1-5

```r
library(quantmod)
```

```
## Warning: package 'quantmod' was built under R version 4.2.3

## Loading required package: xts

## Warning: package 'xts' was built under R version 4.2.3

## Loading required package: zoo

## Warning: package 'zoo' was built under R version 4.2.3

##
## Attaching package: 'zoo'

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

## Loading required package: TTR

## Warning: package 'TTR' was built under R version 4.2.3

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

```r
library(urca)
```

```
## Warning: package 'urca' was built under R version 4.2.3
```

```r
library(forecast)
```

```
## Warning: package 'forecast' was built under R version 4.2.3
```

```r
library(tseries)
```

```
## Warning: package 'tseries' was built under R version 4.2.3
```

```r
library(fGarch)
```

```
## Warning: package 'fGarch' was built under R version 4.2.3
```

```
## NOTE: Packages 'fBasics', 'timeDate', and 'timeSeries' are no longer
## attached to the search() path when 'fGarch' is attached.
##
## If needed attach them yourself in your R script by e.g.,
##         require("timeSeries")

##
## Attaching package: 'fGarch'

## The following object is masked from 'package:TTR':
##
##     volatility
```

```
library(zoo)
library(tseries)
library(rugarch)
```

```
## Warning: package 'rugarch' was built under R version 4.2.3

## Loading required package: parallel

##
## Attaching package: 'rugarch'

## The following object is masked from 'package:stats':
##
##     sigma
```

```
library(ARDL)
```

```
## Warning: package 'ARDL' was built under R version 4.2.3

## To cite the ARDL package in publications:
##
## Use this reference to refer to the validity of the ARDL package.
##
##   Natsiopoulos, Kleanthis, and Tzeremes, Nickolaos G. (2022). ARDL
##   bounds test for cointegration: Replicating the Pesaran et al. (2001)
##   results for the UK earnings equation using R. Journal of Applied
##   Econometrics, 37(5), 1079-1090. https://doi.org/10.1002/jae.2919
##
## Use this reference to cite this specific version of the ARDL package.
##
##   Kleanthis Natsiopoulos and Nickolaos Tzeremes (2023). ARDL: ARDL, ECM
##   and Bounds-Test for Cointegration. R package version 0.2.4.
##   https://CRAN.R-project.org/package=ARDL
```

```
library(vars)
```

```
## Warning: package 'vars' was built under R version 4.2.3
```

```
## Loading required package: MASS

## Warning: package 'MASS' was built under R version 4.2.3

## Loading required package: strucchange

## Warning: package 'strucchange' was built under R version 4.2.3

## Loading required package: sandwich

## Warning: package 'sandwich' was built under R version 4.2.3

## Loading required package: lmtest

## Warning: package 'lmtest' was built under R version 4.2.3
```

```
library(stargazer)
```

```
##
## Please cite as:

##  Hlavac, Marek (2022). stargazer: Well-Formatted Regression and Summary Statistics Tables.

##  R package version 5.2.3. https://CRAN.R-project.org/package=stargazer
```

```
data = na.omit(read.csv('AMZN3.csv'))
qqq = read.csv('QQQ3.csv')
```

1-5

```
closing = na.omit(data$Close) # closing price
log_closing = na.omit(log(data$Close)) # log closing price
log_return = na.omit(diff(log(data$Close))) # log return
time = as.Date(data$Date, format = '%m/%d/%y')

##Check for the trend (the Augmented Dickey-Fuller (ADF) test)
summary(ur.df(log_return, type='trend', lags=20, selectlags="BIC"))
```

```
##
## ###############################################
## # Augmented Dickey-Fuller Test Unit Root Test #
## ###############################################
##
## Test regression trend
##
##
## Call:
```

```
## lm(formula = z.diff ~ z.lag.1 + 1 + tt + z.diff.lag)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.081122 -0.010263 -0.000424  0.010472  0.128190
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -4.419e-04  3.549e-03  -0.125    0.901
## z.lag.1     -9.925e-01  1.016e-01  -9.773   <2e-16 ***
## tt           1.910e-06  2.715e-05   0.070    0.944
## z.diff.lag   6.910e-03  7.235e-02   0.096    0.924
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.02134 on 191 degrees of freedom
## Multiple R-squared:  0.4931, Adjusted R-squared:  0.4852
## F-statistic: 61.94 on 3 and 191 DF,  p-value: < 2.2e-16
##
##
## Value of test-statistic is: -9.7732 31.8397 47.7587
##
## Critical values for test statistics:
##       1pct  5pct 10pct
## tau3 -3.99 -3.43 -3.13
## phi2  6.22  4.75  4.07
## phi3  8.43  6.49  5.47
```

```
# No drift or time trend

adf.test(log_return)
```

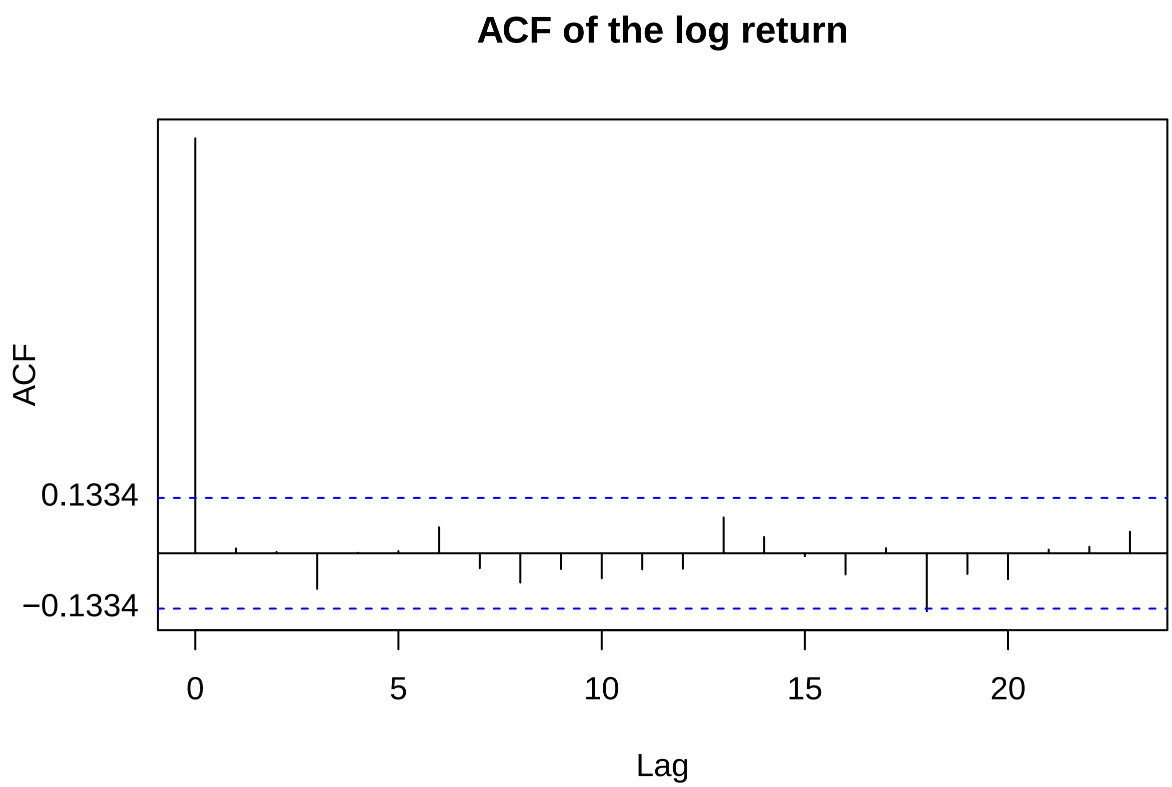
```
## Warning in adf.test(log_return): p-value smaller than printed p-value
```

```
##
##  Augmented Dickey-Fuller Test
##
## data:  log_return
## Dickey-Fuller = -5.615, Lag order = 5, p-value = 0.01
## alternative hypothesis: stationary
```

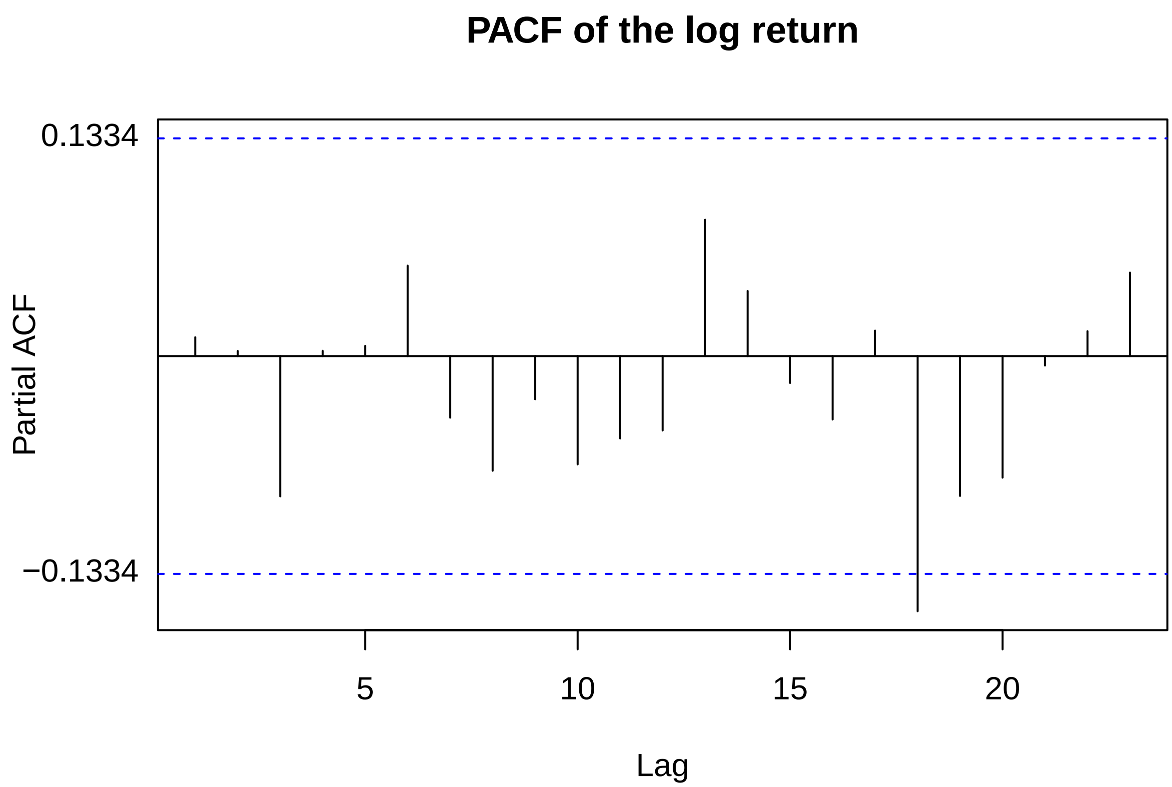
```
# The data is stationary. Difference is not needed.

##Check for the seasonality
Sys.setlocale("LC_TIME","english")
```

```
## [1] "English_United States.1252"
```

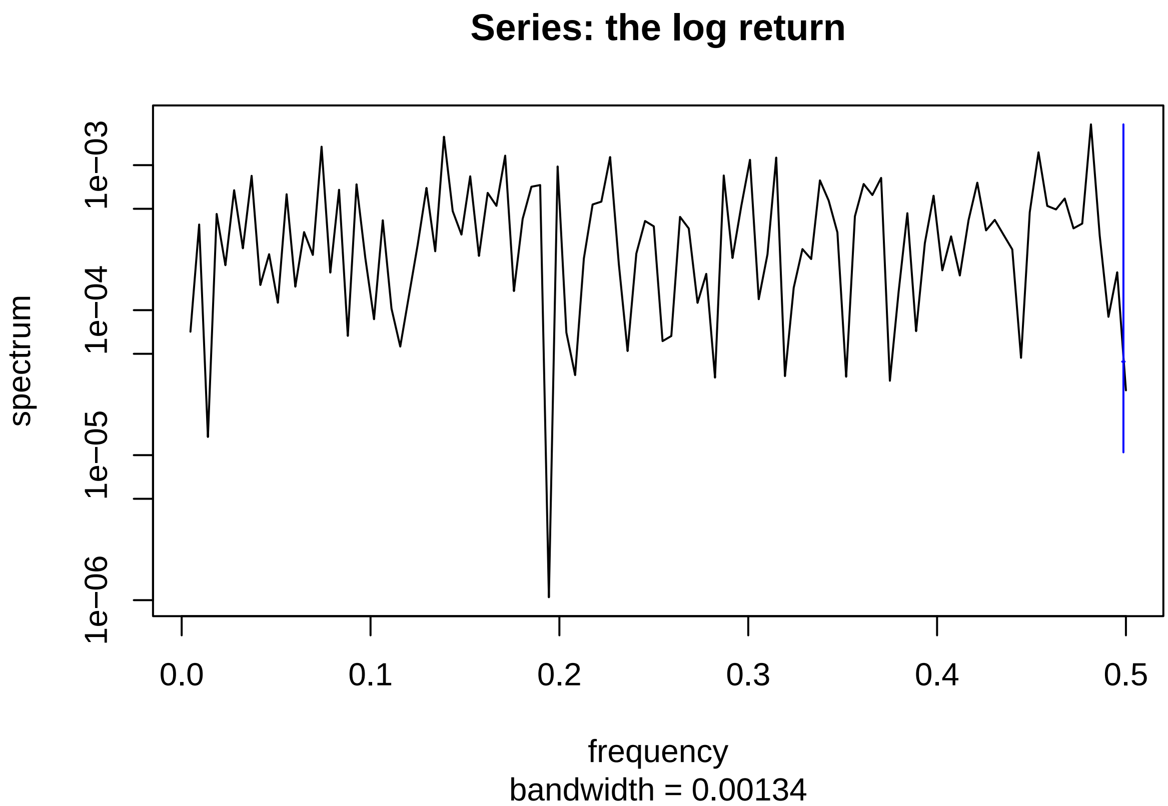
```
n = length(log_return)
acf(log_return,main="ACF of the log return",yaxt="n")
ci=qnorm(c(0.025, 0.975))/sqrt(n)
text(y=ci,par("usr")[1],labels=round(ci,4),pos=2,xpd=TRUE)
```




```
pacf(log_return,main="PACF of the log return",yaxt="n")
text(y=ci,par("usr")[1],labels=round(ci,4),pos=2,xpd=TRUE)
```



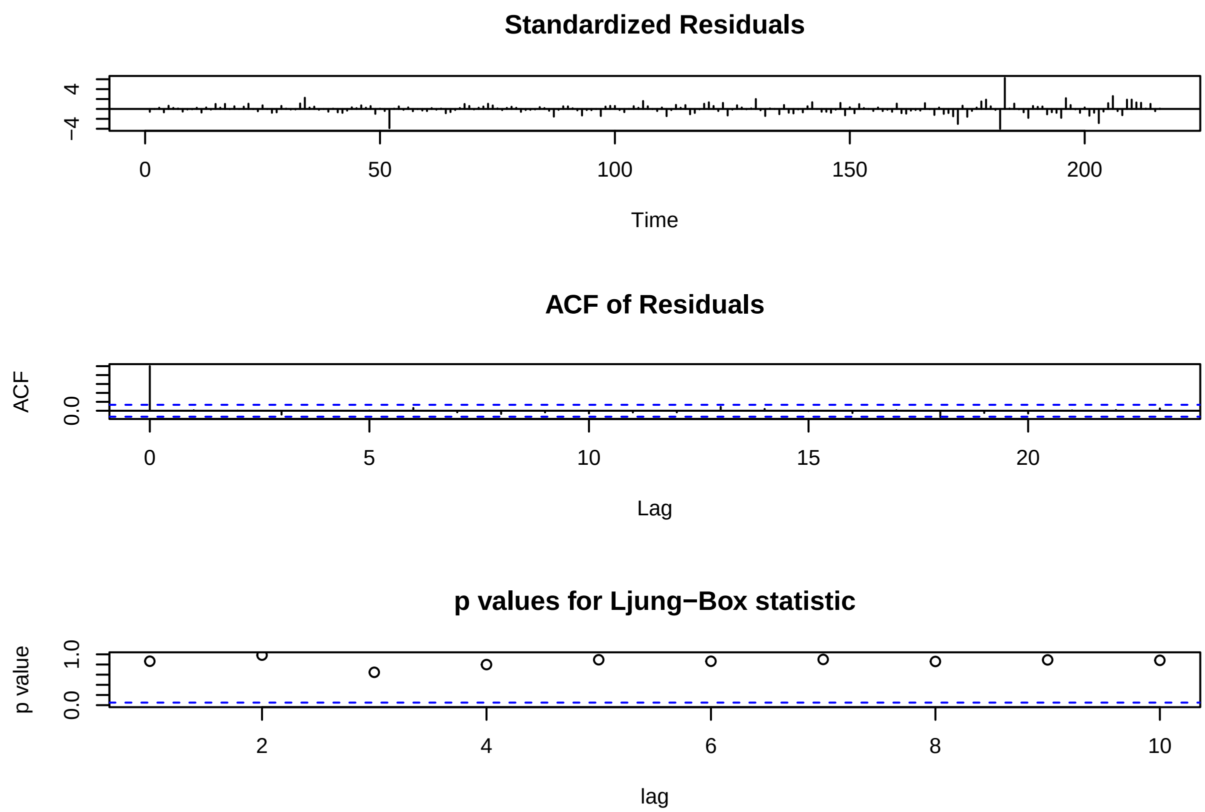

```
spec.pgram(log_return,main="Series: the log return")
```




```
# we cannot find any evidence for seasonality.

# ARIMA
fit = auto.arima(log_return, max.p=25, max.q=25, ic="bic",
                 seasonal=F, lambda=NULL,
                 stepwise=FALSE, approximation=FALSE
)
summary(fit)
```

```
## Series: log_return
## ARIMA(0,0,0) with zero mean
##
## sigma^2 = 0.0004122:  log likelihood = 535.26
## AIC=-1068.53   AICc=-1068.51   BIC=-1065.15
##
## Training set error measures:
##                        ME       RMSE        MAE MPE MAPE      MASE       ACF1
## Training set 3.679287e-06 0.02030252 0.01364303 100  100 0.7113085 0.01156582
```

```
# ARIMA(0,0,0)
# AIC=-1068.53   AICc=-1068.51   BIC=-1065.15
tsdiag(fit)
```




```
shapiro.test(fit$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  fit$residuals
## W = 0.90091, p-value = 9.066e-11
```

```
# The null-hypothesis of this test is that the population is normally distributed.
# The null hypothesis is rejected and there is evidence that the residuals tested
# are not normally distributed.

# ARIMA-Garch
arma_model <- fit
garch_spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1,1)),
                         mean.model = list(armaOrder = c(0,0)))
garch_fit <- ugarchfit(spec = garch_spec, data = arma_model$residuals)
garch_fit
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
## *---------------------------------*
##
## Conditional Variance Dynamics
```

```
## ------------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error   t value Pr(>|t|)
## mu     -0.000052    0.001264 -0.040965 0.967324
## omega   0.000057    0.000059  0.961635 0.336233
## alpha1  0.176791    0.116660  1.515432 0.129663
## beta1   0.692533    0.244439  2.833147 0.004609
##
## Robust Standard Errors:
##         Estimate  Std. Error   t value Pr(>|t|)
## mu     -0.000052    0.001681 -0.030803  0.97543
## omega   0.000057    0.000155  0.367605  0.71317
## alpha1  0.176791    0.246679  0.716686  0.47357
## beta1   0.692533    0.618424  1.119836  0.26278
##
## LogLikelihood : 551.6753
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.0711
## Bayes        -5.0086
## Shibata      -5.0717
## Hannan-Quinn -5.0458
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      1.154  0.2827
## Lag[2*(p+q)+(p+q)-1][2]     1.159  0.4495
## Lag[4*(p+q)+(p+q)-1][5]     1.826  0.6601
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.8176  0.3659
## Lag[2*(p+q)+(p+q)-1][5]    1.7388  0.6813
## Lag[4*(p+q)+(p+q)-1][9]    2.1102  0.8914
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]   0.00738 0.500 2.000  0.9315
## ARCH Lag[5]   0.19069 1.440 1.667  0.9677
## ARCH Lag[7]   0.24533 2.315 1.543  0.9953
##
```

```
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  0.5767
## Individual Statistics:
## mu     0.09939
## omega  0.13649
## alpha1 0.38462
## beta1  0.23453
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.07 1.24 1.6
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           0.2749 0.7837
## Negative Sign Bias  1.2610 0.2087
## Positive Sign Bias  0.2736 0.7847
## Joint Effect        3.5101 0.3195
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     36.04     0.010446
## 2    30     54.56     0.002791
## 3    40     56.96     0.031542
## 4    50     62.24     0.096958
##
##
## Elapsed time : 0.164294
```

```
# infocriteria(garch_fit)
# ARIMA-GARCH would be more apt for modeling time series data with volatility clustering, which is a ch

arma_model <- auto.arima(log_closing)
arma_model # difference --> return
```

```
## Series: log_closing
## ARIMA(0,1,0)
##
## sigma^2 = 0.0004125:  log likelihood = 535.26
## AIC=-1068.53   AICc=-1068.51   BIC=-1065.15
```

```
garch_spec <- ugarchspec(variance.model = list(model = "sGARCH", garchOrder = c(1,1)),
                         mean.model = list(armaOrder = c(0,0)))
garch_fit <- ugarchfit(spec = garch_spec, data = arma_model$residuals)
garch_fit
```

```
##
## *---------------------------------*
## *          GARCH Model Fit        *
```

```
## *---------------------------------*
##
## Conditional Variance Dynamics
## -----------------------------------
## GARCH Model  : sGARCH(1,1)
## Mean Model   : ARFIMA(0,0,0)
## Distribution : norm
##
## Optimal Parameters
## ------------------------------------
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000011    0.001258 0.008966 0.992846
## omega   0.000053    0.000057 0.921653 0.356710
## alpha1  0.171100    0.118432 1.444702 0.148542
## beta1   0.708322    0.242718 2.918297 0.003519
##
## Robust Standard Errors:
##         Estimate  Std. Error  t value Pr(>|t|)
## mu      0.000011    0.001700 0.006636  0.99470
## omega   0.000053    0.000153 0.345321  0.72985
## alpha1  0.171100    0.263670 0.648915  0.51639
## beta1   0.708322    0.629155 1.125831  0.26024
##
## LogLikelihood : 554.8004
##
## Information Criteria
## ------------------------------------
##
## Akaike       -5.0765
## Bayes        -5.0142
## Shibata      -5.0772
## Hannan-Quinn -5.0513
##
## Weighted Ljung-Box Test on Standardized Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                      1.123  0.2892
## Lag[2*(p+q)+(p+q)-1][2]     1.128  0.4586
## Lag[4*(p+q)+(p+q)-1][5]     1.765  0.6748
## d.o.f=0
## H0 : No serial correlation
##
## Weighted Ljung-Box Test on Standardized Squared Residuals
## ------------------------------------
##                         statistic p-value
## Lag[1]                     0.8009  0.3708
## Lag[2*(p+q)+(p+q)-1][5]    1.7219  0.6854
## Lag[4*(p+q)+(p+q)-1][9]    2.1066  0.8919
## d.o.f=2
##
## Weighted ARCH LM Tests
## ------------------------------------
##             Statistic Shape Scale P-Value
## ARCH Lag[3]   0.00871 0.500 2.000  0.9256
```

```
## ARCH Lag[5]   0.20940 1.440 1.667  0.9633
## ARCH Lag[7]   0.27066 2.315 1.543  0.9942
##
## Nyblom stability test
## ------------------------------------
## Joint Statistic:  0.574
## Individual Statistics:
## mu     0.1057
## omega  0.1289
## alpha1 0.3692
## beta1  0.2254
##
## Asymptotic Critical Values (10% 5% 1%)
## Joint Statistic:          1.07 1.24 1.6
## Individual Statistic:     0.35 0.47 0.75
##
## Sign Bias Test
## ------------------------------------
##                    t-value   prob sig
## Sign Bias           0.2348 0.8146
## Negative Sign Bias  1.3131 0.1906
## Positive Sign Bias  0.2821 0.7781
## Joint Effect        3.5975 0.3083
##
##
## Adjusted Pearson Goodness-of-Fit Test:
## ------------------------------------
##   group statistic p-value(g-1)
## 1    20     37.29     0.007307
## 2    30     53.55     0.003641
## 3    40     56.00     0.038131
## 4    50     63.88     0.075110
##
##
## Elapsed time : 0.113224
```

```
forecasted_returns <- ugarchforecast(garch_fit, n.ahead = 1)
last_close_price <- closing[length(closing)]
(price_forecast <-as.numeric(last_close_price*exp(forecasted_returns@forecast$seriesFor)))
```

```
## [1] 3273.027
```

```
(lower_interval <- as.numeric(price_forecast*exp(qnorm(0.025)*forecasted_returns@forecast$sigmaFor)))
```

```
## [1] 3152.746
```

```
(upper_interval <- as.numeric(price_forecast*exp(qnorm(0.975)*forecasted_returns@forecast$sigmaFor)))
```

```
## [1] 3397.896
```

```r
# Print the forecasted closing price and prediction interval
cat("1-day ahead closing price forecast:", price_forecast, "\n")
```

```
## 1-day ahead closing price forecast: 3273.027
```

```r
cat("95% Prediction Interval: (", lower_interval, ", ", upper_interval, ")\n")
```

```
## 95% Prediction Interval: ( 3152.746 ,  3397.896 )
```

```r
# The true value 3295.47 is inside the 95% CI.
```

6-10

```r
lc_AAPL = log_closing
lc_QQQ = na.omit(log(qqq$Close))
ardl_data = data.frame(cbind(lc_AAPL, lc_QQQ))
ardl_model <- ardl(lc_AAPL~lc_QQQ, data = ardl_data, order = c(1,1))
summary(ardl_model)
```

```
##
## Time series regression with "ts" data:
## Start = 2, End = 217
##
## Call:
## dynlm::dynlm(formula = full_formula, data = data, start = start,
##     end = end)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.068513 -0.005829 -0.000144  0.005746  0.105821
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)     0.24277    0.12716   1.909   0.0576 .
## L(lc_AAPL, 1)   0.95057    0.01820  52.235   <2e-16 ***
## lc_QQQ          1.09254    0.07489  14.588   <2e-16 ***
## L(lc_QQQ, 1)   -1.06585    0.07469 -14.271   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01429 on 212 degrees of freedom
## Multiple R-squared:  0.9525, Adjusted R-squared:  0.9519
## F-statistic:  1418 on 3 and 212 DF,  p-value: < 2.2e-16
```

```r
# UECM (Unrestricted Error Correction Model)
uecm_model <- uecm(ardl_model)
summary(uecm_model)
```

```
##
## Time series regression with "ts" data:
## Start = 2, End = 217
##
## Call:
## dynlm::dynlm(formula = full_formula, data = data, start = start,
##     end = end)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.068513 -0.005829 -0.000144  0.005746  0.105821
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)    0.24277    0.12716   1.909  0.05760 .
## L(lc_AAPL, 1) -0.04943    0.01820  -2.716  0.00714 **
## L(lc_QQQ, 1)   0.02669    0.02053   1.300  0.19493
## d(lc_QQQ)      1.09254    0.07489  14.588  < 2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01429 on 212 degrees of freedom
## Multiple R-squared:  0.514,  Adjusted R-squared:  0.5071
## F-statistic: 74.73 on 3 and 212 DF,  p-value: < 2.2e-16
```

```
# RECM (Restricted Error Correction Model)
recm_model <- recm(ardl_model, case = 2)
summary(recm_model)
```

```
##
## Time series regression with "zooreg" data:
## Start = 2, End = 217
##
## Call:
## dynlm::dynlm(formula = full_formula, data = data, start = start,
##     end = end)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.068513 -0.005829 -0.000144  0.005746  0.105821
##
## Coefficients:
##           Estimate Std. Error t value Pr(>|t|)
## d(lc_QQQ)  1.09254    0.07358  14.849  < 2e-16 ***
## ect       -0.04943    0.01768  -2.796  0.00564 **
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.01422 on 214 degrees of freedom
##   (0 observations deleted due to missingness)
## Multiple R-squared:  0.514,  Adjusted R-squared:  0.5094
## F-statistic: 113.2 on 2 and 214 DF,  p-value: < 2.2e-16
```

```r
# VAR
VARselect(ardl_data, lag.max = 4, type = 'const')
```

```
## $selection
## AIC(n)  HQ(n)  SC(n) FPE(n)
##      1      1      1      1
##
## $criteria
##                    1             2             3             4
## AIC(n) -1.713415e+01 -1.710473e+01 -1.707798e+01 -1.706439e+01
## HQ(n)  -1.709588e+01 -1.704095e+01 -1.698870e+01 -1.694959e+01
## SC(n)  -1.703946e+01 -1.694692e+01 -1.685705e+01 -1.678034e+01
## FPE(n)  3.620234e-08  3.728368e-08  3.829544e-08  3.882164e-08
```

```r
# estimation
vare_diff = VAR(ardl_data, p = 1, type = 'const')
summary(vare_diff)
```

```
##
## VAR Estimation Results:
## =========================
## Endogenous variables: lc_AAPL, lc_QQQ
## Deterministic variables: const
## Sample size: 216
## Log Likelihood: 1245.079
## Roots of the characteristic polynomial:
## 0.9599 0.9599
## Call:
## VAR(y = ardl_data, p = 1, type = "const")
##
##
## Estimation results for equation lc_AAPL:
## ========================================
## lc_AAPL = lc_AAPL.l1 + lc_QQQ.l1 + const
##
##            Estimate Std. Error t value Pr(>|t|)
## lc_AAPL.l1  0.96000    0.02568  37.378   <2e-16 ***
## lc_QQQ.l1  -0.01737    0.02867  -0.606   0.5453
## const       0.42676    0.17870   2.388   0.0178 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.02018 on 213 degrees of freedom
## Multiple R-Squared: 0.9049,  Adjusted R-squared: 0.904
## F-statistic:  1013 on 2 and 213 DF,  p-value: < 2.2e-16
##
##
## Estimation results for equation lc_QQQ:
## =======================================
## lc_QQQ = lc_AAPL.l1 + lc_QQQ.l1 + const
##
##            Estimate Std. Error t value Pr(>|t|)
```

```
## lc_AAPL.l1 0.008634   0.016639   0.519    0.604
## lc_QQQ.l1  0.959669   0.018576  51.662   <2e-16 ***
## const      0.168414   0.115765   1.455    0.147
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
##
## Residual standard error: 0.01307 on 213 degrees of freedom
## Multiple R-Squared: 0.9493,  Adjusted R-squared: 0.9488
## F-statistic:  1994 on 2 and 213 DF,  p-value: < 2.2e-16
##
##
##
## Covariance matrix of residuals:
##            lc_AAPL    lc_QQQ
## lc_AAPL 0.0004071 0.0001867
## lc_QQQ  0.0001867 0.0001708
##
## Correlation matrix of residuals:
##          lc_AAPL lc_QQQ
## lc_AAPL  1.0000 0.7078
## lc_QQQ   0.7078 1.0000
```

```r
# residuals test
serial.test(vare_diff)
```

```
##
##  Portmanteau Test (asymptotic)
##
## data:  Residuals of VAR object vare_diff
## Chi-squared = 60.533, df = 60, p-value = 0.4564
```

```r
# forecast of differenced data
varf_diff = predict(vare_diff, n.ahead = 1, ci = 0.95)
exp(varf_diff$fcst$lc_AAPL)
```

```
##                  fcst   lower    upper       CI
## lc_AAPL.fcst 3275.585 3148.58 3407.713 1.040337
```

```r
exp(varf_diff$fcst$lc_QQQ)
```

```
##                 fcst    lower    upper       CI
## lc_QQQ.fcst 359.9891 350.8839 369.3306 1.025949
```

```r
# The true values 3295.47 and 359.35 are inside the 95% CIs.

# 5
upper_interval-lower_interval # range
```

```
## [1] 245.1497
```

```r
abs(price_forecast-3295.47) # error
```

```
## [1] 22.44309
```

```r
# 10
exp(varf_diff$fcst$lc_AAPL)[3]-exp(varf_diff$fcst$lc_AAPL)[2] # range
```

```
## [1] 259.1332
```

```r
abs(exp(varf_diff$fcst$lc_AAPL)[1]-3295.47) # error
```

```
## [1] 19.88482
```

```r
# The 95 CI of VAR(1) is narrower, but the prediction error of ARIMA-garch is lower
# Both have their own advantages
```