)

# Graph monthly AAPL and monthly S&P500 on one coordinate system
# Plot S&P 500
plot(as.zoo(price_compare[, "GSPC.Adjusted"]), screens = 1,
     main = "S&P 500 and AAPL Adjusted Price Overlay",
     xlab = "Year", ylab = "Price", col = "Red")

# Keep working on the same plot
par(new = TRUE)

# Plot AAPL and suppress axis value
plot(as.zoo(price_compare[, "AAPL.Adjusted"]),
     screens = 1,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "Blue")

# Add right-handed axis to display AAPL price
axis(4)

# Add legend
legend("topleft",
       c("S&P 500 Adjusted Price", "AAPL Adjusted Price"),
       lty = 1:1,
       cex = 0.75,
       fill = c("red", "blue"))
```



Remarks
1. Seasonal Trend: Both display similar seasonal trend with up and down spikes in price days after days.
2. Cyclical Trend:
Both is affected by business cycle with similar peaks and troughs. The S&P 500 has more prominent troughs compared to that of AAPL. Especially during the period of March 2020 with the COVID-19 lockdowns started rolling across the countries.
3. Auto-correlation:
Both behave similar in this regard where both rises for some times when they rise and vice versa.
4. Randomness:
Price of both is unpredictable via inspection.
5. Time:
Both follows similar time trend.
6. Structural Break:
Both does not experience any structural break during this time frame. During the COVID-19 March sell-off, price recovered quickly in a V-shaped fashion.
7. Outliers:
SPY looks to have several outliers corresponded to the March 2020 sell-off when the indicies retreated 30% in a short period of a few weeks before quickly recovered.

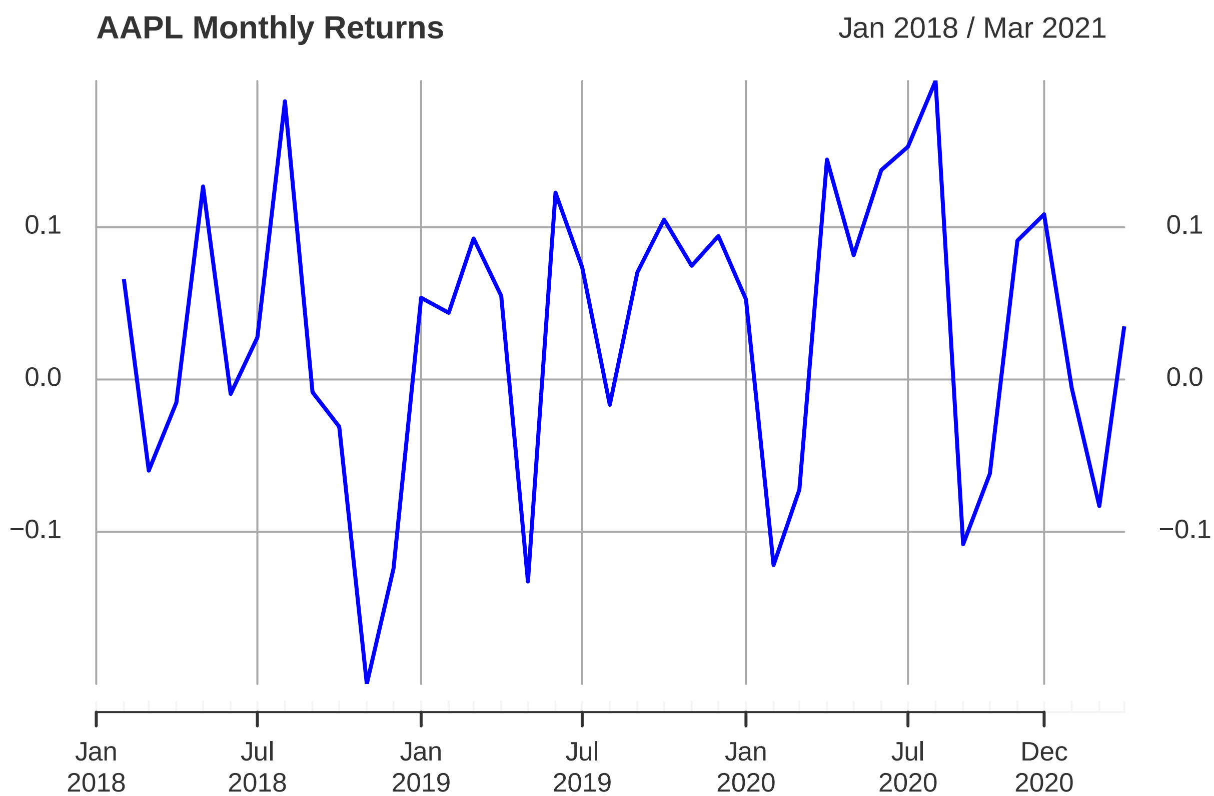
d. Repeat a to c for the monthly returns to AAPL and S&P 500.

Adjusting data to monthly:

```
# Get adjusted returns data
rAAPL <- diff(log(to.monthly(AAPL)$AAPL.Adjusted))
rGSPC <- diff(log(to.monthly(GSPC)$GSPC.Adjusted))
```

d.a. Graph your AAPL monthly returns against time (scatter diagram). Comment on the existence of time trend, seasonal trend, cyclical trend, autocorrelation, randomness, structural breaks, and outliers.

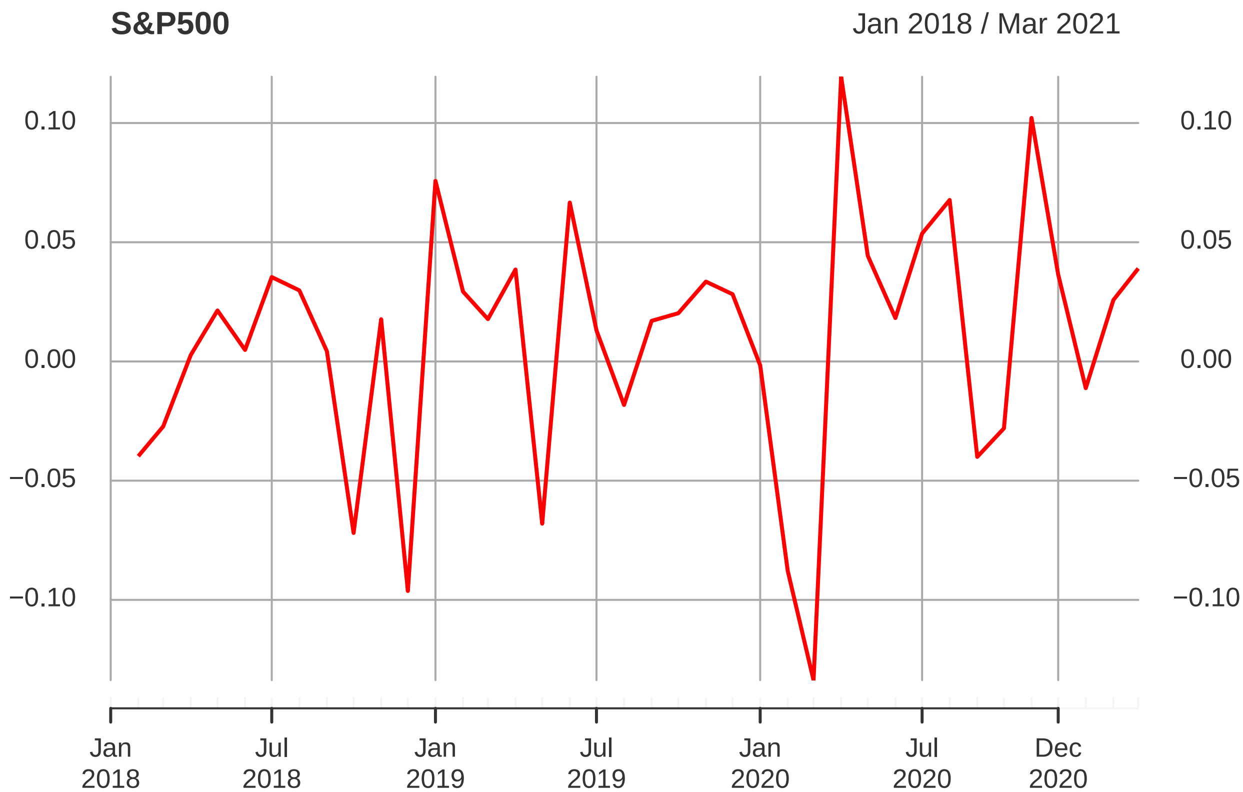
```
plot(rAAPL, main="AAPL Monthly Returns", col="Blue")
```



Time Trend: We can not observe a clear time trend here. **Seasonal Trend:** AAPL monthly returns displays seasonal trend with up and down spikes.
Cyclical Trend: We cannot observe a clear cyclical trend that coincides with business cycles here even though there are peaks and troughs.
Autocorrelation: We cannot observe autocorrelation characteristics.
Randomness: AAPL monthly returns can be observed to be unpredictable.
Structural Breaks: We cannot observe any structural break here.
Outliers: AAPl monthly returns have several clear outliers during this period.

d.b. Graph S&P 500. Comment on the existence of time trend, seasonal trend, cyclical trend, autocorrelation, randomness, structural breaks, and outliers.

```
plot(rGSPC, main="S&P500", col="Red")
```

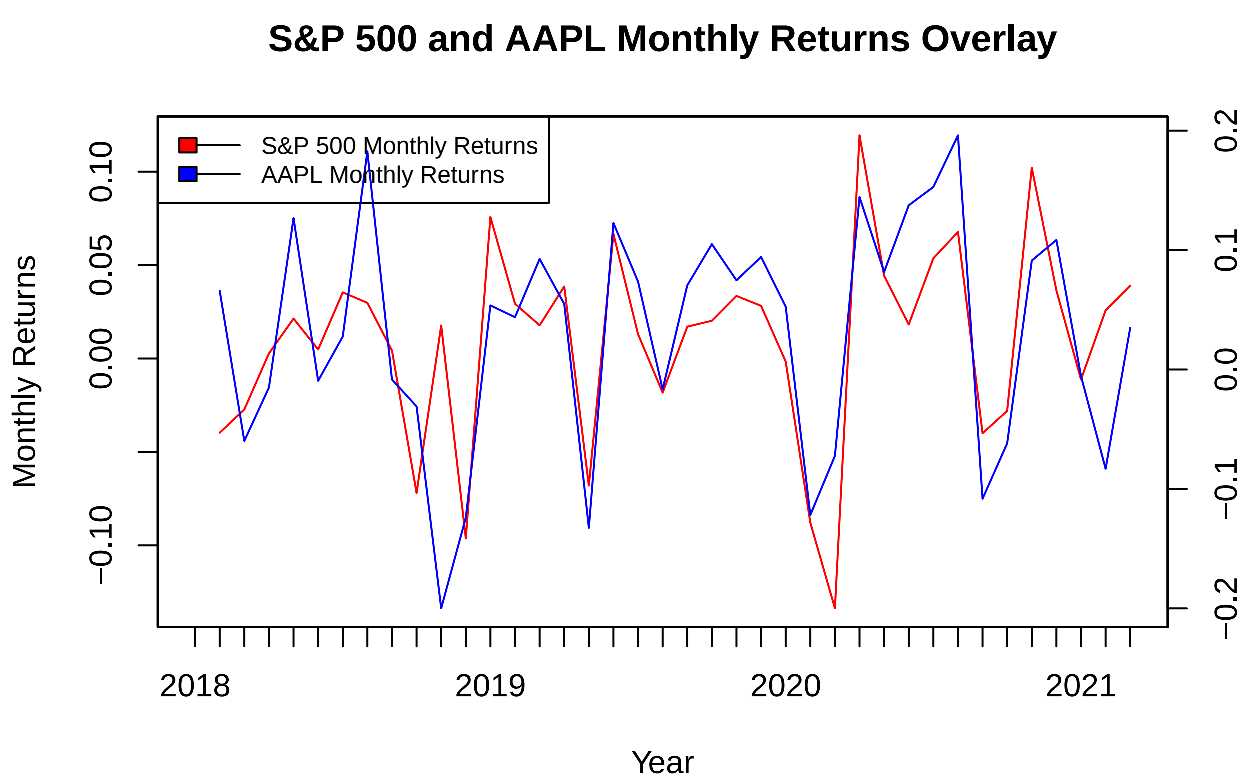


Time Trend: We can not observe a clear time trend here. **Seasonal Trend:** S&P 500 monthly returns displays seasonal trend with up and down spikes.
Cyclical Trend: We cannot observe a clear cyclical trend that coincides with business cycles here even though there are peaks and troughs.
Autocorrelation: We cannot observe autocorrelation characteristics.
Randomness: S&P 500 monthly returns can be observed to be unpredictable.
Structural Breaks: We cannot observe any structural break here.
Outliers: S&P 500 monthly returns have several clear outliers during this period.

d.c. Graph your variable against the market index S&P 500 on x-y axis. Comment on the behavior and the relationship between the two variables.

```r
# Initialize xts objects contain adjusted price for S&P 500 and AAPL and merge
rgspc_xts <- as.xts(rGSPC[,"GSPC.Adjusted"])
raapl_xts <- as.xts(rAAPL[,"AAPL.Adjusted"])
monthlyR_compare <- merge.xts(rgspc_xts, raapl_xts)

# Graph monthly AAPL and monthly S&P500 on one coordinate system
# Plot S&P 500
plot(as.zoo(monthlyR_compare[, "GSPC.Adjusted"]), screens = 1,
     main = "S&P 500 and AAPL Monthly Returns Overlay",
     xlab = "Year", ylab = "Monthly Returns", col = "Red")

# Keep working on the same plot
par(new = TRUE)
```

```r
# Plot AAPL and suppress axis value
plot(as.zoo(monthlyR_compare[, "AAPL.Adjusted"]),
     screens = 1,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "Blue")

# Add right-handed axis to display AAPL price
axis(4)

# Add legend
legend("topleft",
       c("S&P 500 Monthly Returns", "AAPL Monthly Returns"),
       lty = 1:1,
       cex = 0.75,
       fill = c("red", "blue"))
```



Remarks
For the most part, both monthly returns of AAPL and S&P 500 follow each other in lock-step saved for a few periods where AAPL monthly returns become more volatile. S&P 500 experienced larger draw down in returns compared to AAPL in the March 2020 "pandemic sell-off".

e. Compare the risk and return of AAPL with the risk and return to S&P 500.

Calculations:

```r
# Calculate statistics
AAPL_meanR <- mean(rAAPL, na.rm=TRUE)
GSPC_meanR <- mean(rGSPC, na.rm=TRUE)

AAPL_riskR <- sqrt(var(rAAPL, na.rm=TRUE))
GSPC_riskR <- sqrt(var(rGSPC, na.rm=TRUE))
```

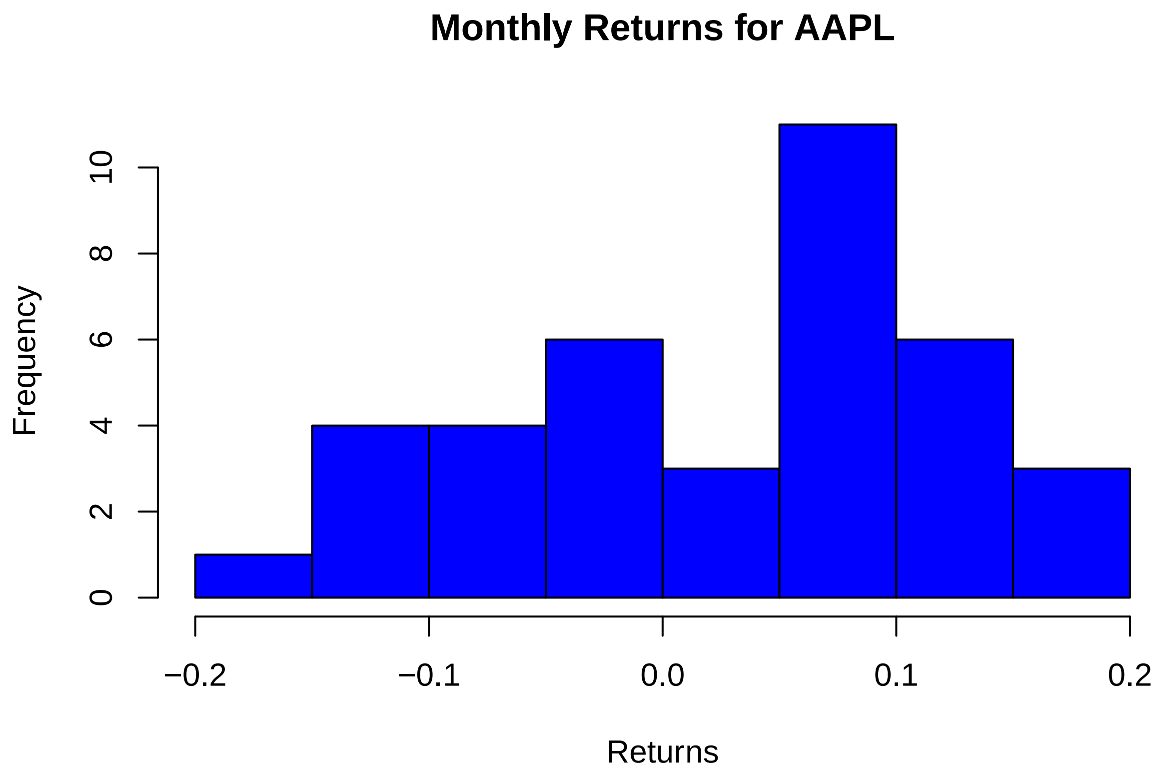
Comparison:

Underlying	Expected Monthly Returns	Risk
S&P 500	0.0089169	0.0528529
AAPL	0.0299605	0.0959594

We can observe AAPL has higher expected monthly returns but also higher risk than the S&P 500.

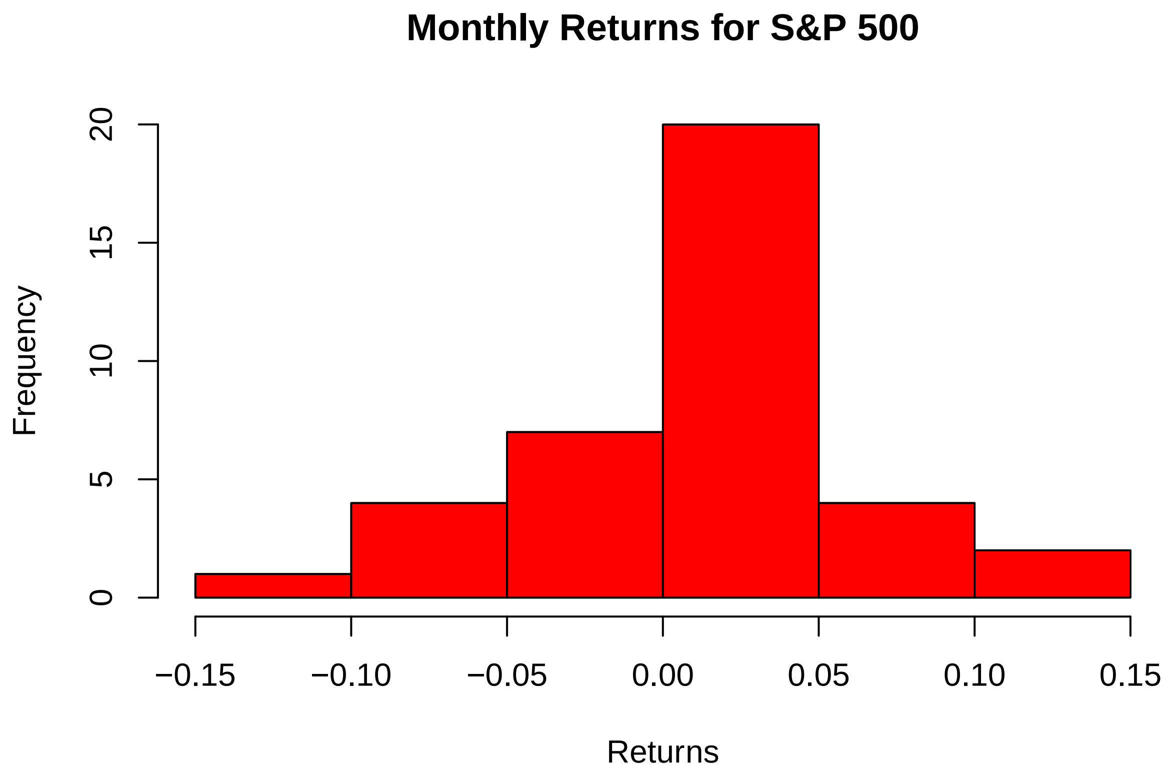
f. Plot histograms of returns to AAPL and returns to S&P 500. Comment on the distribution of the returns.

```r
hist(rAAPL,
     main='Monthly Returns for AAPL',
     xlab='Returns',
     col='blue',
    )
```



We can observe that AAPL has a fat left tail (to the downside).

```
hist(rGSPC,
     main='Monthly Returns for S&P 500',
     xlab='Returns',
     col='red',
    )
```



We can observe that compared to AAPL, S&P 500 resembles, but not necessary is, a normal distribution with fairly even tails.

g. Test whether the distributions of returns to AAP and returns to S&P 500 are normal or not.

We run Shapiro-Wilk normality test on AAPL monthly returns:

```
shapiro.test(as.vector(rAAPL))
```

```
##
##  Shapiro-Wilk normality test
##
## data:  as.vector(rAAPL)
## W = 0.96862, p-value = 0.3559
```

We can see that p-value > .05 implying that the distribution of AAPL monthly returns is not significantly different from a normal distribution, and thus, we can assume normality.

Similar with S&P 500:

```
shapiro.test(as.vector(rGSPC))
```

```
##
```

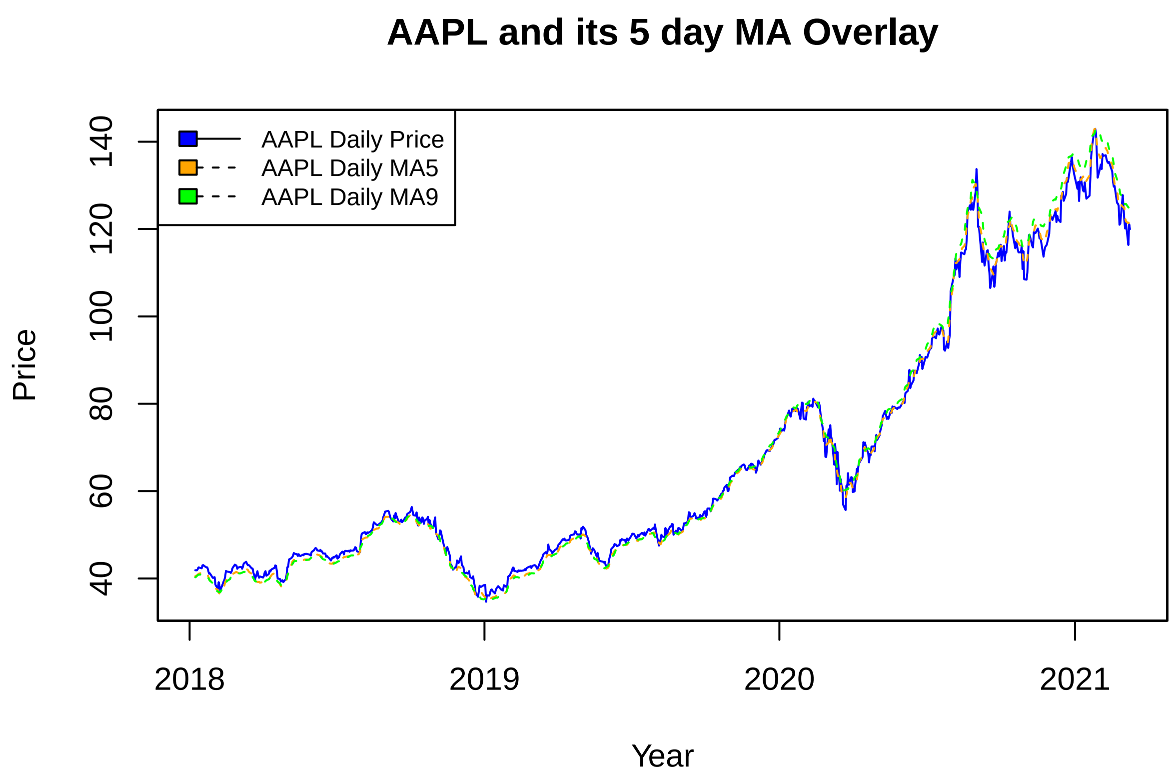
```
##  Shapiro-Wilk normality test
##
## data:  as.vector(rGSPC)
## W = 0.95624, p-value = 0.143
```

Similarly, we can see that the distribution of S&P 500 monthly returns is not significantly different from a normal distribution, and thus, we can assume normality.

h. Fit $MA(5)$ and $MA(9)$ on AAPL data and compare the accuracy criterion of the fits.

Fitting $MA(5)$:

```r
library(forecast)

ma5 <- ma(adjAAPL, order=5)  # Get MA5 in ts format
ma9 <- ma(adjAAPL, order=9)  # Get MA9 in ts format
ma5 <- as.xts(ma5)  # Convert to xts
ma9 <- as.xts(ma9)  # Convert to xts
index(ma5) <- index(adjAAPL)  # Replace MA5 index with adjAAPL index
index(ma9) <- index(adjAAPL)  # Replace MA9 index with adjAAPL index
adjAAPL_wMA <- na.omit(merge.xts(adjAAPL, ma5, ma9))  # Merge adjAAPL and its MA5 to a single object

plot(as.zoo(adjAAPL_wMA[, "AAPL.Adjusted"]), screens = 1,
     main = "AAPL and its 5 day MA Overlay",
     xlab = "Year", ylab = "Price", col = "blue")

par(new=TRUE)

# Plot MA5 and suppress axis value
plot(as.zoo(adjAAPL_wMA[, "ma5"]),
     screens = 1,
     lty=2,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "orange")

par(new=TRUE)

# Plot MA9 and suppress axis value
plot(as.zoo(adjAAPL_wMA[, "ma9"]),
     screens = 1,
     lty=2,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "green")

# Add legend
legend("topleft",
       c("AAPL Daily Price", "AAPL Daily MA5", "AAPL Daily MA9"),
       lty=c(1,2,2),
       cex = 0.75,
       fill = c("blue", "orange", "green"))
```



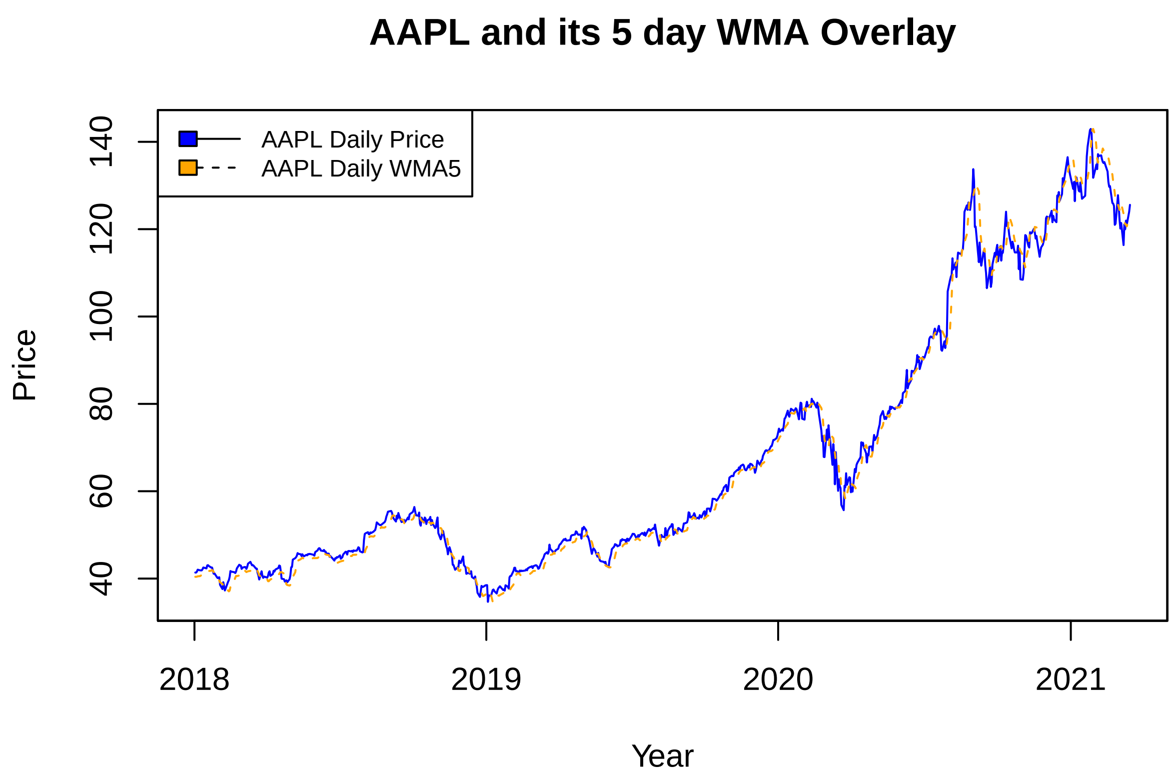
We can observe that both MAs smooth out the actual price graph and remove some of its sharp movements.

i. Fit WMA(5) on AAPL data and compare its accuracy of the fit with MA(5) and MA(9)

Add WMA(5)

```
library(pracma)

wma5 <- movavg(adjAAPL, n=5, type="w")  # Calculate WMA5
adjAAPL_wWMA <- merge.xts(adjAAPL, wma5)  # Merge WMA5 to AAPL daily price xts object

plot(as.zoo(adjAAPL_wWMA[, "AAPL.Adjusted"]), screens = 1,
     main = "AAPL and its 5 day WMA Overlay",
     xlab = "Year", ylab = "Price", col = "blue")

par(new=TRUE)

# Plot MA5 and suppress axis value
plot(as.zoo(adjAAPL_wWMA[, "wma5"]),
     screens = 1,
     lty=2,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "orange")
```

```r
# Add legend
legend("topleft",
       c("AAPL Daily Price", "AAPL Daily WMA5"),
       lty=c(1,2),
       cex = 0.75,
       fill = c("blue", "orange"))
```



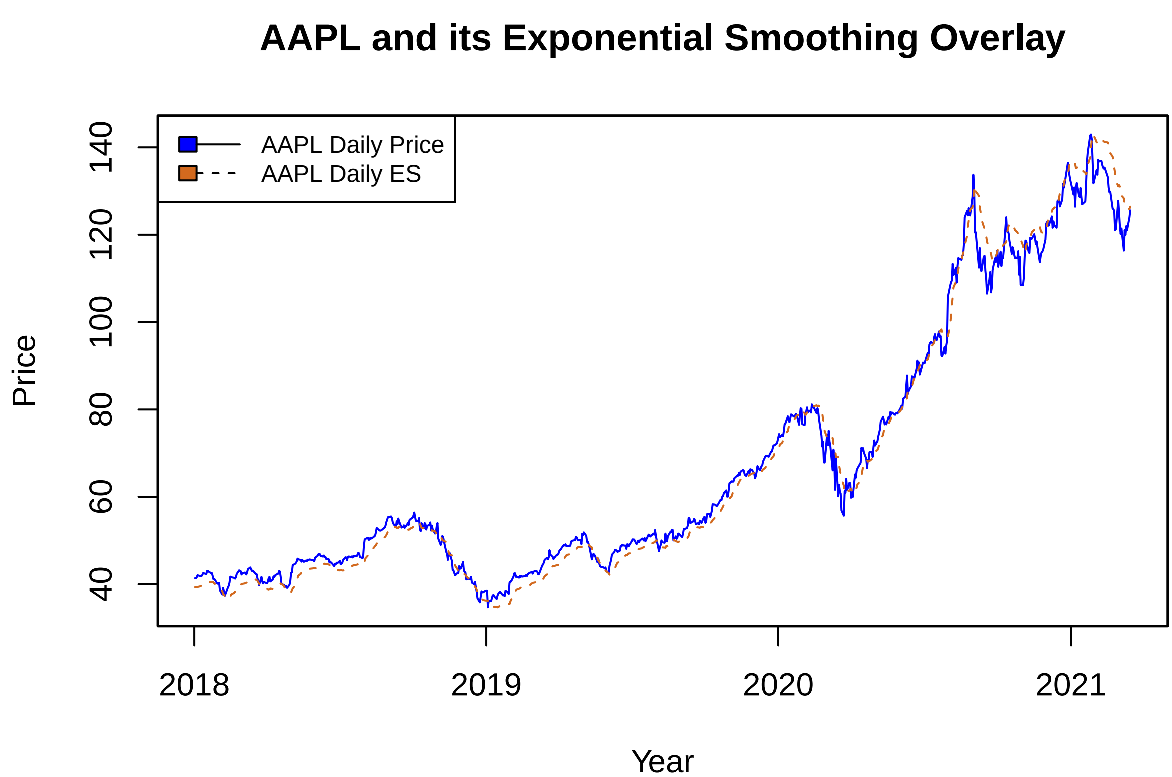
We can abserve that WMA5 fits daily price a lot better than MA5 and MA9.

j. Fit simple ES to AAPL data and compare its accuracy of the fit with MA(5) and MA(9).

Simple ES

```r
#Simple Exponential Smoothing with 1 period ahead
fit1 <- ses(adjAAPL, alpha=0.2, initial="simple", h=1)
es <- as.xts(fitted(fit1))
index(es) <- index(adjAAPL)
adjAAPL_wES <- merge.xts(adjAAPL, es)  # Merge ES to AAPL daily price xts object

plot(as.zoo(adjAAPL_wES[, "AAPL.Adjusted"]), screens = 1,
     main = "AAPL and its Exponential Smoothing Overlay",
     xlab = "Year", ylab = "Price", col = "blue")
```

```r
par(new=TRUE)

# Plot ES and suppress axis value
plot(as.zoo(adjAAPL_wES[, "es"]),
     screens = 1,
     lty=2,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "chocolate")

# Add legend
legend("topleft",
       c("AAPL Daily Price", "AAPL Daily ES"),
       lty=c(1,2),
       cex = 0.75,
       fill = c("blue", "chocolate"))
```



We can observe that ES5 also fit daily price quite well, but poorer than WMA5.

k. Do a one-period ahead forecasting of AAPL price for using simple ES model.

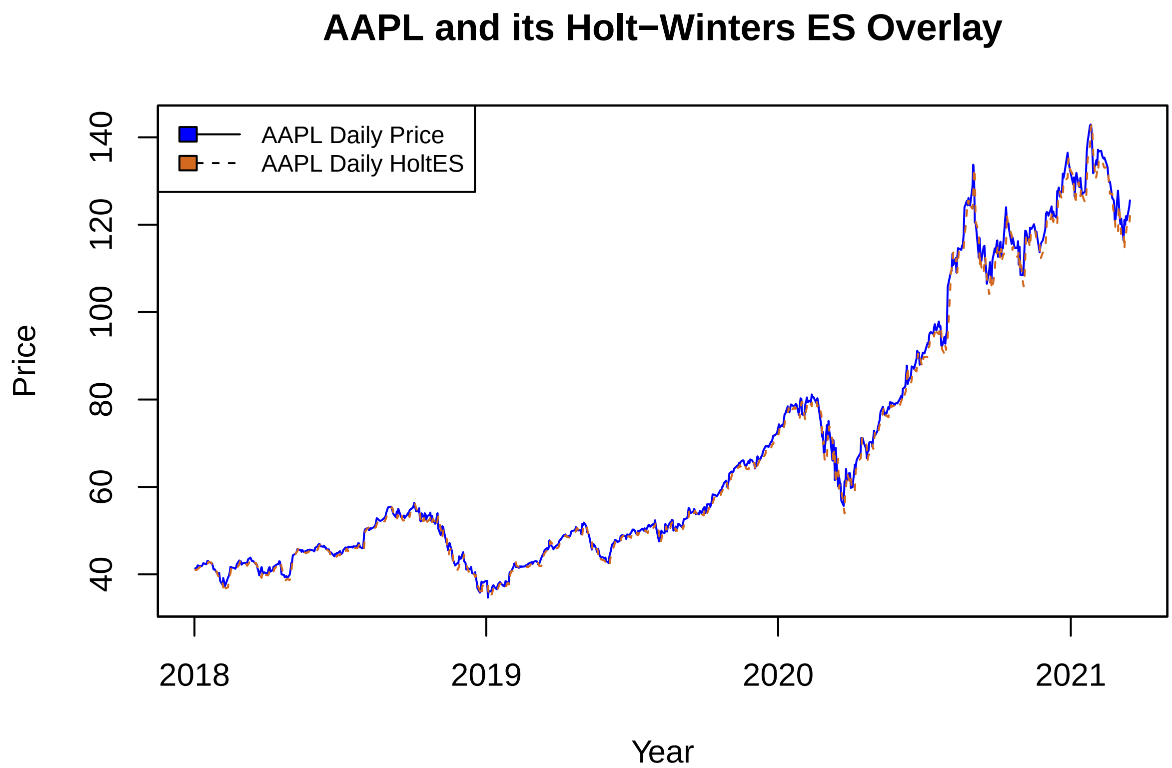
```r
fit1
```

```
##      Point Forecast     Lo 80     Hi 80     Lo 95     Hi 95
```

```
## 807        122.7628 119.3411 126.1845 117.5297 127.9959
```

Fit Holt-Winter ES to AAPL data and compare its accuracy of the fit with MA(5) and MA(9).

```r
#Simple Exponential Smoothing with 1 period ahead
holt1 <- holt(adjAAPL, alpha=0.8, beta=0.2, initial="simple", exponential=TRUE, h=3)
holt1ES <- as.xts(fitted(holt1))
index(holt1ES) <- index(adjAAPL)
adjAAPL_wHoltES <- merge.xts(adjAAPL, holt1ES)  # Merge Holt ES to AAPL daily price xts object

plot(as.zoo(adjAAPL_wHoltES[, "AAPL.Adjusted"]), screens = 1,
     main = "AAPL and its Holt-Winters ES Overlay",
     xlab = "Year", ylab = "Price", col = "blue")

par(new=TRUE)

# Plot ES and suppress axis value
plot(as.zoo(adjAAPL_wHoltES[, "holt1ES"]),
     screens = 1,
     lty=2,
     xaxt = "n", yaxt = "n",
     xlab = "", ylab = "",
     col = "chocolate")

# Add legend
legend("topleft",
       c("AAPL Daily Price", "AAPL Daily HoltES"),
       lty=c(1,2),
       cex = 0.75,
       fill = c("blue", "chocolate"))
```


AAPL and its Holt−Winters ES Overlay
AAPL Daily Price
AAPL Daily HoltES
Price
140
120
100
80
60
40
2018
2019
2020
2021
Year

“‘