Stock Price Prediction – Multiple Linear regression and Adaline Neural Network

Dataset
Downloaded excel files from Yahoo finance contains stocks from different domains for the period 29-Mar-2016 to 25-Mar-2019 (three financial years).

Barclays: A bank Finance industry
RollsRoyce: Auto Industry
UKOilGas: Energy domain
Aviva: An insurance company

Columns of interest
Date
Close(the closing price of a stock for the corresponding date)

Handling missing data

For the columns selected the close price has missing values for the below dates other than weekends. The dates for the missed values are the same for the four stocks.

Date	Day of the week	Market Holiday
2016-05-02	Monday	Labour Day
2016-05-30	Monday	Memorial Day
2016-08-29	Monday	Summer Bank Holiday
2016-12-26	Monday	Christmas Day
2016-12-27	Tuesday	Boxing Day
2017-01-02	Monday	New Year's Day
2017-04-14	Friday	Good Friday
2017-04-17	Monday	Easter Day
2017-05-01	Monday	Labour Day
2017-05-29	Monday	Memorial Day
2017-08-28	Monday	Summer Bank Holiday
2017-12-25	Monday	Christmas Day
2017-12-26	Tuesday	Boxing Day
2018-01-01	Monday	New Year's Day
2018-03-30	Friday	Good Friday
2018-04-02	Monday	Labour Day
2018-05-07	Monday	Early May Bank Holiday
2018-05-28	Monday	Memorial Day
2018-08-27	Monday	Summer Bank Holiday
2018-12-25	Tuesday	Christmas Day
2018-12-26	Wednesday	Boxing Day
2019-01-01	Tuesday	New Year's Day

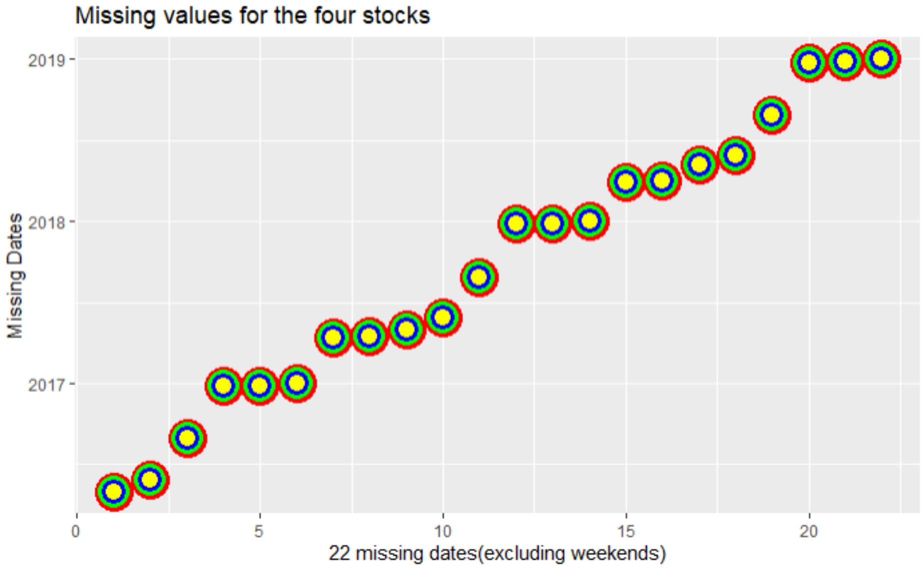
Table 1

From the table above Table 1, we observe 22 missing days for the period 29-Mar-2016 to 25-Mar-2019 (three financial years).

These missing dates correspond to the market holidays. When a holiday falls on a weekend, market closures are decided by two rules:

- If the holiday falls on a Saturday, the market will close on the preceding Friday.
- If the holiday falls on a Sunday, the market will close on the subsequent Monday[1].

For the period I have downloaded the data the missing data corresponds with the bank holidays. However, there could be other missing days when Yahoo Finance or Google finance does not hold data on a particular day due to technical reasons (which is very rare).



🟥	Aviva
🟩	RollsRoyce
🟦	Barclays
🟨	UKOilGas

Fig 1

From Fig 1 above we observe that the four stocks have missing closed price value on the same date.

Approaches for handling missing data

If one value is missing at the time ‘t’ and no values are missing at ‘t-1’ and ‘t+1’ one can calculate the geometric mean of the prices at ‘t-1’ and ‘t+1’ as

$$\mathbf{P_t = sqrt(P_{t-1} * P_{t+1})}$$

where P_{t-1} is the close price on the previous day of the missing value day
P_{t+1} is the close price on the next day of the missing value day

If two values are missing at the time ‘t’ and ‘t+1’ and no values are missing at ‘t-1’ and ‘t+2’ then

$$\mathbf{P_t = cube\ root\ ((P_{t-1})^2 * P_{t+1})}$$

$$\mathbf{P_{t+1} = cube\ root\ (P_{t-1} * (P_{t+2})^2)}$$

The third approach would be to fill the missing value with the previous day's closed value or the next day's closed value.

Filling missing data with closest existing past value

I used the last observed carry forward function **locf()** in R to insert the last observed value into the missing field. Below are the four plots with the missing values inserted. The dots in the line graph resemble the locf values inserted.

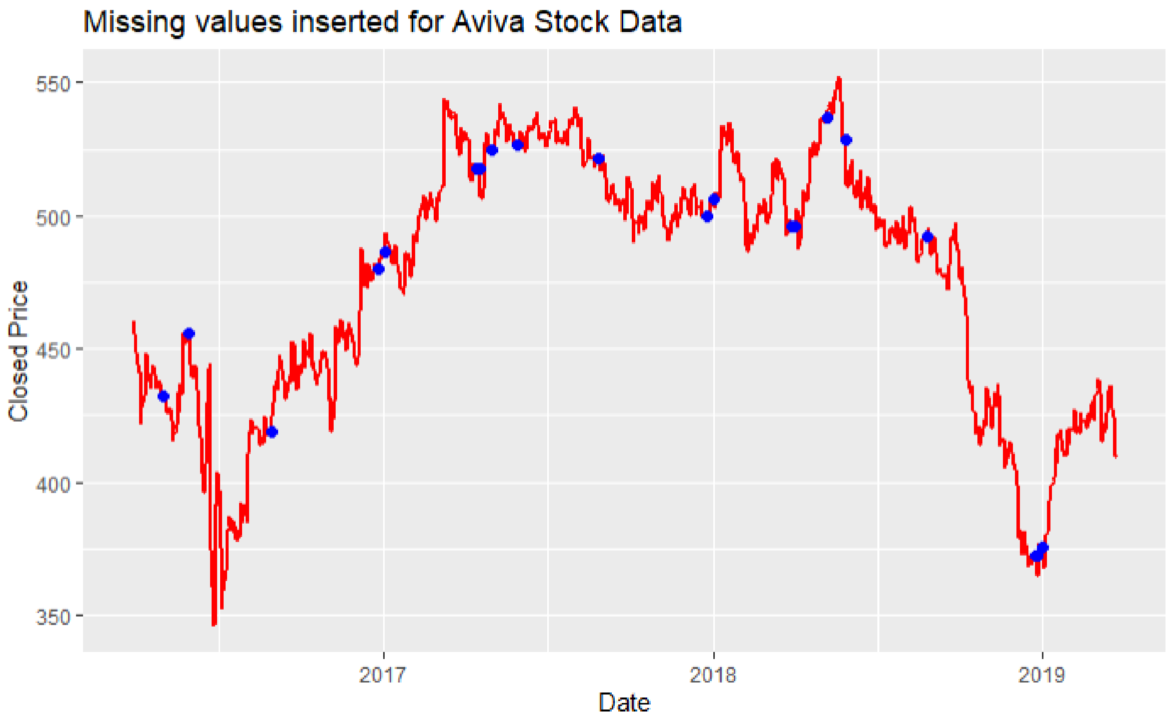

Missing values inserted for Aviva Stock Data
Closed Price
550
500
450
400
350
2017
2018
2019
Date

Fig 2

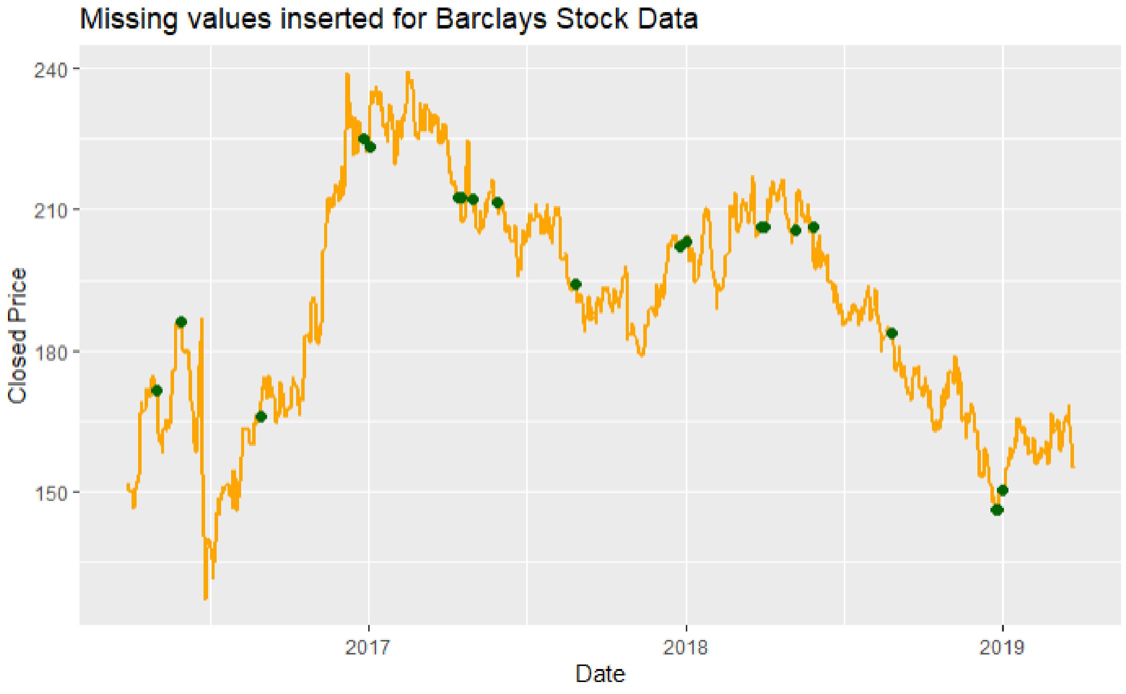

Missing values inserted for Barclays Stock Data
Closed Price
240
210
180
150
2017
2018
2019
Date

Fig 3

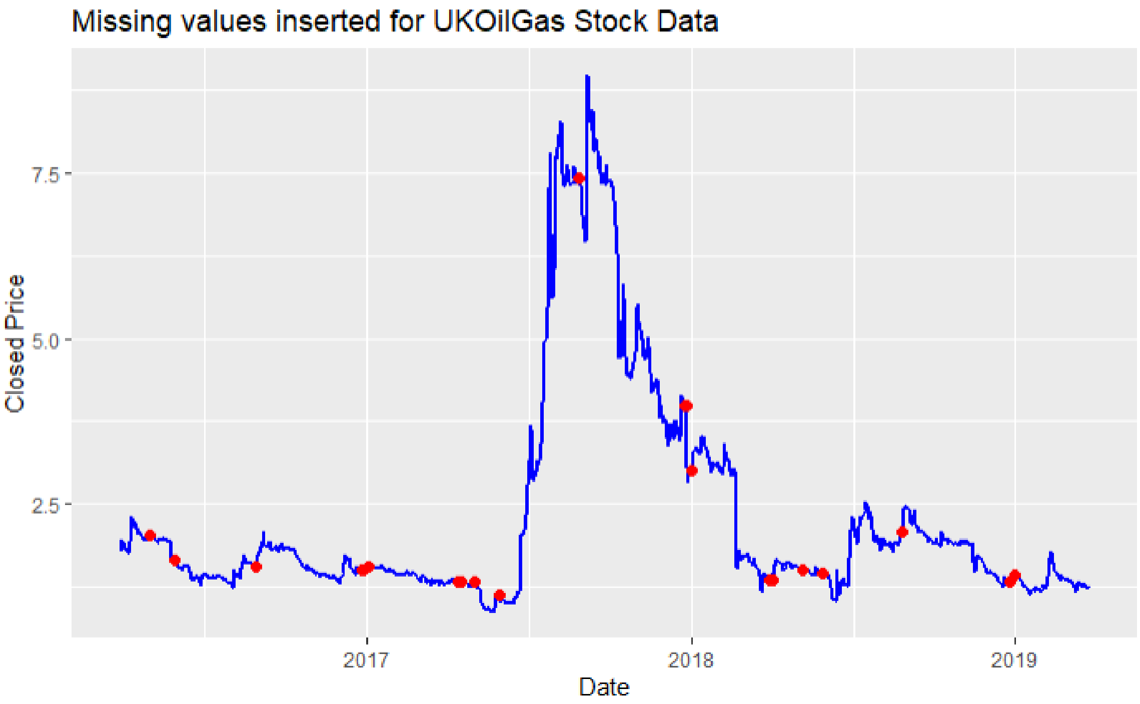


Fig 4

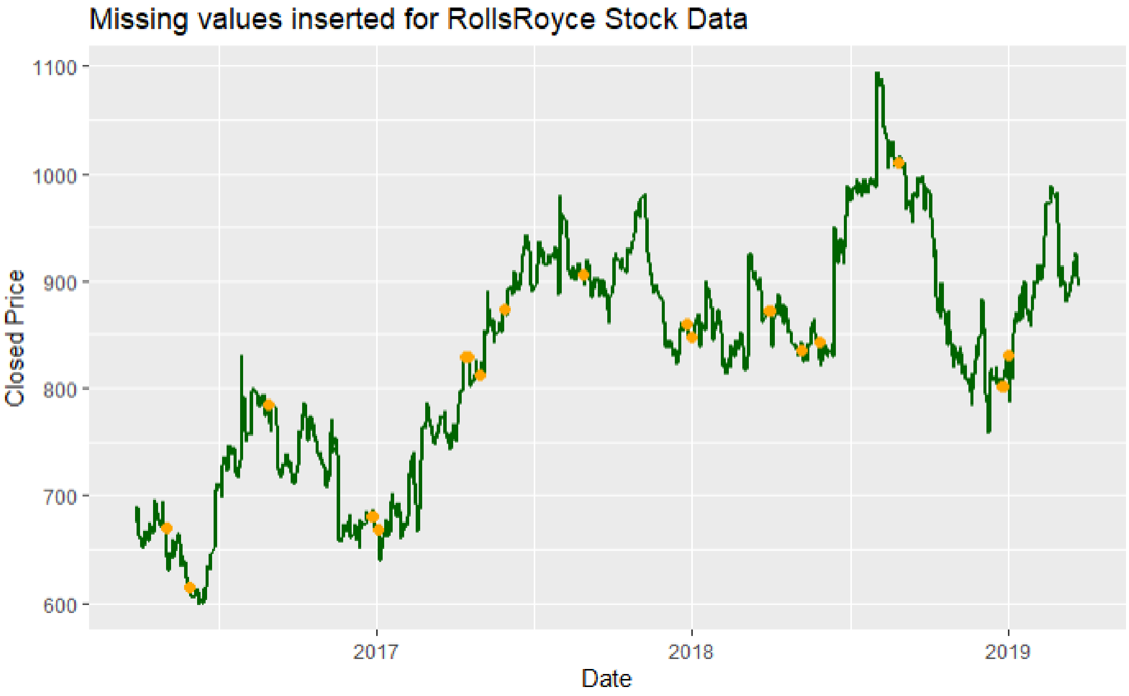


Fig 5

Log returns of a time series

The log-returns of a time series is given as:

$$\mathbf{y_i = \ln(x_i/x_{i-1})}$$

where y_i is the log returned signal
ln is the natural logarithm
x_i is the current close price at time t
x_{i-1} is the close price at time t-1

Computing log-returns of a time series gives us the relative change in the current closed price value of a stock compared to its value the previous day. Stock prices are based on returns and returns are based on percentages therefore we use log returns for the computation.

Below are the log-return plots for the four stocks. From these plots, we observe that the log-return value fluctuates between positive and negative values indicating a relatively positive change and negative change of the close price at time t as compared to time t-1.

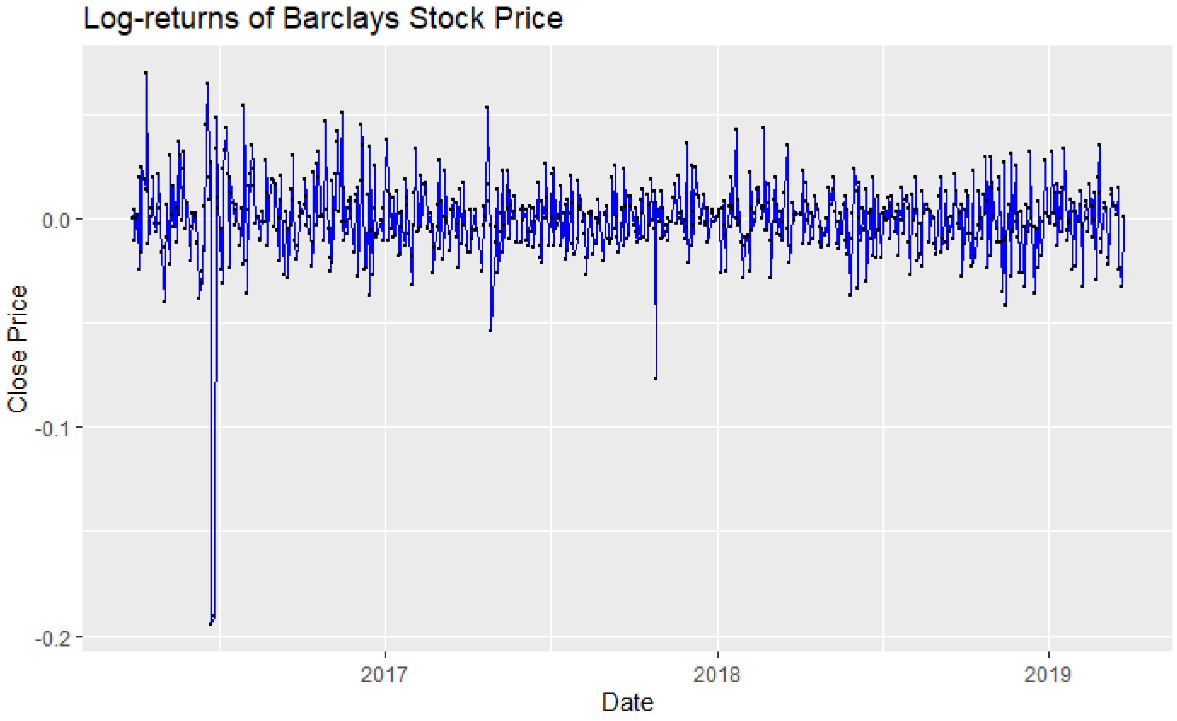


Fig 6

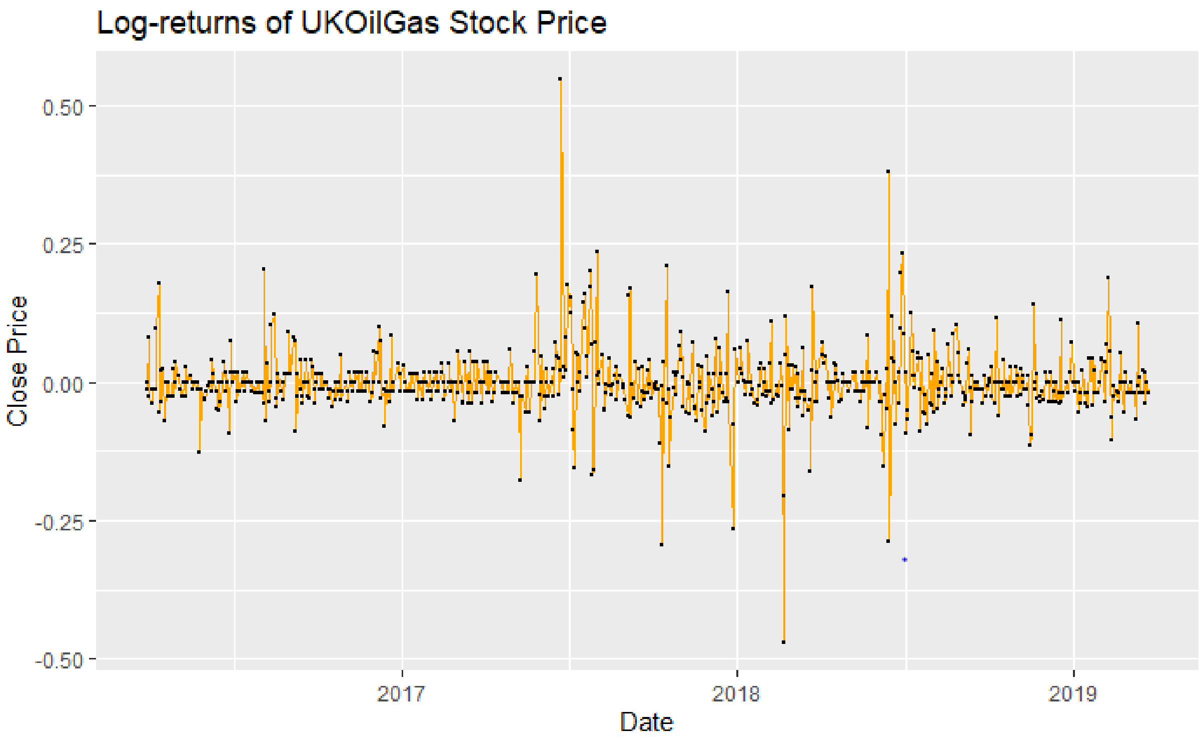

Log-returns of UKOilGas Stock Price
0.50
0.25
0.00
-0.25
-0.50
Close Price
2017
2018
2019
Date

Fig 7

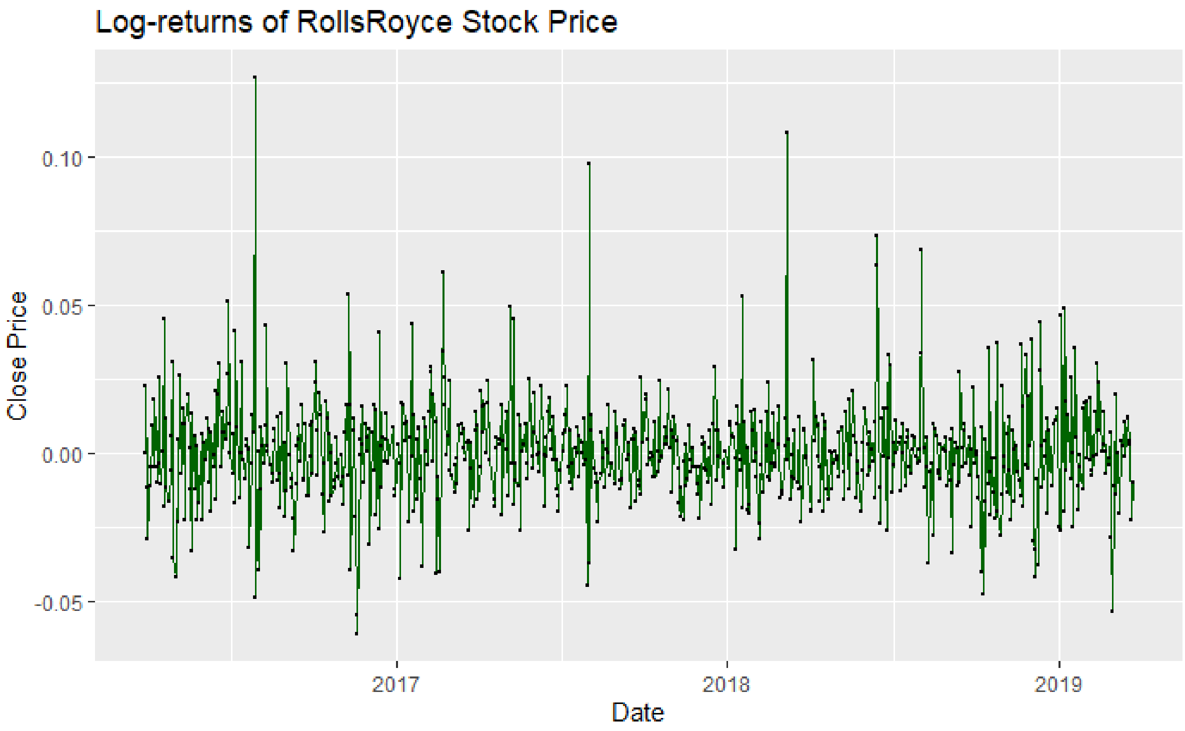

Log-returns of RollsRoyce Stock Price
0.10
0.05
0.00
-0.05
Close Price
2017
2018
2019
Date

Fig 8

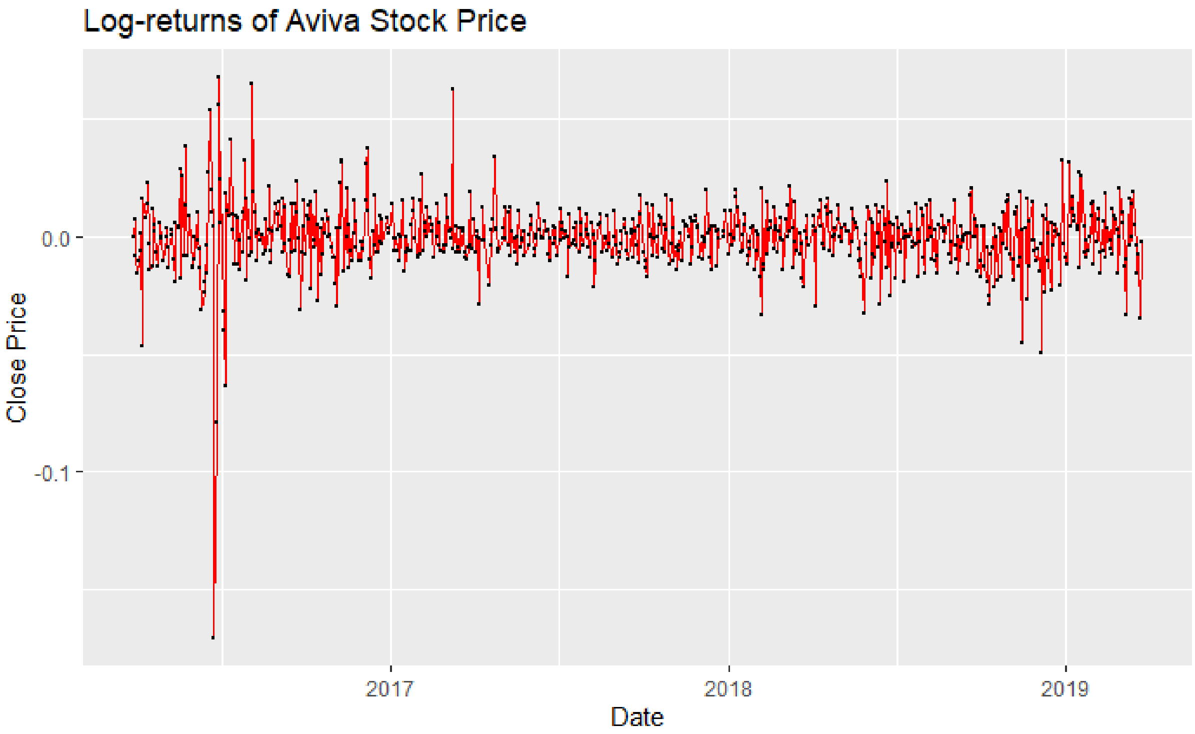


Fig 9

Normalization to z-score

To obtain the z-score of the log-returned signal we transform the data to have a mean of zero and standard deviation 1. The data points can be standardized to find the z-score with the below formula

$$z_i = \frac{x_i - \bar{x}}{s}$$ [2]

where, xi is a data point (x1, x2...xn)

x̄ is the sample mean.

S is the sample standard deviation.

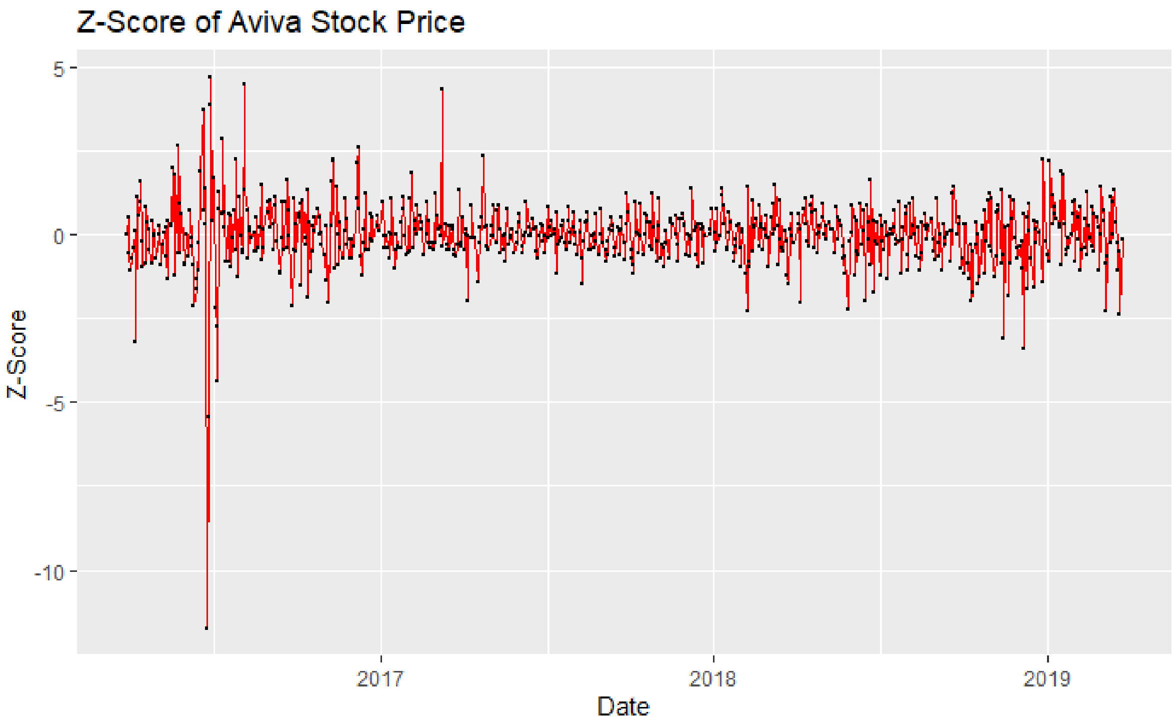

Z-Score of Aviva Stock Price
5
0
-5
-10
Z-Score
2017
2018
2019
Date

Fig 10

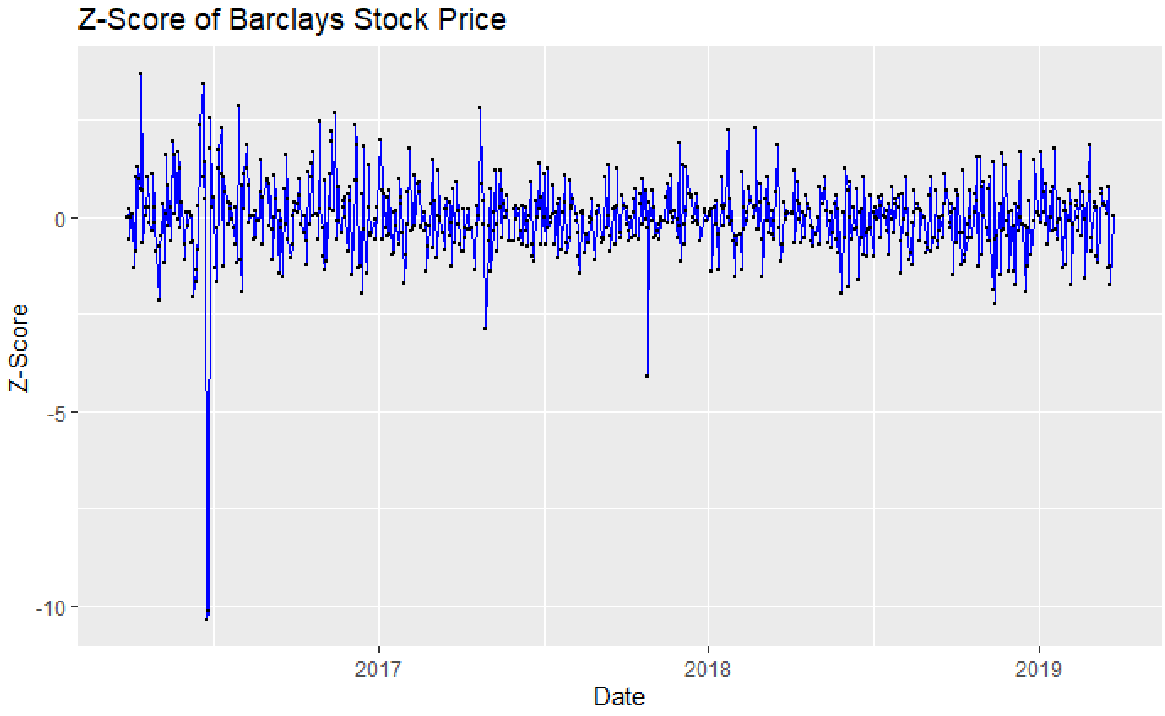

Z-Score of Barclays Stock Price
0
-5
-10
Z-Score
2017
2018
2019
Date

Fig 11

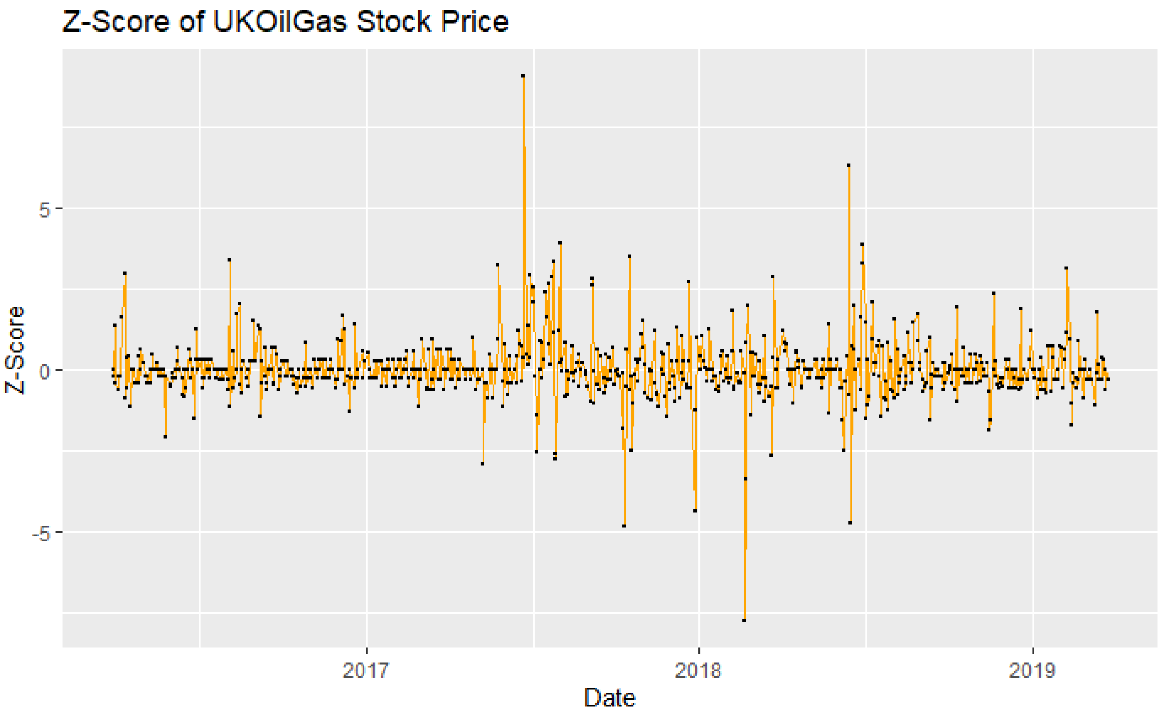

Z-Score of UKOilGas Stock Price
Z-Score
5
0
-5
2017
2018
2019
Date

Fig 12

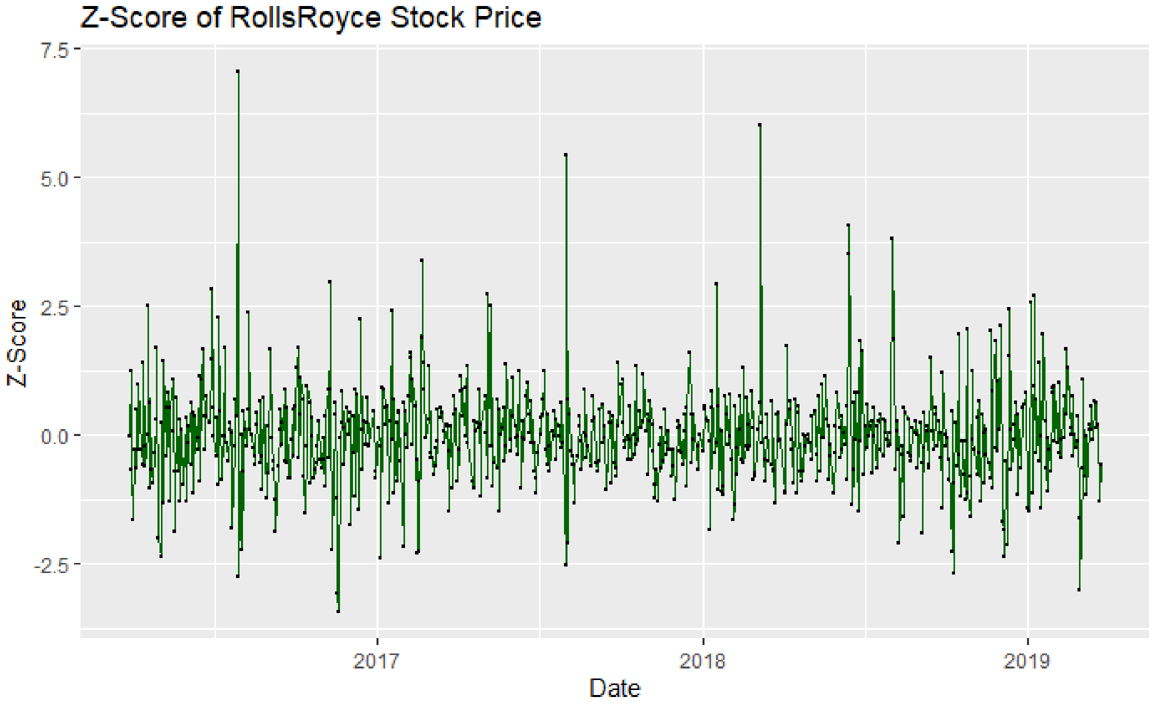

Z-Score of RollsRoyce Stock Price
Z-Score
7.5
5.0
2.5
0.0
-2.5
2017
2018
2019
Date

Fig 13

Mean and Variance

Stock Name	Mean of actual time series	Mean of the log returned signal	Mean of the z-score signal
Aviva	475.1562	0.00014	1.484896e-18
RollsRoyce	829.7833	0.00037	1.552783e-18
Barclays	189.3234	3.908054e-05	-7.134141e-18
UKOilGas	2.3209	0.00052	1.293812e-17

Table 2

Stock Name	Variance of actual time series	Variance of the log returned signal	Variance of the z-score signal
Aviva	47.55886	0.01451	1
RollsRoyce	105.8442	0.01793	1
Barclays	24.11111	0.01878	1
UKOilGas	1.697596	0.06038	1

Table 3

Table 2 above, contains the mean of the actual time series, log-returned series and the z-score series. We observe that the mean is almost/close to zero for the z-score signal.

Table 3 above, contains the variance of the actual time series, log-returned series and the z-score series. We observe that the variance is one for the z-score signal.

Linear Regression Prediction

I have used Multiple linear regression to create the predictor module for the four stocks. The stock to be predicted at time 't+1' forms the dependent variable. The stock to be predicted at time t and the other 3 stocks at time t are the independent variables which the dependent variable (The stock to be predicted at time 't+1') depends on.

The basic multiple linear regression model is as follows:

$$\hat{}\mathbf{Y} = \mathbf{b_o} + \mathbf{b_1x_1} + \mathbf{b_2x_2} + \mathbf{b_3x_3} + \mathbf{b_4x_4}$$

where, yn	is the value to be predicted – in our case it is g(t+1)
bo	is the intercept
b1, b2, b3, b4	are the coefficients of the slopes of the three independent variables – in our case it is the predicted signal itself at time 't' and the other 3 stocks at time t.

Below are the plots for the 4 different stocks predicted using the linear model lm() function in R. A closer look into the first 100 points of the plot is provided.

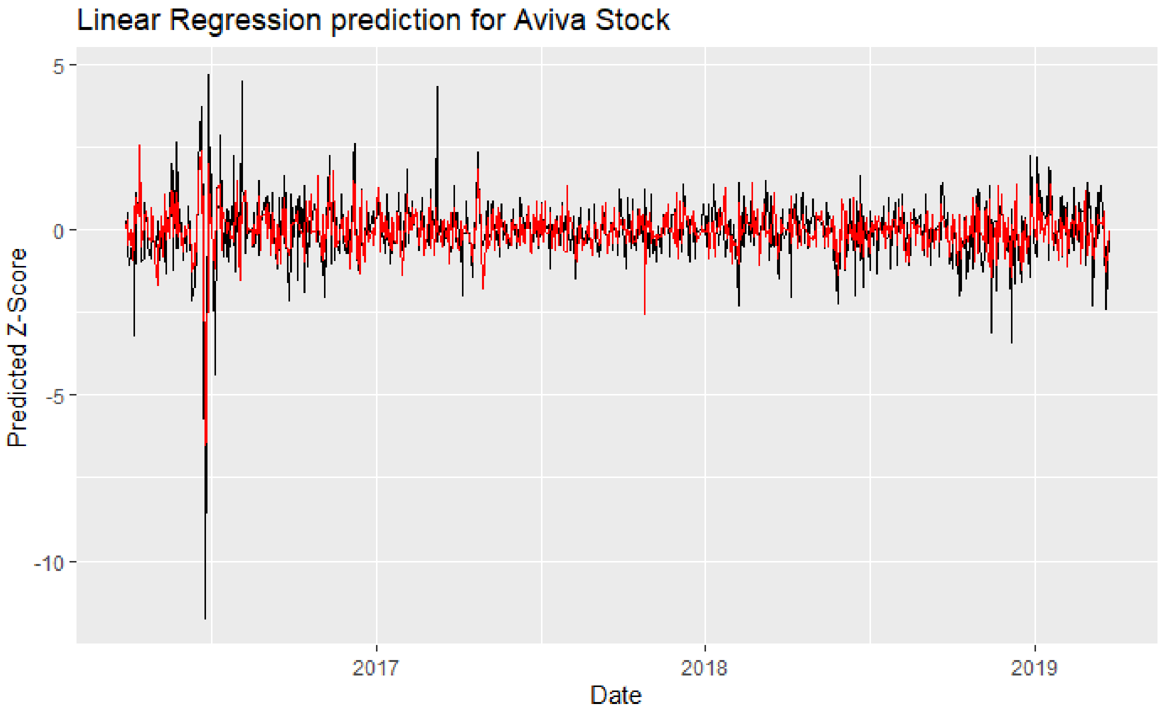


Fig 14

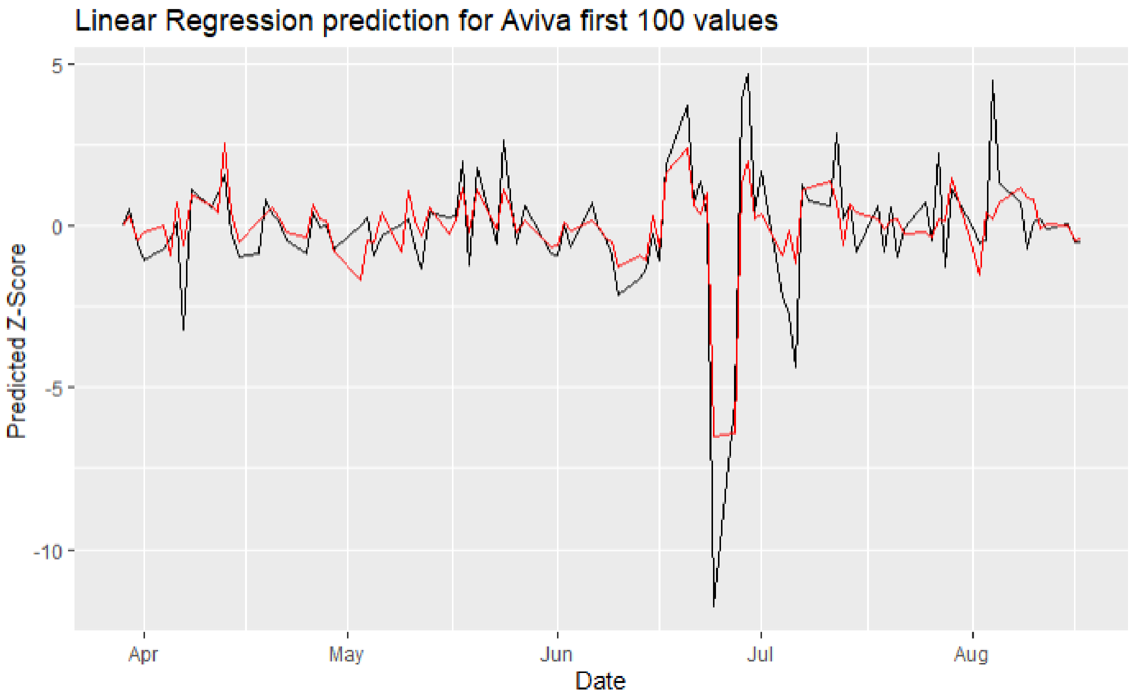


Fig 15

Coefficients	Estimate	Std. Error	t value	Pr(>|t|)
Intercept	6.346e-17	2.714e-02	0.000	1.000
UKOilGas	1.343e-02	2.728e-02	0.492	0.623
Barclays	6.349e-01	2.755e-02	23.045 <	2e-16 ***
RollsRoyce	1.265e-01	2.744e-02	4.609	4.76e-06 ***

Table 4

From table 4 we observe that the independent variables Barclays and RollsRoyce are most significant for the prediction of Aviva stocks. This means that Barclays and RollsRoyce share a significant linear relationship with Aviva. This is mainly due to the domain that they belong as Aviva and Barclays belong to the financial domain and will suffer the same impacts due to market dynamics. The UKOilGas is very less significant. The Residual standard error is the quality of the fit. The Residual standard error for this model is 0.7472.

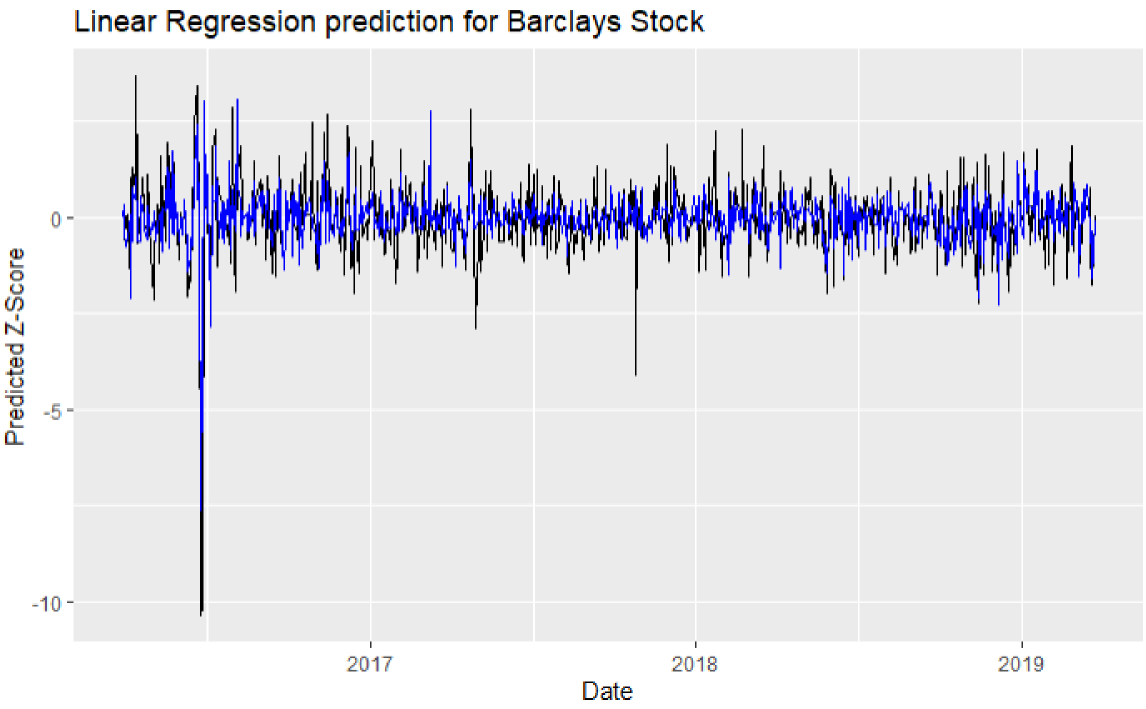


Fig 16

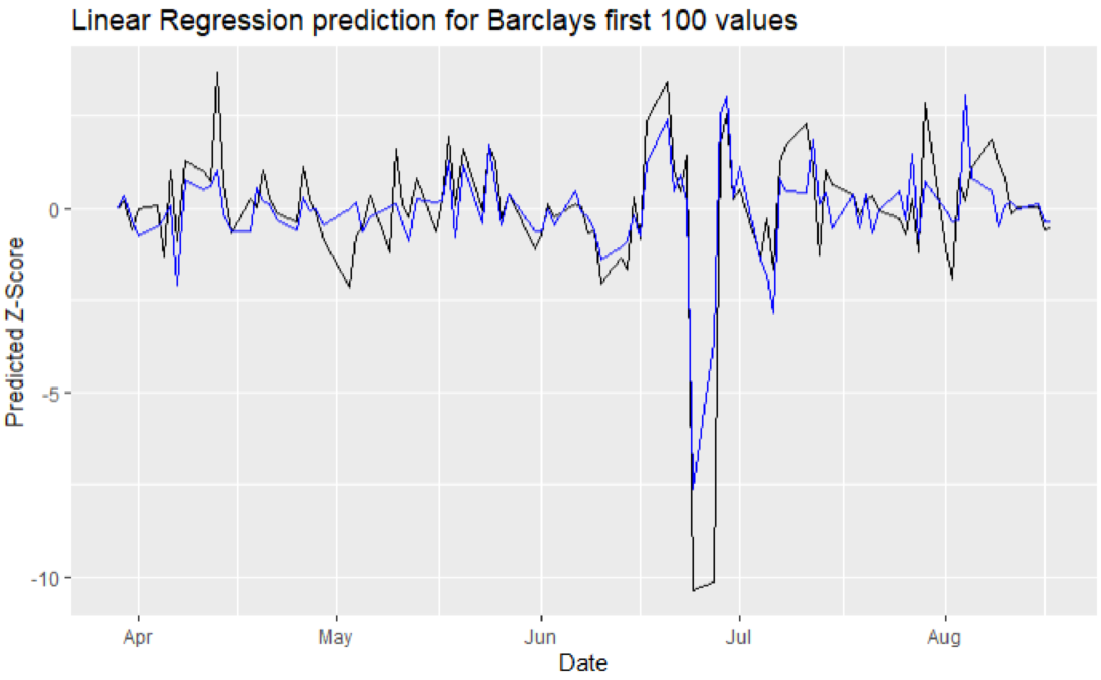


Fig 17

Coefficients	Estimate	Std. Error	t value	Pr(>|t|)
Intercept	-5.017e-17	2.748e-02	0.000	1.000
UKOilGas	4.384e-02	2.758e-02	1.590	0.112
Aviva	6.509e-01	2.824e-02	23.045	<2e-16 ***
RollsRoyce	1.194e-04	2.817e-02	0.004	0.997

Table 5

From table 5 we observe that the independent variables Aviva is the most significant for the prediction of Barclays stocks. This means that Aviva shares a significant linear relationship with Barclays. The Residual standard error is the quality of the fit. The Residual standard error for this model is 0.7565.

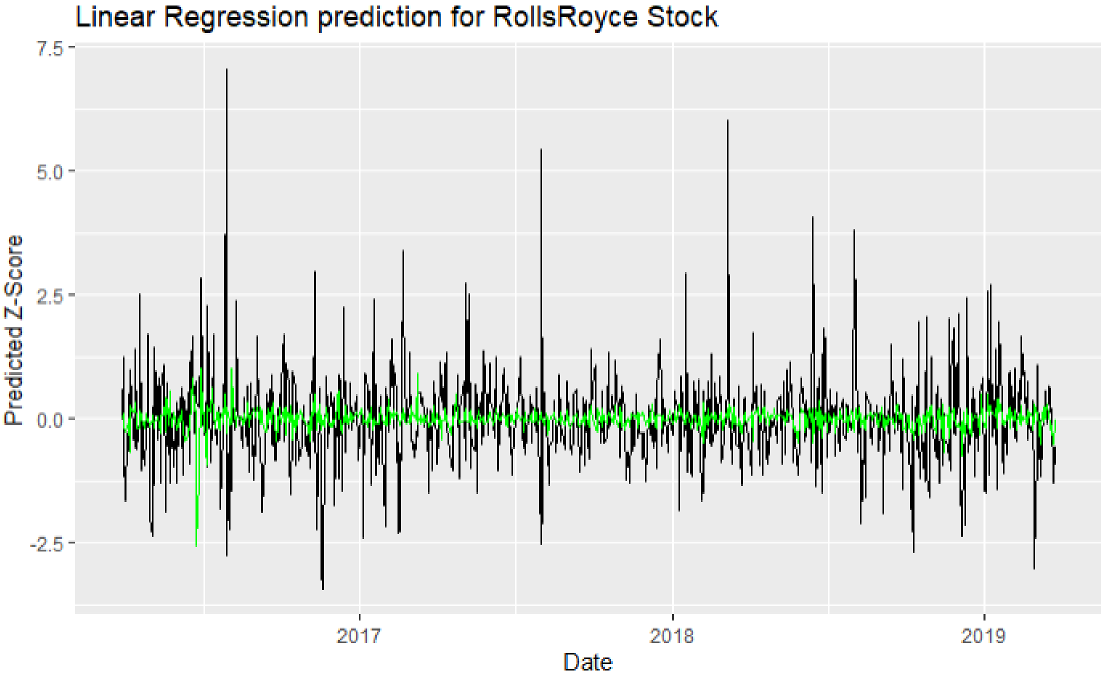


Fig 18

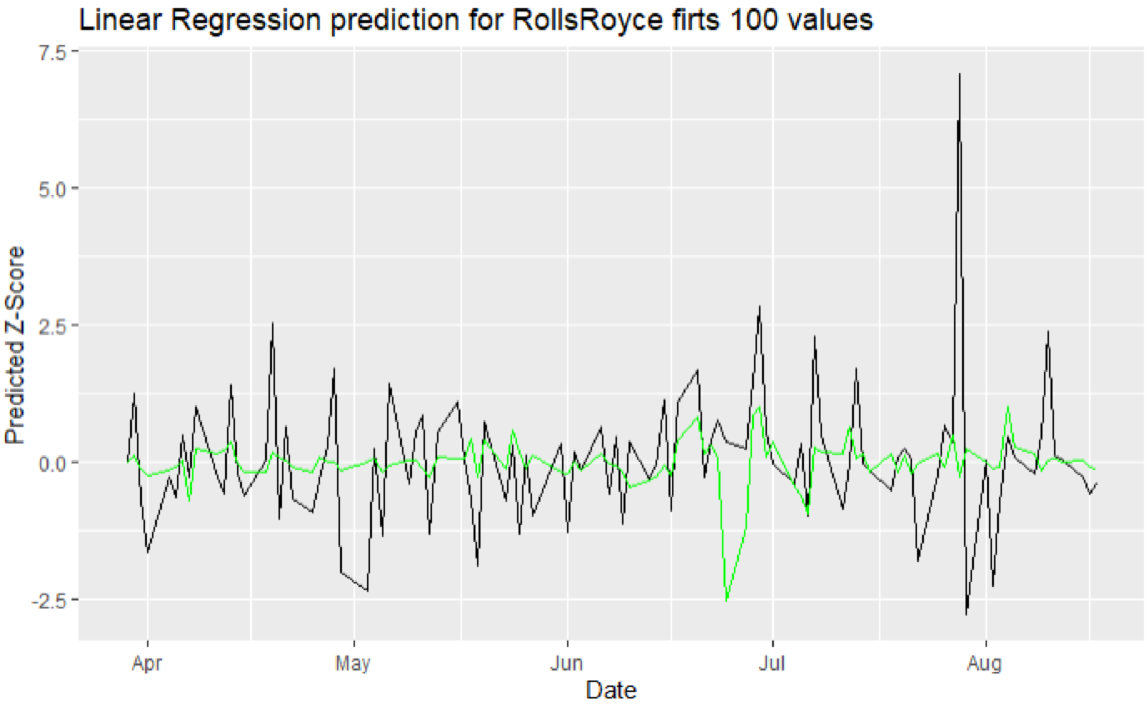


Fig 19

Coefficients	Estimate	Std. Error	t value	Pr(>|t|)
Intercept	-1.602e-17	3.552e-02	0.000	1.000
UKOilGas	8.339e-03	3.571e-02	0.234	0.815
Aviva	2.166e-01	4.701e-02	4.609	4.76e-06 ***
Barclays	1.996e-04	4.708e-02	0.004	0.997

Table 6

From table 6 we observe that the independent variables Aviva is the most significant for the prediction of RollsRoyce stocks. This means that Aviva shares a significant linear relationship with Rolls-Royce. The Residual standard error is the quality of the fit. The Residual standard error for this model is 0.978.

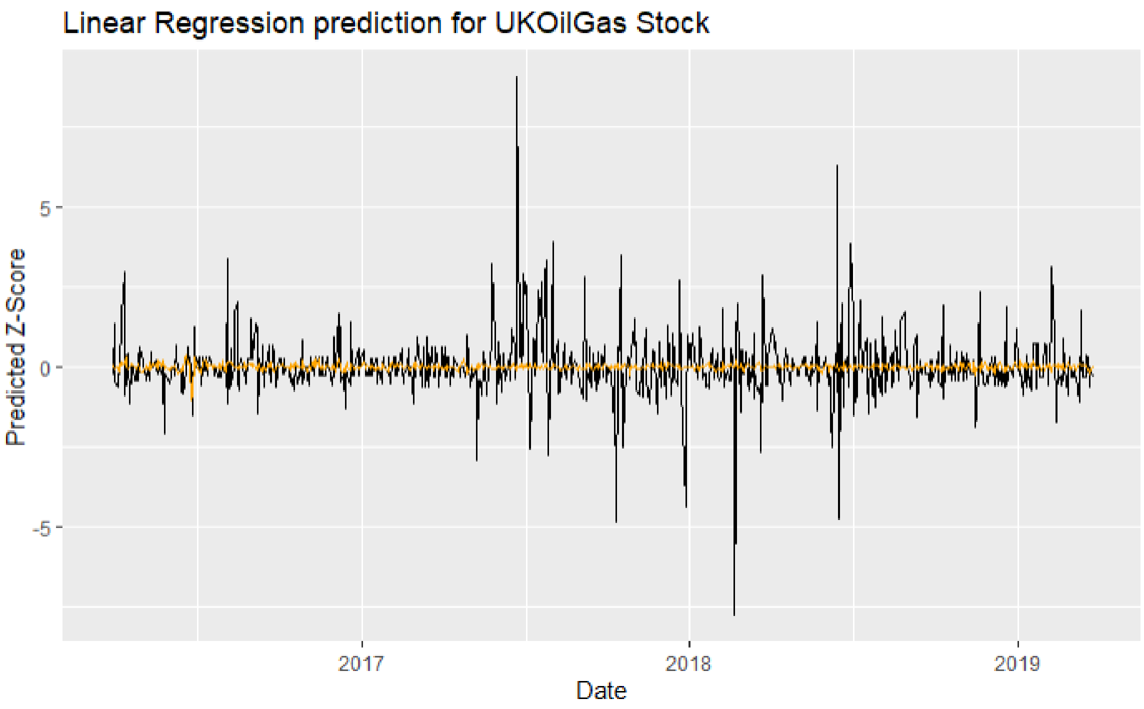


Fig 20

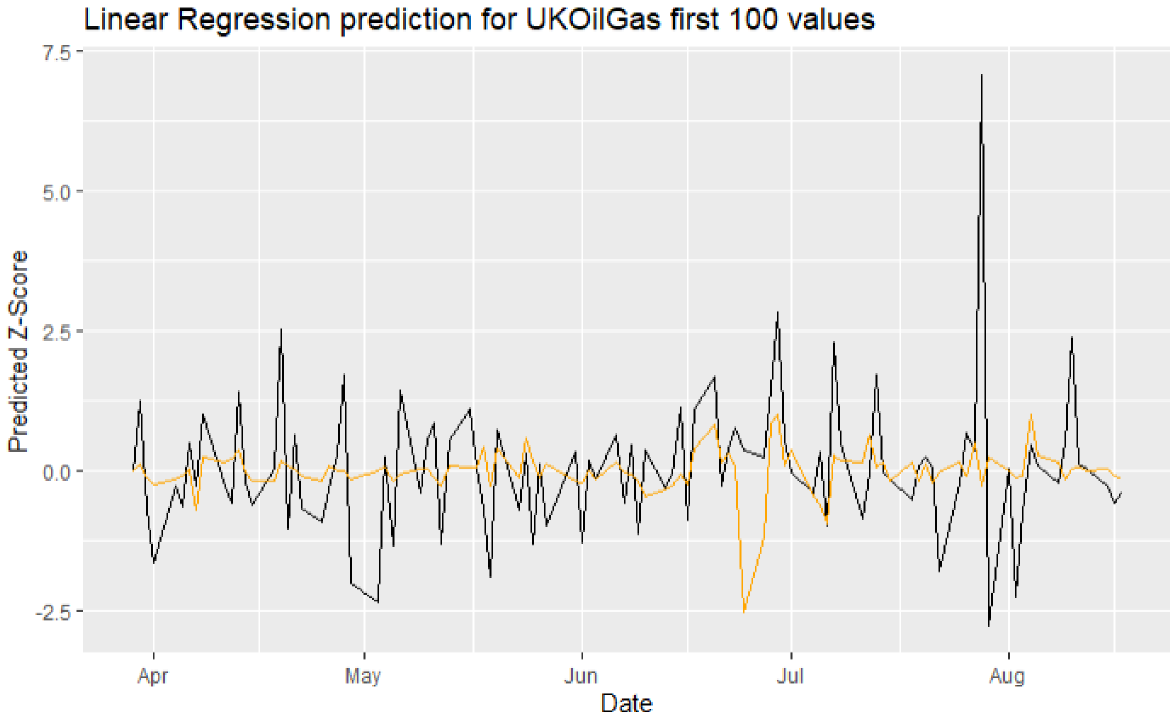


Fig 21

Coefficients	Estimate	Std. Error	t value	Pr(>|t|)
Intercept	6.217e-18	3.623e-02	0.000	1.000
RollsRoyce	8.674e-03	3.714e-02	0.234	0.815
Aviva	2.393e-02	4.860e-02	0.492	0.623
Barclays	7.620e-02	4.793e-02	1.590	0.112

Table 7

From table 7 we observe that none of the independent variables share a significant linear relationship with UKOilGas stocks and therefore from fig 21 we observe that the prediction is very bad. This is mainly due to the domain that the stocks belong to. The Residual standard error is the quality of the fit. The Residual standard error for this model is 0.9974.

From the data above we observe that the error in prediction is higher for UKOilGas as there is no linear relationship with any of the other stocks. This error is the least for prediction of Aviva stocks as it shares significant linear relationships with 2 other stocks.
If we use independent variables from the same domain for prediction of stocks it will result in higher accuracy and reduced residual standard error.

<u>Plots for g(t), g(t+1) and residuals</u>

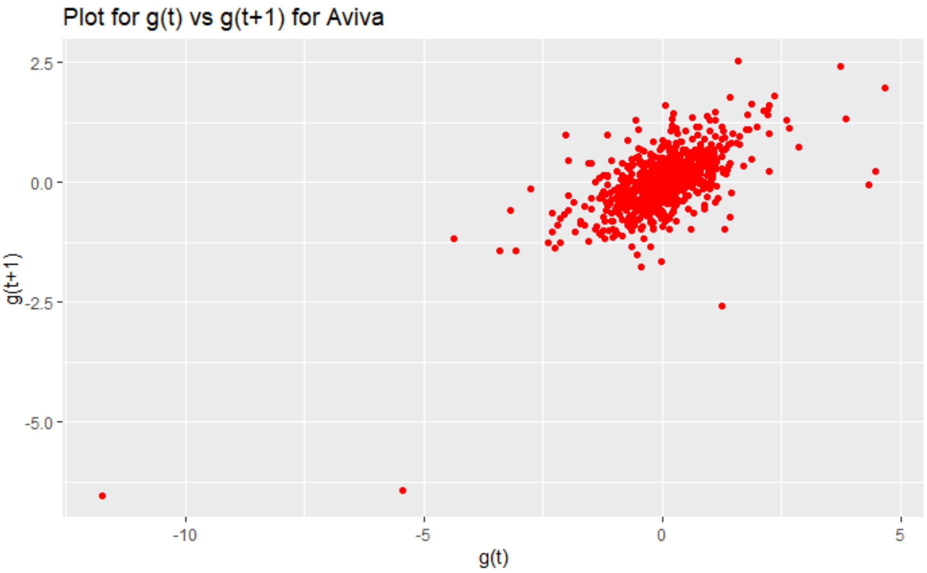


Fig 22

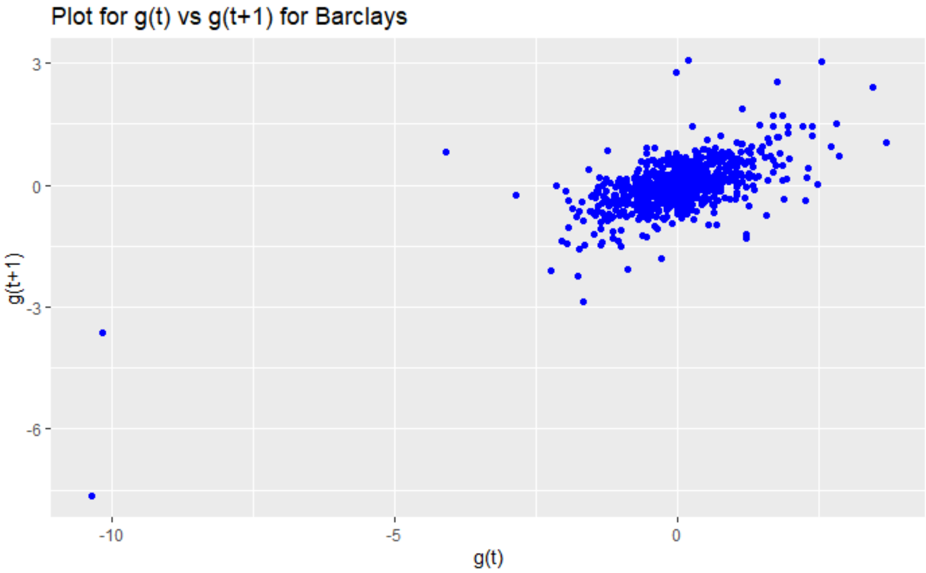


Fig 23

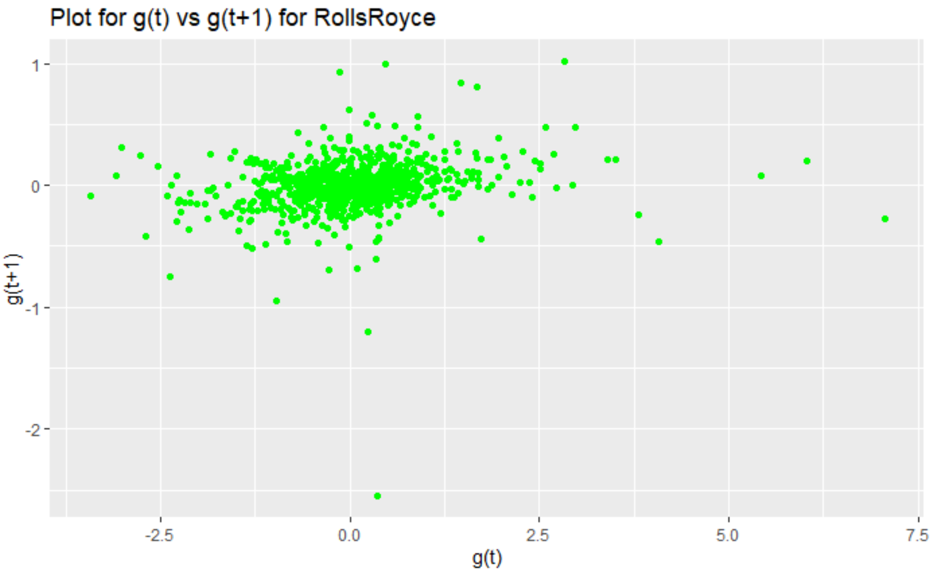

Plot for g(t) vs g(t+1) for RollsRoyce
1
0
-1
-2
g(t+1)
-2.5
0.0
2.5
5.0
7.5
g(t)

Fig 24

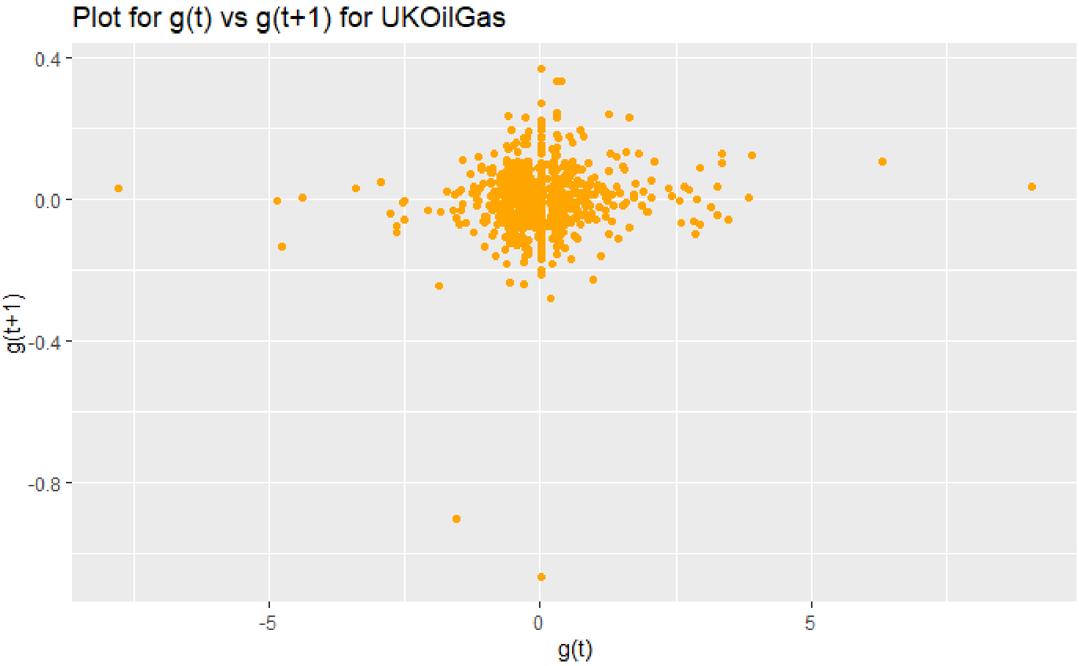

Plot for g(t) vs g(t+1) for UKOilGas
0.4
0.0
-0.4
-0.8
g(t+1)
-5
0
5
g(t)

Fig 25

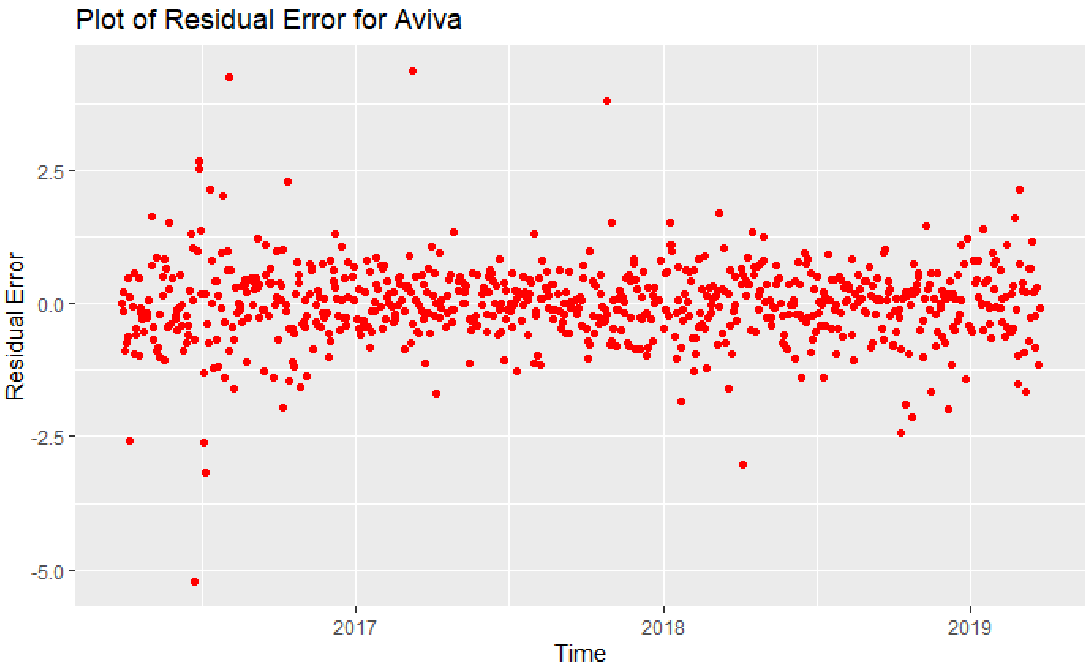

Plot of Residual Error for Aviva
Residual Error
2.5
0.0
-2.5
-5.0
2017
2018
2019
Time

Fig 26

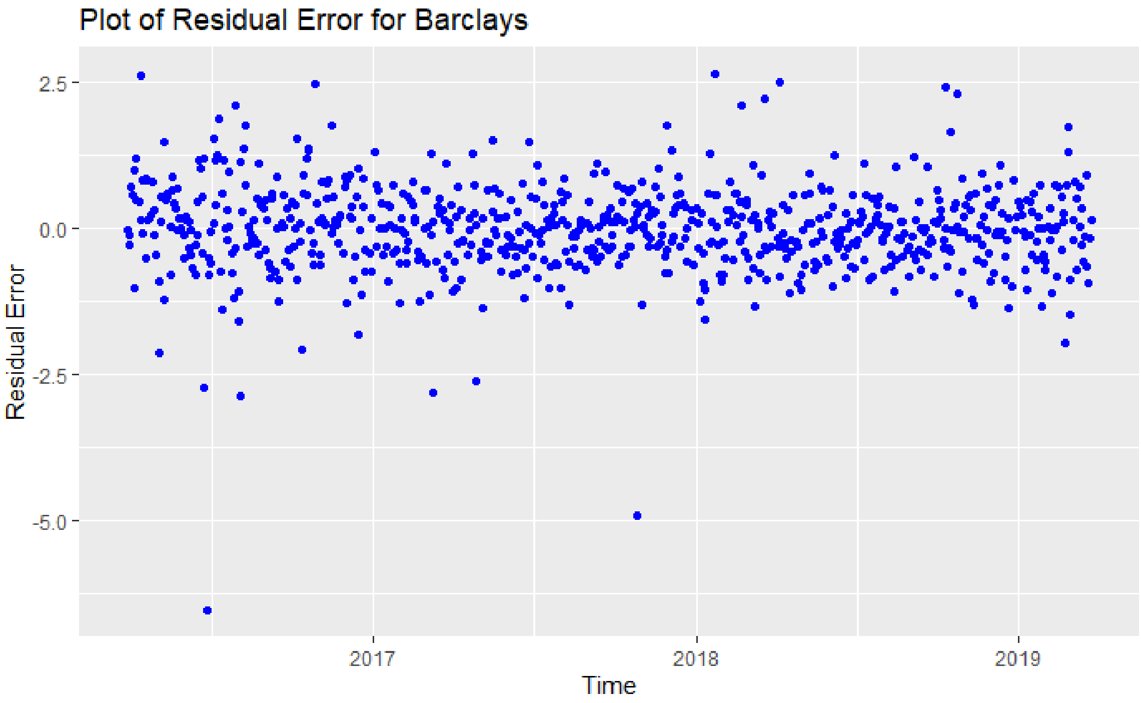

Plot of Residual Error for Barclays
Residual Error
2.5
0.0
-2.5
-5.0
2017
2018
2019
Time

Fig 27

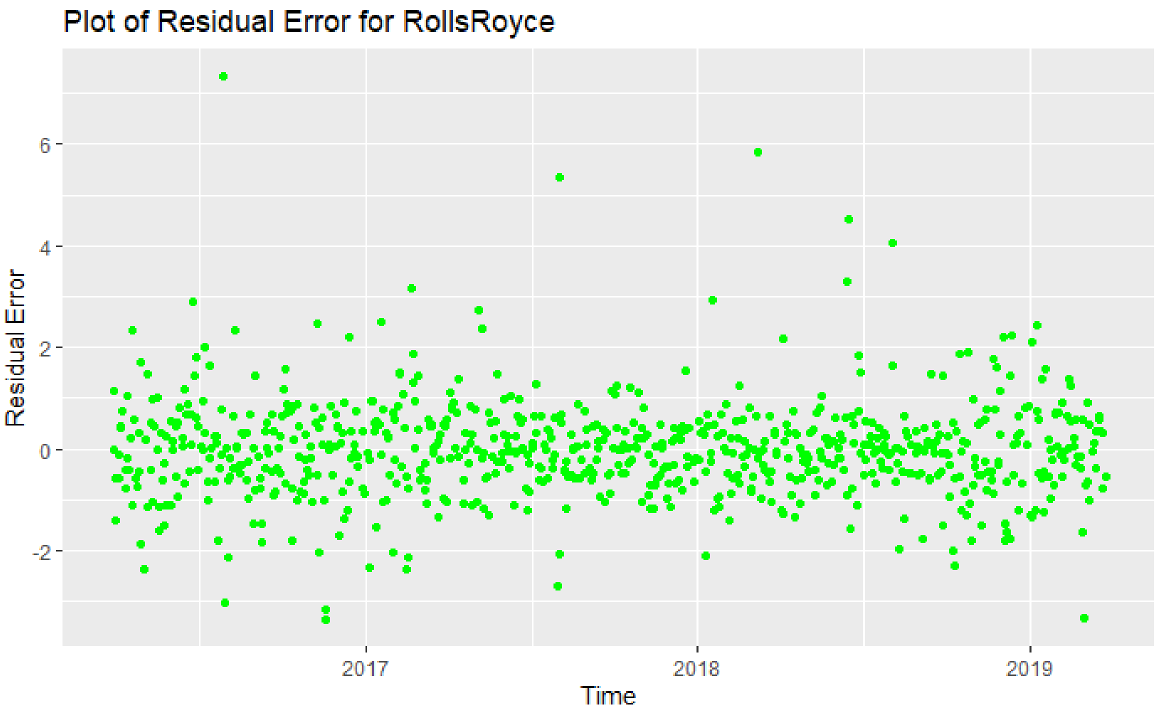

Plot of Residual Error for RollsRoyce
Residual Error
6
4
2
0
-2
2017
2018
2019
Time

Fig 28

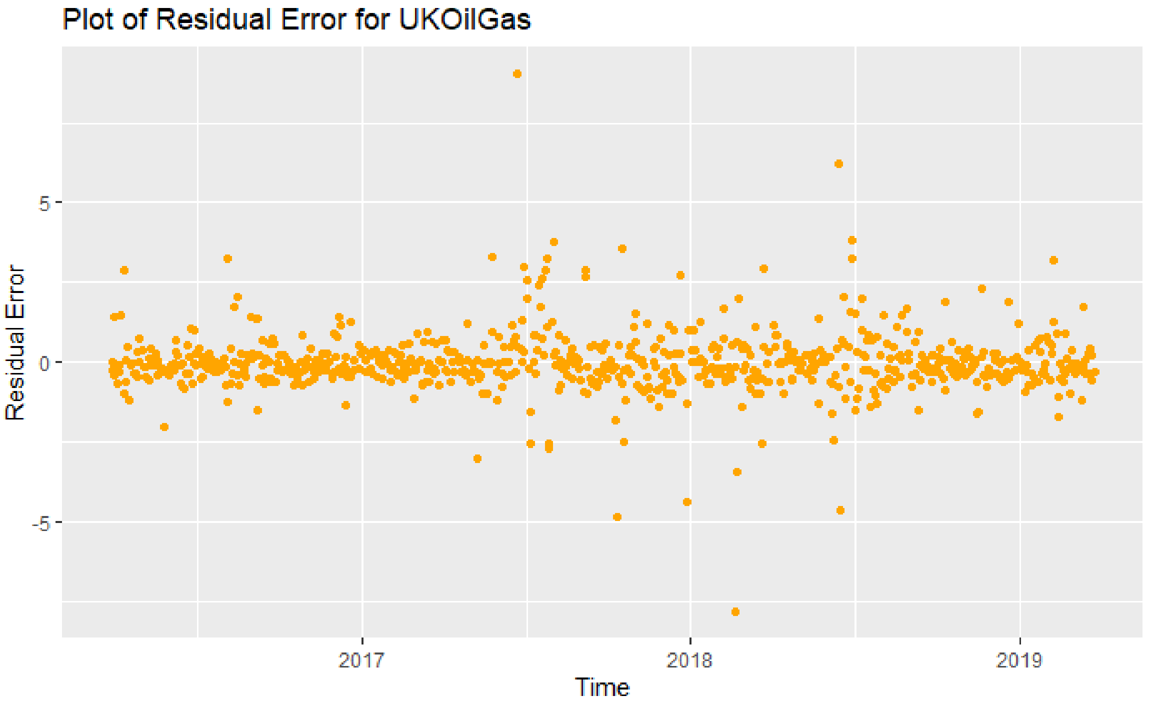

Plot of Residual Error for UKOilGas
Residual Error
5
0
-5
2017
2018
2019
Time

Fig 29

Adaline Neural Network Prediction

A neural network consists of input layer, hidden layer and the output layer. In the fig 22 we observe a neural network.

In this neural network we are feeding the signals

g(t) in our case it is the signal to be predicted at time t

$d1(t)$,$d2(t)$,$d3(t)$ in our case are the 3 other stocks at time t

and the output signal after the error has been calculated.

The Adaline neural network has nodes that accept the input vector ‘x’ and the weight vector ‘w’ and computes the output as follows:

$$y = \sum_{j=1}^{n} x_j w_j + \theta.$$

Where, y is the output of the model. This output is compared with the g(t+1) the actual predicted value. The subtraction gives us the error ‘e’. This error is then feedback to the input by readjusting the weight as

w(new) = w + learning_rate* (e)*x ------ (1)

teta is the bias of the network (which behaves as a threshold).

n is the total number of inputs.

The neural network first has to be trained. During the training at the end of every iteration the error is computed and feedback to the input. The network converges or stops training when the error reduces to zero.

I have divided the data into training and testing. For training I have used 70% of the data(542 samples). For testing I have used 30% of the remaining data (216 samples).

Neural network for Aviva Prediction

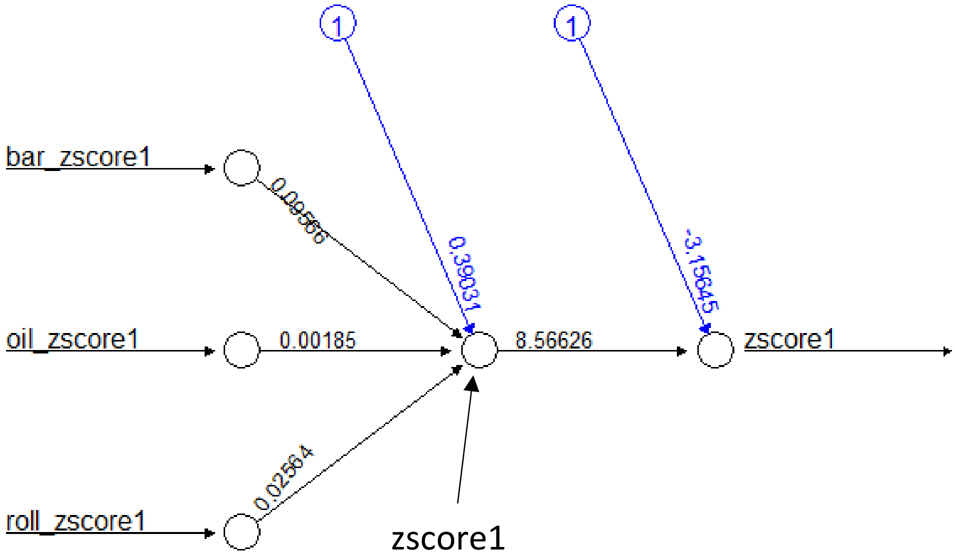


Fig 30

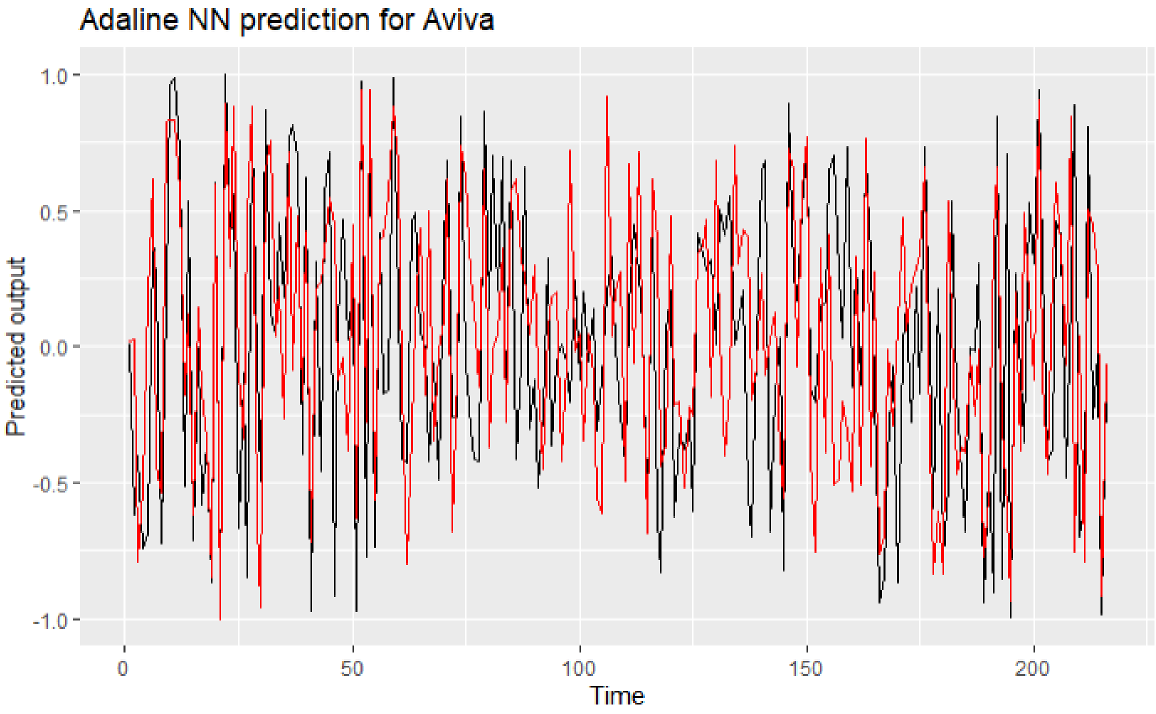


Fig 31

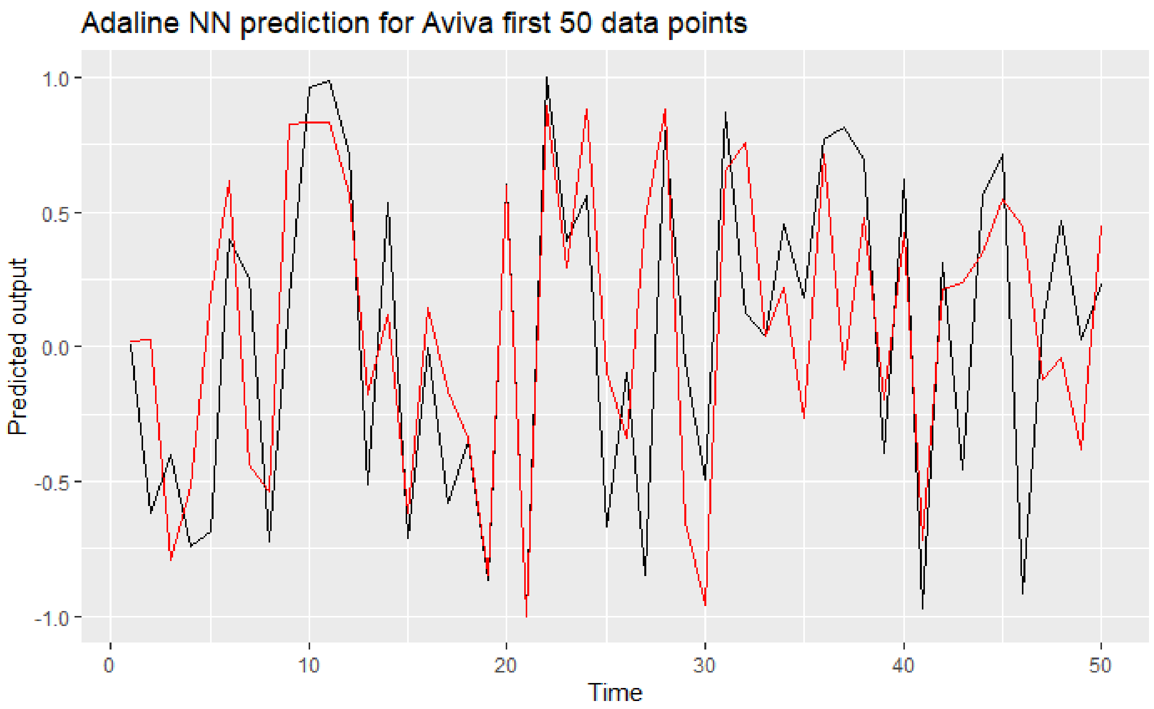


Fig 32

Neural Network for Barclays

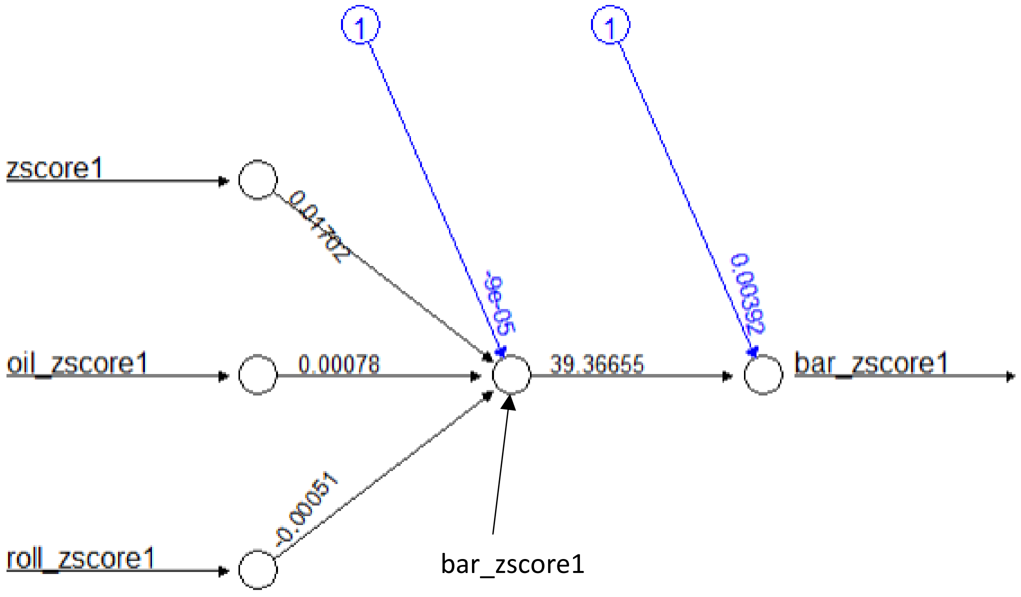


Fig 33

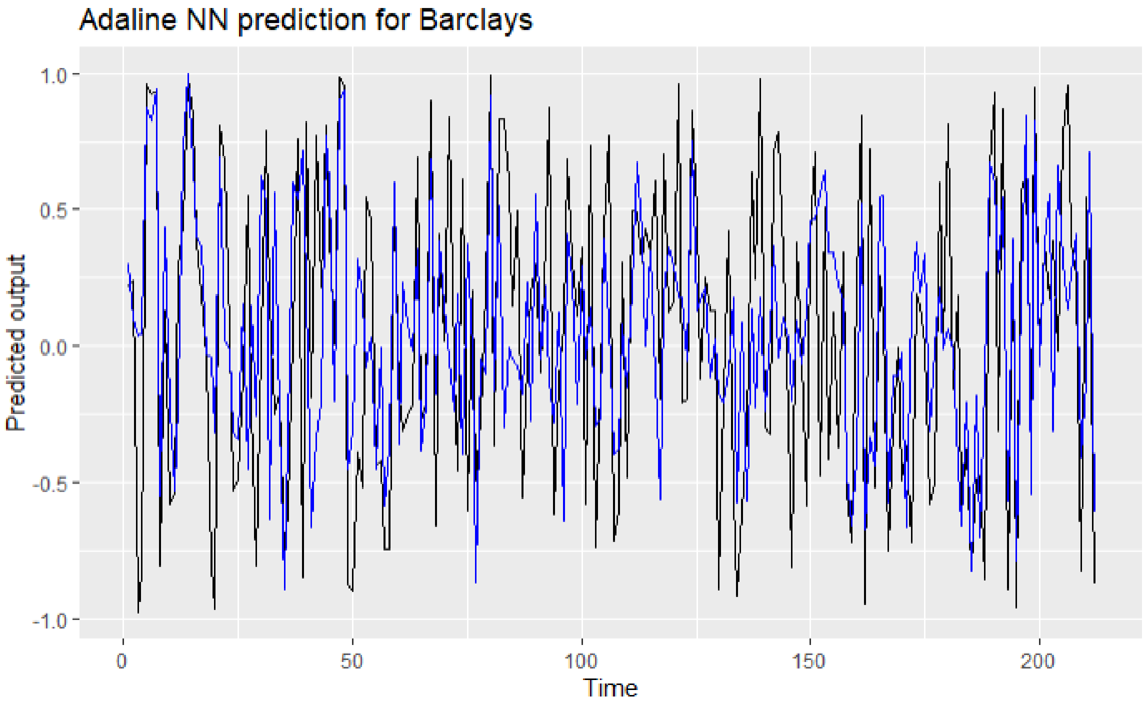

Adaline NN prediction for Barclays
1.0
0.5
0.0
-0.5
-1.0
Predicted output
0
50
100
150
200
Time

Fig 34

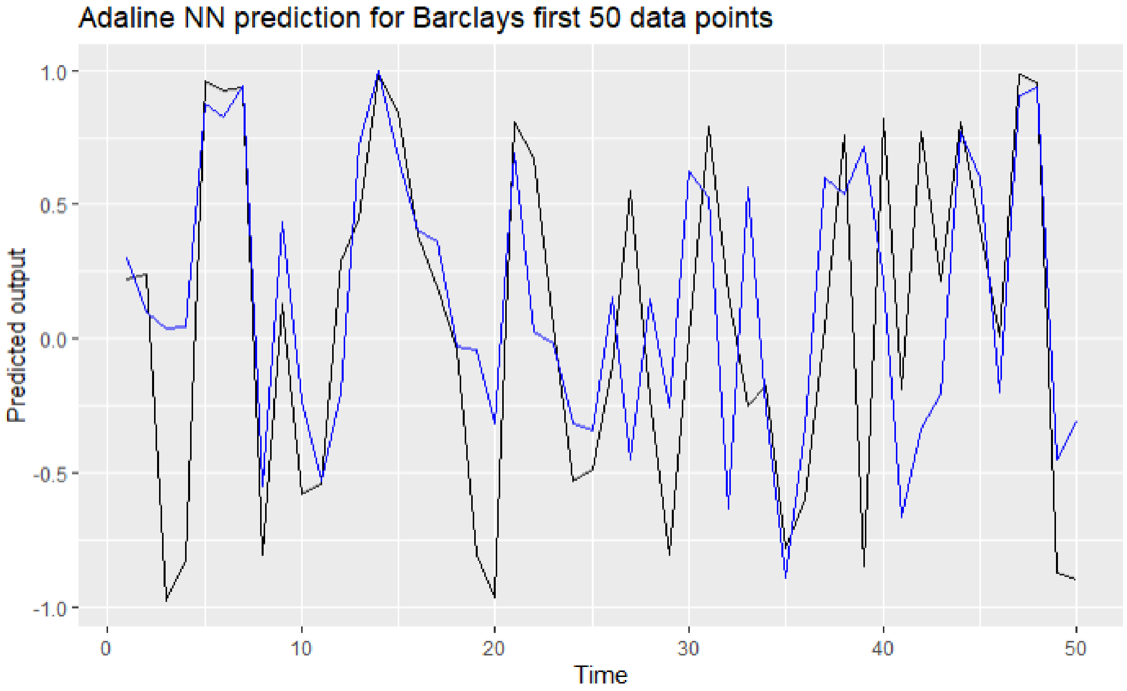

Adaline NN prediction for Barclays first 50 data points
1.0
0.5
0.0
-0.5
-1.0
Predicted output
0
10
20
30
40
50
Time

Fig 35

Neural Network for Rolls Royce

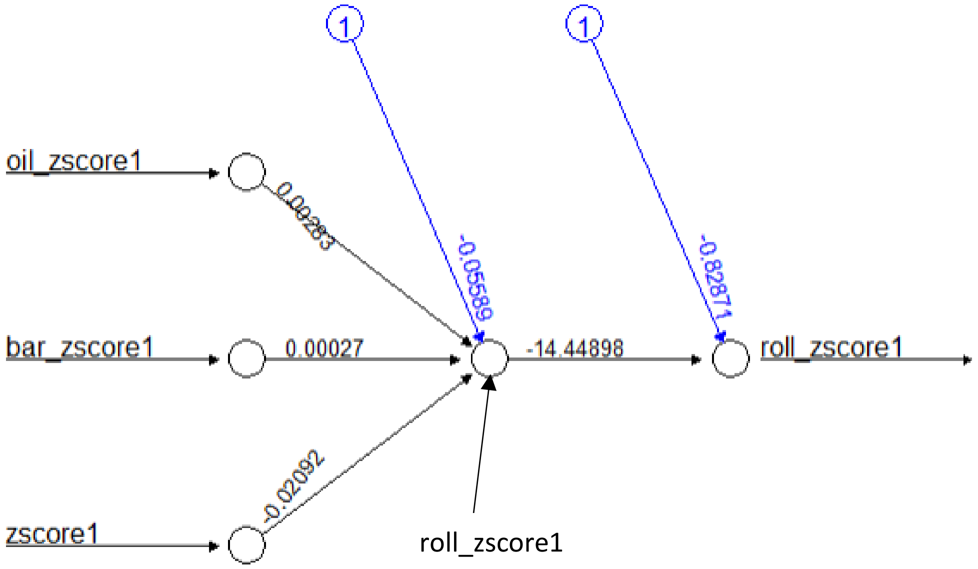


Fig 36

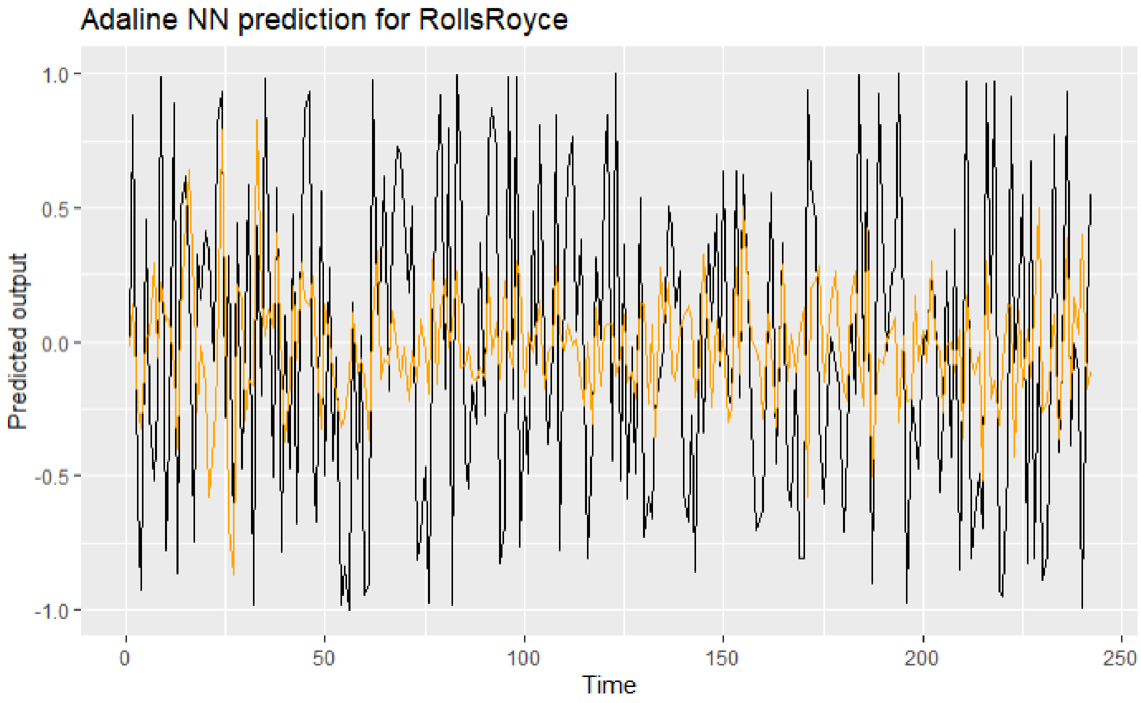


Fig 37

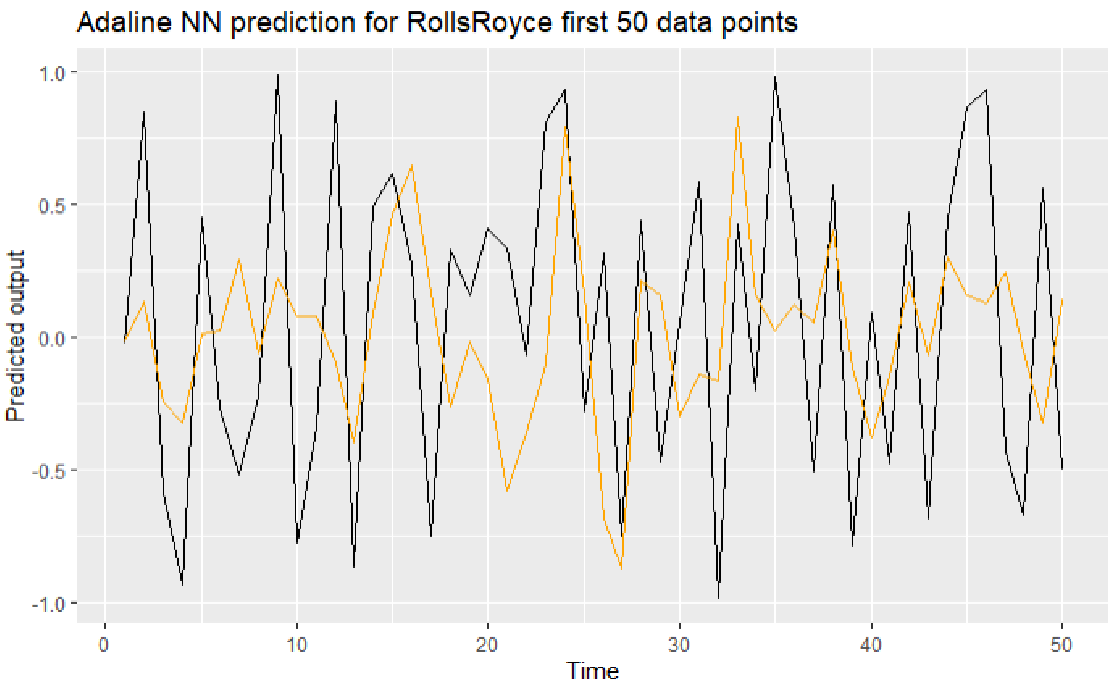


Fig 38

Neural Network for UKOilGas

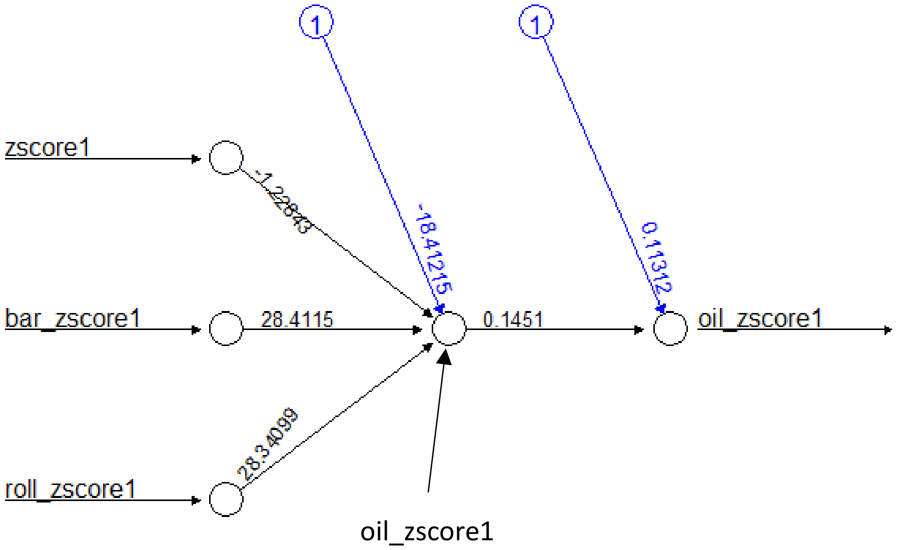


Fig 39

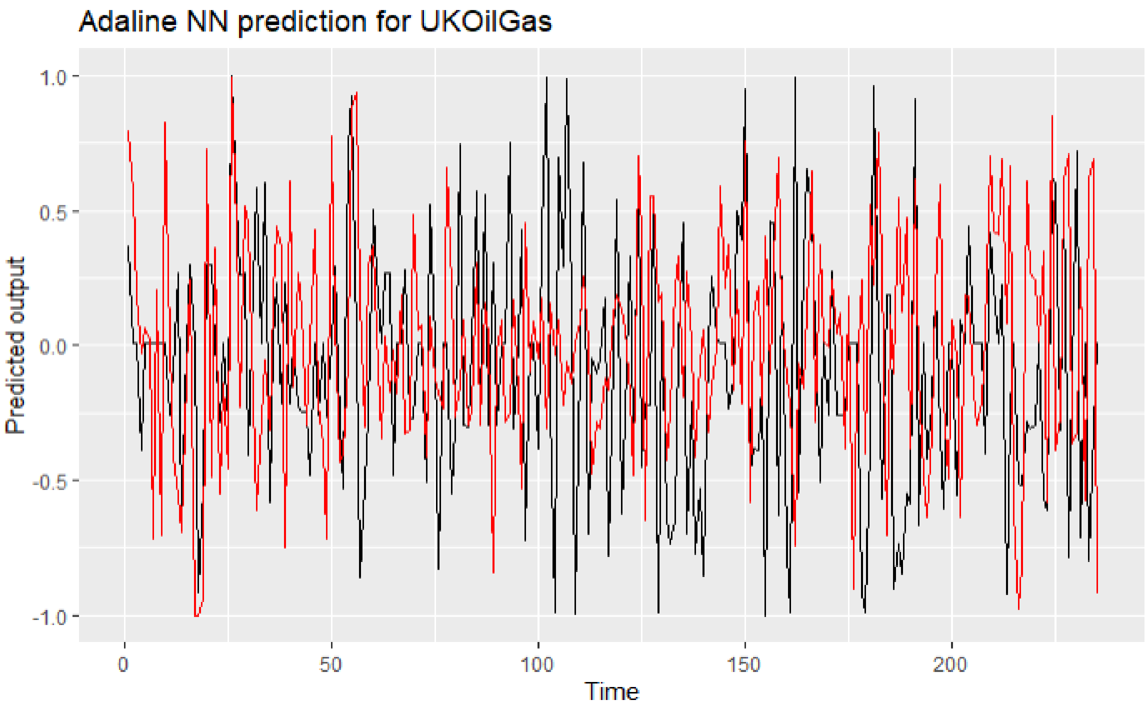

Adaline NN prediction for UKOilGas
Predicted output
1.0
0.5
0.0
-0.5
-1.0
0
50
100
150
200
Time

Fig 40

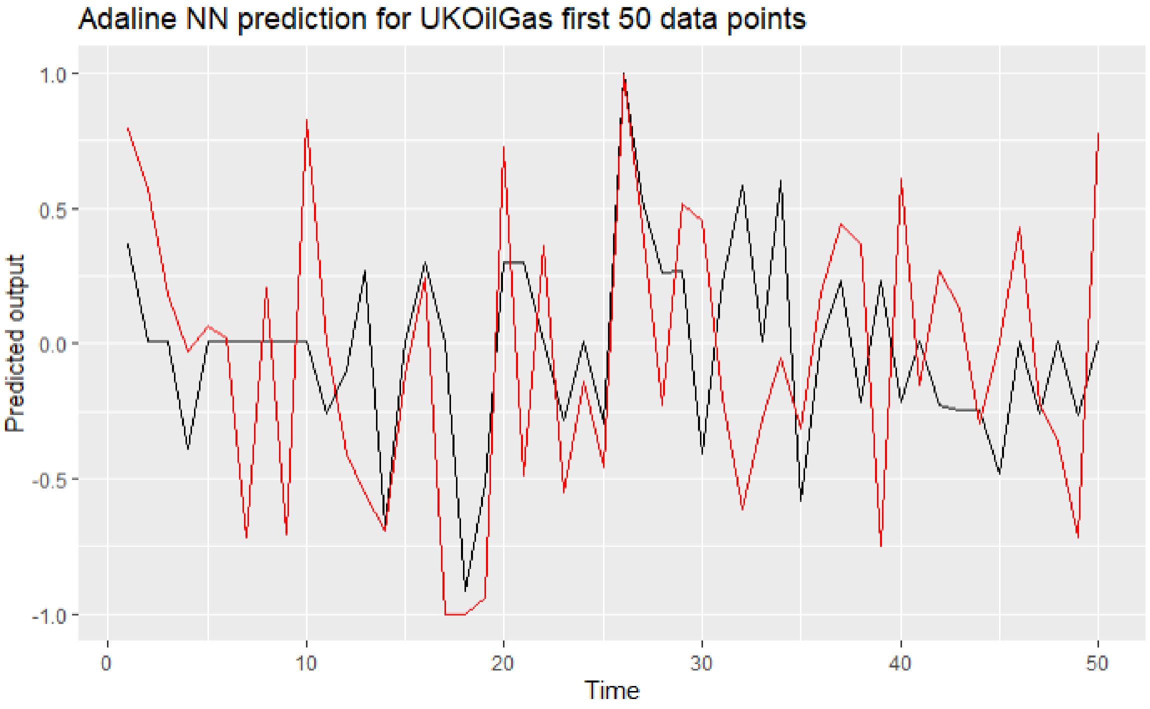

Adaline NN prediction for UKOilGas first 50 data points
Predicted output
1.0
0.5
0.0
-0.5
-1.0
0
10
20
30
40
50
Time

Fig 41

Stock Name	Root Mean Square Error (RMSE)
Aviva	0.3214
Barclays	0.4559
RollsRoyce	0.5580
UKOilGas	0.5998

Table 8

From Table 8 above we observe that the root mean square errors for the neural network prediction is the highest for UKOilGas and the least for Aviva. In the previous solution we have seen that there is no clear relationship observed between the other stocks and UKOilGas which is the reason for bad predictions in this stock.

Learning rate

Learning rate in a neural network controls how fast or slowly a neural network learns or trains itself[3]. In an Adaline neural network, if the error (difference between the target output and the predicted output) is not zero, the weights are updated with the formula:

w(next iteration) = w(current iteration) + learning_rate* (e)*x(current iteration) ------ (1)

The range of this learning_rate parameter ranges between 0.0 and 1.0. Instead of updating the weight with the full amount, it is scaled by this learning rate as can be seen from the equation above. For e.g. if the learning_rate = 0.1 it means that in the next iteration the weight is updated by 10% of the estimated weight error[3].

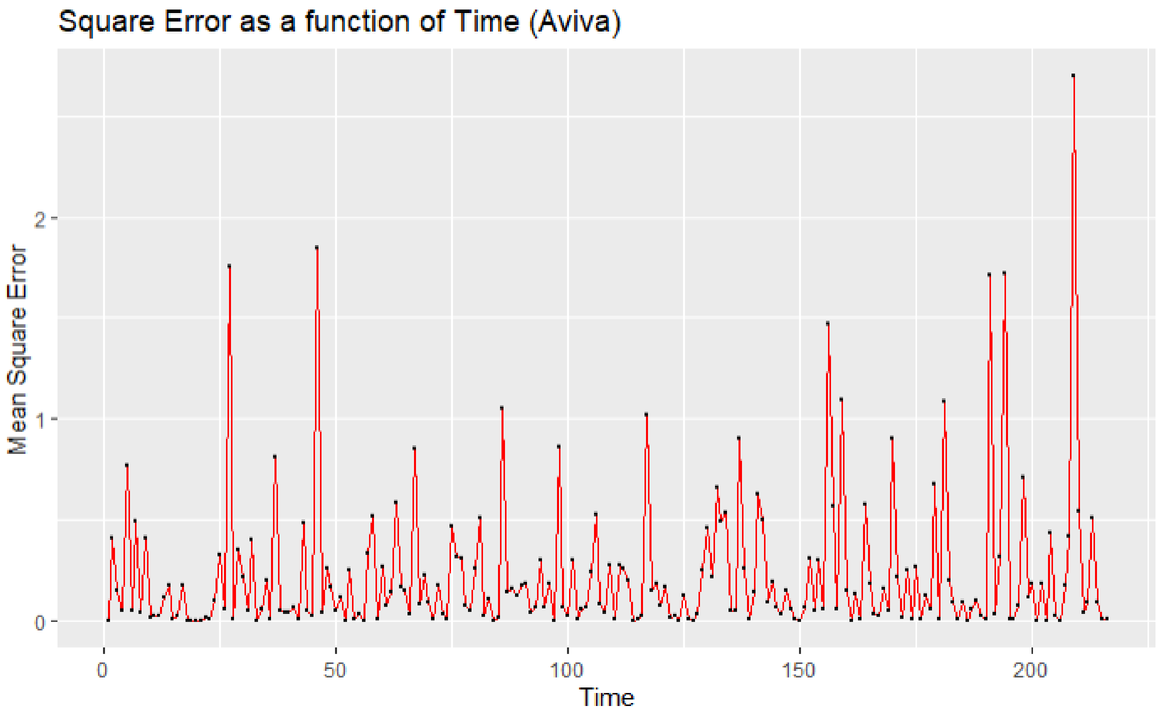
A larger learning rate of approximately 0.9 will train the model faster with sub-optimal performance. A smaller learning rate <0.5 will train the model slowly. It can give an optimal solution however, the training may never converge and can get stuck on a sub-optimal solution.

Stock Names	Number of Steps for Learning Rate = 0.2	Number of Steps for Learning Rate = 0.5	Number of Steps for Learning Rate = 0.8
Aviva	4.9532 x 10^08	5.3916 x 10^06	6.391600e+04
Barclays	4.3728 x 10^07	6.8560 x 10^05	6.468000e+03
RollsRoyce	2.5134 x 10^08	2.4545 x 10^06	3.613600e+04
UKOilGas	1.0185 x 10^07	1.7596 x 10^05	2.023000e+03

Table 9

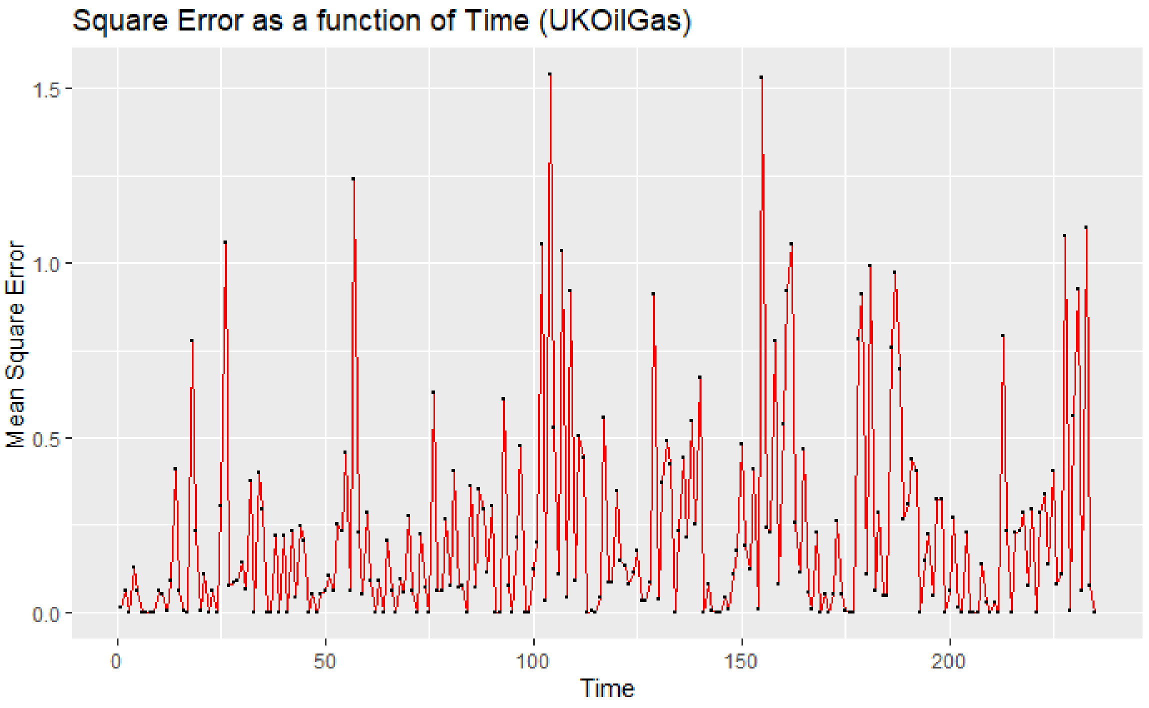
From table 9 we observe that as the learning rate increases the number of steps taken for the network to get trained reduces.

Below are the plots for the Square Errors as a function of Time for the 4 time series.



Square Error as a function of Time (Aviva)

Fig 42



Square Error as a function of Time (UKOilGas)

Fig 43

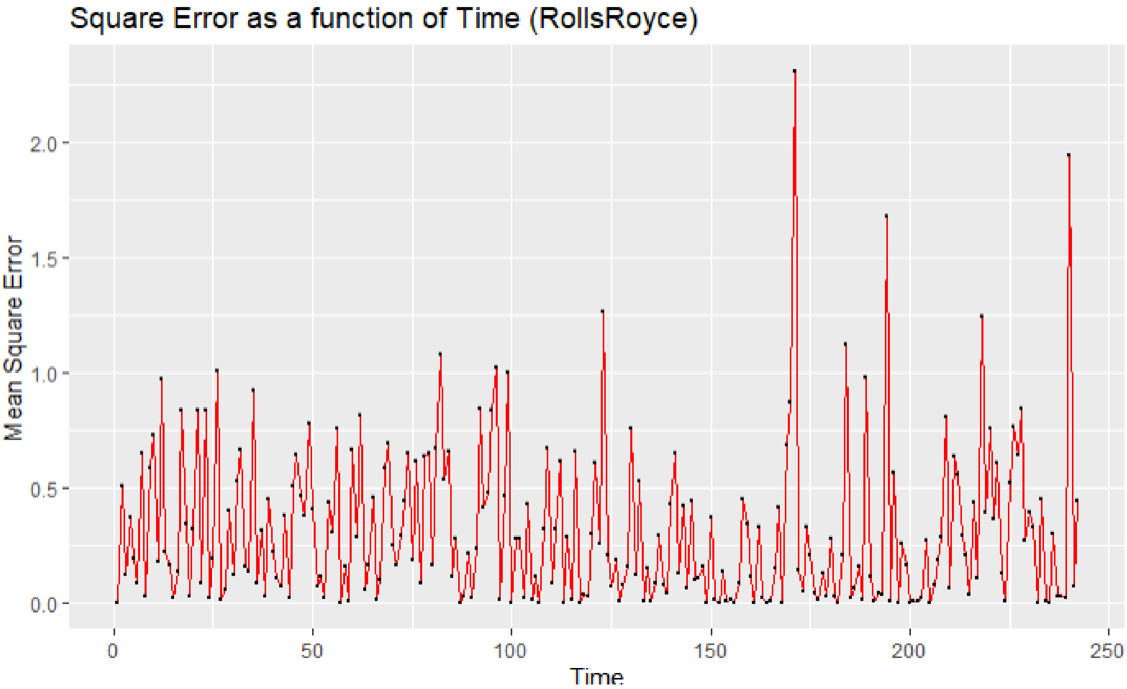

Square Error as a function of Time (RollsRoyce)
Mean Square Error
2.0
1.5
1.0
0.5
0.0
0
50
100
150
200
250
Time

Fig 44

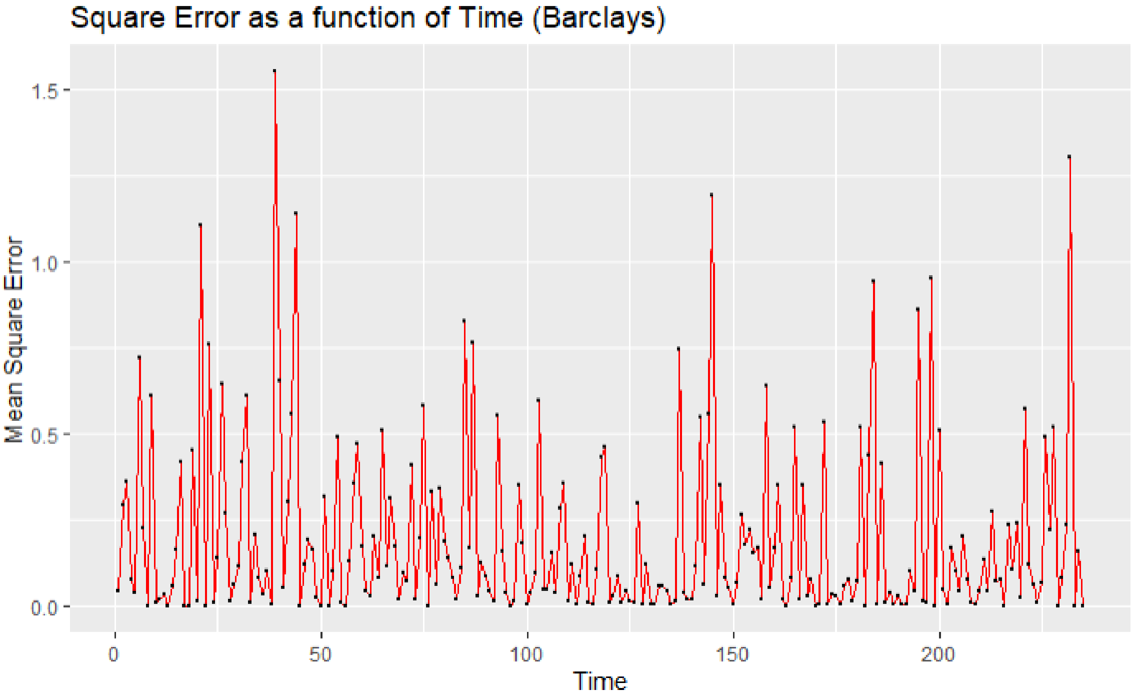

Square Error as a function of Time (Barclays)
Mean Square Error
1.5
1.0
0.5
0.0
0
50
100
150
200
Time

Fig 45

Bibliography

[1] "2018 Stock Market Holidays and Bond Market Holidays." [Online]. Available: https://finance.yahoo.com/news/2018-stock-market-holidays-bond-184004911.html. [Accessed: 22-Apr-2020].

[2] "Normalized Data / Normalization - Statistics How To." [Online]. Available: https://www.statisticshowto.com/normalized/. [Accessed: 22-Apr-2020].

[3] "Understand the Impact of Learning Rate on Neural Network Performance." [Online]. Available: https://machinelearningmastery.com/understand-the-dynamics-of-learning-rate-on-deep-learning-neural-networks/. [Accessed: 03-May-2020].