Econometrics III
Assignment Part III

Thao Le (523716)
David Gyaraki (582340)

Contents

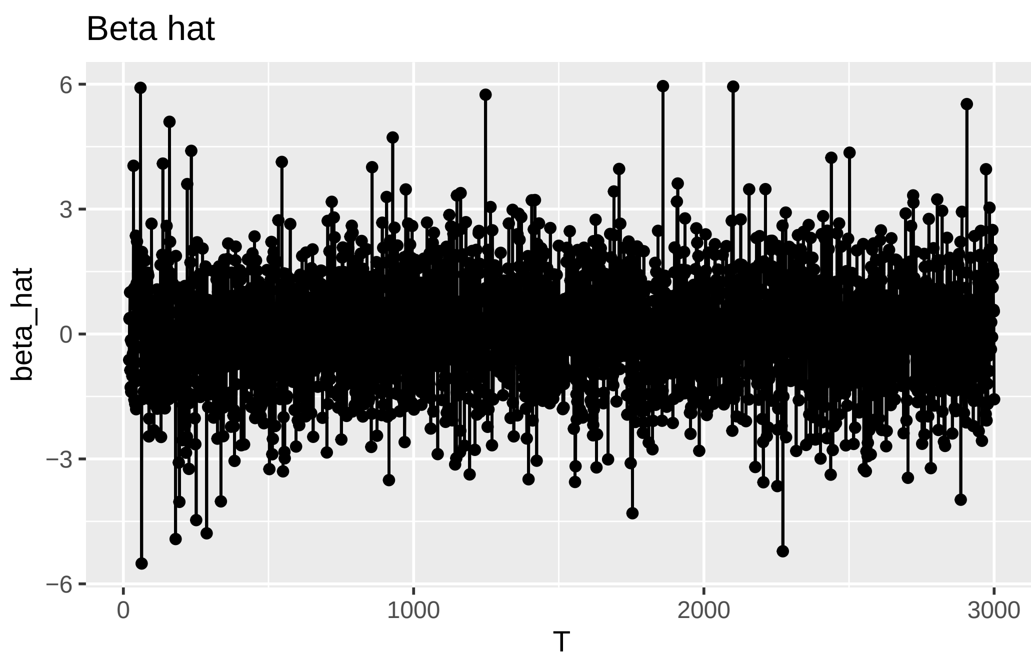
```r
# load packages
if(!require(pacman)){install.packages("pacman")}

p_load(devtools,tidyverse,dplyr,ggplot2,latex2exp,stargazer, fixest,
 ↪  modelsummary, knitr, readr, tseries, lmtest, forecast, dynlm, vars,
 ↪  gridExtra)

dfAssign_p3 <- as.data.frame(read_csv("data/data_assign_p3.csv"))
```

1 Question 1

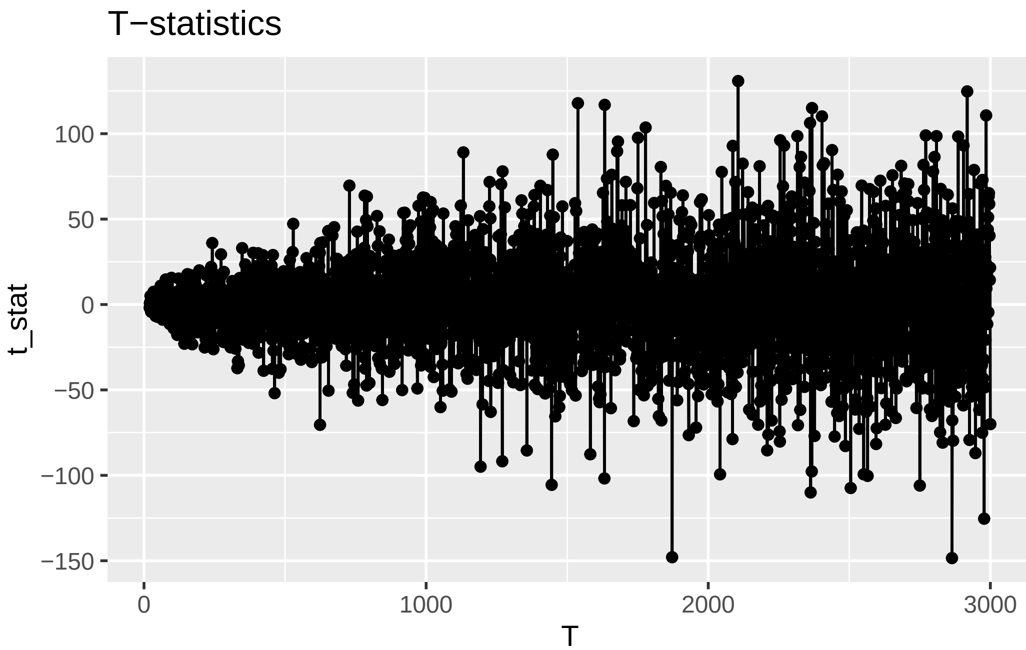
```r
# Set parameter for each walk
sigma_u = 1
sigma_v = 0.5

# Function to simulate X_t, Y_t
simulate_2RW = function(n_steps, sigma_u, sigma_v){
  # Simulate the random steps for each walk and store in vectors
  steps1 <- rnorm(n_steps, mean = 0, sd = sigma_u)
  steps2 <- rnorm(n_steps, mean = 0, sd = sigma_v)

  Y <- cumsum(steps1)
  X <- cumsum(steps2)
  return(list(X_t=X,Y_t=Y))
}

# Set simulation parameters
max_T <- 3000
step <- 1
init_step = 20

# Initialize dataframe to store results
sim_results <- data.frame(
  T = numeric(),
  beta_hat = numeric(),
  t_stat = numeric(),
  R_squared = numeric(),
  stringsAsFactors = FALSE
```

```
)

# For each sample size T from T = 200,400,..., 1000
for (T in seq(init_step, max_T, 1)) {
  lRWs = simulate_2RW(T, sigma_u, sigma_v)
  X_t = lRWs$X_t
  Y_t = lRWs$Y_t

  # Perform linear regression and get summary stats
  model <- lm(Y_t ~ X_t)
  summary <- tidy(model)
  beta_hat <- summary$estimate[2]
  t_stat <- summary$statistic[2]
  R_squared <- summary(model)$r.squared
  # Store results
  sim_results <- sim_results %>%
    add_row(T = T,
            beta_hat = beta_hat,
            t_stat = t_stat,
            R_squared = R_squared)
}
```

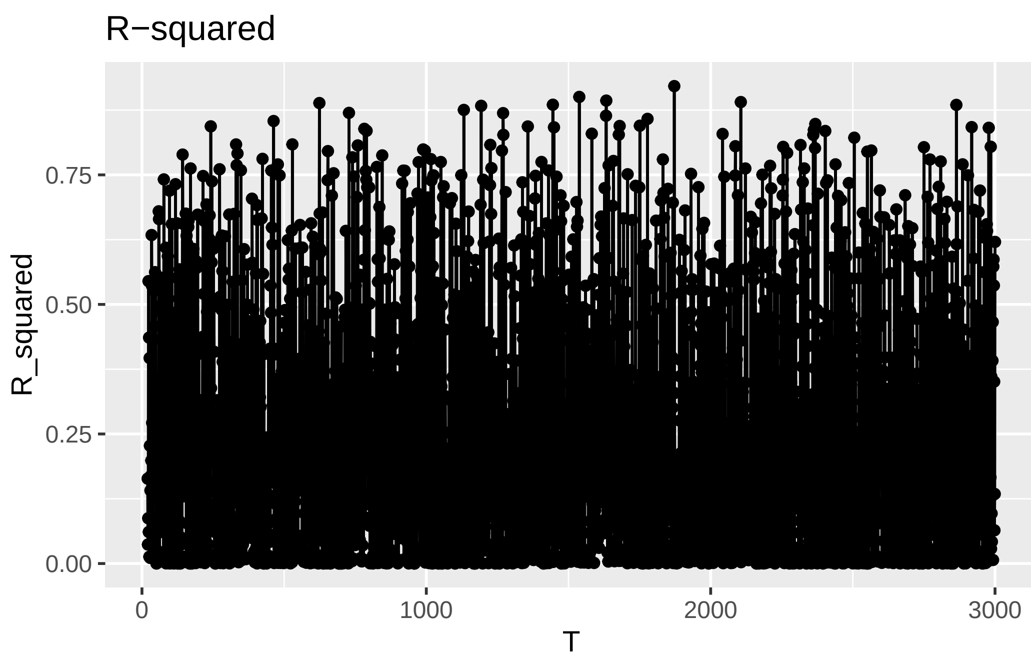
```
ggplot(sim_results, aes(x = T, y = beta_hat)) +
  geom_point() +
  geom_line() +
  ggtitle("Beta hat")
```




```r
ggplot(sim_results, aes(x = T, y = t_stat)) +
  geom_point() +
  geom_line() +
  ggtitle("T-statistics")
```




```r
ggplot(sim_results, aes(x = T, y = R_squared)) +
  geom_point() +
  geom_line() +
  ggtitle("R-squared")
```



From the scatter plot of values of $\hat{\beta}$, T-statistics, and R-squared values with increasing sample sizes, we can verify that:

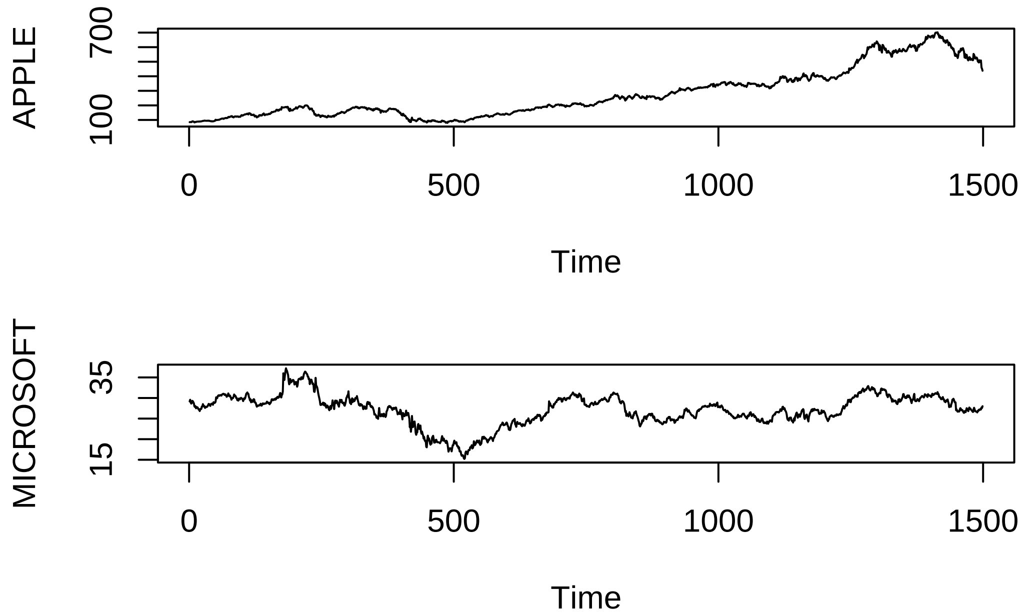
$$\hat{\beta} \xrightarrow{d} RV$$

$$\text{T-statistics} \xrightarrow{p} \infty$$

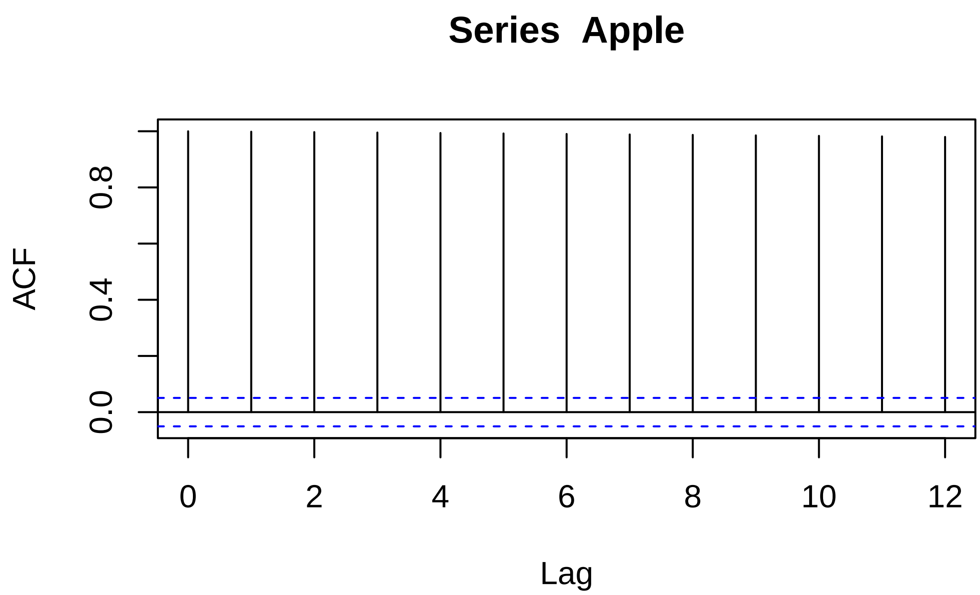
$$R^2 \xrightarrow{d} RV$$

2 Question 2

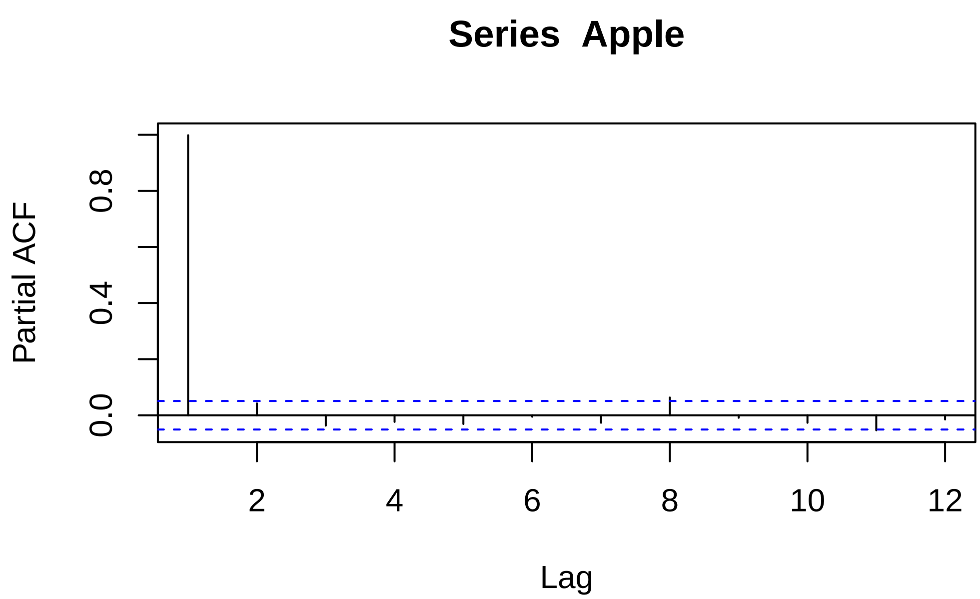
```r
# Plot both time series
par(mfrow = c(2, 1))
par(mar = c(4, 4, 2, 1) + 0.1)  # Adjust margins
plot(dfAssign_p3$APPLE, type = "l", xlab = "Time", ylab = "APPLE")
plot(dfAssign_p3$MICROSOFT, type = "l", xlab = "Time", ylab =
↪  "MICROSOFT")
```



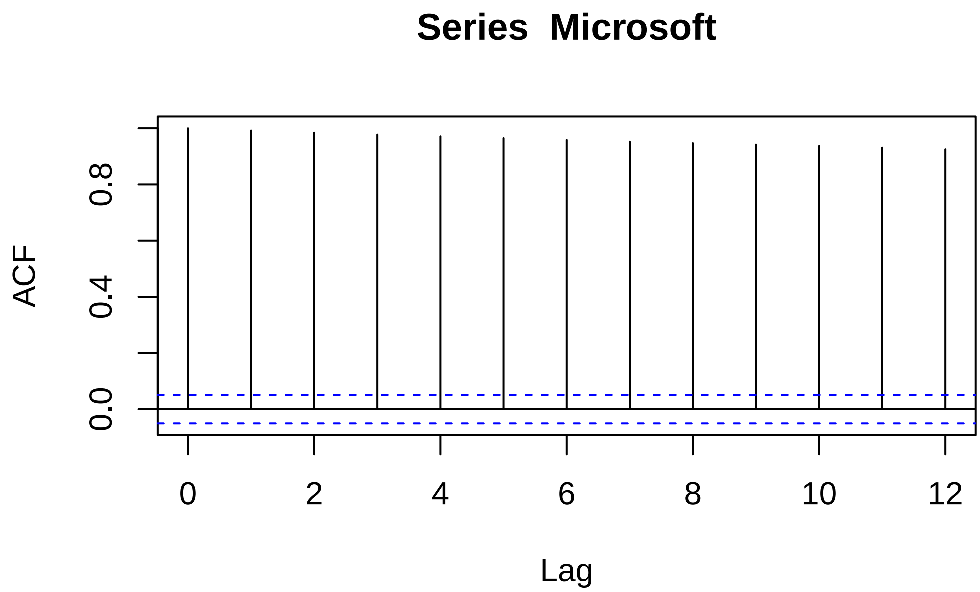

```
# Calculate and Plot ACF and PACF for Apple
Apple = dfAssign_p3[["APPLE"]]
acf(Apple, 12, pl=T)
```




```r
pacf(Apple, 12, pl=T)
```



```r
# Calculate ACF and PACF for Microsoft
Microsoft = dfAssign_p3[["MICROSOFT"]]
acf(Microsoft, 12, pl=T)
```




```r
pacf(Microsoft, 12, pl=F)
```

```
Partial autocorrelations of series 'Microsoft', by lag

     1      2      3      4      5      6      7      8      9     10     11 
 0.992  0.006  0.053  0.031  0.002 -0.002  0.016  0.025  0.054 -0.019 -0.039 
    12 
-0.012 
```

We can see that all 12 lags of the autocorrelation function (ACF) are significant but only one lag of the partial autocorrelation function (PACF) is significant, it suggests that the dynamic of these stocks may have a high degree of autocorrelation but can be adequately modeled using a simple autoregressive (AR) model with one lag. To be specific:

The high degree of autocorrelation indicated by the significant ACF lags suggests that past values of the stock prices are highly correlated with its current values. This can indicate that the stock price is predictable to some extent and that past values may provide useful information for forecasting future values.

The low number of significant lags in the PACF suggests that the significant ACF lags can be adequately explained by a simple AR(1) model. This implies that the current value of the time series can be explained by its previous value and that other factors such as trend, seasonality, or exogenous variables may not be significant.

We suspect that this also means that the time series can be integrated with order 1.

3 Question 3

```
ADF_Apple = ur.df(dfAssign_p3$APPLE, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_Exxon = ur.df(dfAssign_p3$EXXON_MOBIL, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_Ford = ur.df(dfAssign_p3$FORD, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_GenElectric = ur.df(dfAssign_p3$GEN_ELECTRIC, type =  "none", lags =
↪  10,
      selectlags = "BIC")
ADF_Intel = ur.df(dfAssign_p3$INTEL, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_Micro = ur.df(dfAssign_p3$MICROSOFT, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_Netflix = ur.df(dfAssign_p3$NETFLIX, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_Nokia = ur.df(dfAssign_p3$NOKIA, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_SP500 = ur.df(dfAssign_p3$SP500, type =  "none", lags = 10,
      selectlags = "BIC")
ADF_Yahoo = ur.df(dfAssign_p3$YAHOO, type =  "none", lags = 10,
      selectlags = "BIC")
```

	ADF test statictis	lags used	p-value	10 $\%$ critical value
Apple	0.892	1	0.372	-1.62
Exxon Mobil	0.264	2	0.792	-1.62
Ford	0.132	1	0.895	-1.62
Gen Electric	-1.218	1	0.174	-1.62
Intel	-0.252	1	0.800	-1.62
Microsoft	-0.313	1	0.754	-1.62
Netflix	0.136	1	0.892	-1.62
Nokia	-1.183	3	0.237	-1.62
SP500	-0.045	1	0.964	-1.62
Yahoo	-1.179	1	0.239	-1.62

Table 1: ADF test results for all 10 stocks

We carried out the ADF unit-root test for all 10 time series using the G2S approach based on the Schwart Information Criterion. This test did not include a trend in the model. The G2S approach finds that including 1 lag in the analysis would fit the time series the most.

At 90$\%$ confidence level, the results show that we did not reject the null hypothesis for any of the time series, suggesting that there is a unit-root presence and the time-series are non-stationary.

This result is in accordance with our expectation as we did not expect to reject any of the null hypothesis for the time series. This is because these time series are stock prices and usually follow a random walk process. Random walk processes are non-stationary and they should have a unit root.

4 Question 4

First, we need to check whether to include a drift (constant term α) in the random walk processes of Apple and Microsoft stock prices. To do this, we rewrite the random walk model by taking the first difference:

$$\Delta X_t = \alpha + \epsilon_t$$

Then, the first difference can be described by the constant term plus a Gaussian noise term. To test the significance of α, we proceed with a normal linear regression below:

```
apple_lm =lm(diff(Apple) ~ 1)
summary(apple_lm)
```

```
Call:
lm(formula = diff(Apple) ~ 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-49.046  -2.876   0.164   2.839  52.794 

Coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   0.2358     0.1729   1.364    0.173

Residual standard error: 6.692 on 1497 degrees of freedom
```

```
microsoft_lm =lm(diff(Microsoft) ~ 1)
summary(microsoft_lm)
```

```
Call:
lm(formula = diff(Microsoft) ~ 1)

Residuals:
    Min      1Q  Median      3Q     Max 
-2.0792 -0.2392  0.0008  0.2408  4.4408 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) -0.0007744  0.0120066  -0.064    0.949

Residual standard error: 0.4647 on 1497 degrees of freedom
```

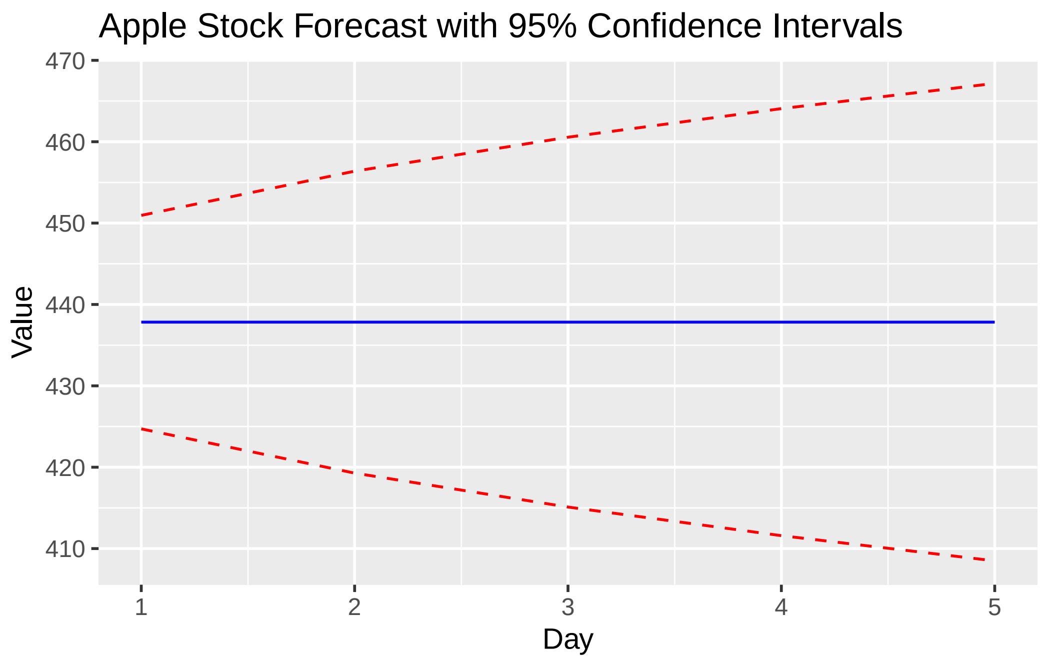
The regression results shows that the constant term is not significant at any levels of significant. Thus, we decide to not include a drift in the models. The final model to be fitted to these stock prices is:

$$X_t = X_{t-1} + \epsilon_t$$

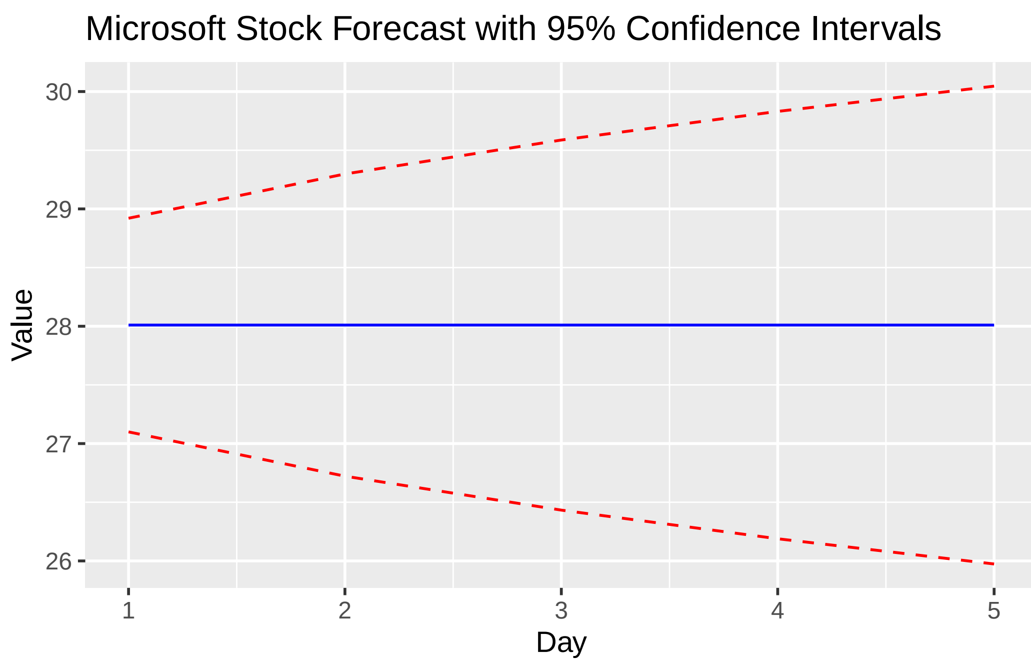
To produce the forecast for these stock prices, we have:

```
Apple_forecast <- rwf(Apple,drift = FALSE, h = 5, level = c(95))
Apple_df <- data.frame(Value = Apple_forecast$mean, Lower =
↪  Apple_forecast$lower, Upper = Apple_forecast$upper)
```

```r
colnames(Apple_df) = c("Value","Lower","Upper")
ggplot(Apple_df, aes(x = 1:length(Apple_df$Value))) +
  geom_line(aes(y = Value), color = "blue") +
  geom_line(aes(y = Lower), color = "red", linetype = "dashed") +
  geom_line(aes(y = Upper), color = "red", linetype = "dashed") +
  labs(x = "Day", y = "Value", title = "Apple Stock Forecast with 95%
   ↪ Confidence Intervals")
```




```r
MCS_forecast <- rwf(Microsoft,drift = FALSE, h = 5, level = c(95))
MCS_df <- data.frame(Value = MCS_forecast$mean, Lower =
 ↪ MCS_forecast$lower, Upper = MCS_forecast$upper)
colnames(MCS_df) = c("Value","Lower","Upper")
ggplot(MCS_df, aes(x = 1:length(MCS_df$Value))) +
  geom_line(aes(y = Value), color = "blue") +
  geom_line(aes(y = Lower), color = "red", linetype = "dashed") +
  geom_line(aes(y = Upper), color = "red", linetype = "dashed") +
  labs(x = "Day", y = "Value", title = "Microsoft Stock Forecast with
   ↪ 95% Confidence Intervals")
```

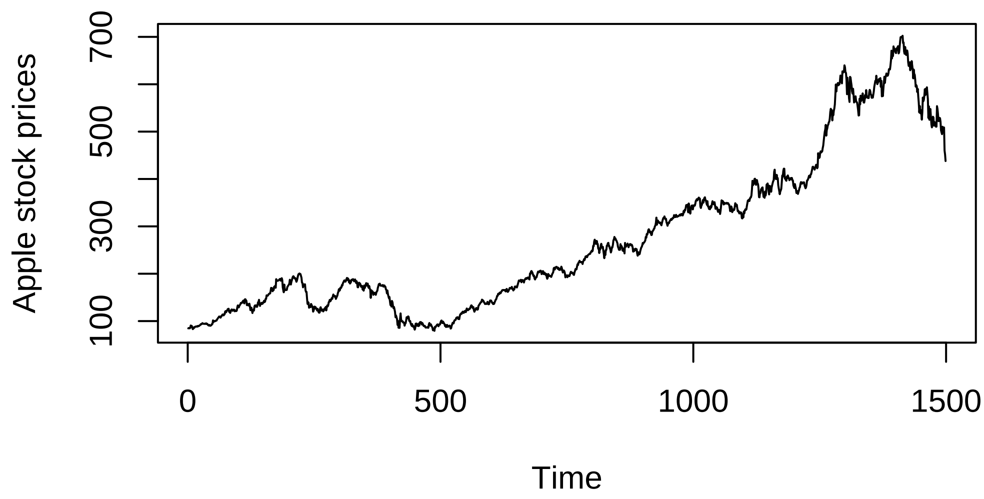


According to the forecast, we can see that since there is no statistically significant evidence of a drift, the stock of these companies could fluctuates up or down. However, Since there is a higher evidence of a positive drift for Apple stocks (looking at the plots of stock prices below) and the p-values of having a positive constant term in the tests above, it is more likely that there exists a positive trend in Apple stocks. Moreover, the price of Apple stocks seems to be higher than that of Microsoft. Generally speaking, Apple's stocks is more valuable than Microsoft's stock.

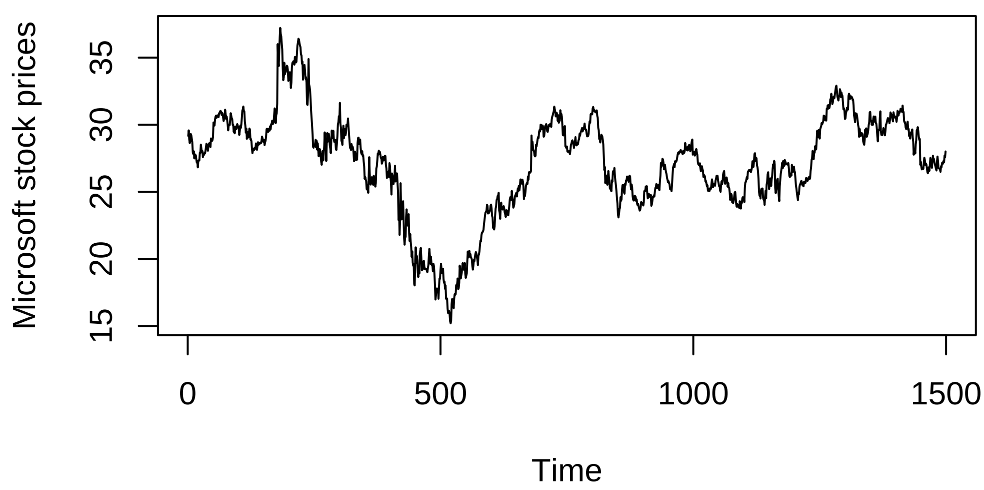
However, looking at the plots below, the last several periods of Apple stocks seems to follow a downward trends, so it might be that Apple's stock can decrease soon. Thus, I would not dare to give investment advice based solely on this forecast and would check more background information of the companies' current prospects.

Because there is no drift included in the model, we cannot say their stock values is expected to increase or decrease.

```
plot(Apple, type = "l", xlab = "Time", ylab = "Apple stock prices")
```




```
plot(Microsoft, type = "l", xlab = "Time", ylab ="Microsoft stock
↪  prices")
```



5 Question 5

```r
Exxon = dfAssign_p3$EXXON_MOBIL
Micro_Exxon = lm(Microsoft ~ Exxon)
summary(Micro_Exxon)
```

```
Call:
lm(formula = Microsoft ~ Exxon)

Residuals:
     Min       1Q   Median       3Q      Max 
-10.2357  -1.7355   0.2203   1.9912   8.3225 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)    
(Intercept) 11.244927   0.712719   15.78   <2e-16 ***
Exxon        0.202856   0.009054   22.40   <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 3.229 on 1497 degrees of freedom
Multiple R-squared:  0.2511,	Adjusted R-squared:  0.2506 
F-statistic: 501.9 on 1 and 1497 DF,  p-value: < 2.2e-16
```

Based on the simple linear regression above, we find statistically significant evidence of contemporaneous relations between two stock prices. However, since both stock prices are non-stationary I(1) (according to teh ADF tests above), this result is not reliable since it is subjected to the spurious regression problems.

The variables in spurious regression appear to be statistically significant and have a high correlation coefficient, but there is no causal relationship between them. Spurious regression happens when two time series have unrelated trends yet are highly connected due to random chance. Because both time series are going upwards over time, their correlation coefficient will also grow, resulting in the impression of a relationship between them. This relationship, however, is deceptive, and any inferences formed from it will be untrustworthy.

Thus, we do not agree to the claim that changes in Microsoft stock prices are largely explained by fluctuations in the stock price of Exxon Mobile.