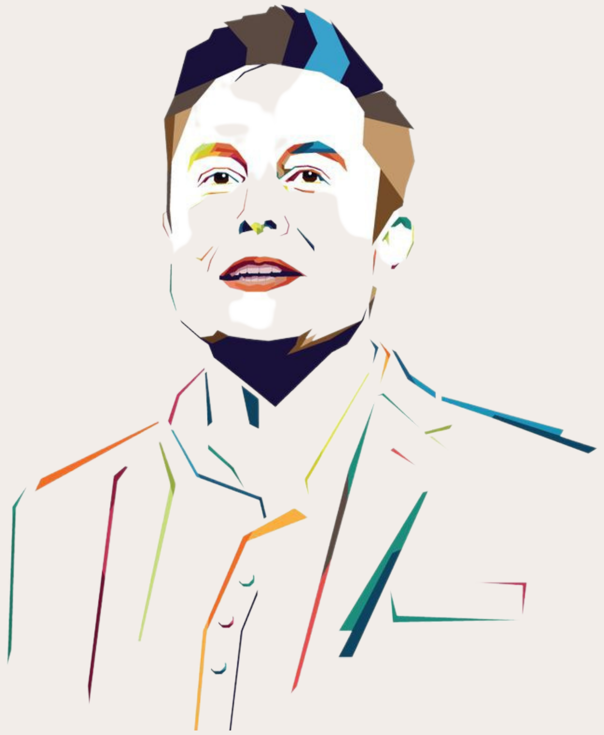


ARCH MODEL

RAVISH KAMATH
VI NGUYEN
YILIN CHEN

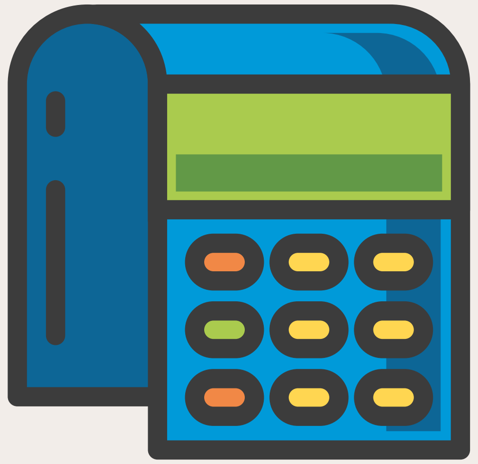


Table of Content

ARCH

autoregressive conditionally heteroscedastic

An ARCH model is a model for the variance of a time series. ARCH models are used to describe a changing, possibly volatile variance.

Where does it come from

ARCH models were created in the context of econometric and finance problems having to do with the amount that investments or stocks increase (or decrease) per time period, so there's a tendency to describe them as models for that type of variable.

What are we interested in

The variable of interest in these problems are

- The proportion gained or lost since the last time

$$y_t = \frac{x_t - x_{t-1}}{x_{t-1}}$$

- The logarithm of the ratio of this time's value to last time's value

$$log(\frac{x_t}{x_{t-1}}) = log(x_t) - log(x_{t-1})$$

- ARCH model could be used for any series that has periods of increased or decreased variance.

  For example: a property of residuals after an ARIMA model has been fit to the data.

Example of high volatility

ARCH(1)

Suppose that we are modeling the variance of a series y_t . The ARCH(1) model for the variance of model y_t is that conditional on y_{t-1}, the variance at time t is

$$\mathrm{Var}(y_t|y_{t-1}) = \sigma_t^2 = \alpha_0 + \alpha_1 y_{t-1}^2$$

If we assume that the series has mean = 0 (this can always be done by centering), the ARCH model could be written as

$$\begin{aligned} y_t &= \sigma_t \epsilon_t \\ \sigma_t^2 &= \alpha_0 + \alpha_1 y_{t-1}^2 \end{aligned}$$

where

$$\begin{aligned} &\epsilon_t \sim iidN(0,1) \\ &\alpha_0 \ , \ \alpha_1 > 0 \end{aligned}$$

Properties of ARCH (1)

- If $0 < \alpha_1 < 1$, the process y_t itself is white noise, and its unconditional distribution is symmetrically distributed around zero; this distribution is leptokurtic

- If $3\alpha_1^2 < 1$, the square of the process, y_t^2, follows a casual AR(1) model with ACF given by $\rho_{y^2}(h) = \alpha_1^h \geq 0$, for all $h > o$

- If $3\alpha_1 \geq 1$ but $\alpha_1 < 1$, then y_t^2 is strictly stationary with infinite variance

Estimation of parameters α_0 & α_1

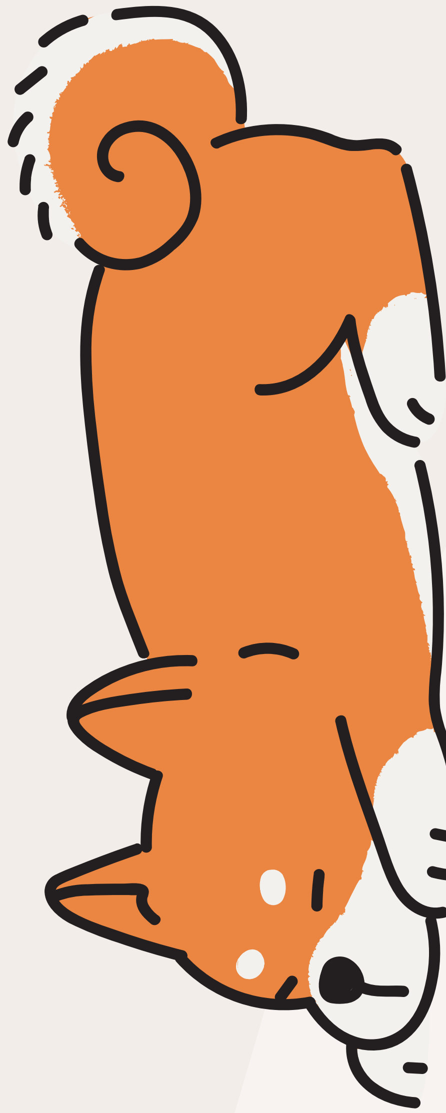
The estimators α_0 and α_1 of the ARCH(1) model is typically accomplished by conditional MLE, the conditional likelihood of the data $y_2, \ldots, y_n$ given y_1, is given by:

$$L(\alpha_0, \alpha_1 | y_1) = \prod_{t=2}^{n} f_{\alpha_0,\alpha_1}(y_1 | y_{t-1})$$

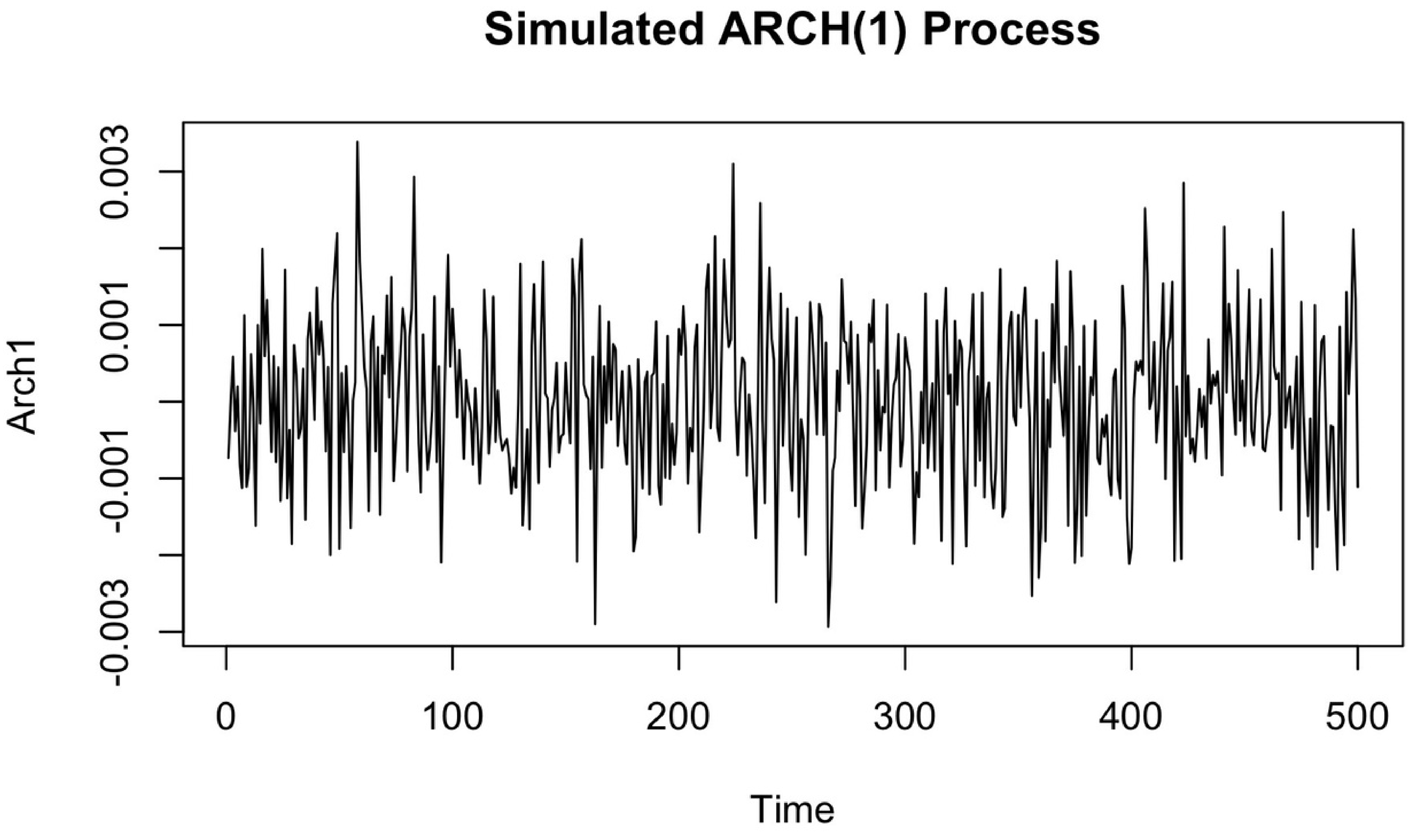
$$l(\alpha_0, \alpha_1) - lnL(\alpha_0, \alpha_1 | y_1)$$

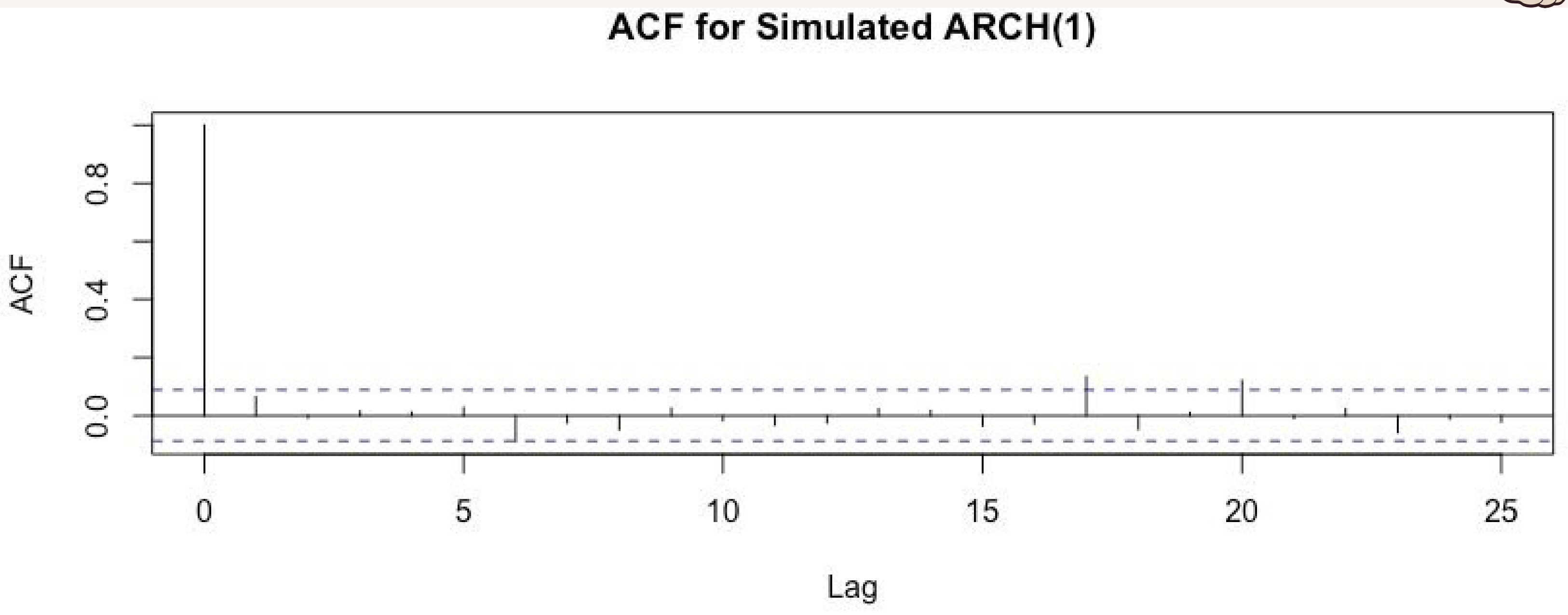
where the density $f_{\alpha_0,\alpha_1}(y_t|y_{t-1})$ is the normal density, specified earlier. Hence the criterian function to be minimized, $l(\alpha_0, \alpha_1) - lnL(\alpha_0, \alpha_1|y_1)$ is given this Yule-Walker equations:

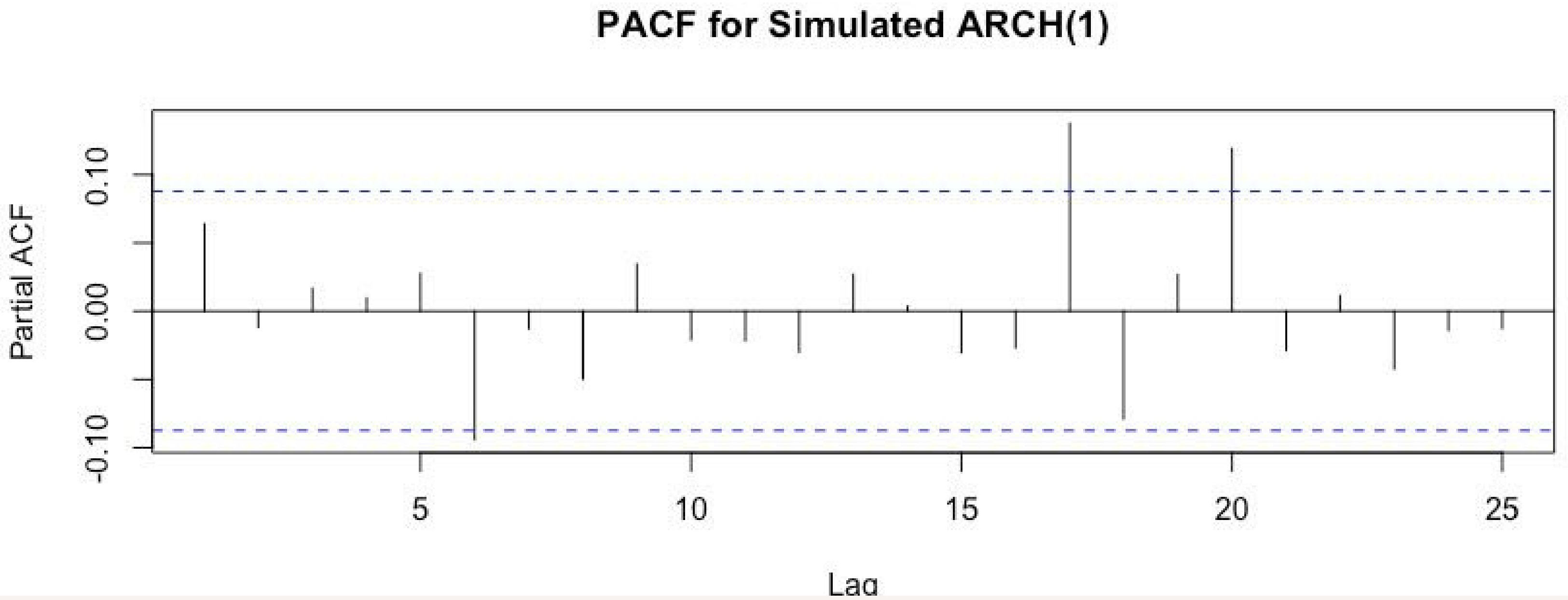
$$l(\alpha_0, \alpha_1) = \frac{1}{2} \sum_{n}^{t=2} ln(\alpha_0 + \alpha_1 y_{t-1}^2) + \frac{1}{2} \sum_{n}^{t=2} \frac{y_t^2}{\alpha_0 + \alpha_1 y_{t-1}^2}$$



Simulation







500 data point simulation for ARCH(1) Process

```r
archspec = garchSpec(model = list(alpha = c(0.1), beta = 0))
Arch1 = garchSim(spec = archspec, n = 500)
ts.plot(Arch1, main = 'Simulated ARCH(1) Process')
par(mfrow = c(2,1))
acf(Arch1^2, lag.max = 25, type="correlation",
    main="ACF for Simulated ARCH(1)" )
acf(Arch1^2, lag.max = 25, type="partial",
    main="PACF for Simulated ARCH(1)" )
```

SOOO,
we are *temporarily* done with the daunting theories

Swipe for some exciting stuff

BITCOIN

(the used to be hot piece of cake)

Dec-2017
$20,000

Jan-2018
bellow $12,000

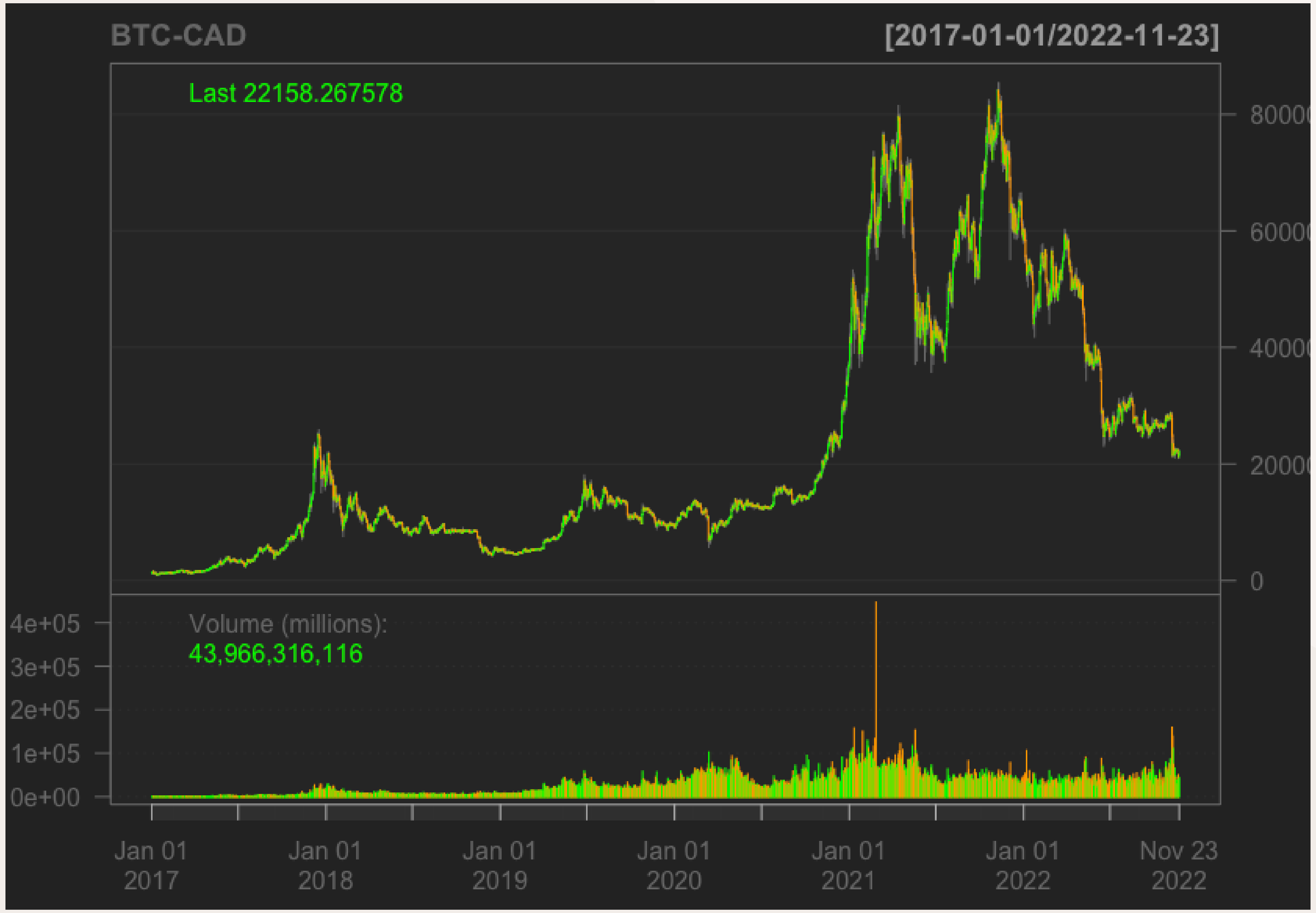
2018
Inactive $

Mar-2020
bellow $4,000

May-2021
$64,000

2022
$70,000

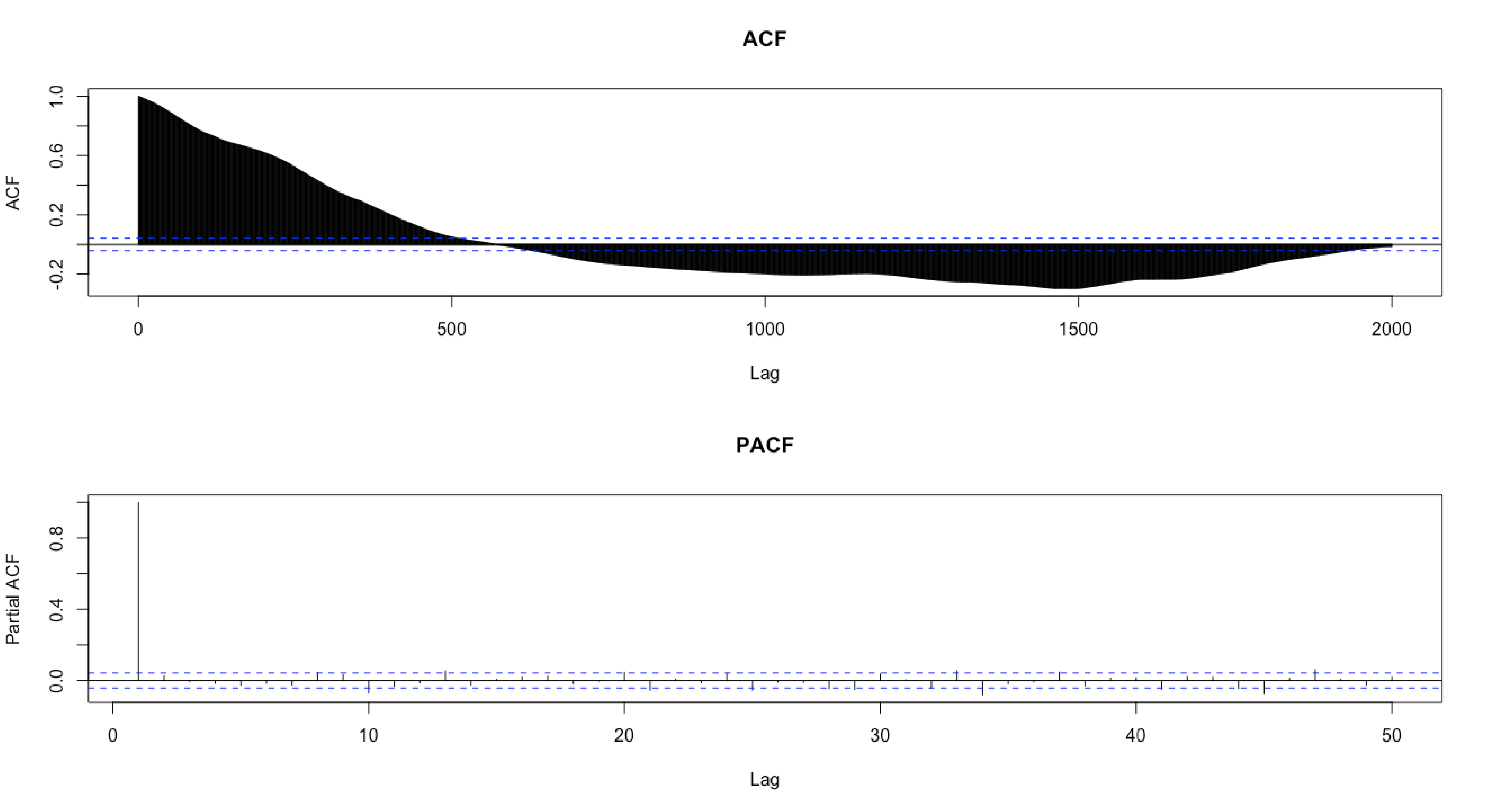
We've got proof

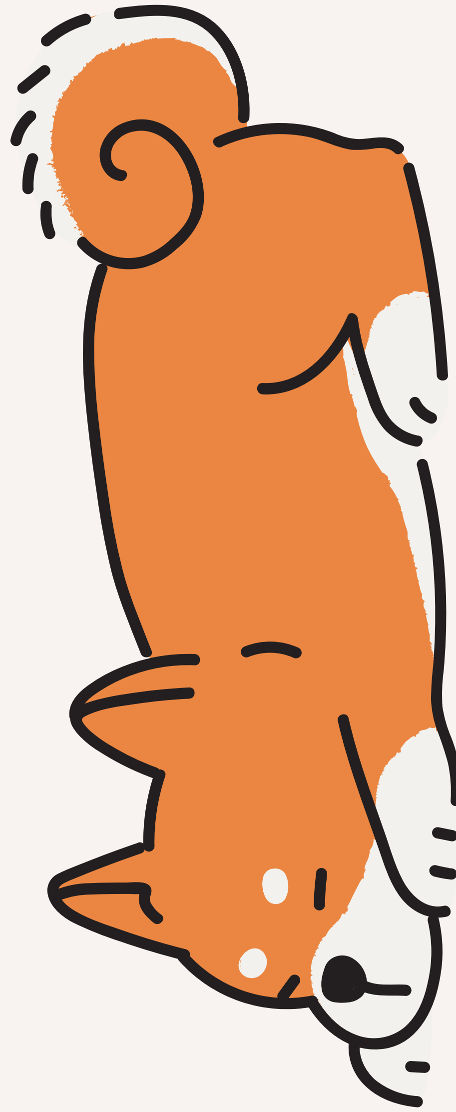


BTC-CAD (2017-Present) Time Series Plot

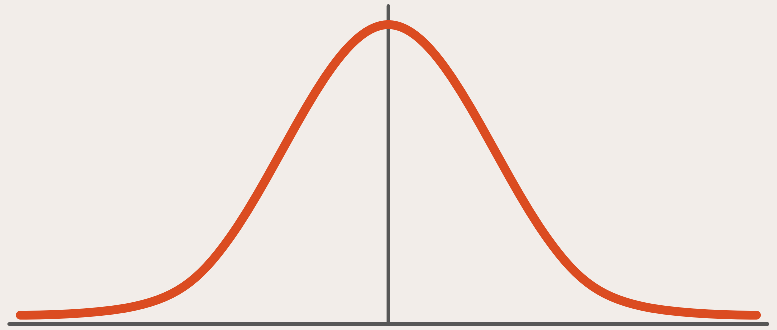
BTC-CAD (March - July, 2021) Time Series Plot

We've got proof

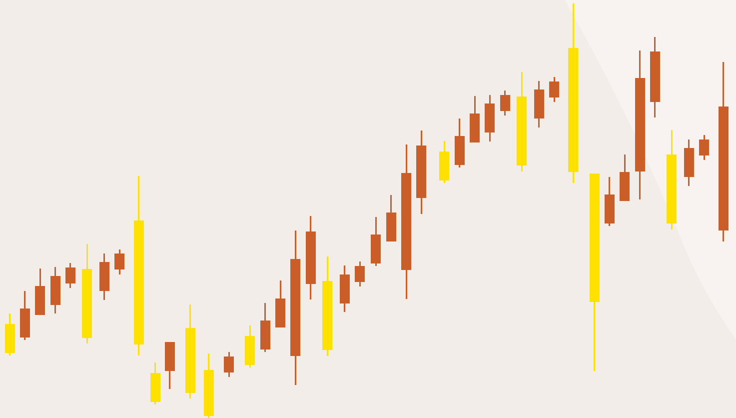




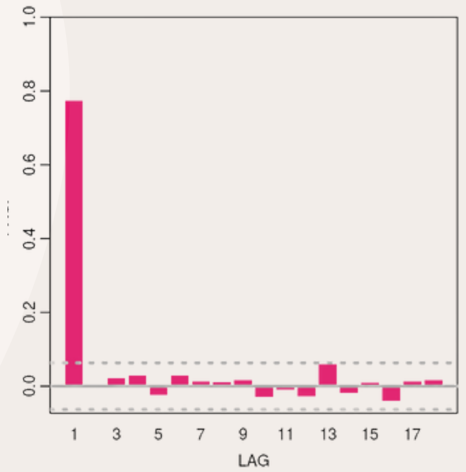
Statistical Analysis



Step 1-Checking for Normality

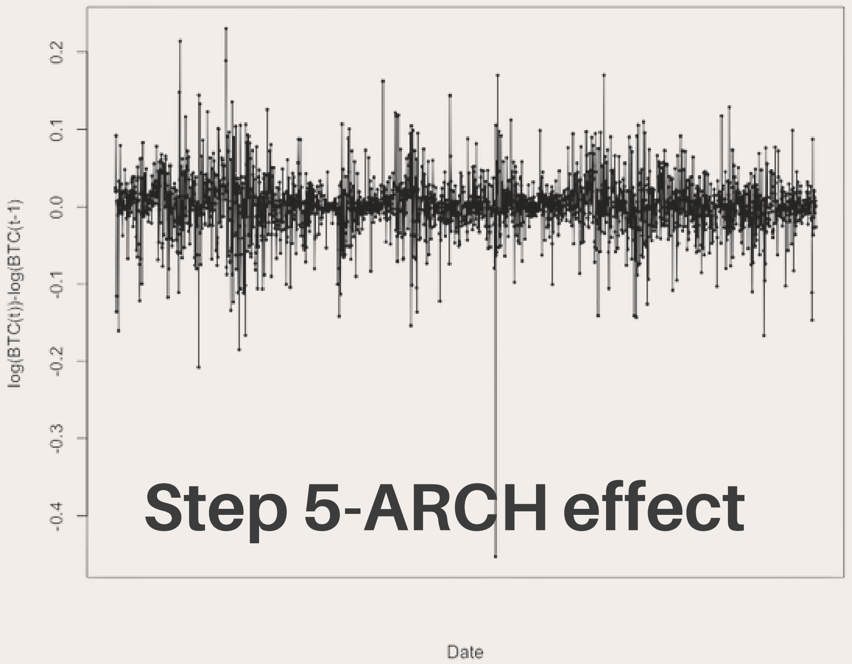


Step 2- Checking for Stationarity



Step 3: Determining ARMA

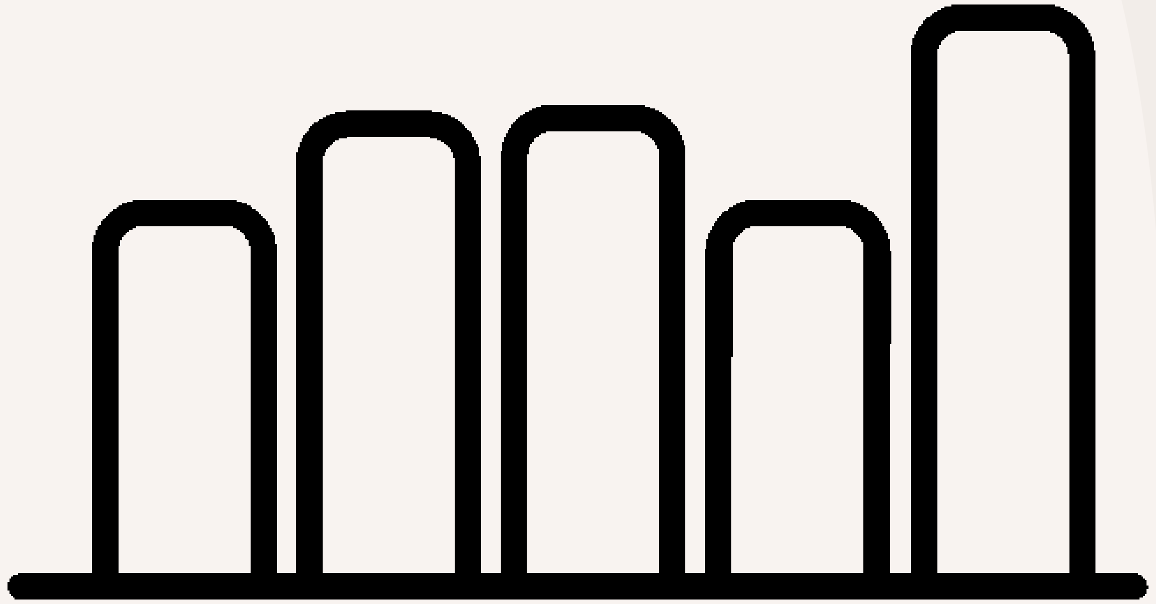
Step 4-Estimating the mean Equation



Step 5-ARCH effect

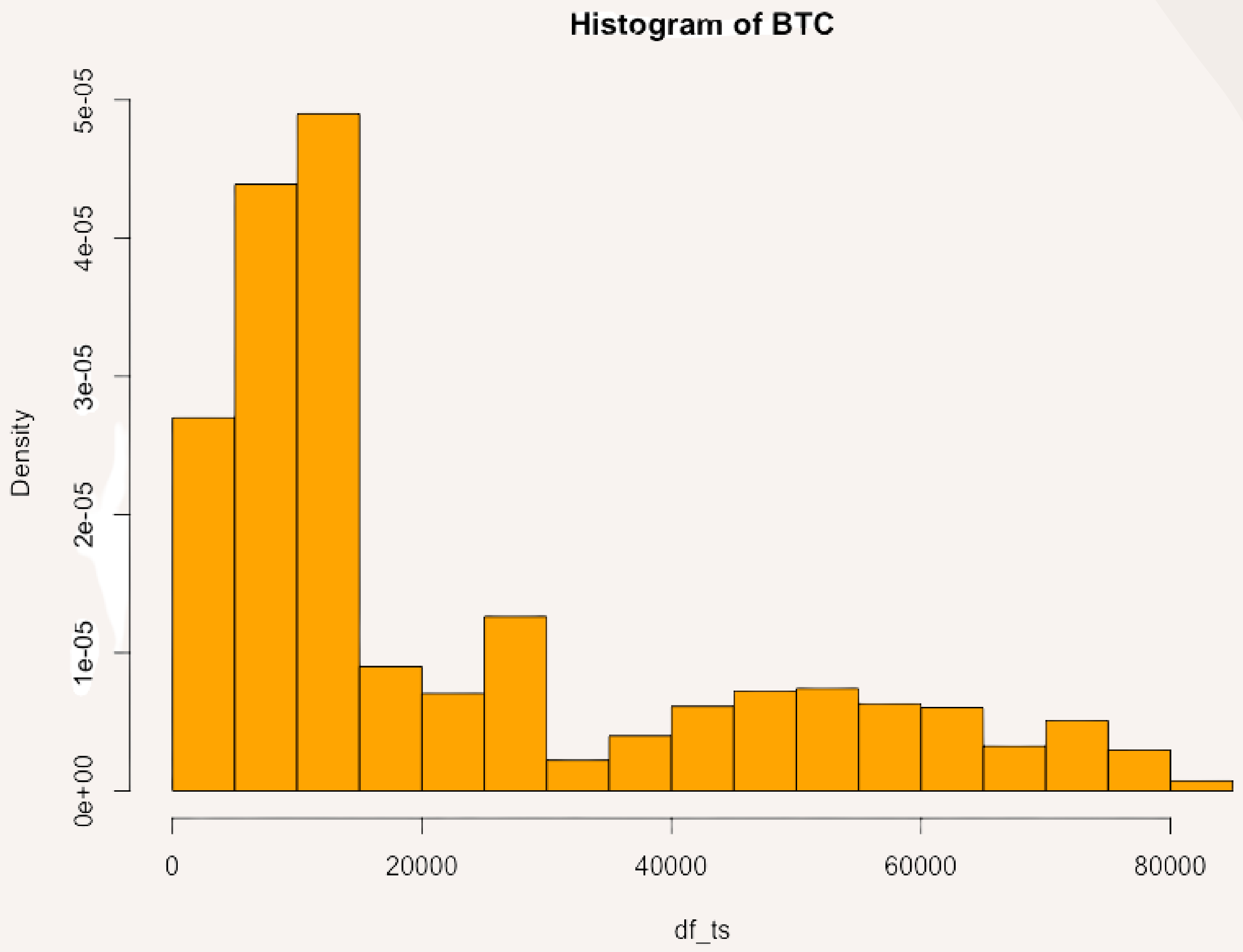
Step 6-Estimating the ARCH equation

Step 7-Forecasting



Step 1-Checking for normality

Step 1



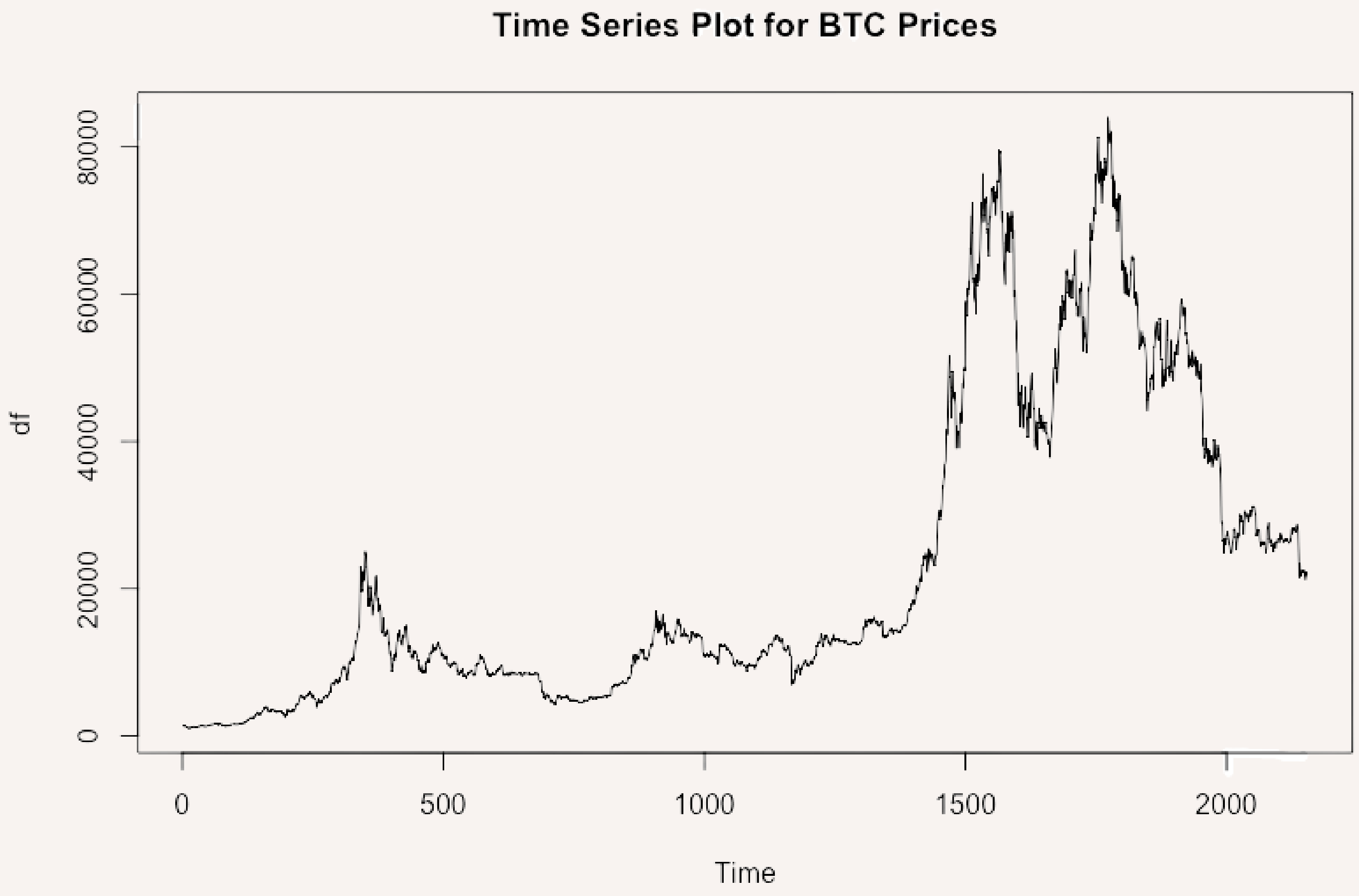
01 Data clearly does not appear to be normal

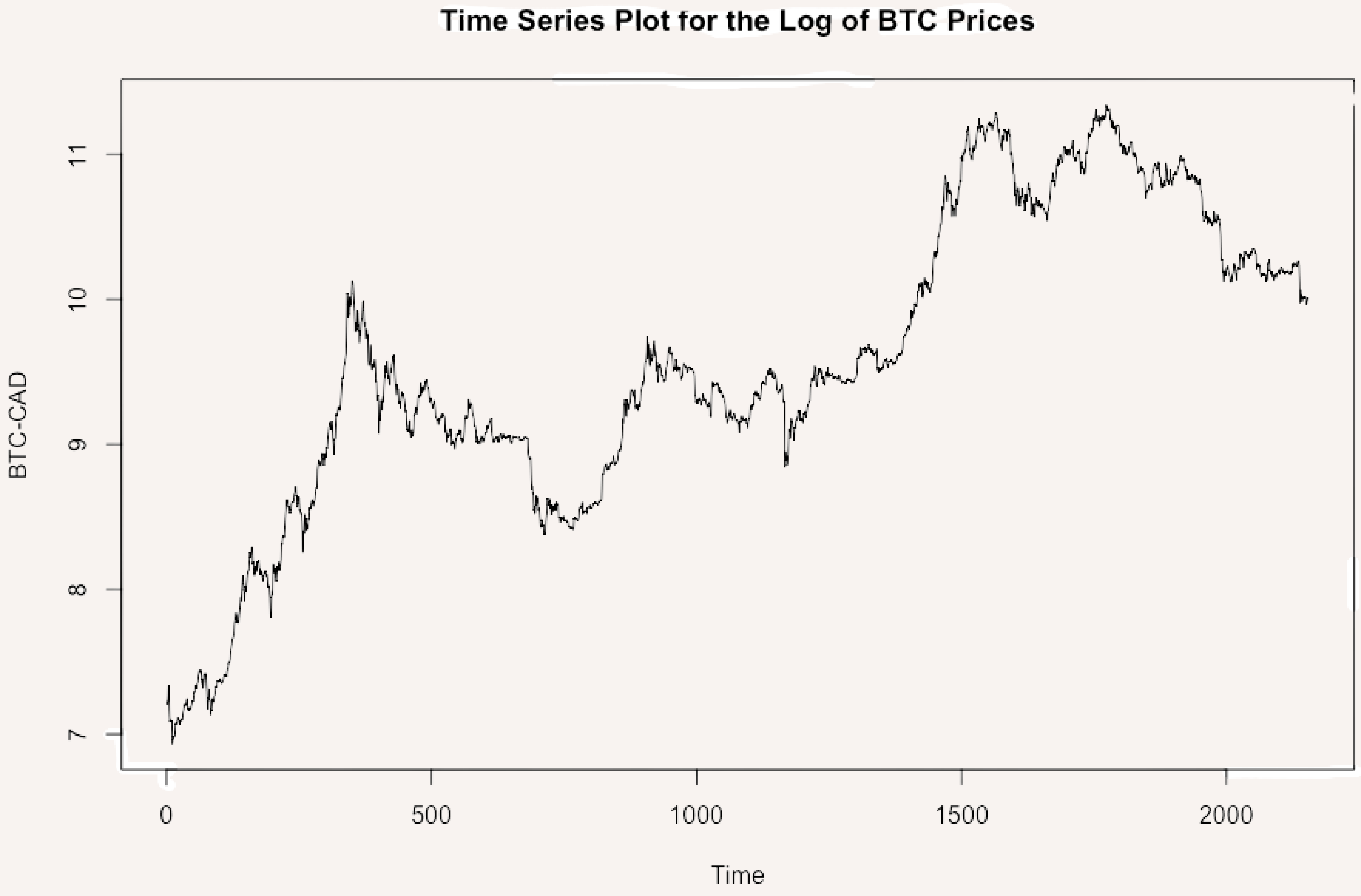
02 Use the Shapiro Test to confirm our observation

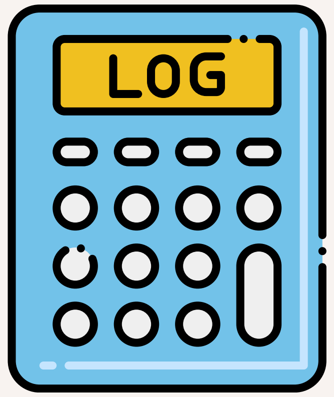
```
        Shapiro-Wilk normality
        test

data:  df_ts
W = 0.81689, p-value <
2.2e-16
```

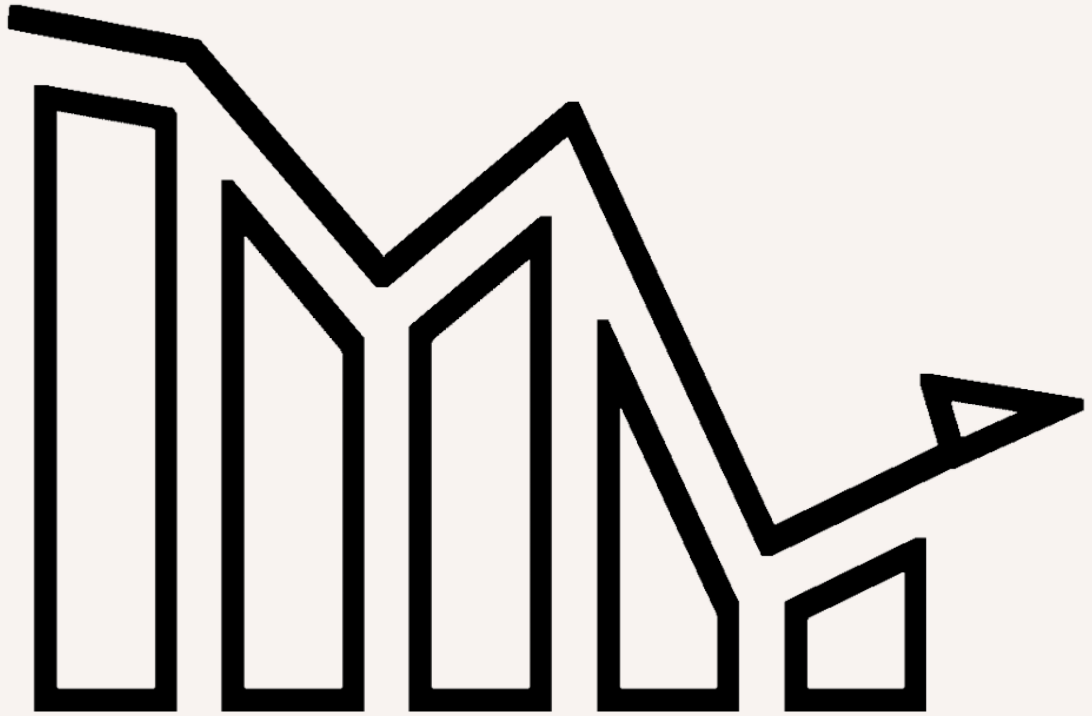
03 To overcome non-normality, we will take the log of our data, to normalize it

Before and after log transformation









Step 2-Checking for Stationarity

Step 2-Checking for Stationarity

Hypothesis test:

H_0 : Data has constant variance

```
> Auto.VR(logdf)
$stat
[1] 325.5075

$sum
[1] 1073.879
```

Reject the null hypothesis, since there's a high ratio test

Need to add an ARCH model with an ARMA model

H_0: Non-stationary

```
> adf.test(logdf$`BTC-CAD`, k = 3)

        Augmented Dickey-Fuller Test

data:  logdf$`BTC-CAD`
Dickey-Fuller = -1.4644, Lag order = 3, p-value
= 0.8051
alternative hypothesis: stationary
```

Fail to reject the null hypotheses, since our p-value is quite large

There is no evidence of stationarity

Step 2-Checking for Stationarity

First Difference

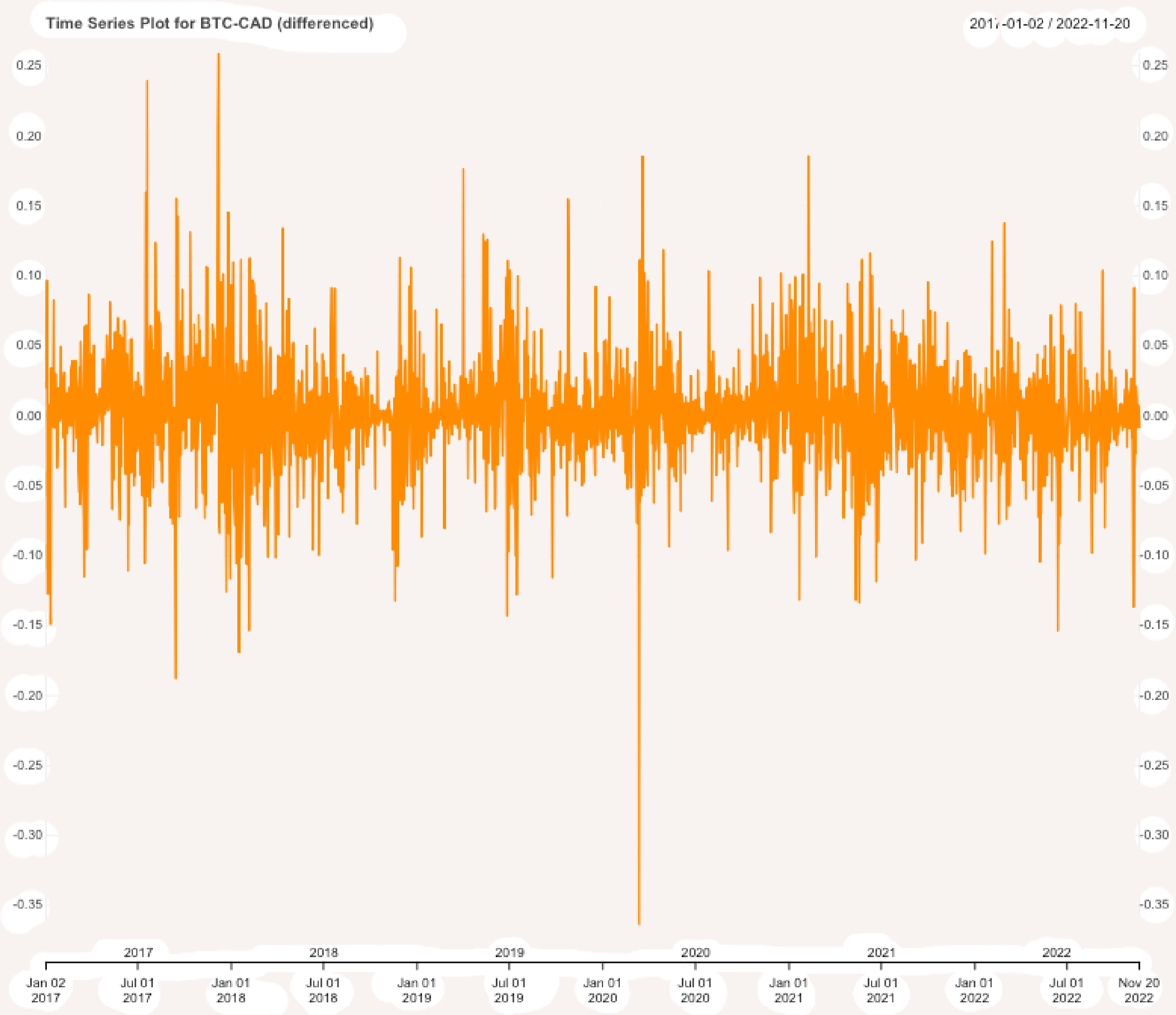
```
> logdiffdf = diff(logdf)
> logdiffdf = na.remove(logdiffdf)
```

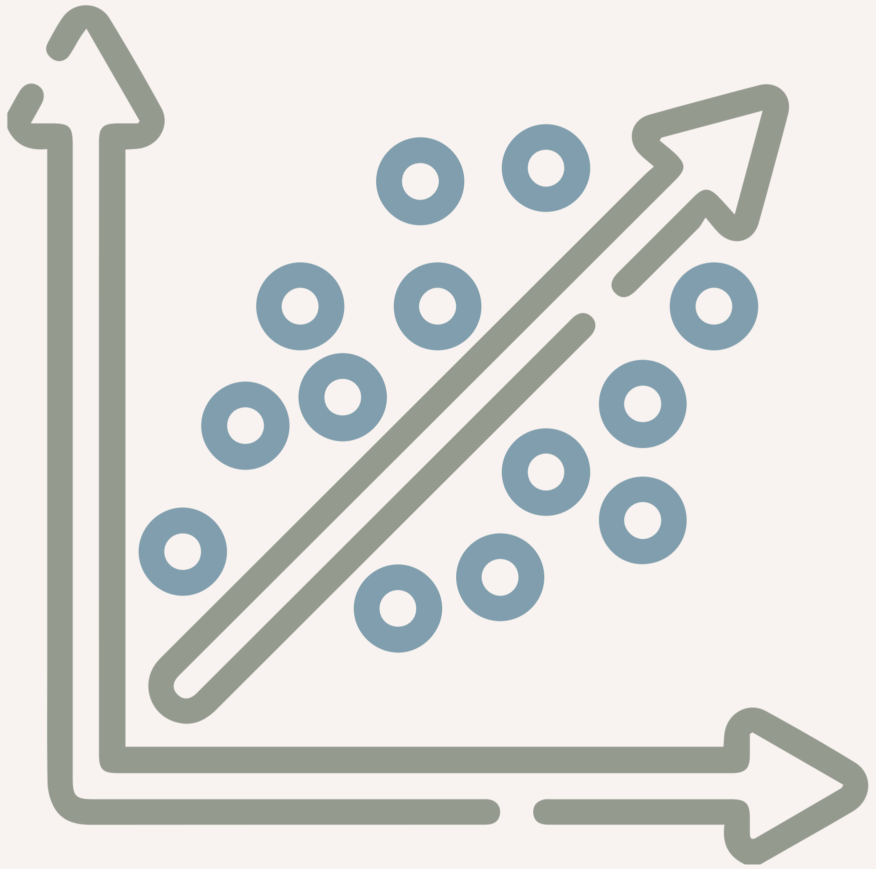
Check for stationarity

H_0 : Non-stationary

```
        Augmented Dickey-Fuller Test

data:  logdiffdf
Dickey-Fuller = -22.697, Lag order = 3, p-value =
0.01
alternative hypothesis: stationary
```

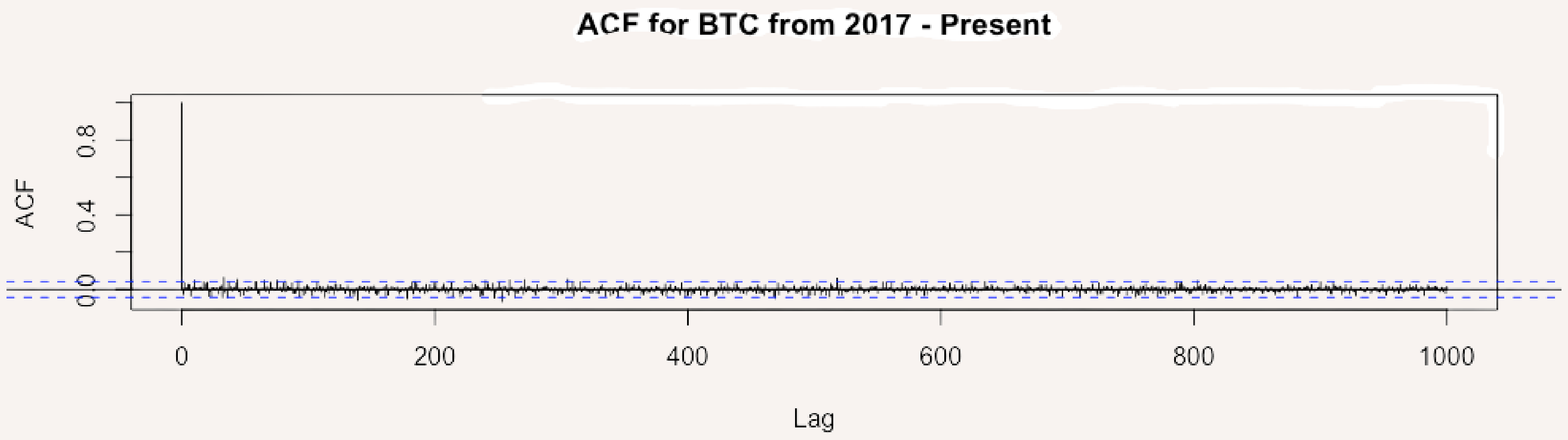
There is evidence of stationarity

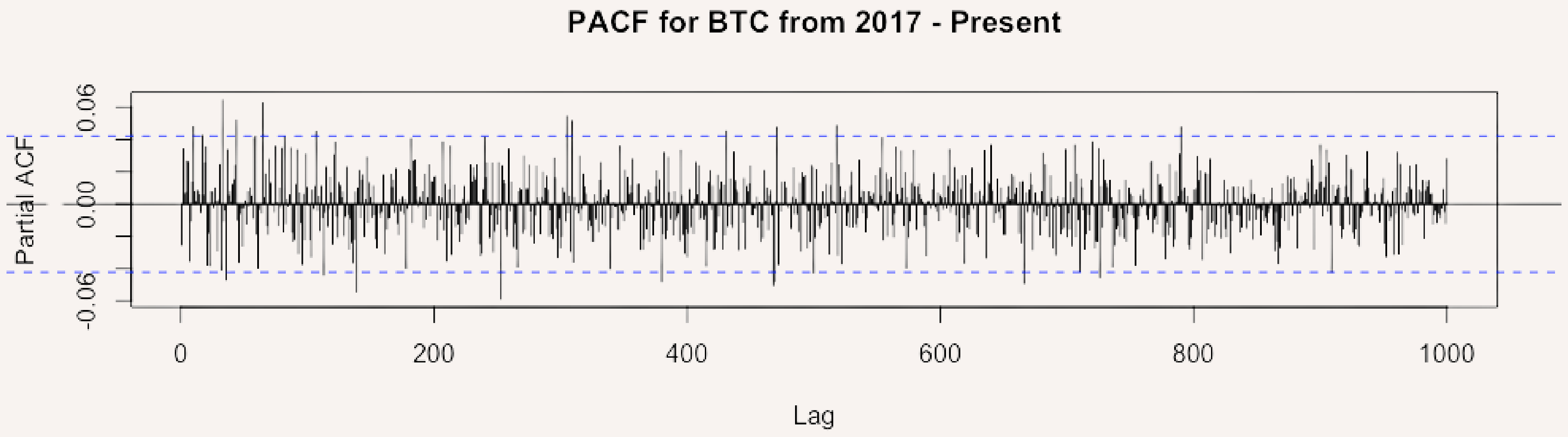




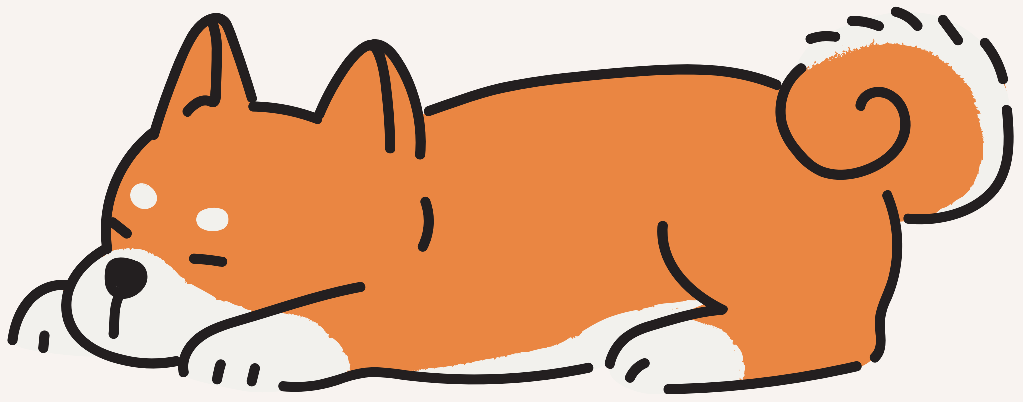
Step 3-Determining ARMA

Step 3 - Determining ARMA before



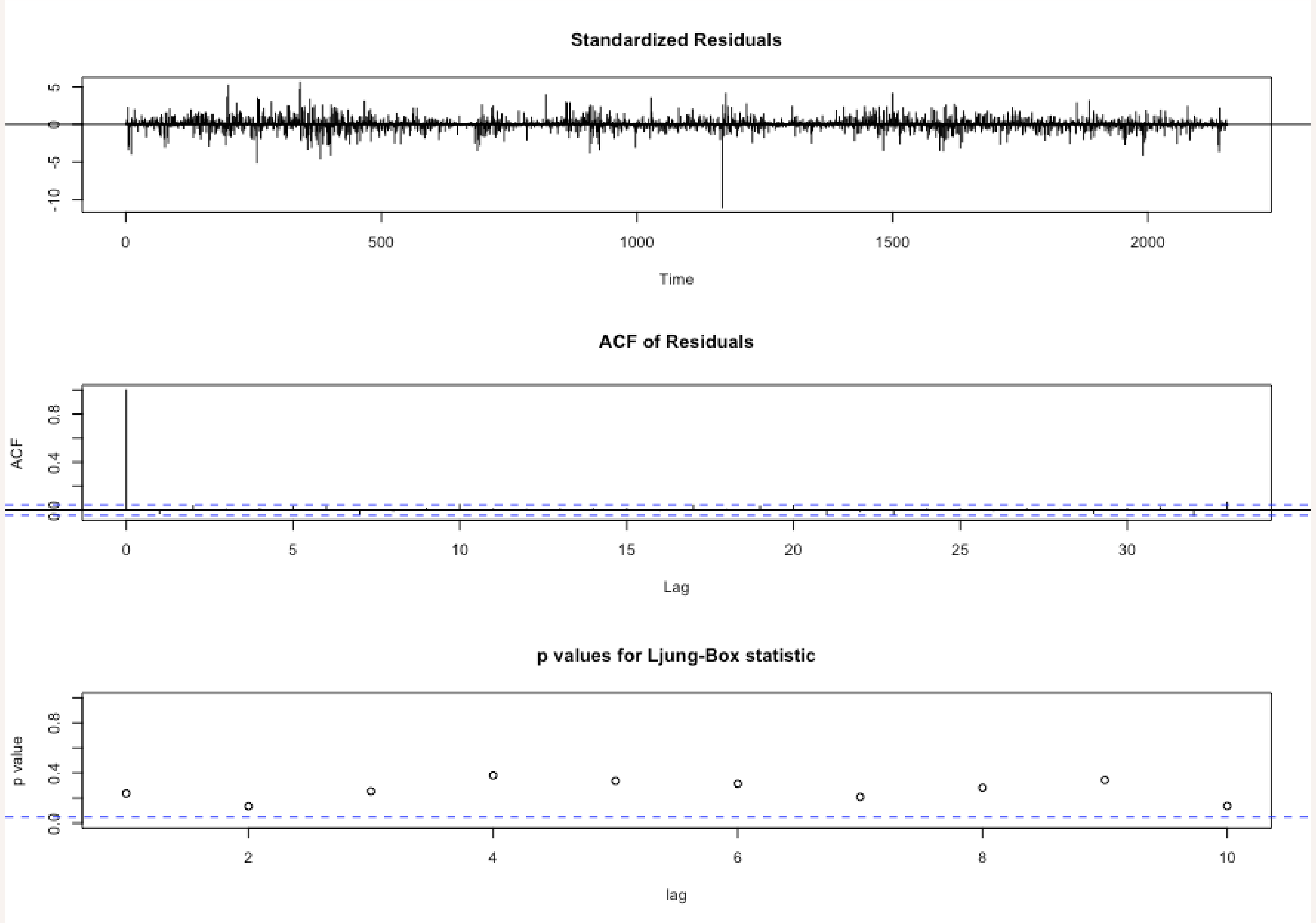



```
> auto.arima(logdiffdf)
Series: logdiffdf
ARIMA(0,0,0) with non-zero mean

Coefficients:
        mean
      0.0013
s.e.  0.0009

sigma^2 = 0.001675:  log likelihood = 3828.44
AIC=-7652.87   AICc=-7652.87   BIC=-7641.52
```



Step 3 - Determining ARMA

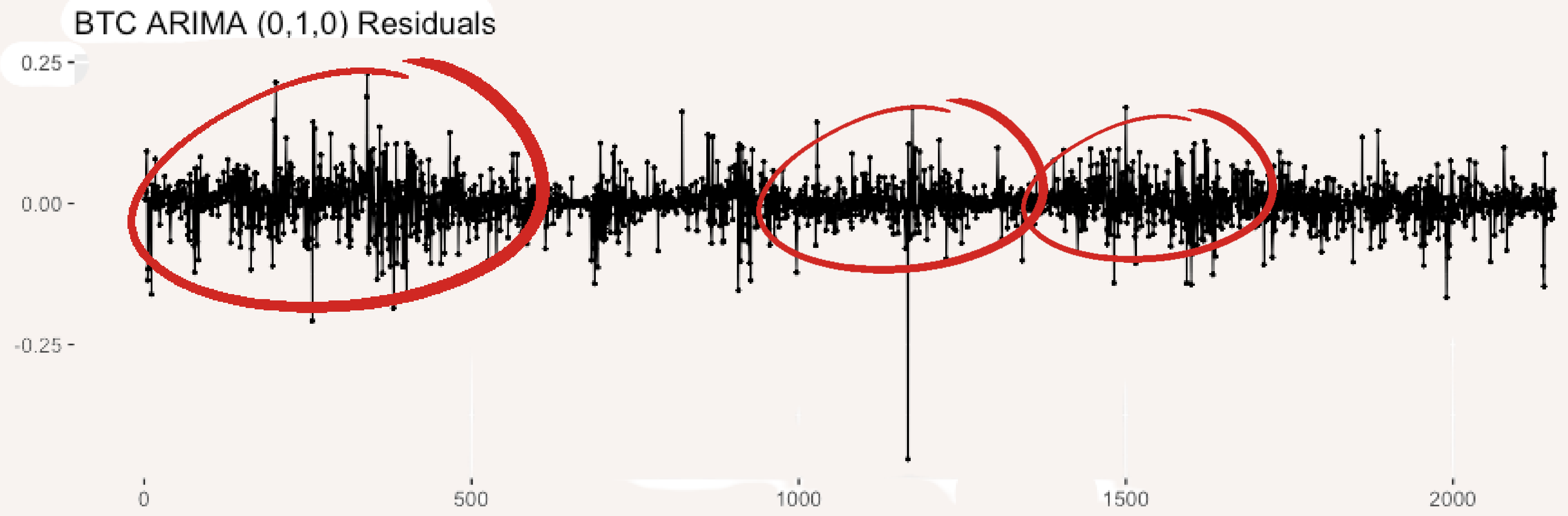
```
> arima010 = arima(logdf, order = c(0,1,0))
> summary(arima010)

Call:
arima(x = logdf, order = c(0, 1, 0))


sigma^2 estimated as 0.001676:  log likelihood = 3827.35,  aic = -7652.7
```

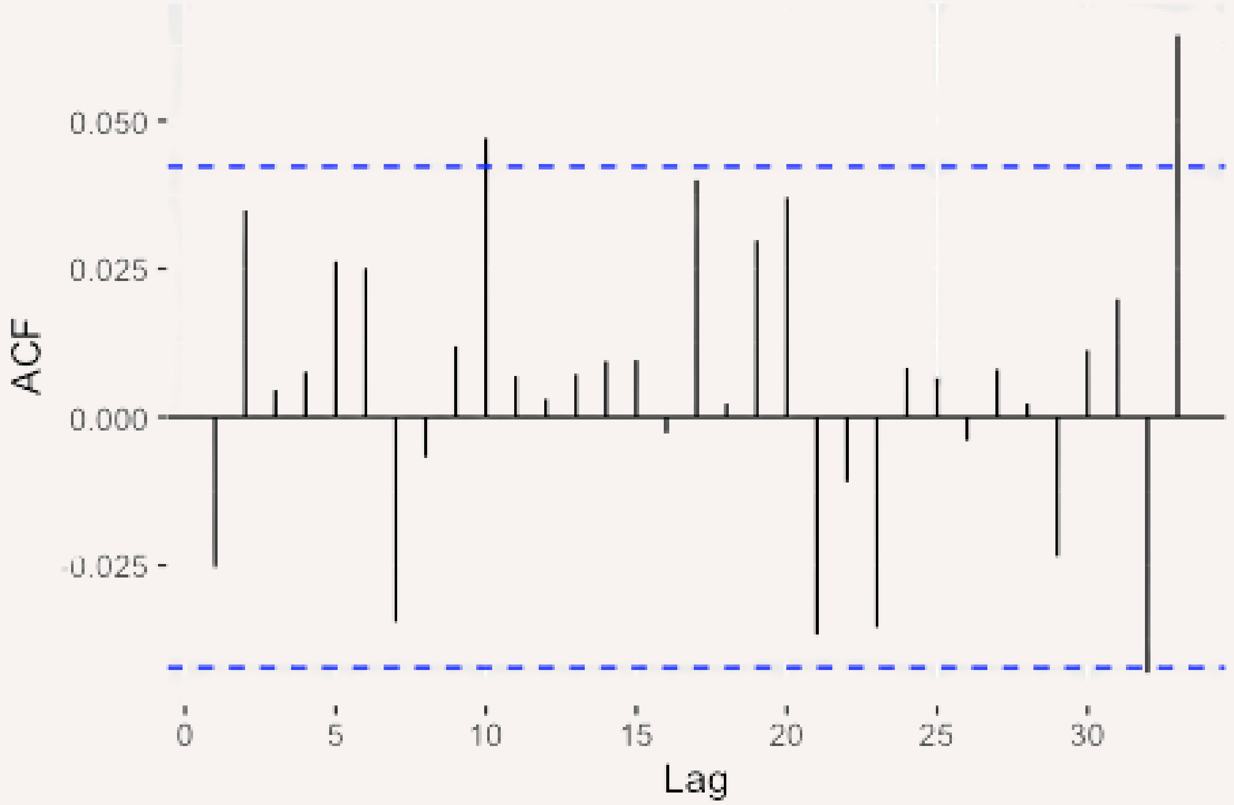


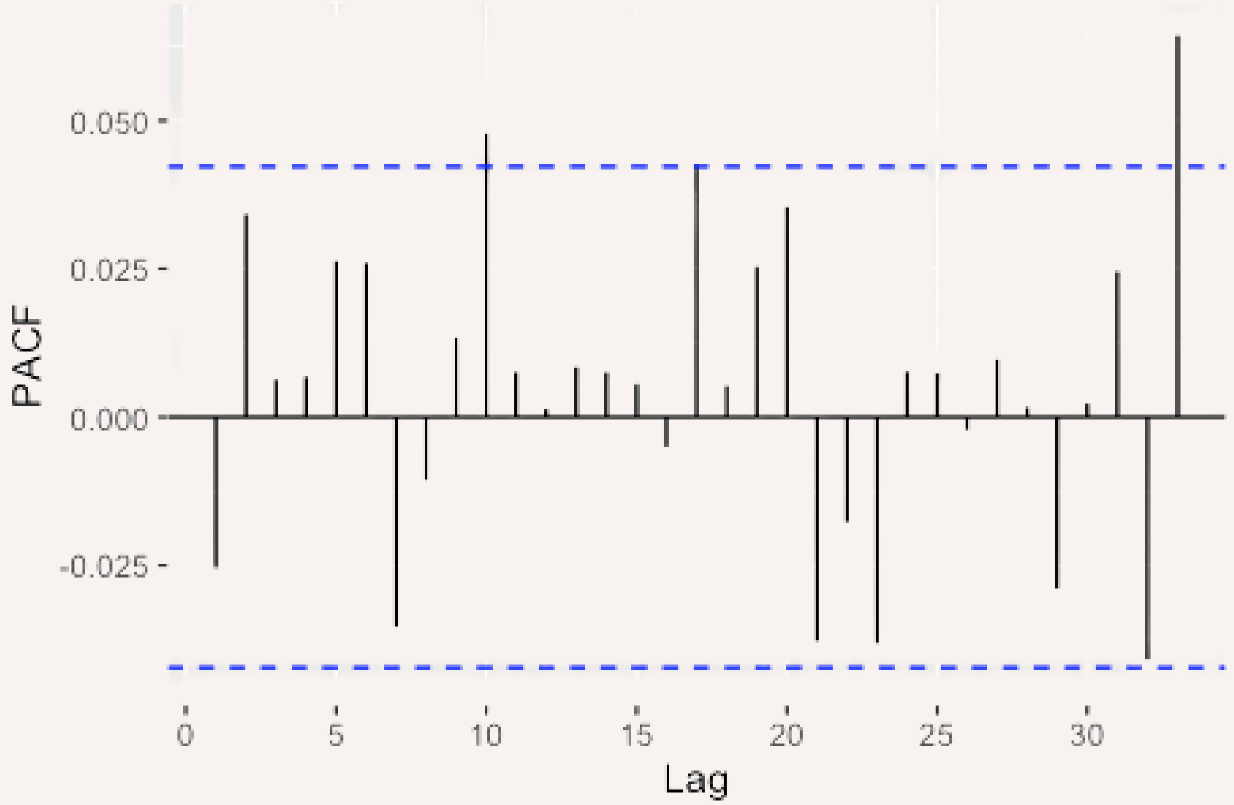
Step 3 - Determining ARMA

Residual Plot:

- Clear Signs of Volatility at Multiple Points







Step 4- Estimating the mean Equation

Step 4- Estimating the mean Equation

```
> BTC_mean = dynlm(arima010$residuals ~ 1)
> summary(BTC_mean)

Time series regression with "ts" data:
Start = 1, End = 2155

Call:
dynlm(formula = arima010$residuals ~ 1)

Residuals:
     Min       1Q   Median
-0.45394 -0.01715  0.00026
      3Q      Max
 0.01851  0.22928

Coefficients:
             Estimate
(Intercept) 0.0013021
             Std. Error
(Intercept)  0.0008813
             t value Pr(>|t|)
(Intercept)   1.477     0.14

Residual standard error: 0.04091 on 2154 degrees of freedom
```

The mean value is insignificant.
It means that we are able to capture any mean effect by using the differencing

Step 5- ARCH EFFECT

Step 5- ARCH EFFECT

- Estimate the square of the residuals

```
> esq = ts(resid(BTC_mean)^2)
```

- Create the regression of the squared residuals with the lag of residual squares

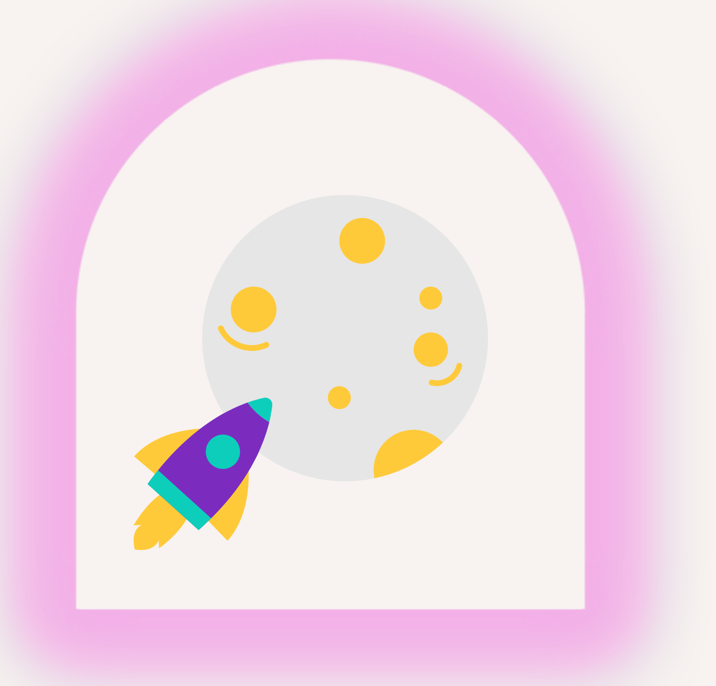
```
> BTC_ARCH_EF = dynlm(esq ~ L(esq))
> summary(BTC_ARCH_EF)

Time series regression with "ts" data:
Start = 2, End = 2155

Call:
dynlm(formula = esq ~ L(esq))

Residuals:
      Min        1Q    Median        3Q       Max 
-0.009457 -0.001517 -0.001296 -0.000261  0.204540 

Coefficients:
             Estimate Std. Error t value Pr(>|t|)
(Intercept) 0.0015213  0.0001289  11.805  < 2e-16
L(esq)      0.0911418  0.0214668   4.246 2.27e-05

(Intercept) ***
L(esq)      ***

Signif. codes:
0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.005744 on 2152 degrees of freedom
Multiple R-squared:  0.008307,	Adjusted R-squared:  0.007846 
F-statistic: 18.03 on 1 and 2152 DF,  p-value: 2.272e-05
```

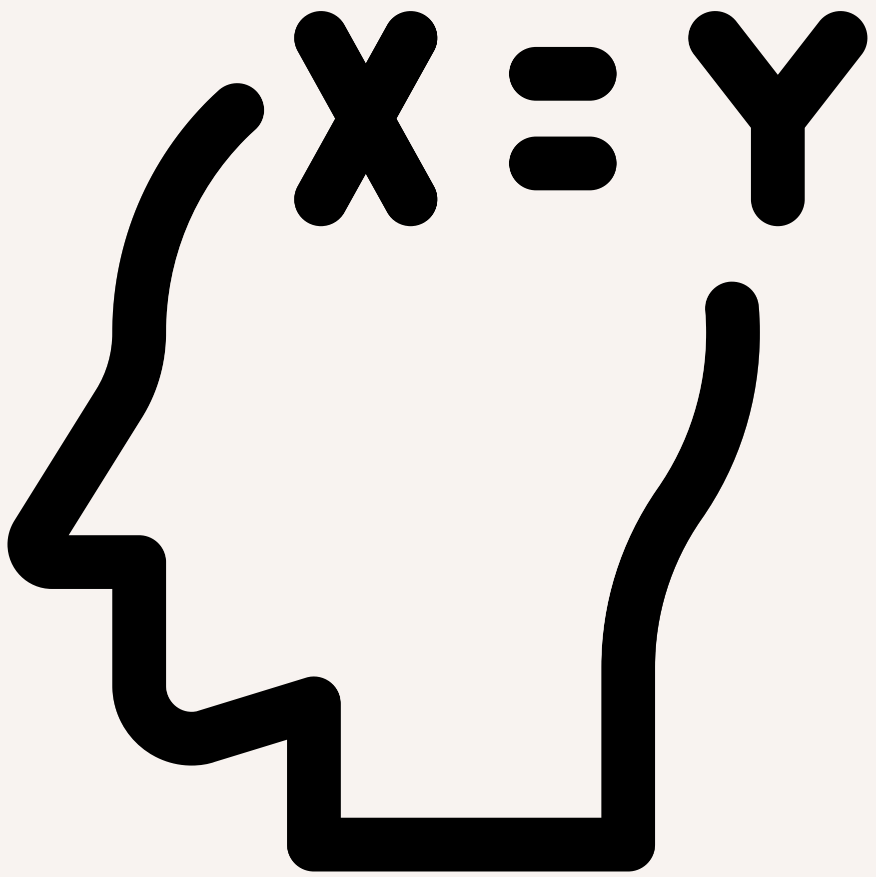
- Further confirm the ARCH effect using

Hypothesis test:
H_0 = No ARCH effect

```
> ArchTest(arima010$residuals, lags = 1, demean = T)

	ARCH LM-test; Null hypothesis: no ARCH effects

data:  arima010$residuals
Chi-squared = 17.893, df = 1, p-value =
2.337e-05
```

There is evidence of ARCH effect.





Step 6-Estimating the ARCH equation

Step 6-Estimating the ARCH equation

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

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(0,1)
Mean Model      : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
       Estimate  Std. Error    t value Pr(>|t|)
mu     0.001377    0.000891 1.5454e+00  0.12226
omega  0.000001    0.000000 5.7467e+01  0.00000
beta1  0.999000    0.000008 1.2888e+05  0.00000

Robust Standard Errors:
       Estimate  Std. Error    t value Pr(>|t|)
mu     0.001377    0.001010     1.3636   0.1727
omega  0.000001    0.000000    13.5579   0.0000
beta1  0.999000    0.000081 12361.6953   0.0000

LogLikelihood : 3734.837

Information Criteria
------------------------------------

Akaike       -3.5474
Bayes        -3.5393
Shibata      -3.5474
Hannan-Quinn -3.5444
```

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

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(0,2)
Mean Model      : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
       Estimate  Std. Error    t value Pr(>|t|)
mu     0.001360    0.000891 1.5270e+00  0.12676
omega  0.000001    0.000000 7.4731e+01  0.00000
beta1  0.000007    0.000071 9.4168e-02  0.92497
beta2  0.998993    0.000002 4.0329e+05  0.00000

Robust Standard Errors:
       Estimate  Std. Error    t value Pr(>|t|)
mu     0.001360    0.001010 1.3466e+00  0.17810
omega  0.000001    0.000000 2.6329e+01  0.00000
beta1  0.000007    0.000196 3.4196e-02  0.97272
beta2  0.998993    0.000041 2.4131e+04  0.00000

LogLikelihood : 3736.173

Information Criteria
------------------------------------

Akaike       -3.5477
Bayes        -3.5369
Shibata      -3.5477
Hannan-Quinn -3.5438
```

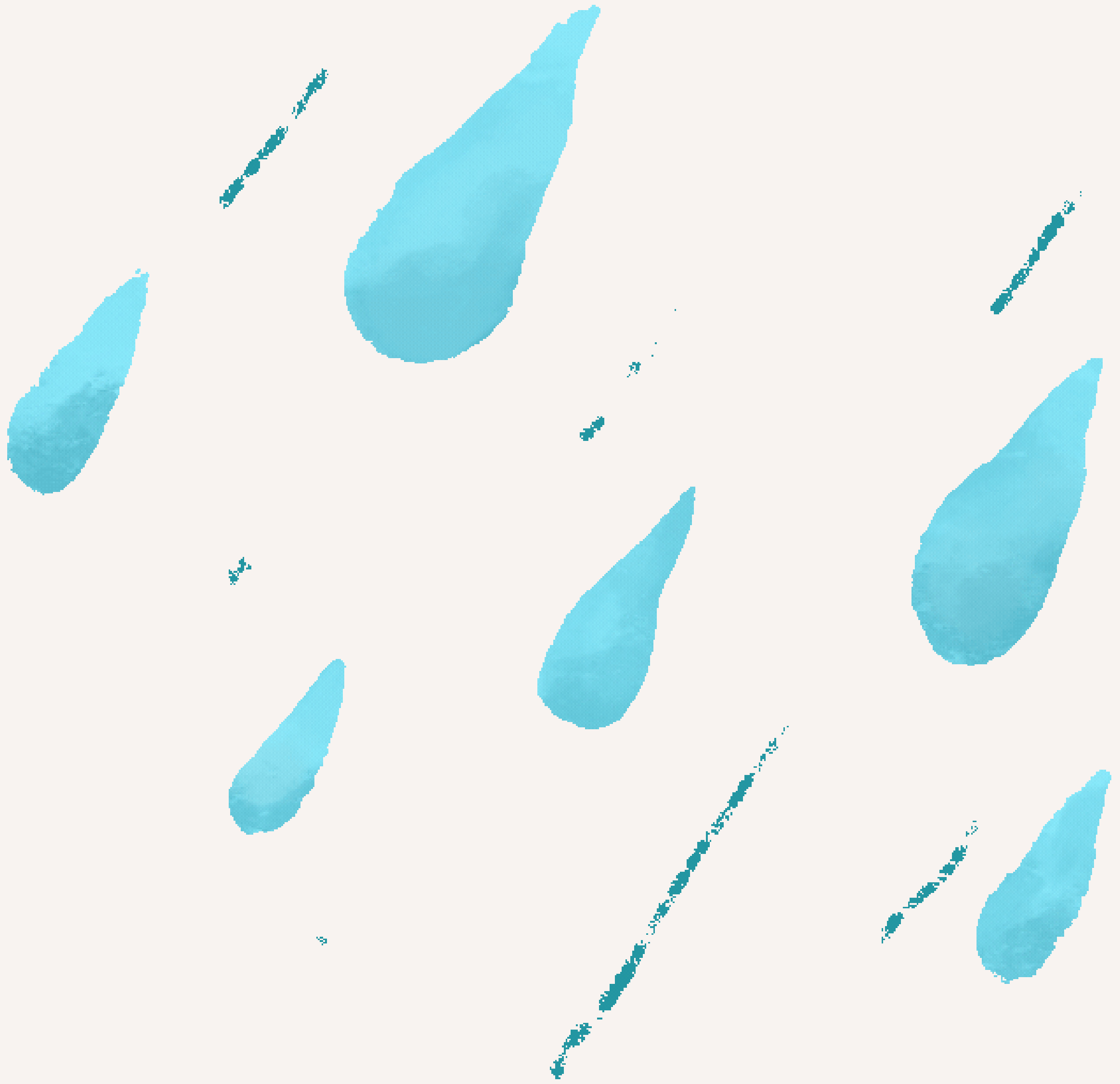
```
*---------------------------------*
*          GARCH Model Fit        *
*---------------------------------*

Conditional Variance Dynamics
-----------------------------------
GARCH Model     : sGARCH(0,3)
Mean Model      : ARFIMA(0,0,0)
Distribution    : norm

Optimal Parameters
------------------------------------
       Estimate  Std. Error    t value Pr(>|t|)
mu     0.001353    0.000890 1.5196e+00  0.12860
omega  0.000001    0.000000 7.3760e+01  0.00000
beta1  0.000009    0.032674 2.7500e-04  0.99978
beta2  0.000751    0.032669 2.2975e-02  0.98167
beta3  0.998240    0.000003 2.9355e+05  0.00000

Robust Standard Errors:
       Estimate  Std. Error    t value Pr(>|t|)
mu     0.001353    0.001011 1.3378e+00  0.18094
omega  0.000001    0.000000 4.1054e+01  0.00000
beta1  0.000009    0.002356 3.8160e-03  0.99696
beta2  0.000751    0.002285 3.2843e-01  0.74259
beta3  0.998240    0.000037 2.7217e+04  0.00000

LogLikelihood : 3736.664

Information Criteria
------------------------------------

Akaike       -3.5472
Bayes        -3.5338
Shibata      -3.5472
Hannan-Quinn -3.5423
```

Step 6-Estimating the ARCH equation

Therefore, the model for ARCH(1)

$$y_t = \sigma_t \epsilon_t, \text{ with } \sigma_t = \sqrt{0.000001 + 0.999000 y_{t-1}^2}, \text{ and } \epsilon_t \sim iidN(0,1)$$

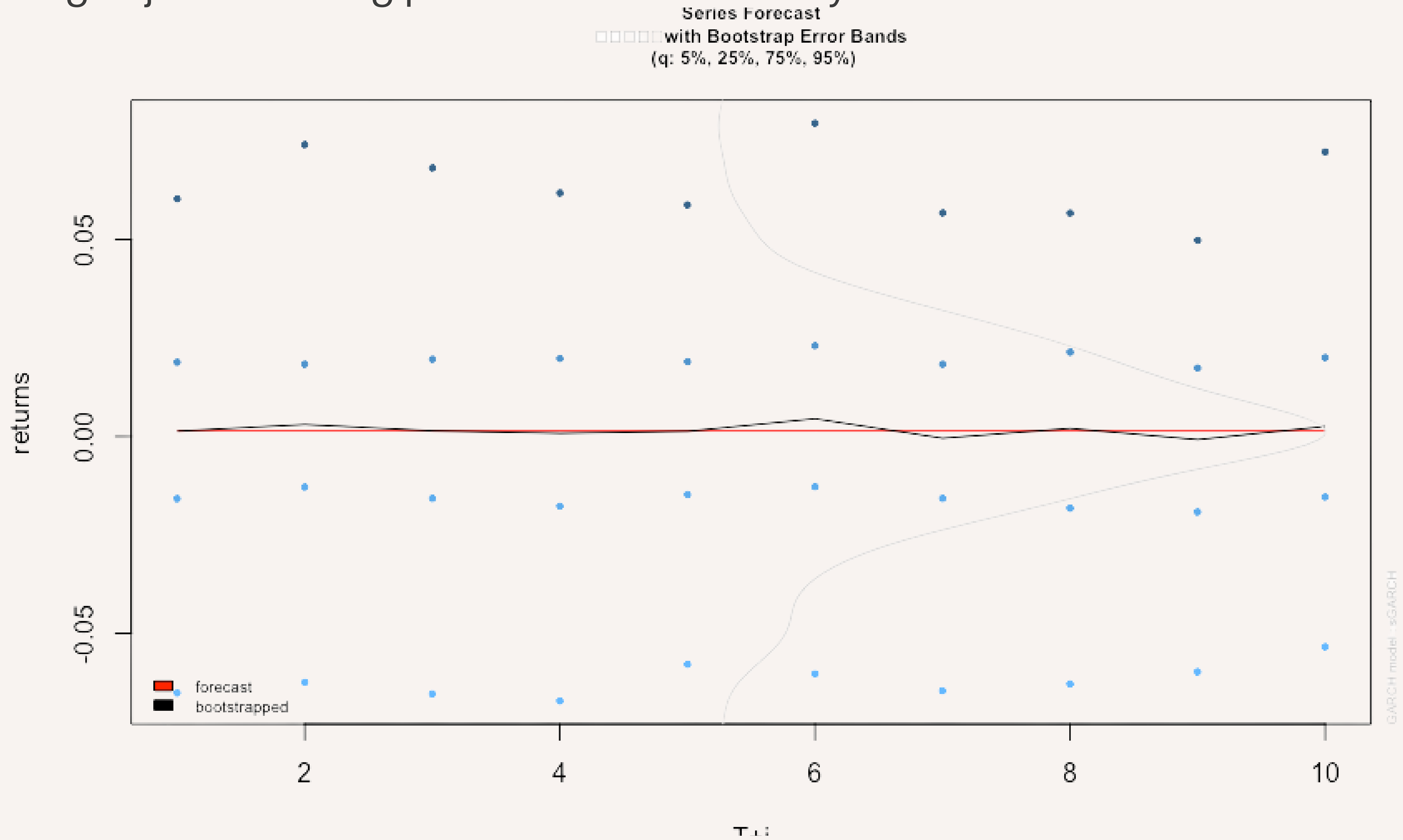
```
Optimal Parameters
------------------------------------
        Estimate  Std. Error    t value Pr(>|t|)
mu      0.001377    0.000891 1.5454e+00  0.12226
omega   0.000001    0.000000 5.7467e+01  0.00000
beta1   0.999000    0.000008 1.2888e+05  0.00000
```



Step 7- Forecast

Step 7- Forecast

Forecasting adjusted closing prices in the next 10 days.



Generalizations

An ARCH(m) process is one for which the variance at time t is conditional on observations at the previous m times, and the relationship is

$$\mathrm{Var}(y_t | y_{t-1}, \ldots, y_{t-m}) = \sigma_t^2 = \alpha_0 + \alpha_1 y_{t-1}^2 + \cdots + \alpha_m y_{t-m}^2.$$

THIS ENDS OUR PRESENTATION


Q & A