

Project Dissertation

Analyzing the Interdependency of Financial Market's Instruments
within the Indian Economy using Machine Learning Algorithms

Submitted in partial fulfilment of the requirements for the degree of

Master of Business Administration (M.B.A)

with specialization in Business Analytics



Supervisor:

Mr. Anurag Chaturvedi
Assistant Professor

Submitted By:


Saurabh Kumar	Dhwani Dhingra
2K19/BMBA/15	2K19/BMBA/07

CERTIFICATE

This is to certify that **Dhwani Dhingra (2K19/BMBA/07)**, and **Saurabh Kumar (2K19/BMBA/15)** are bona fide students at University School of Management and Entrepreneurship, Delhi Technological University, Delhi and have successfully completed the project work as prescribed by the Delhi Technological University in the partial fulfilment of the requirement of Master of Business Administration (MBA), Business Analytics Program for the academic year 2019-2021.

The Project Work titled as “*Analyzing the Interdependency of Financial Markets’ Instruments within the Indian Economy using Machine Learning Algorithms*”.

Project Guide



Mr. Anurag Chaturvedi Assistant Professor

University School of Management & Entrepreneurship

Delhi Technological University

(Please note students only took guidance for methodology and research questions. Report and Plag check is not under my purview)

DECLARATION

We, the undersigned, hereby declare that the project report entitled, Analyzing the Interdependency of Financial Markets' Instruments within the Indian Economy using Machine Learning Algorithms submitted by us to the Delhi Technological University, in partial fulfilment of the requirement for the award of degree of Master of Business Administration (M.B.A) with specialization in Business Analytics under the guidance of Mr. Anurag Chaturvedi, is our original work and the conclusions drawn therein are based on the material collected by us. The Report submitted is our own work and has not been duplicated from any other source. We shall be responsible for any unpleasant moment/situation.

Place: New Delhi

Date: 31st May 2021

Dhwani Dhingra

(2K19/BMBA/07)

Saurabh Kumar

(2K19/BMBA/15)

ACKNOWLEDGEMENT

A successful project is the result of teamwork and coordination that includes not only the group of developers who put forth the ideas, logic, and efforts but also those who guide them. So, at the completion of the project, we feel obliged to extend our gratitude towards all those who made valuable contributions throughout our research study.

We would like to express our sincere thanks to Mr. Anurag Chaturvedi, our supervisor who have provided us excellent knowledge and support throughout our research study.

We would like to express our sincere thanks to Dr. Deepti Aggrawal, our course coordinator and all the other staff members who have provided us excellent knowledge and support throughout our Post Graduation.

We are very much thankful to our parents, siblings, friends, and colleagues for their continuous support.

Dhwani Dhingra
(2K19/BMBA/07)

Saurabh Kumar
(2K19/BMBA/15)

TABLE OF CONTENT

CERTIFICATE	1
DECLARATION	2
ACKNOWLEDGEMENT	3
TABLE OF CONTENT	4
ABSTRACT	5
SECTION 1 - INTRODUCTION	6
SECTION 2 - LITERATURE REVIEW	8
SECTION 3: PROPOSED MODEL	10
SECTION 4: METHODOLOGY	12
SECTION 5: RESULT	40
SECTION 6: CONCLUSION	41
REFERENCES	42
APPENDIX	43

ABSTRACT

Over the past decades, researchers around the globe have been trying to understand the volatile behavior of the equity market by analyzing the stock prices, daily percentage returns, traded volumes, bid ask spread of the stock, impact of macroeconomic conditions, impact of the bond market, sentiment analysis of the post and tweets on the various social media platforms. These types of studies have been performed on various stock exchanges like New York Stock Exchange, NASDAQ, London Stock Exchange, and various others. Researchers have found that it is not possible to study one market to derive insights and apply its findings over another market. The primary objective for our study is to analyze the interdependency of the financial market's instruments within the Indian Economy using Machine Learning Algorithms. The instruments of financial markets could be Equity Market, Bond Market, Derivatives Market, Commodity Market while we focus on the Equity Market and a segment of Bond Market called Corporate Bond Market. Machine Learning Algorithms ranges from Regression Analysis, Classification, Clustering, Association Rule Mining, Support Vector Machines, Random Forest to Deep Learning Techniques such as Recurrent Neural Networks, Convolutional Neural Networks etc. In our study, the major focus is on the Application of Regression Analysis and Generalized Addictive Models with Cubic Regression Spline to develop our model. For Performance Evaluation of our Models, we have restricted ourselves to Root Mean Square Error Loss (RMSE) and Mean Absolute Error Loss (MAE) along with the Correlational Analysis to provide insights at the pre-modeling stage.

Keywords: Equity Market, Corporate Bond Market, Regression Analysis, Generalized Addictive Models.

Equity Market is a marketplace through which shares of public-owned companies are issued and traded through exchanges or over-the-counter markets. This market provides the financial support that these companies require to grow their business who are the issuers of the equity. While on the other hand, it also helps the investors in receiving an ownership in the company with the potential to gain maximum return on their investment. According to Researchers around the globe, Indian Economy is a developing economy with immensely volatile equity market. This volatile behavior of the Indian Equity Market could be due to the implication of a single variable or due to the collaboration of multiple variables. Such variables could be among the following - number of securities available, number of trades, volume of trades and total traded value, prices of securities, securities' return on investment, impact of the macroeconomic conditions such as Gross Domestic Product (GDP), Inflation, Interest Rates, Employment Rate, Monetary Policy, Fiscal Policy etc. One could say that Indian Financial Market has risen to a whole new level even during the COVID-19 pandemic, but that does not imply that the volatile aspect is depreciating. It is due to this volatile nature, prediction of equity market to a certain degree is necessary to prepare for the black swan event if it happens.

Corporate Bond Market is a marketplace through which the companies receive capital to run their businesses by issuing corporate bonds to the investors. When an investor buys a corporate bond, the companies provide several interest payments at a fixed or a variable interest rate along with the original investment when the bond is matured. Researchers have been trying to correlate the Corporate Bond Market with the Equity Market in various countries for the past few decades. Major studies have been conducted with respect to countries with developed economies such as United States, United Kingdom, France, Germany, Japan etc. But each study has its limitation that it can never be generalized over another countries' economic environment. There are various reasons for such limitations, some of them could be the impact of macroeconomic variables like Gross Domestic Product (GDP), Consumer Price Index (CPI), Wholesale Price Index (WPI), Interest Rates, Employment Rate, Monetary Policy, Fiscal Policy etc. Researchers believe that the Equity Market may have some correlation or dependency with Bond Market, Derivatives Market, Commodity Market etc.

The primary objective for our study is to analyze the interdependency of the financial market's instruments within the Indian Economy using Machine Learning Algorithms. The instruments of financial markets could be Equity Market, Bond Market, Derivatives Market, Commodity Market while we focus on the Equity Market and a segment of Bond Market called Corporate Bond Market. Machine Learning Algorithms ranges from Regression Analysis, Classification, Clustering, Association Rule Mining, Support Vector Machines, Random Forest to Deep Learning Techniques such as Recurrent Neural Networks, Convolutional Neural Networks etc. In our study, the

major focus is on the Application of Regression Analysis and Generalized Addictive Models with Cubic Regression Spline to develop our model. For the Performance Evaluation of our Models, we have restricted ourselves to use Root Mean Square Error Loss (RMSE) and Mean Absolute Error Loss (MAE) along with the Correlational Analysis to provide insights at the pre-modeling stage.

The structure of the study is as follows –

- Section 2 provides the literature review needed to understand the problem statement and outlines the research gap that the study tries to fulfil.
- Section 3 discusses the proposed model that needs to be developed to carry out the study.
- Section 4 explores the developed model itself.
- Section 5 discusses the result of the study.
- Section 6 concludes the study with possibly outlining the future work.
- Section 7 cites the references used in the study.
- Section 8 provides the model developed in R.

SECTION 2 - LITERATURE REVIEW

This section reviews and explores the various research work that had been conducted in the relevant field. The research works are briefly discussed and provides an insight to the wide range of models that could have been utilized for conducting the study.

Yakup Kara, Melek Acar Boyacioglu, Omer Kaan Baykan (2011) developed and compared the performance of the two efficient classification models in predicting the direction of the movement in the daily price of the Istanbul Stock Exchange National 100 Index. The two models that they developed were based on the Artificial Neural Networks (ANN) and Support Vector Machines (SVM). They concluded that the training and the holdout performance of the ANN Model developed varied in the range of 84.53% to 99.64% and 71.17% to 82.85% respectively. The training and the holdout performance of the polynomial SVM Model varied in the range of 72.63% to 100% and 72.26% to 82.48% respectively. They also stated that both ANN and SVM are useful in predicting stock prices, where the average prediction performance of the ANN Model was 75.74%, which was found significantly better than that of SVM Model which was 71.52%.

Eunsuk Chong, Chulwoo Han, Frank C. Park (2017) suggests that the deep learning networks can be effectively used for stock market analysis and prediction. They used the Deep Neural Networks (DNN), Auto-Regressive (AR) and Artificial Neural Network (ANN) for analysing the frequency intraday stock returns and examined the effects of three unsupervised feature extraction methods such as Principal Component Analysis (PCA), Autoencoder (AE), and the Restricted Boltzmann Machine (RBM) on the network's ability to predict the future market behaviour. They used the performance metrics such as Normalised Mean Squared Error (NMSE), Root Mean Squared Error (RMSE), Mean Absolute Error (MAE) and the Mutual Information (MI) to validate their findings. They reported that the ANN Model performed worst in the RMSE and MI metrics and best in MAE metrics.

Hiransha et. Al. (2018) analysed the day-wise closing prices of MARUTI, HCL and AXIS BANK from National Stock Exchange (NSE) and BANK OF AMERICA and CHESAPEAKE ENERGY from New York Stock Exchange (NYSE) by developing various models of deep learning architecture such as Multilayer Perceptron (MLP), Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) and Convolutional Neural Network (CNN) and compared the models with Auto Regressive Integrated Moving Average (ARIMA) and validating their results by Mean Absolute Percentage Error (MAPE) metric. They concluded that the models could identify patterns existing in NSE as well as NYSE and were outperforming ARIMA model. They also concluded that CNN was the best performer among the deep learning models.

Robert Johnson and Philip Young (2002) examined the stock and bond market volatility in the United Kingdom (U.K). They suggested that their findings could have implications for the portfolio asset allocations decisions for the global investors. They also indicated that assuming the trends observed in the U.S would also be present in the other developed markets would be dangerous and may have serious implications in optimising the investment's return.

This section explores the proposed model for the study. This section will provide information about the hypothesis formulated for the study, variables considered, and tools used for conducting the research work.

Hypothesis Formulation:

Before developing any model, hypothesis formation is necessary which would form the base for the research work. Therefore, the hypothesis formulated for the study is as follows -

Null Hypothesis	There is no significant evidence that the Corporate Bond Market have impact over the Equity Market.
Alternative Hypothesis	There is significant evidence that the Corporate Bond Market have impact over the Equity Market.

Table 3.1: Hypothesis Formulation

Variables:

After the hypothesis is formulated, the variables considered for conducting the research study are listed -

Cb_securities	The no. of securities available in the Corporate Bonds Market for a particular trading day.
CB_trades	The number of trades taking place in the Corporate Bonds Market for a particular trading day.
CB_trd_val_cr	The total amount of trades traded (in crores) in the Corporate Bonds Market for a particular trading day.
E_securities	The no. of securities available in the Equity Market for a particular trading day.
E_trades	The number of trades taking place in the Equity Market for a particular trading day.
E_trd_qty_lac	The volume of trades traded (in lakhs) in the Equity Market for a particular trading day.
E_Trld_val_cr	he total amount of trades traded (in crores) in the Equity Market for a particular trading day.

Table 3.2: Variables Considered for the Study

Tools Used:

R Programming Language:

R is a programming language and environment for statistical computing and graphics supported by the R Foundation for Statistical Computing. R is considered among the best tools available in the market for conducting data science experiments due to its robust data handling capabilities and versatility with programming. It is widely used by statisticians and data miners for developing statistical software and data analysis.

Blueprint of the Model:

The hypotheses defined in the above section is broad and vague. This broad and vagueness feature of the Hypothesis could be made more understandable by specifying the blueprint for the study. To study the impact of Corporate Bond Market, we first need to understand the delay in the time-period required to study that impact. As a researcher, it would be irrelevant to study the impact of Corporate Bond Market over the Equity Market on the very same day because both these markets will provide their results in the form of the variables under study at the end of the trading time for a particular day. Therefore, the lagging day approach is required to study the impact. The highly researched lagging days are 1-day, 3-days, 5-days, 7-days, 14-days, 21-days, 28-days, 50-days, 100-days and so on and so forth. For this study, we have used 1-day, 3-days, 5-days, 7-days, 14-days, 21-days, 28-days lagging approach. We have applied the lagging day approach to both Corporate Bonds Market and Equity Market to provide insights over the present day Equity Market.

The variable selected for representing the Corporate Bonds Market and the Equity Market is the total amount of trades (in crores) for each market. The reason for selection of this variable is that even though there is constant number of securities available, and/or constant number of trades taking place, the value of trades for a particular day will vary because the denomination of the securities traded would vary and this variation would reflect the volatility in the Equity Market as well as the Corporate Bonds Market.

This section explores the developed model. The steps involved for developing the model are as follows - Data Acquisition phase, Data Cleaning and Pre-processing phase, Model Building phase and Model Performance Phase.

Data Acquisition Phase:

The data was collected from the official website of the National Stock Exchange. The time-period for the study is from 1st April 2014 to 28th May 2021. The total no. of trading days within this timeframe is 1766 days.

```
> data[1:5,]
# A tibble: 5 x 8
  Date                Cb_securities CB_trades CB_trd_val_cr E_securities E_trades E_trd_qty_lac E_Trd_val_cr
  <dtm>              <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>    <dbl>
1 2021-05-28 00:00:00      155      305      7764.    2052 22869269    38562.    74125.
2 2021-05-27 00:00:00      172      453      7526.    2040 26181965    46469.    120763.
3 2021-05-26 00:00:00      NA       NA       NA       2056 22452204    35475.    69055.
4 2021-05-25 00:00:00      164      322      7447.    2057 23681217    35652.    73500.
5 2021-05-24 00:00:00      149      259      3899.    2093 24072214    38223.    78112.
```

Figure 4.1: The top 5 rows of the dataset in R.

Data Cleaning and Pre-processing Phase:

Before applying any model to the data collected, it is advisable to check for any missing values, or irrelevant data present within the dataset. The data collected for the study had some (43) missing values. These missing values may be due to the difference in the trading dates for Equity Market as well as Corporate Bonds Market. It would be inappropriate to manipulate these missing values. Thus, it is highly recommended that these missing values are removed before applying any operation. The reason for such action is the volatile nature of these financial markets. If manipulated, it may cause overfitting or underfitting the model.

After cleaning the dataset, it is advisable to pre-process the dataset to derive some basic insights. The pre-processing phase outlines the graphical representation of the variables, estimating the correlational analysis of the variables along with the Scatter Plot of the variables deriving some relational insights at an initial stage.

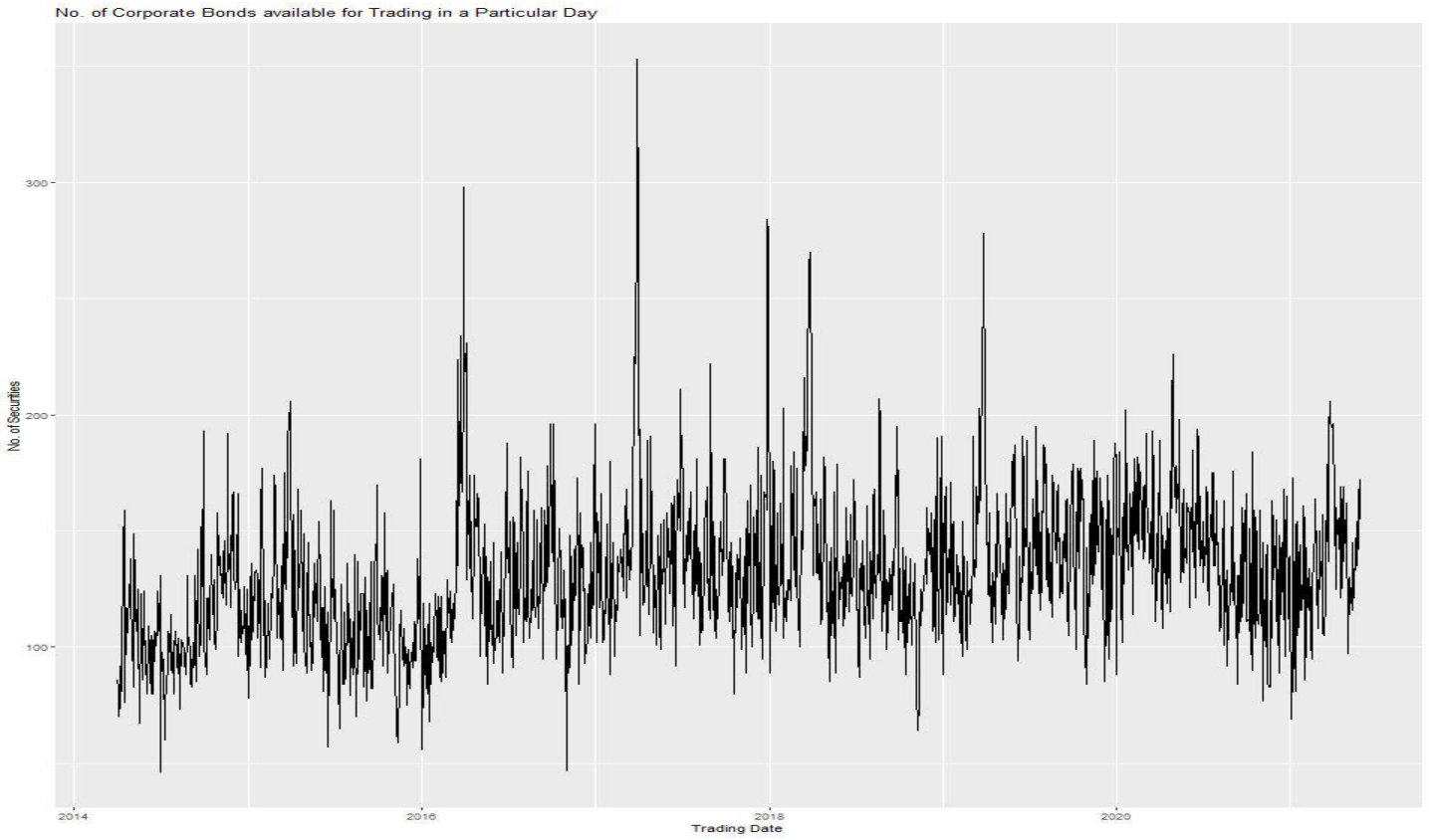


Figure 4.2: Number of Corporate Bonds securities available for Trading in a particular day.

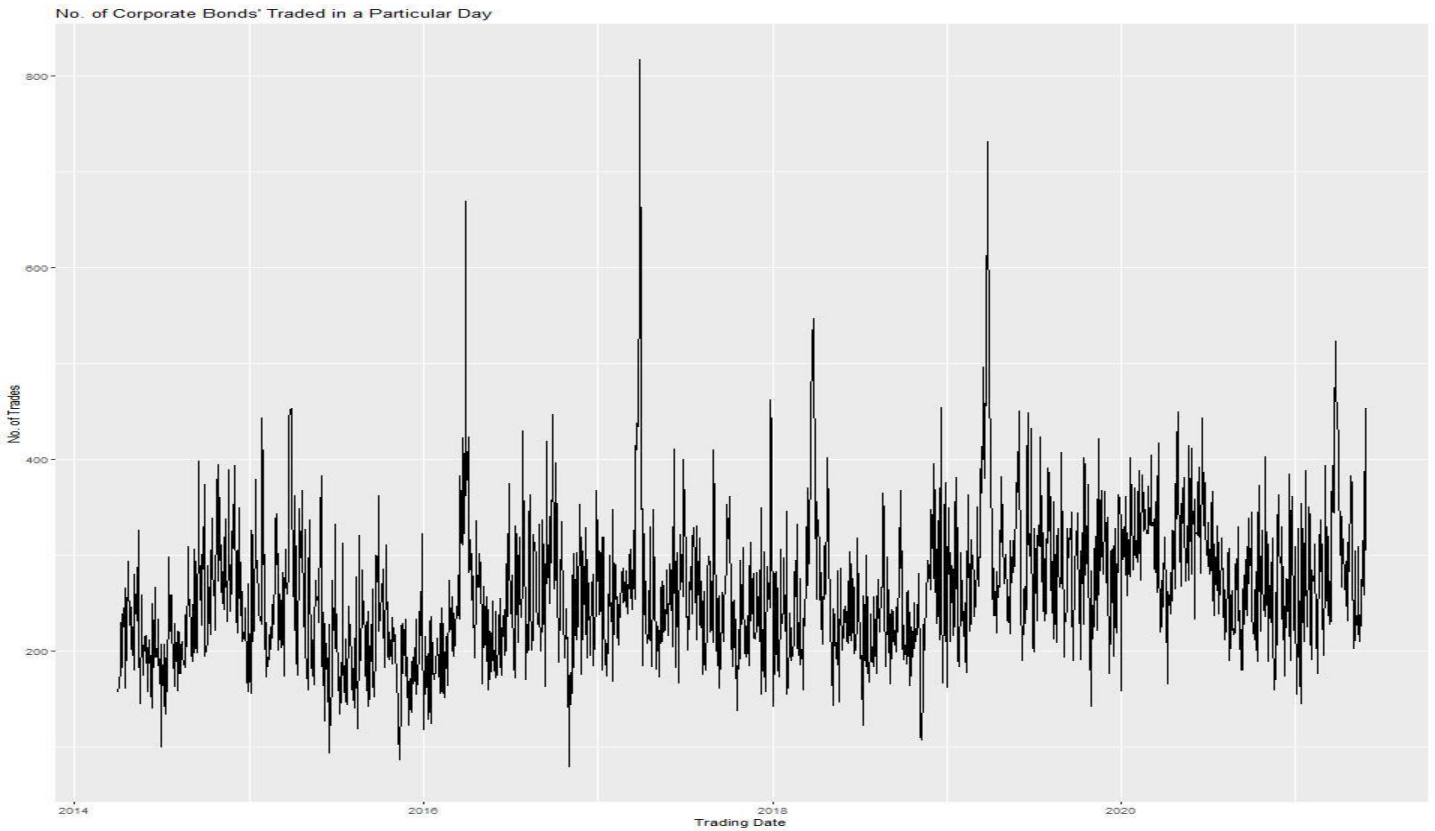


Figure 4.3: Number of Corporate Bonds traded in a particular day.

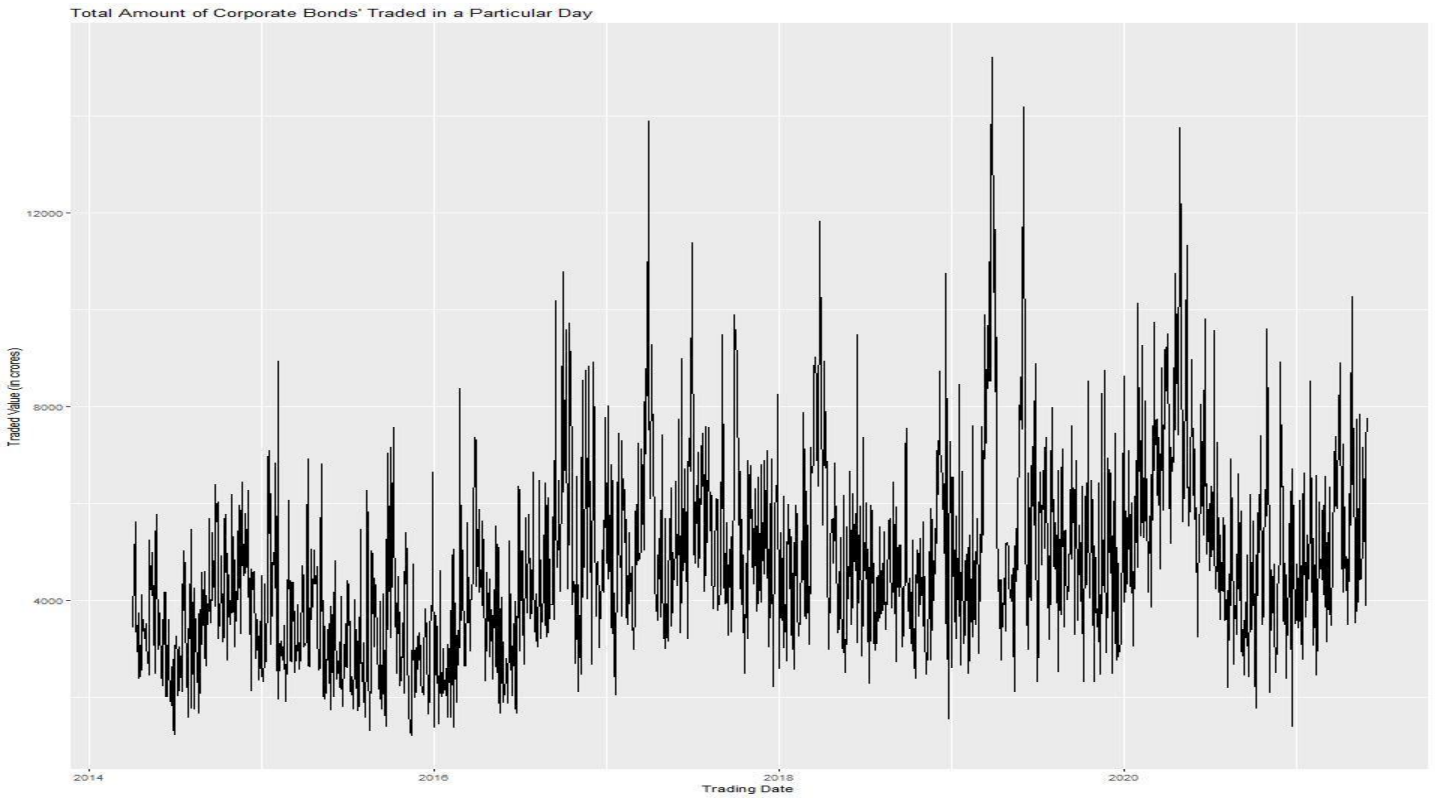


Figure 4.4: Total Amount of Corporate Bonds traded in a particular day.

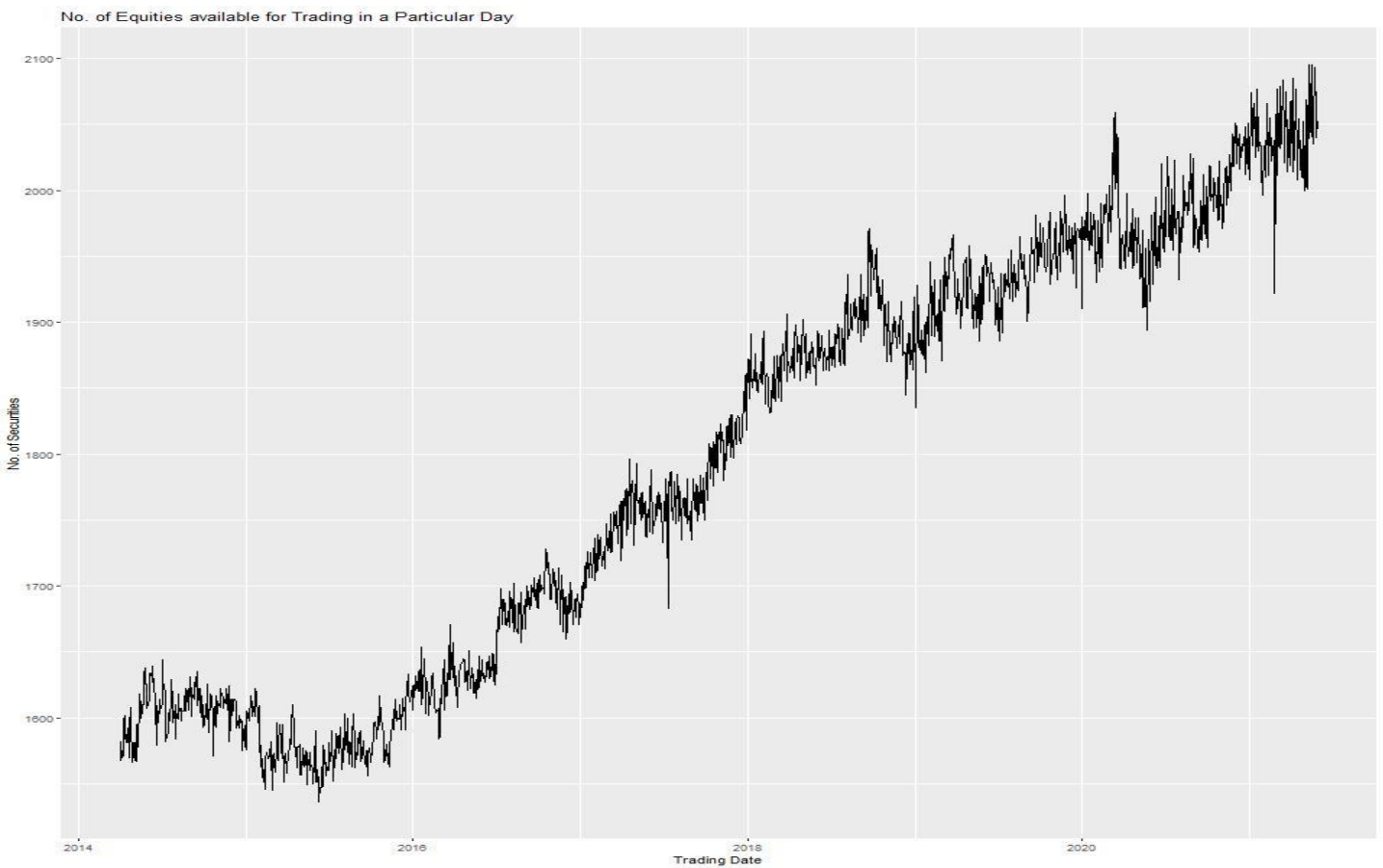


Figure 4.5: Number of Equities available for trading in a particular day.

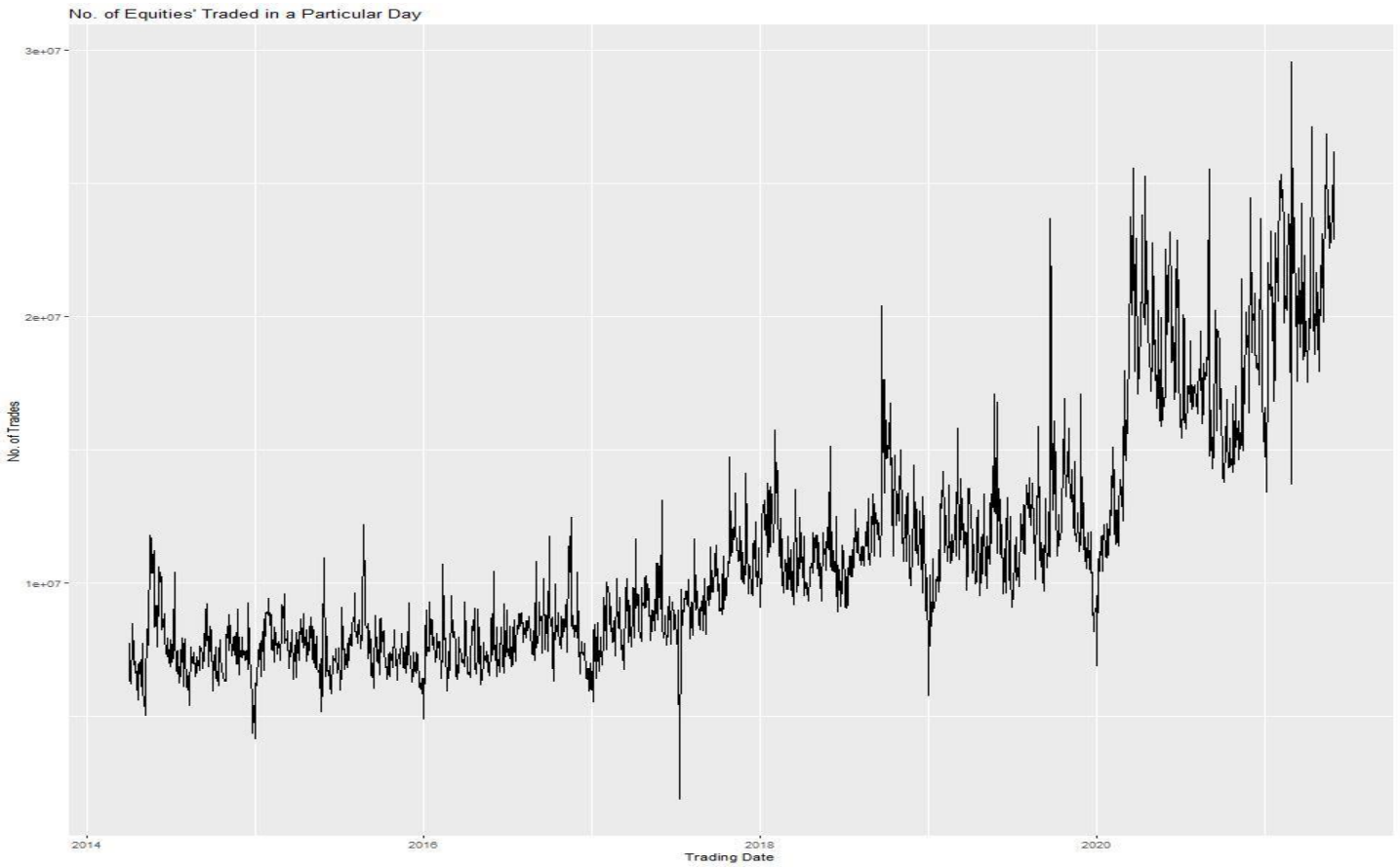


Figure 4.6: Number of Equities traded in a particular day.

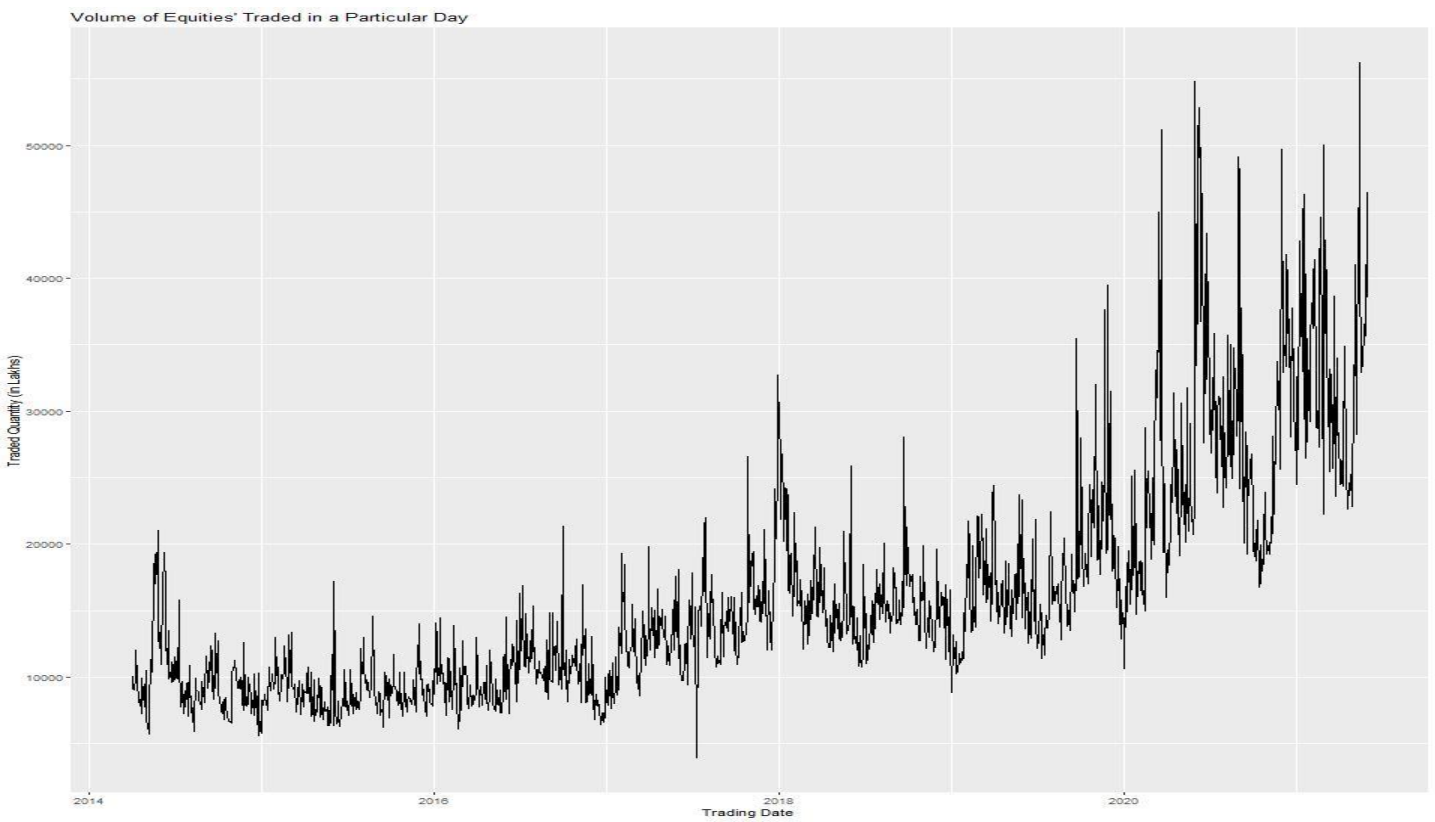


Figure 4.7: Volume of Equities traded in a particular day.

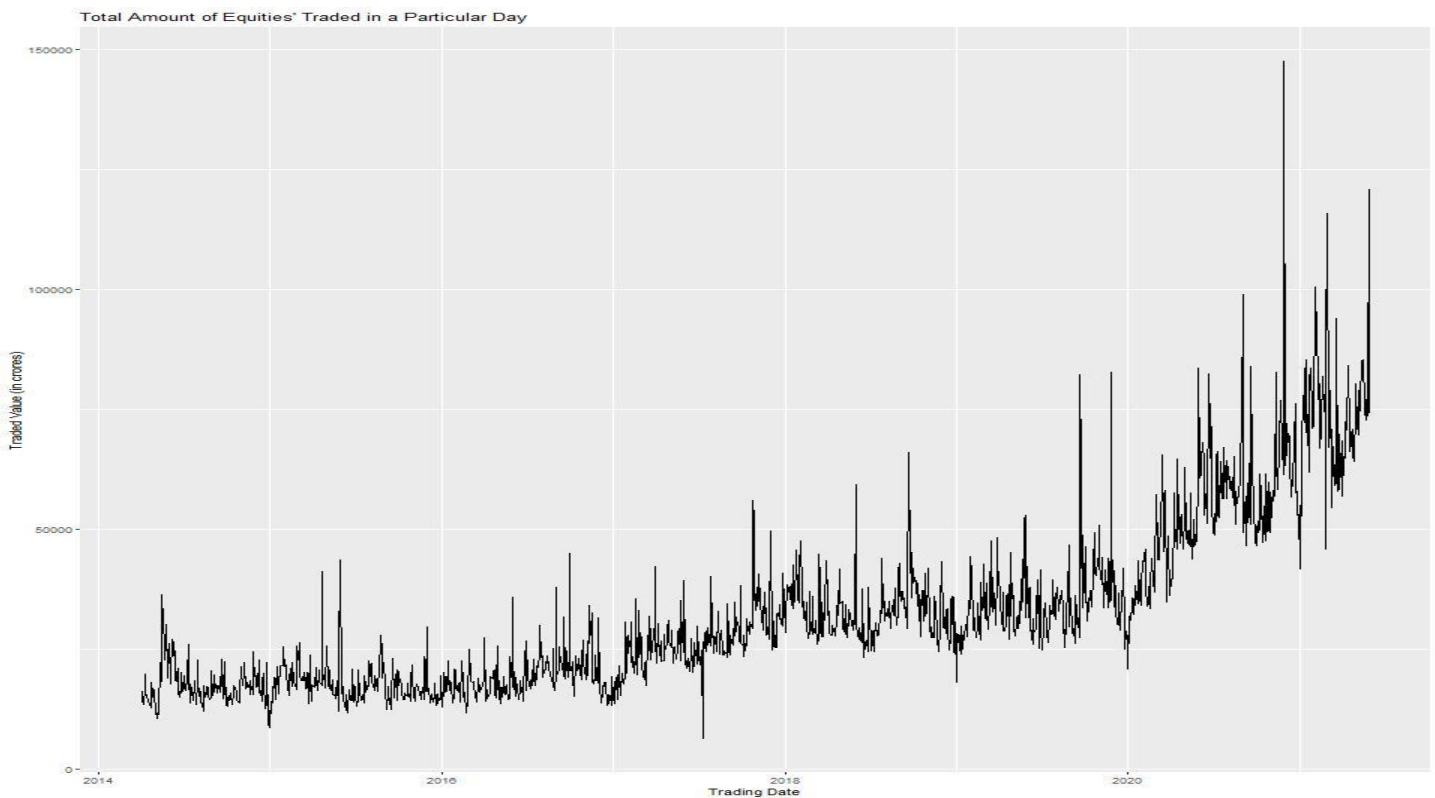


Figure 4.8: Total Amount of Equities traded in a particular day.

The graphs (Figure 4.2, 4.3 and 4.4) depicts that the volatile nature of the Corporate Bonds Markets. It also reflects that the corporate bonds market has been rising at a slow rate rather than its counterpart – Equity Market. The graphs (Figure 4.5, 4.6, 4.7 and 4.8) depicts the facts that the equity market has been rising even during the COVID-19 pandemic, but the concern of volatility is also clearly visible through the graphs.

	Cb_securities	CB_trades	CB_trd_val_cr	E_securities	E_trades	E_trd_qty_lac	E_TrD_val_cr
Cb_securities	1.000	0.868	0.683	0.320	0.224	0.238	0.237
CB_trades	0.868	1.000	0.747	0.299	0.251	0.263	0.268
CB_trd_val_cr	0.683	0.747	1.000	0.372	0.298	0.276	0.283
E_securities	0.320	0.299	0.372	1.000	0.820	0.775	0.812
E_trades	0.224	0.251	0.298	0.820	1.000	0.922	0.962
E_trd_qty_lac	0.238	0.263	0.276	0.775	0.922	1.000	0.927
E_TrD_val_cr	0.237	0.268	0.283	0.812	0.962	0.927	1.000

Figure 4.9: Correlational Analysis between the variables.

The correlation matrix shows that the variables have linear correlation among each other but there is low positive correlation between the variables representing the Corporate Bonds Market and the variables representing the Equity Market. This could be visualized more effectively through scatter plot.

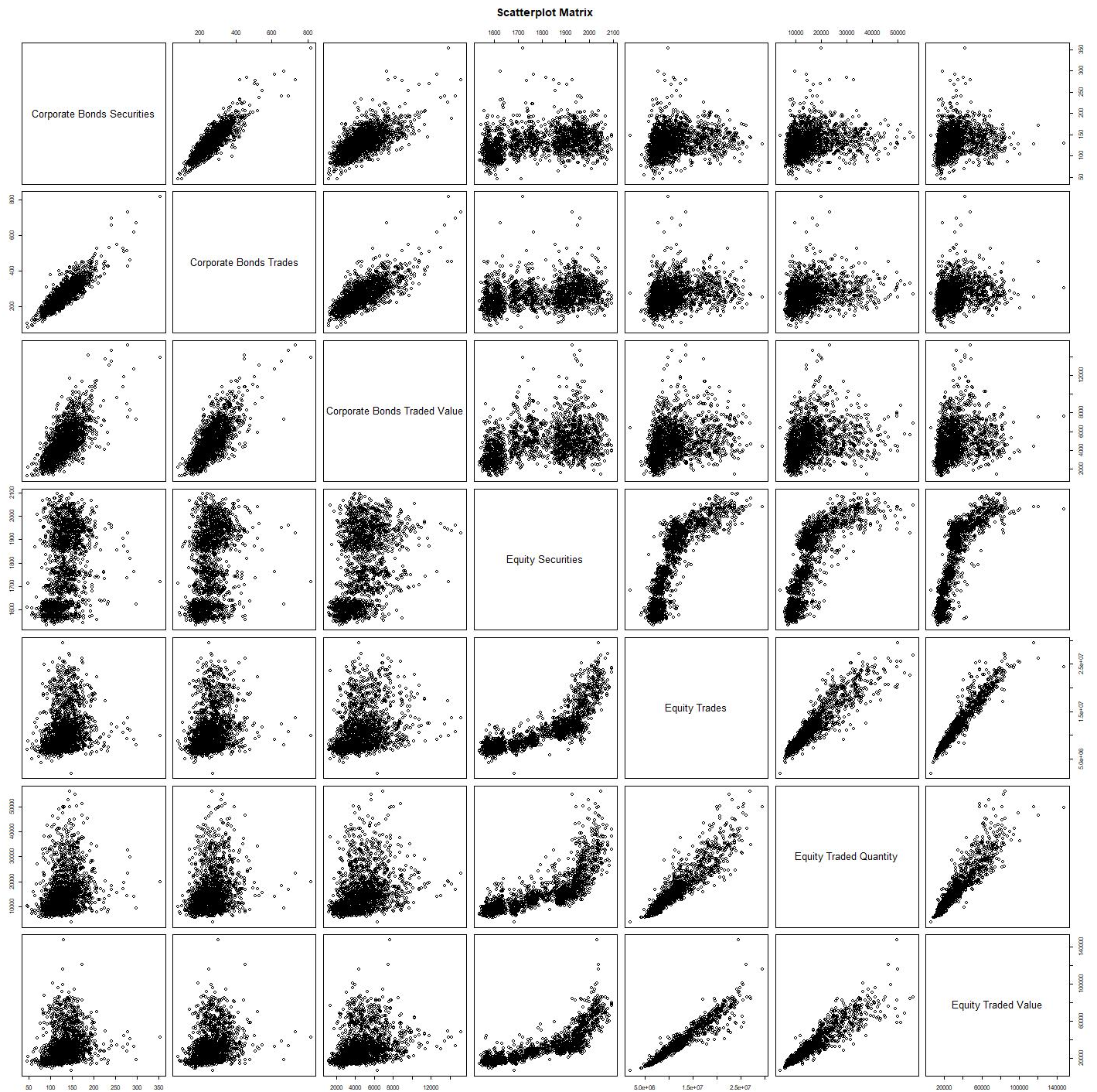


Figure 4.10: Pairwise Scatterplot of the Variables.

Pairwise Scatterplot is used to represent the scatterplot of variables among each other. The above figure reflects that there might lie significant correlation among the variables along with the outliers present in the dataset. These outliers might impact the relation between these markets but neglecting these outliers could never help the researchers account for extreme conditions such as Black Swan Events.

Model Building Phase:

This section explores the model developed using Linear Regression Analysis and Generalized Additive Models with Cubic Regression Spline for 1-day, 3-days, 5-days, 7-days, 14-days, 21-days, and 28-days lagging approach. This phase is sub-divided into 7 events depicting each lagging approach.

1-day Lagging Approach –

The variables used under this approach are 1-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Trđ_val_cr) and 1-day Lagged Equity Market Variable (E_Trđ_val_cr_lag).

	CB_trd_val_cr	E_Trđ_val_cr	E_Trđ_val_cr_lag
CB_trd_val_cr	1.000	0.275	0.282
E_Trđ_val_cr	0.275	1.000	0.910
E_Trđ_val_cr_lag	0.282	0.910	1.000

Figure 4.11: Correlational Analysis between the variables.

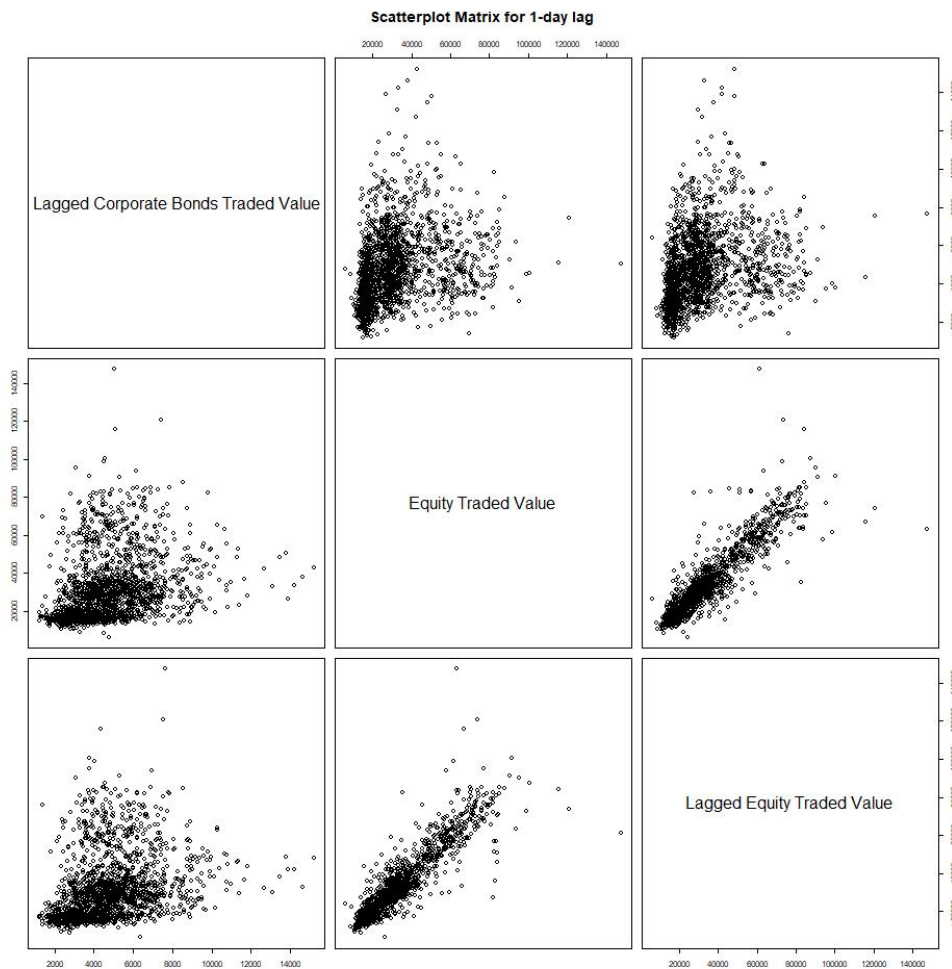


Figure 4.12: Pairwise Scatterplot of the variables.

```
> print(summary(lin_reg_lag1))
```

Call:

```
lm(formula = lag_1day__data1$E_Tr_d_val_cr ~ lag_1day__data1$CB_tr_d_val_cr +  
lag_1day__data1$E_Tr_d_val_cr_lag)
```

Residuals:

```
   Min       1Q   Median       3Q      Max  
-73887 -2850   -461    2239   88916
```

Coefficients:

```
                Estimate Std. Error t value Pr(>|t|)  
(Intercept)      2.149e+03  5.137e+02   4.183 3.02e-05 ***  
lag_1day__data1$CB_tr_d_val_cr  1.880e-01  9.482e-02   1.983  0.0476 *  
lag_1day__data1$E_Tr_d_val_cr_lag 9.051e-01  1.045e-02  86.606 < 2e-16 ***  
---
```

Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7188 on 1719 degrees of freedom

Multiple R-squared: 0.8277, Adjusted R-squared: 0.8275

F-statistic: 4128 on 2 and 1719 DF, p-value: < 2.2e-16

Figure 4.13: Summary of Linear Regression Model.

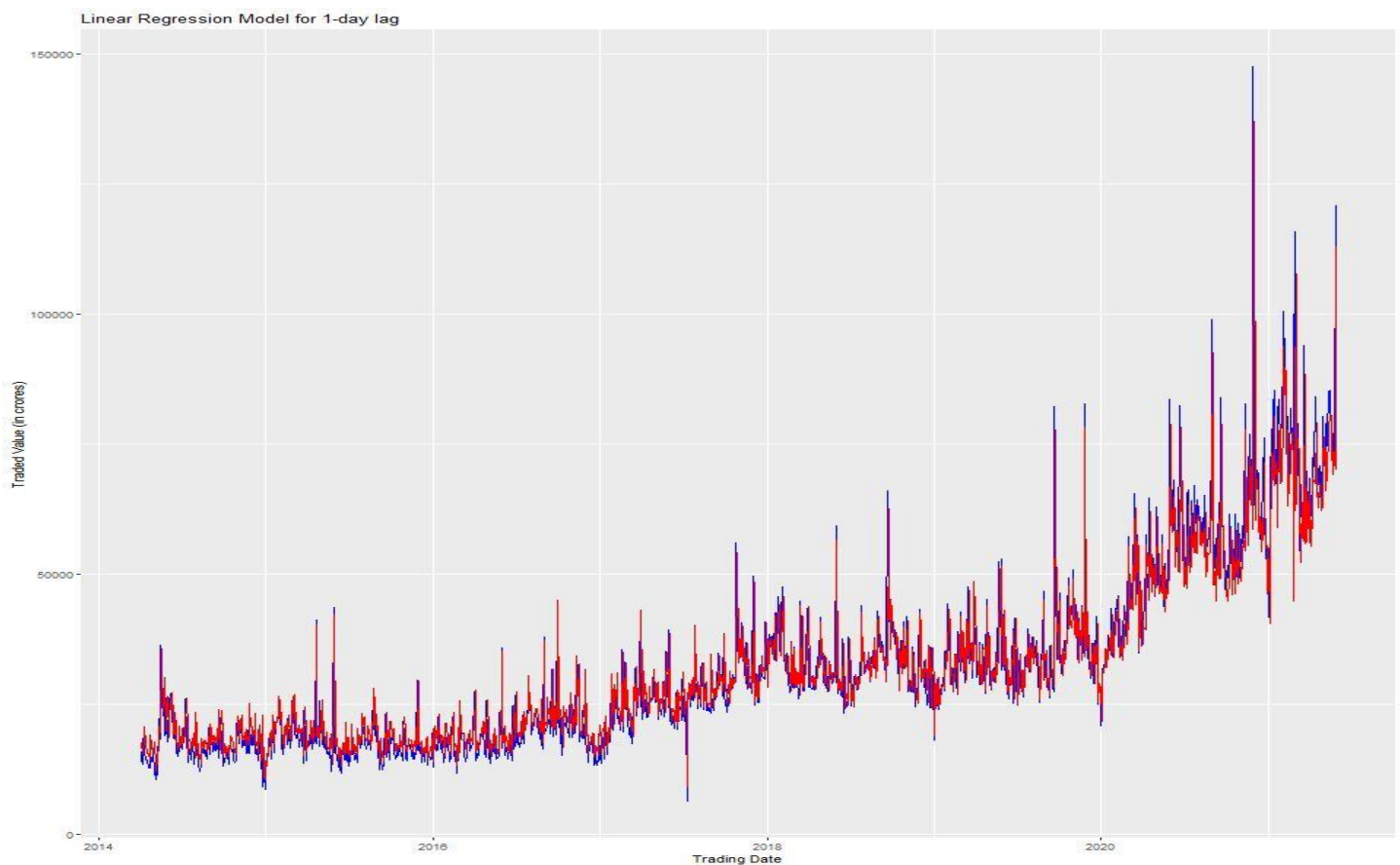


Figure 4.14: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.

```

> print(summary(gam_cr_lag1))

Family: gaussian
Link function: identity

Formula:
lag_1day__data1$E_Trđ_val_cr ~ s(lag_1day__data1$CB_trđ_val_cr,
  bs = "cr") + s(lag_1day__data1$E_Trđ_val_cr_lag, bs = "cr")

Parametric coefficients:
      Estimate Std. Error t value Pr(>|t|)
(Intercept) 31860.4    161.4   197.4  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_1day__data1$CB_trđ_val_cr)  1.626  2.041   2.621  0.0718 .
s(lag_1day__data1$E_Trđ_val_cr_lag) 5.826  6.677 1317.114 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.85   Deviance explained = 85.1%
GCV = 4.5072e+07  scale est. = 4.485e+07  n = 1722

```

Figure 4.15: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

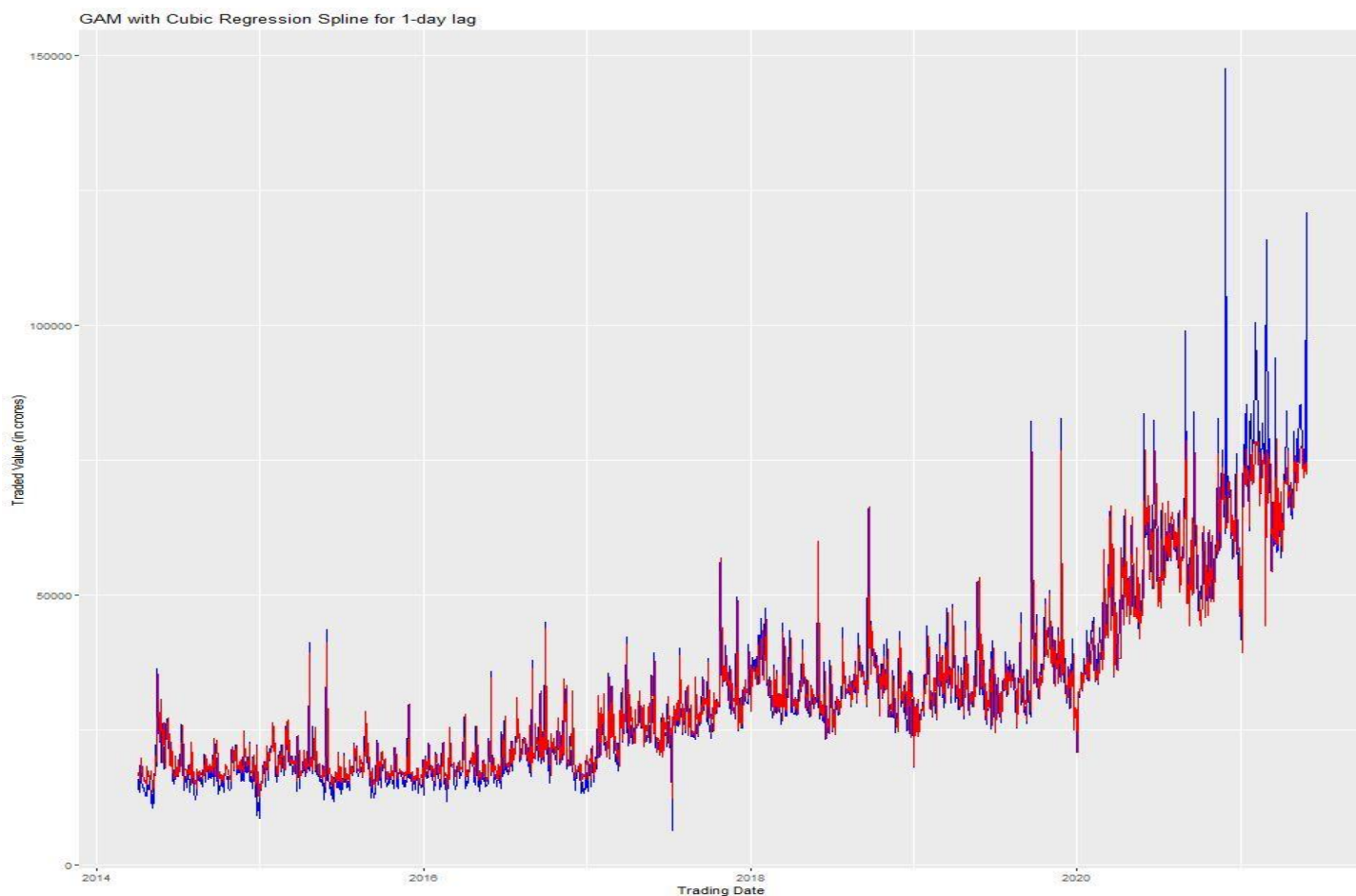


Figure 4.16: Plotting the Predicted Values obtained from the GAM over the Original Value.

3-day Lagging Approach –

The variables used under this approach are 3-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Trđ_val_cr) and 3-day Lagged Equity Market Variable (E_Trđ_val_cr_lag).

	CB_trd_val_cr	E_Trđ_val_cr	E_Trđ_val_cr_lag
CB_trd_val_cr	1.000	0.253	0.278
E_Trđ_val_cr	0.253	1.000	0.893
E_Trđ_val_cr_lag	0.278	0.893	1.000

Figure 4.17: Correlational Analysis between the variables.

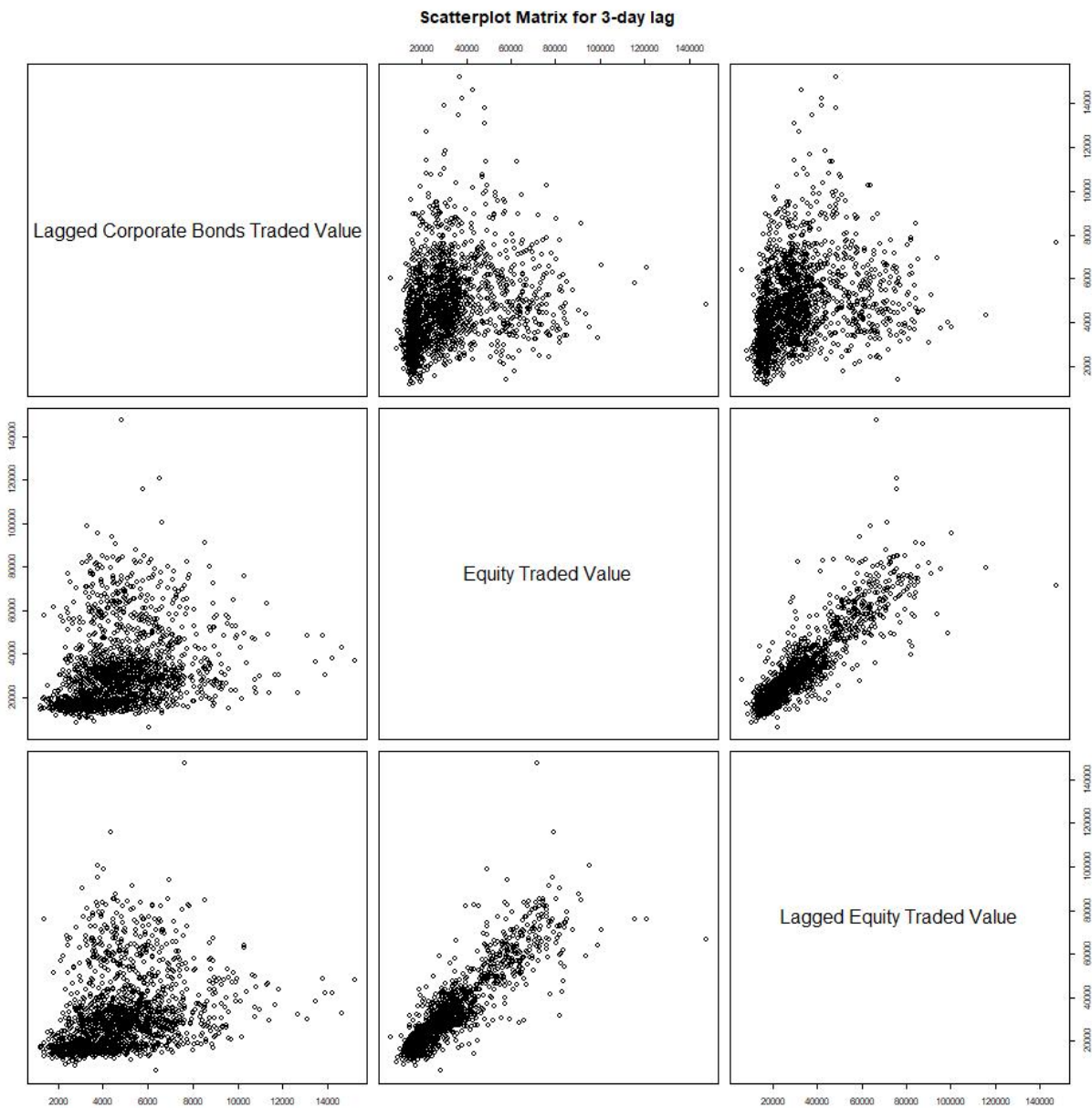


Figure 4.18: Pairwise Scatterplot of the variables.


```

> print(summary(lin_reg_lag3))

Call:
lm(formula = lag_3day__data1$E_Trđ_val_cr ~ lag_3day__data1$CB_trđ_val_cr +
    lag_3day__data1$E_Trđ_val_cr_lag)

Residuals:
    Min       1Q   Median       3Q      Max
-64692  -3615   -743    2469   84355

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      3.073e+03  5.586e+02   5.502 4.32e-08 ***
lag_3day__data1$CB_trđ_val_cr  4.033e-02  1.028e-01   0.392   0.695
lag_3day__data1$E_Trđ_val_cr_lag  9.012e-01  1.142e-02  78.876 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7792 on 1717 degrees of freedom
Multiple R-squared:  0.7975,    Adjusted R-squared:  0.7973
F-statistic: 3381 on 2 and 1717 DF,  p-value: < 2.2e-16

```

Figure 4.19: Summary of Linear Regression Model.

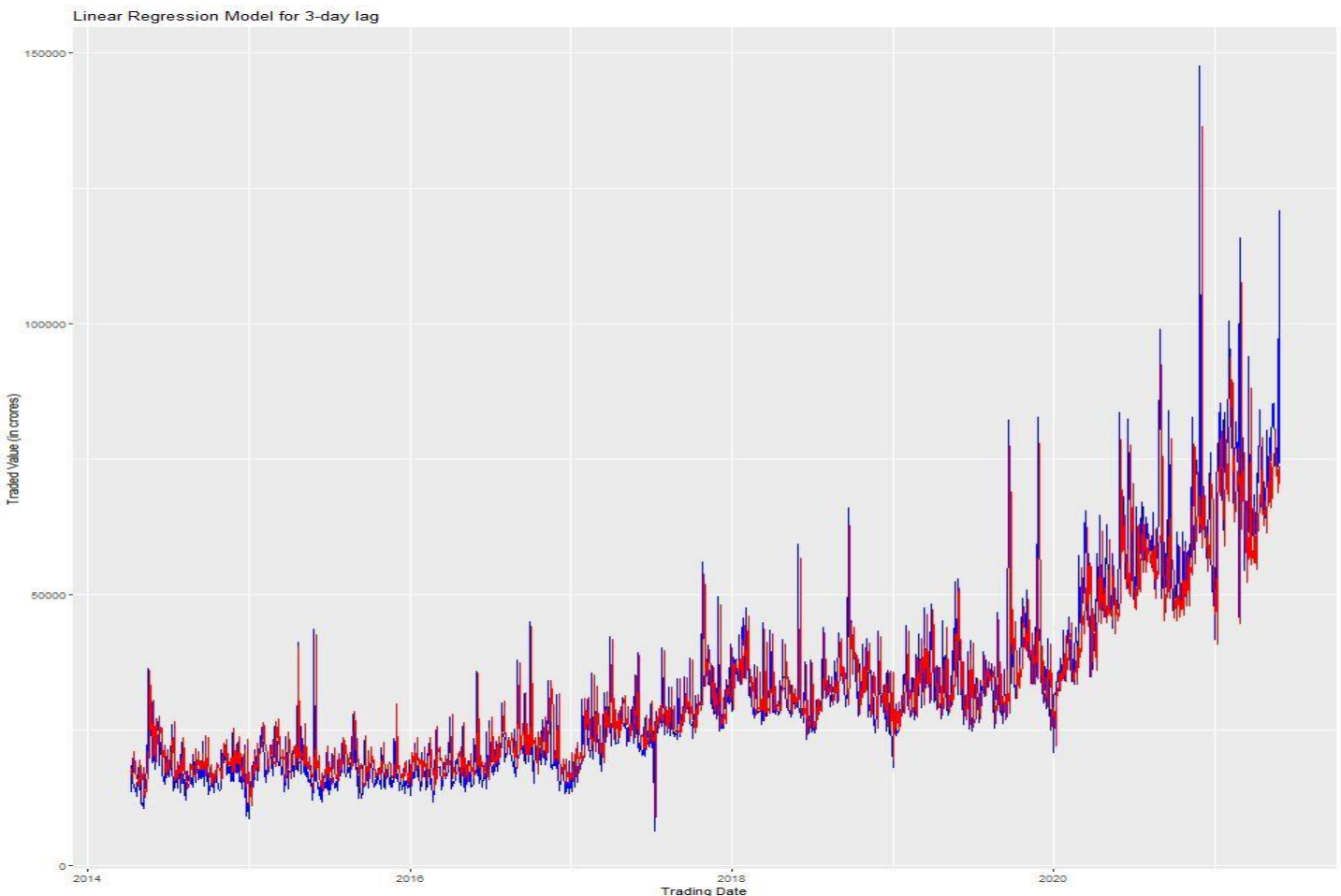


Figure 4.20: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.

```

> print(summary(gam_cr_lag3))

Family: gaussian
Link function: identity

Formula:
lag_3day__data1$E_Trd_val_cr ~ s(lag_3day__data1$CB_trd_val_cr,
  bs = "cr") + s(lag_3day__data1$E_Trd_val_cr_lag, bs = "cr")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)   31880         177   180.2  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_3day__data1$CB_trd_val_cr)  2.154  2.718  1.333  0.355
s(lag_3day__data1$E_Trd_val_cr_lag) 7.466  8.258 858.198 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) =  0.82  Deviance explained = 82.1%
GCV = 5.4197e+07  Scale est. = 5.3862e+07  n = 1720

```

Figure 4.21: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

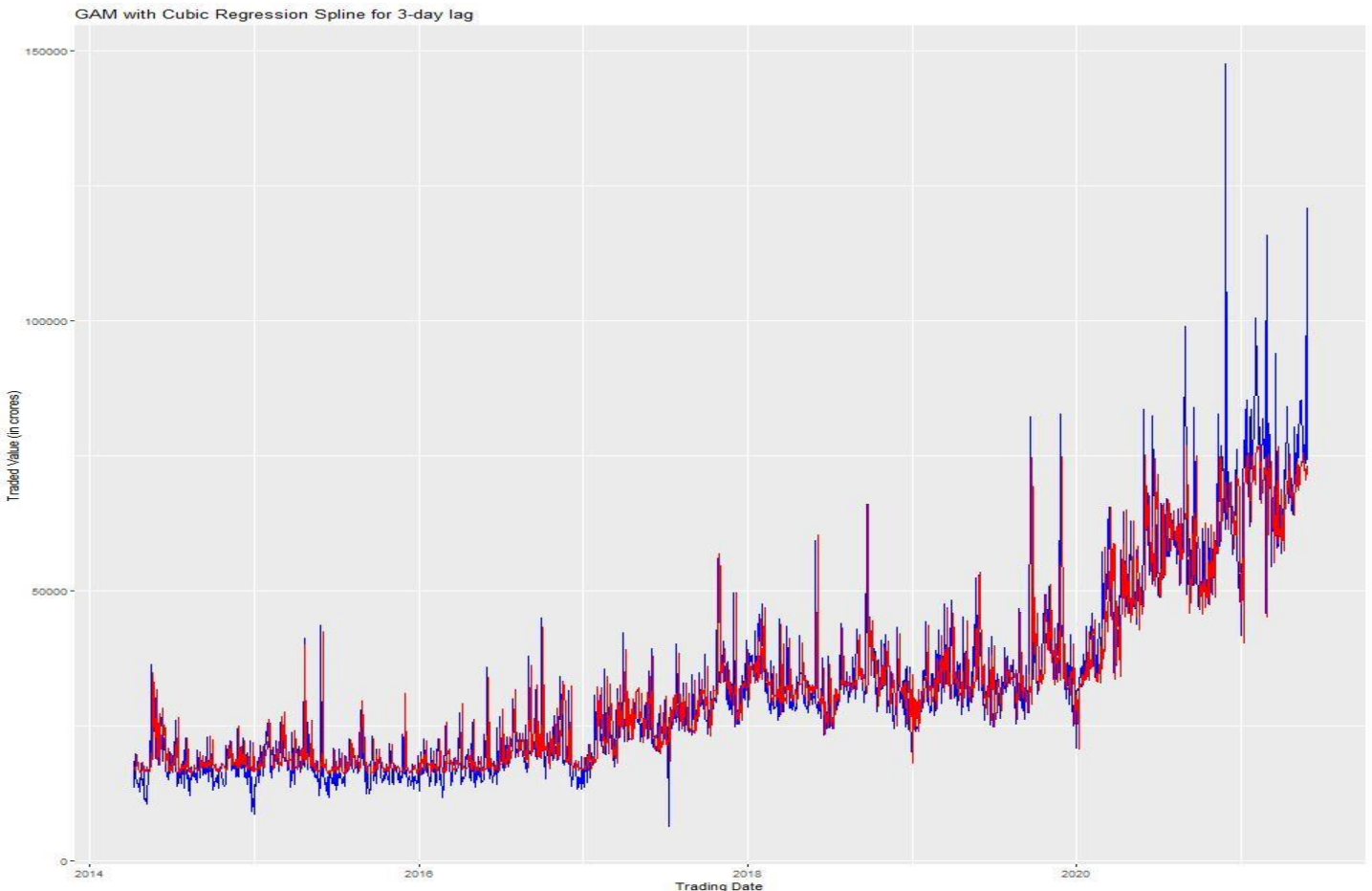


Figure 4.22: Plotting the Predicted Values obtained from the GAM over the Original Value.

5-day Lagging Approach –

The variables used under this approach are 5-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Tr_d_val_cr) and 5-day Lagged Equity Market Variable (E_Tr_d_val_cr_lag).

	CB_trd_val_cr	E_Tr_d_val_cr	E_Tr_d_val_cr_lag
CB_trd_val_cr	1.000	0.254	0.279
E_Tr_d_val_cr	0.254	1.000	0.881
E_Tr_d_val_cr_lag	0.279	0.881	1.000

Figure 4.23: Correlational Analysis between the variables.

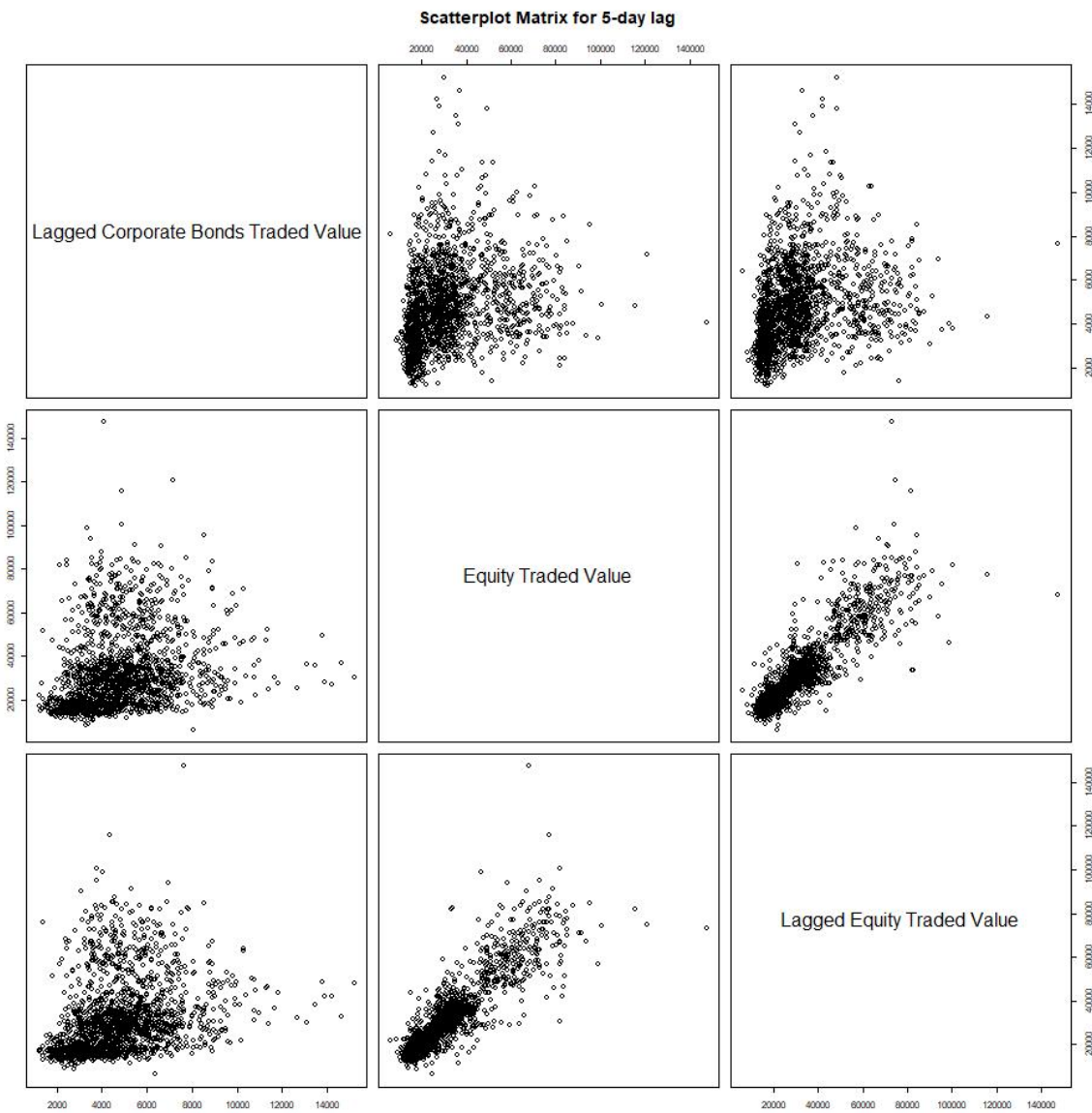


Figure 4.24: Pairwise Scatterplot of the variables.

```
> print(summary(lin_reg_lag5))
```

```
Call:
```

```
lm(formula = lag_5day__data1$E_Trđ_val_cr ~ lag_5day__data1$CB_trđ_val_cr +  
    lag_5day__data1$E_Trđ_val_cr_lag)
```

```
Residuals:
```

```
    Min      1Q  Median      3Q     Max  
-67157 -3628  -769    2957  78742
```

```
Coefficients:
```

```
                Estimate Std. Error t value Pr(>|t|)  
(Intercept)      3.263e+03  5.874e+02   5.555 3.21e-08 ***  
lag_5day__data1$CB_trđ_val_cr  8.194e-02  1.081e-01   0.758   0.448  
lag_5day__data1$E_Trđ_val_cr_lag 8.910e-01  1.206e-02  73.893 < 2e-16 ***  
---
```

```
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
Residual standard error: 8188 on 1715 degrees of freedom
```

```
Multiple R-squared:  0.7764,    Adjusted R-squared:  0.7762
```

```
F-statistic: 2978 on 2 and 1715 DF,  p-value: < 2.2e-16
```

Figure 4.25: Summary of Linear Regression Model.

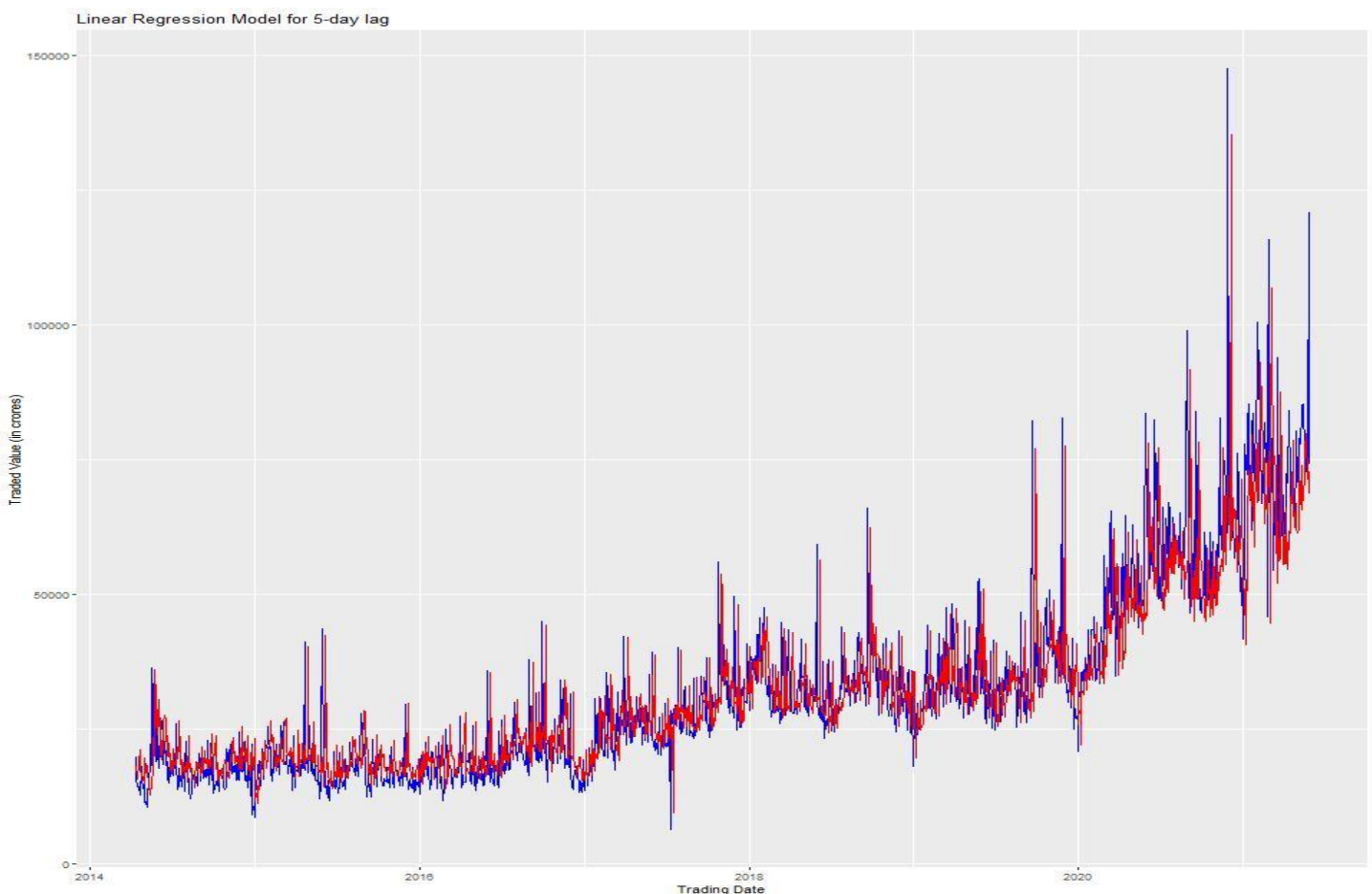


Figure 4.26: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.

```

> print(summary(gam_cr_lag5))

Family: gaussian
Link function: identity

Formula:
lag_5day__data1$E_Trd_val_cr ~ s(lag_5day__data1$CB_trd_val_cr,
  bs = "cr") + s(lag_5day__data1$E_Trd_val_cr_lag, bs = "cr")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  31899.3     186.1    171.4  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_5day__data1$CB_trd_val_cr)  2.234  2.815  1.891  0.106
s(lag_5day__data1$E_Trd_val_cr_lag) 7.854  8.550 727.921 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.801  Deviance explained = 80.2%
GCV = 5.9916e+07  scale est. = 5.9529e+07  n = 1718

```

Figure 4.27: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

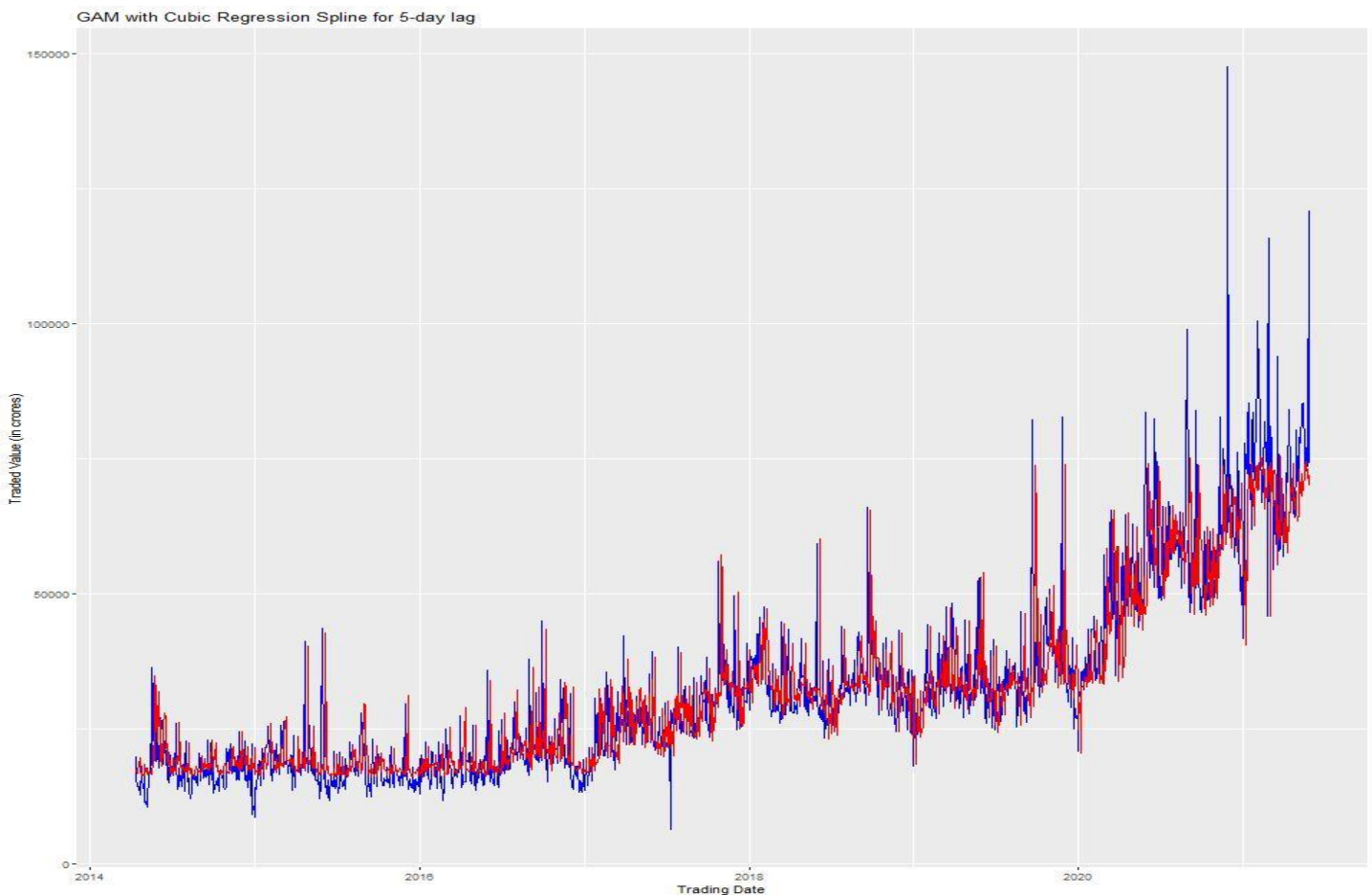


Figure 4.28: Plotting the Predicted Values obtained from the GAM over the Original Value.

7-day Lagging Approach –

The variables used under this approach are 7-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Trđ_val_cr) and 7-day Lagged Equity Market Variable (E_Trđ_val_cr_lag).

	CB_trd_val_cr	E_Trđ_val_cr	E_Trđ_val_cr_lag
CB_trd_val_cr	1.000	0.241	0.278
E_Trđ_val_cr	0.241	1.000	0.867
E_Trđ_val_cr_lag	0.278	0.867	1.000

Figure 4.29: Correlational Analysis between the variables.

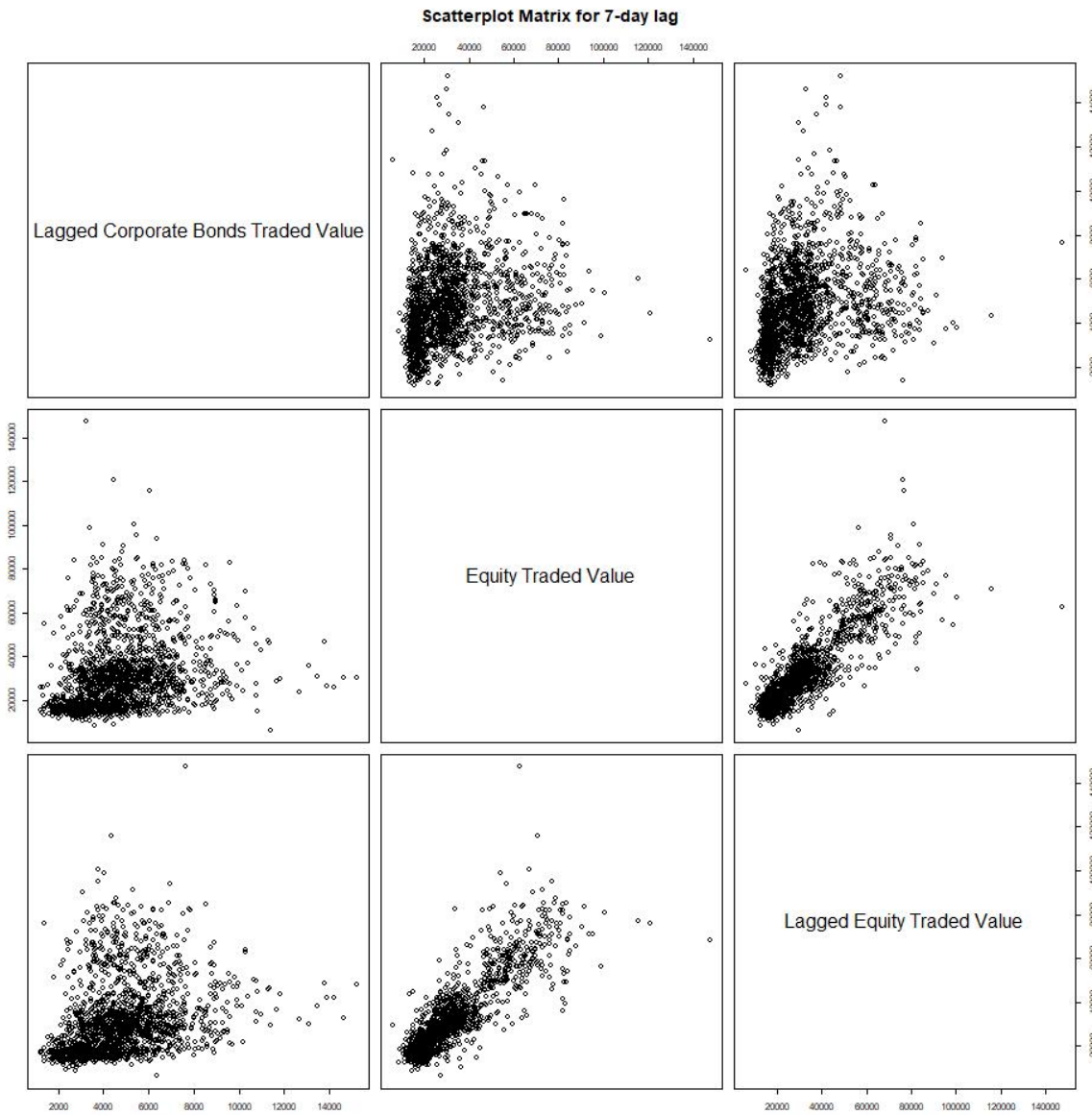


Figure 4.30: Pairwise Scatterplot of the variables.

```

> print(summary(lin_reg_lag7))

Call:
lm(formula = lag_7day__data1$E_Trđ_val_cr ~ lag_7day__data1$CB_trđ_val_cr +
    lag_7day__data1$E_Trđ_val_cr_lag)

Residuals:
    Min       1Q   Median       3Q      Max
-71475  -3957   -761    3335   83066

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      3.972e+03  6.190e+02   6.418 1.79e-10 ***
lag_7day__data1$CB_trđ_val_cr  4.402e-03  1.138e-01   0.039  0.969
lag_7day__data1$E_Trđ_val_cr_lag 8.822e-01  1.273e-02  69.292 < 2e-16 ***
---
signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8619 on 1713 degrees of freedom
Multiple R-squared:  0.7523,    Adjusted R-squared:  0.7521
F-statistic: 2602 on 2 and 1713 DF,  p-value: < 2.2e-16

```

Figure 4.31: Summary of Linear Regression Model.

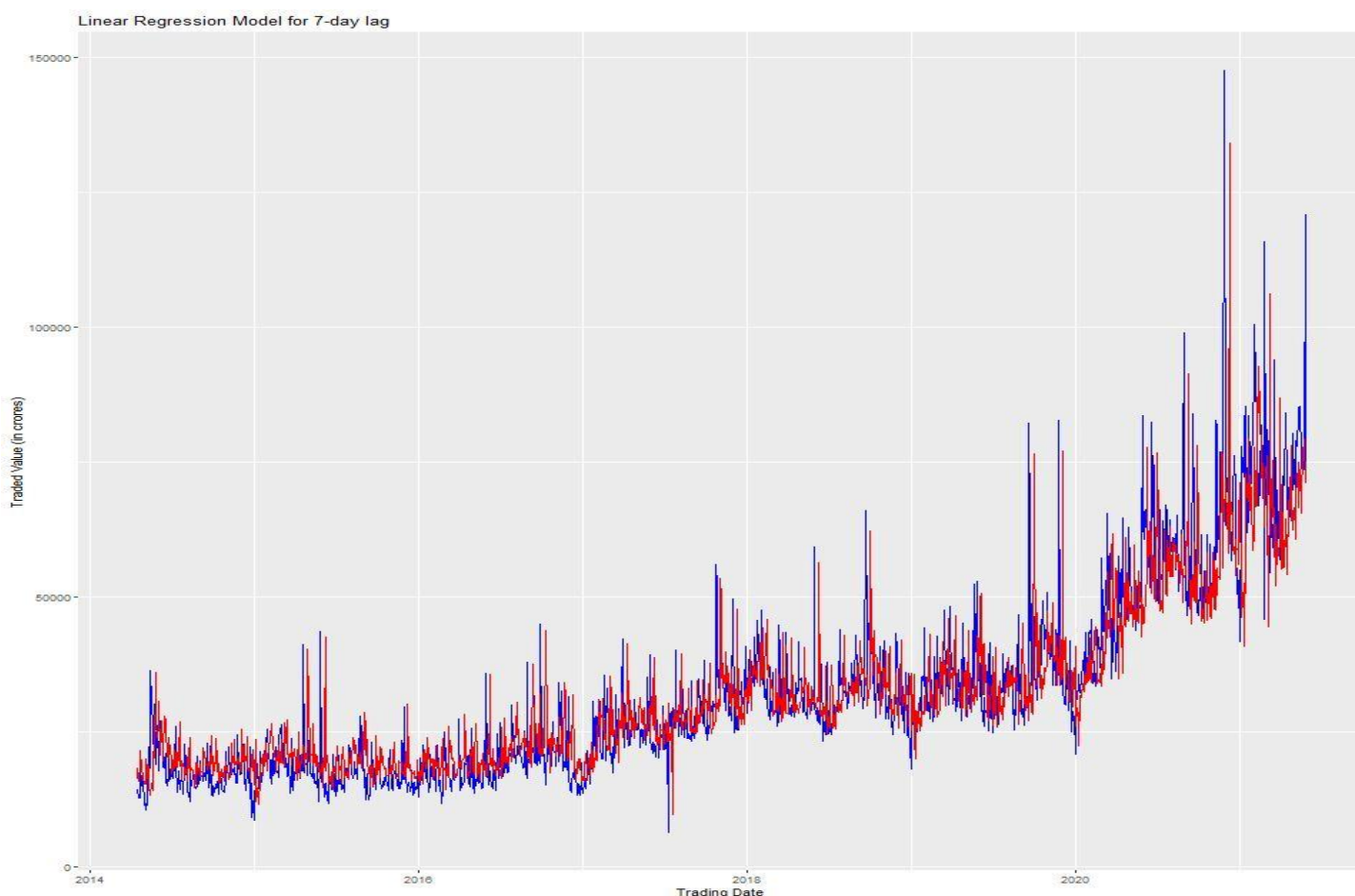


Figure 4.32: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.


```

> print(summary(gam_cr_lag7))

Family: gaussian
Link function: identity

Formula:
lag_7day__data1$E_Trld_val_cr ~ s(lag_7day__data1$CB_trld_val_cr,
  bs = "cr") + s(lag_7day__data1$E_Trld_val_cr_lag, bs = "cr")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  31915.6      195.5    163.2  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_7day__data1$CB_trld_val_cr)  2.473  3.094   3.526  0.0132 *
s(lag_7day__data1$E_Trld_val_cr_lag) 7.093  7.940 693.506 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.781  Deviance explained = 78.2%
GCV = 6.6016e+07  scale est. = 6.561e+07  n = 1716

```

Figure 4.33: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

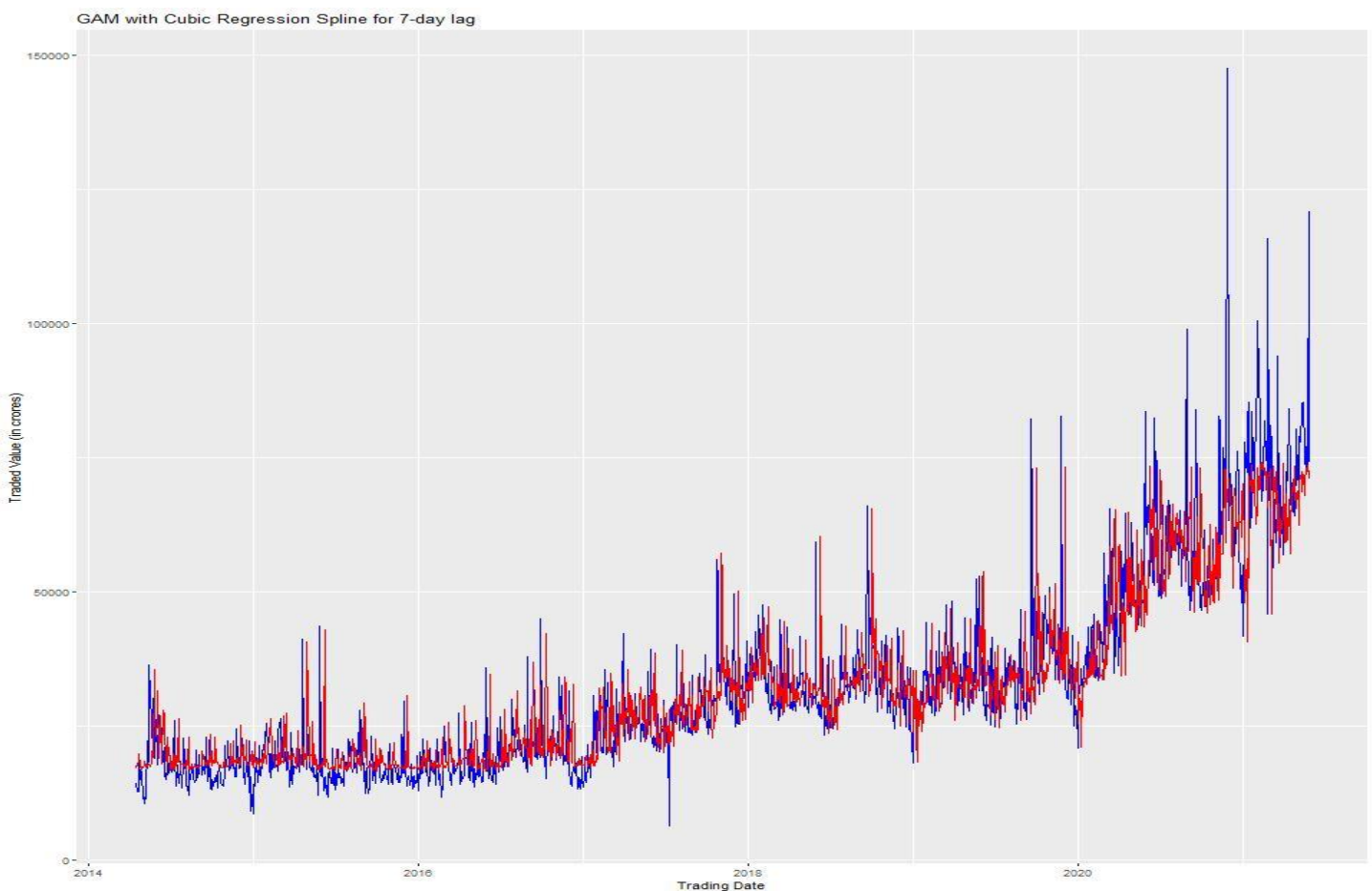


Figure 4.34: Plotting the Predicted Values obtained from the GAM over the Original Value.

14-day Lagging Approach –

The variables used under this approach are 14-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Trđ_val_cr) and 14-day Lagged Equity Market Variable (E_Trđ_val_cr_lag).

	CB_trd_val_cr	E_Trđ_val_cr	E_Trđ_val_cr_lag
CB_trd_val_cr	1.000	0.255	0.278
E_Trđ_val_cr	0.255	1.000	0.868
E_Trđ_val_cr_lag	0.278	0.868	1.000

Figure 4.35: Correlational Analysis between the variables.

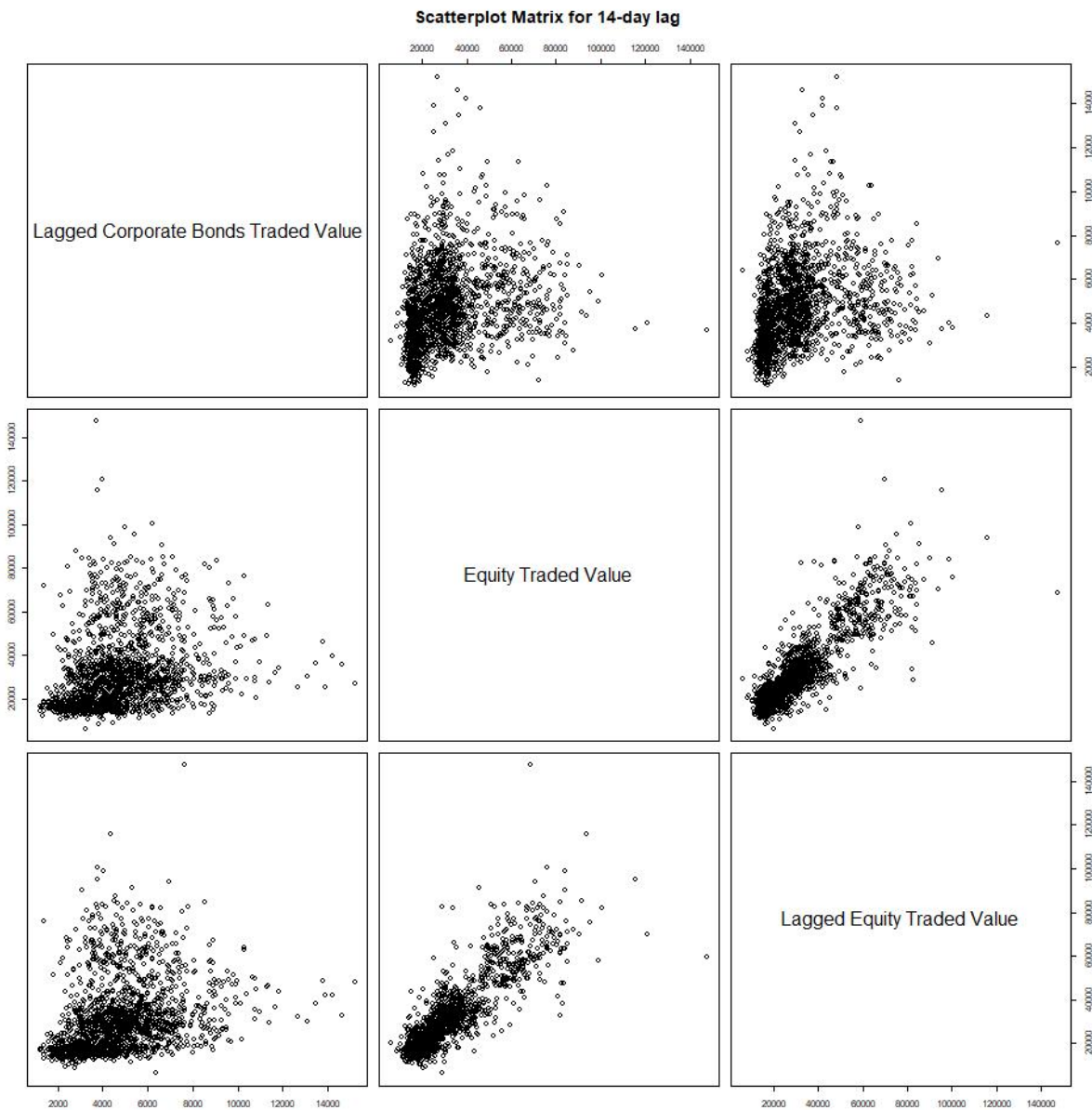


Figure 4.36: Pairwise Scatterplot of the variables.

```

> print(summary(lin_reg_lag14))

Call:
lm(formula = lag_14day__data1$E_Trđ_val_cr ~ lag_14day__data1$CB_trđ_val_cr +
    lag_14day__data1$E_Trđ_val_cr_lag)

Residuals:
    Min       1Q   Median       3Q      Max
-67050  -4424  -1036   3368   90778

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      3.278e+03  6.205e+02   5.283 1.44e-07 ***
lag_14day__data1$CB_trđ_val_cr  1.379e-01  1.138e-01   1.212   0.226
lag_14day__data1$E_Trđ_val_cr_lag  8.918e-01  1.294e-02  68.896 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 8610 on 1706 degrees of freedom
Multiple R-squared:  0.7528,    Adjusted R-squared:  0.7525
F-statistic: 2598 on 2 and 1706 DF,  p-value: < 2.2e-16

```

Figure 4.37: Summary of Linear Regression Model.

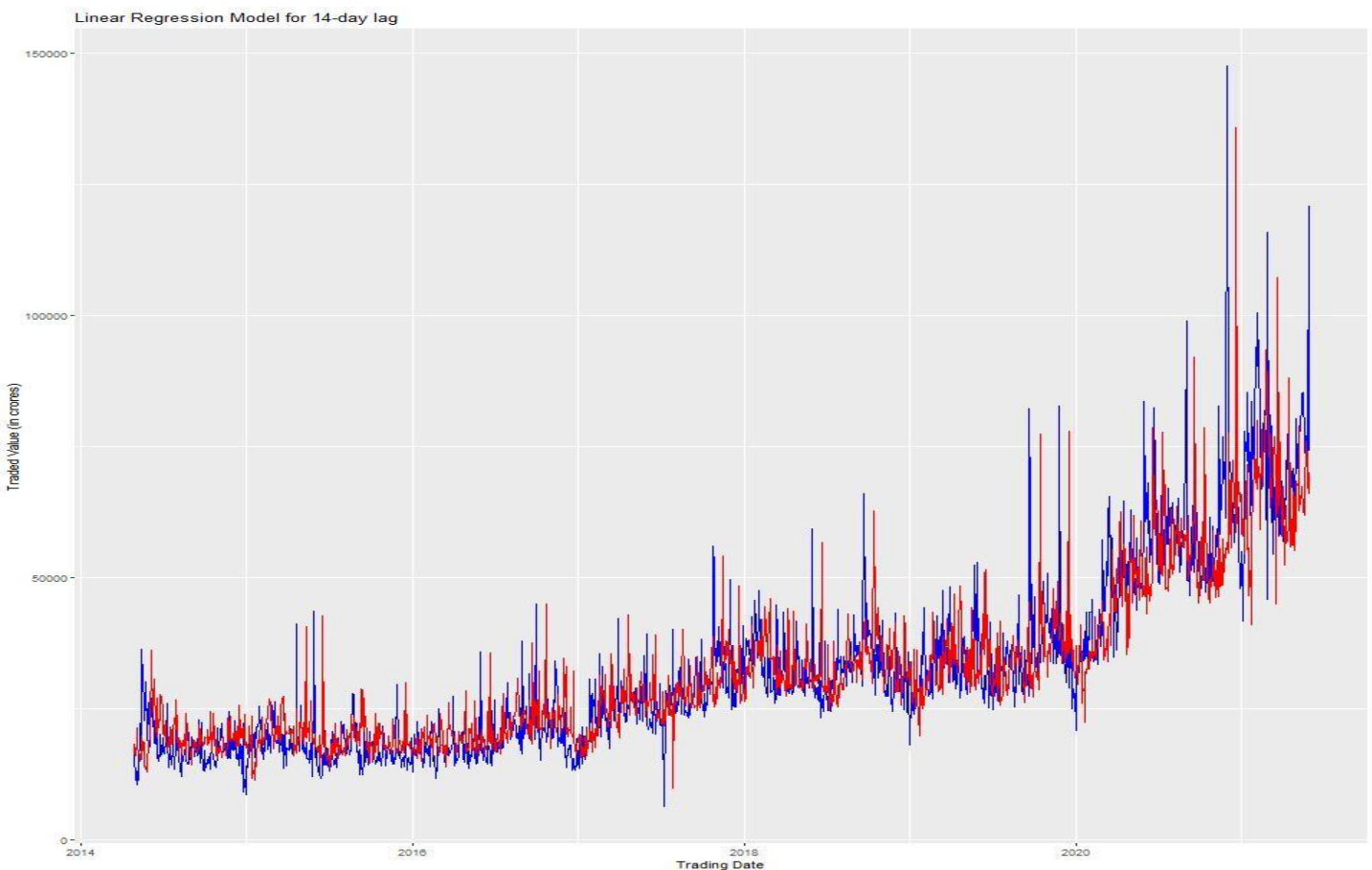


Figure 4.38: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.


```

> print(summary(gam_cr_lag14))

Family: gaussian
Link function: identity

Formula:
lag_14day__data1$E_Tr_d_val_cr ~ s(lag_14day__data1$CB_tr_d_val_cr,
  bs = "cr") + s(lag_14day__data1$E_Tr_d_val_cr_lag, bs = "cr")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  31986.5      198.1    161.5  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_14day__data1$CB_tr_d_val_cr)  2.231  2.810  1.785  0.14
s(lag_14day__data1$E_Tr_d_val_cr_lag) 6.942  7.803 680.346 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.776  Deviance explained = 77.7%
GCV = 6.7482e+07  scale est. = 6.708e+07  n = 1709

```

Figure 4.39: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

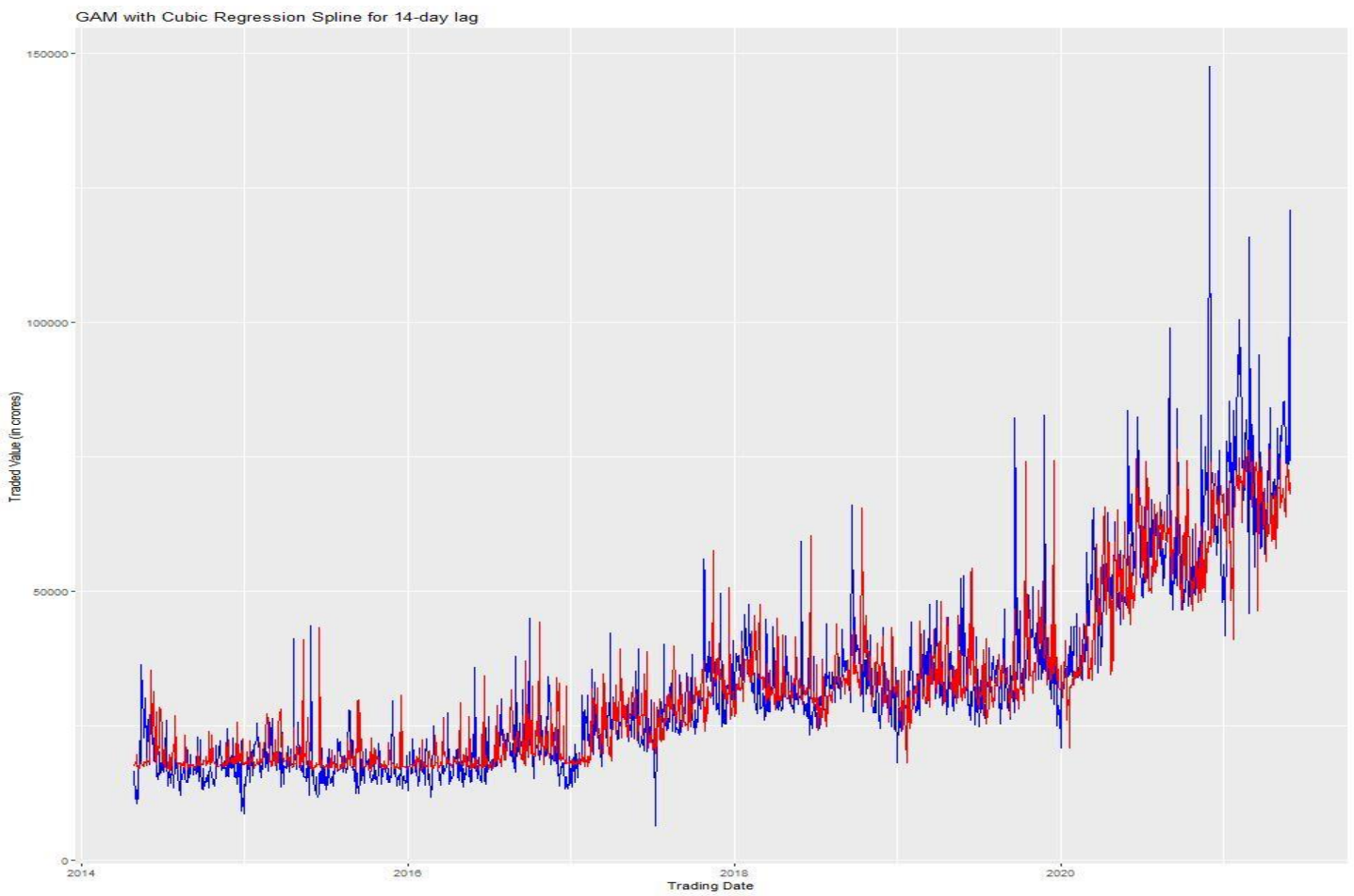


Figure 4.40: Plotting the Predicted Values obtained from the GAM over the Original Value.

21-day Lagging Approach –

The variables used under this approach are 21-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Tr_d_val_cr) and 21-day Lagged Equity Market Variable (E_Tr_d_val_cr_lag).

	CB_trd_val_cr	E_Tr_d_val_cr	E_Tr_d_val_cr_lag
CB_trd_val_cr	1.000	0.261	0.278
E_Tr_d_val_cr	0.261	1.000	0.838
E_Tr_d_val_cr_lag	0.278	0.838	1.000

Figure 4.41: Correlational Analysis between the variables.

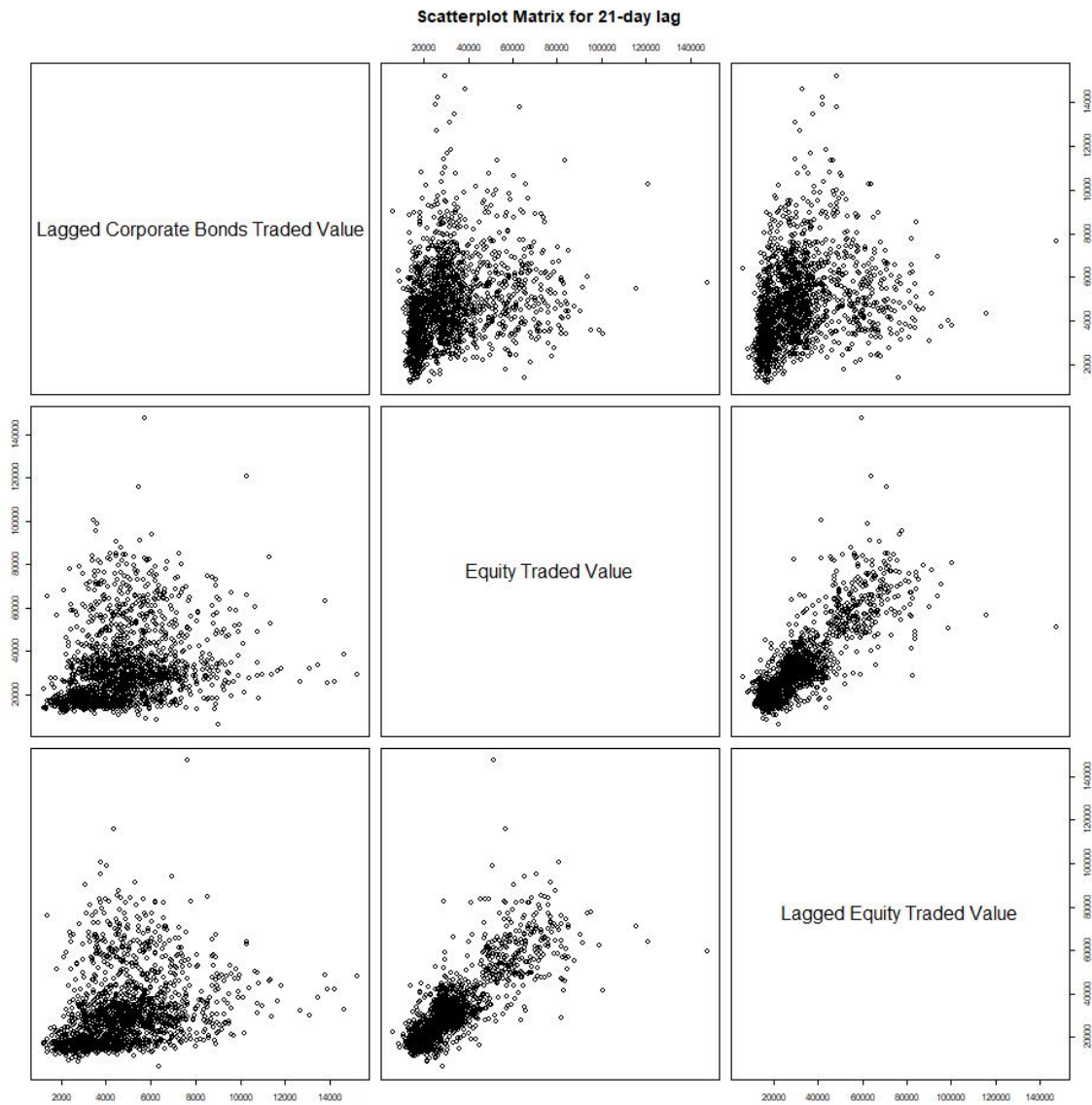


Figure 4.42: Pairwise Scatterplot of the variables.

```

> print(summary(lin_reg_lag21))

Call:
lm(formula = lag_21day__data1$E_Trđ_val_cr ~ lag_21day__data1$CB_trđ_val_cr +
    lag_21day__data1$E_Trđ_val_cr_lag)

Residuals:
    Min       1Q   Median       3Q      Max
-82350  -4535  -1075   3647   90418

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      3.625e+03  6.821e+02   5.315 1.21e-07 ***
lag_21day__data1$CB_trđ_val_cr  2.783e-01  1.249e-01   2.227  0.0261 *
lag_21day__data1$E_Trđ_val_cr_lag 8.667e-01  1.438e-02  60.276 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9433 on 1699 degrees of freedom
Multiple R-squared:  0.7031,    Adjusted R-squared:  0.7027
F-statistic: 2011 on 2 and 1699 DF,  p-value: < 2.2e-16

```

Figure 4.43: Summary of Linear Regression Model.

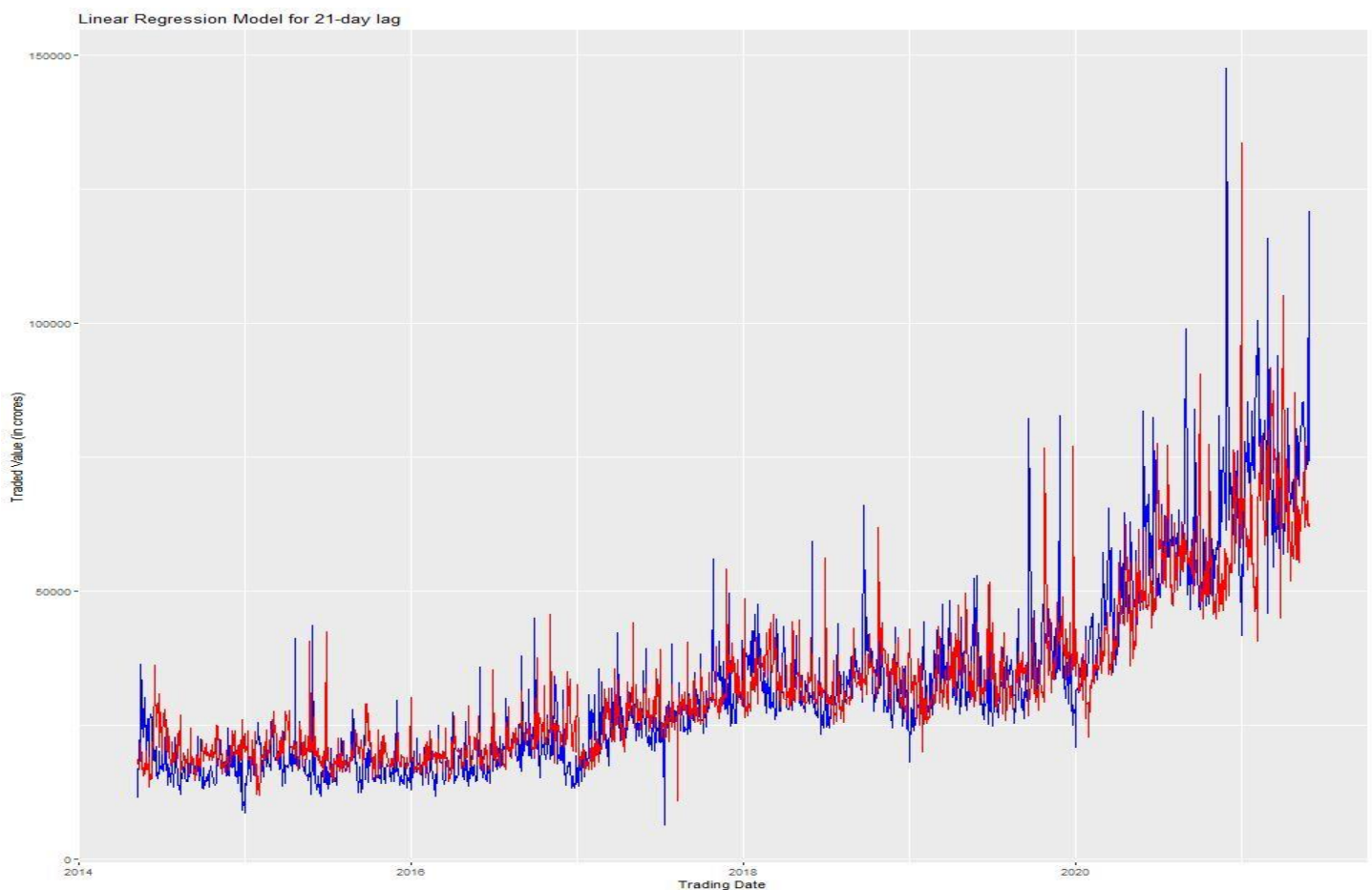


Figure 4.44: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.

```

> print(summary(gam_cr_lag21))

Family: gaussian
Link function: identity

Formula:
lag_21day__data1$E_Trđ_val_cr ~ s(lag_21day__data1$CB_trđ_val_cr,
  bs = "cr") + s(lag_21day__data1$E_Trđ_val_cr_lag, bs = "cr")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept)  32064.7      209.2    153.3  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_21day__data1$CB_trđ_val_cr)  6.369  7.275  1.891  0.0697 .
s(lag_21day__data1$E_Trđ_val_cr_lag) 7.479  8.273 546.788 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.751  Deviance explained = 75.3%
GCV = 7.511e+07  scale est. = 7.4455e+07  n = 1702

```

Figure 4.45: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

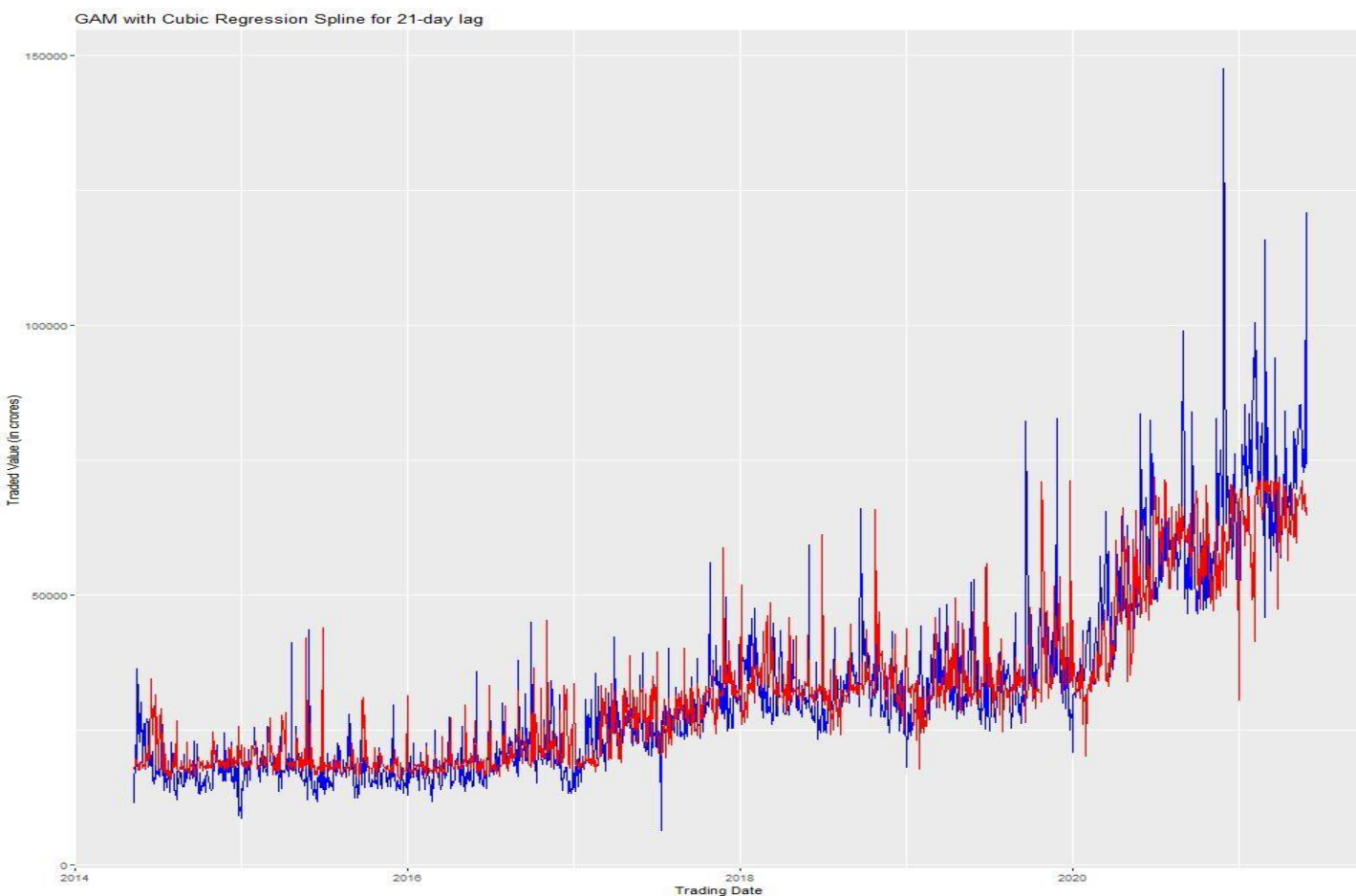


Figure 4.46: Plotting the Predicted Values obtained from the GAM over the Original Value.

28-day Lagging Approach –

The variables used under this approach are 28-day Lagged Corporate Bond variable (CB_trd_val_cr), Equity Market variable (E_Trđ_val_cr) and 28-day Lagged Equity Market Variable (E_Trđ_val_cr_lag).

	CB_trd_val_cr	E_Trđ_val_cr	E_Trđ_val_cr_lag
CB_trd_val_cr	1.000	0.262	0.273
E_Trđ_val_cr	0.262	1.000	0.838
E_Trđ_val_cr_lag	0.273	0.838	1.000

Figure 4.47: Correlational Analysis between the variables.

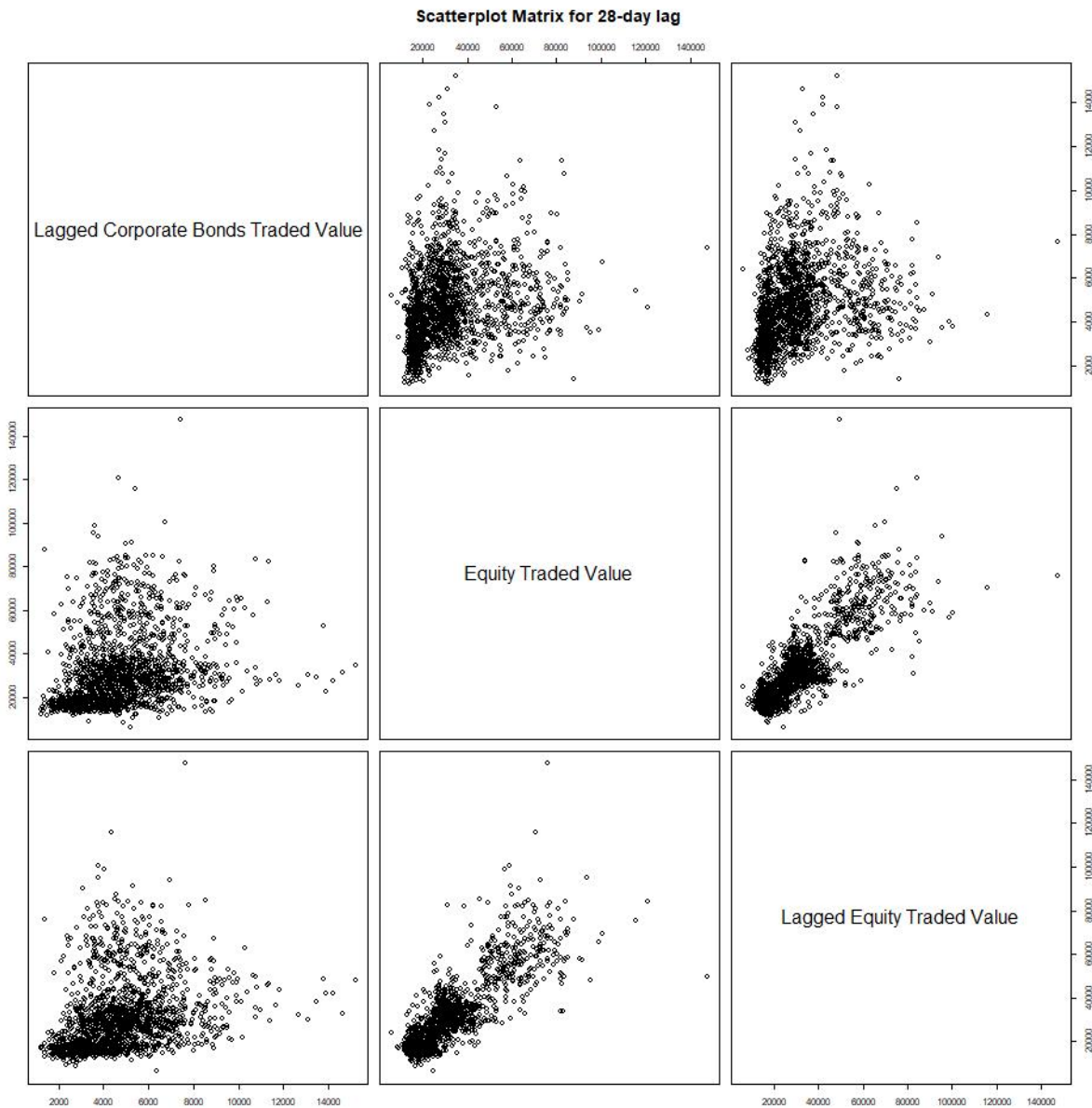


Figure 4.48: Pairwise Scatterplot of the variables.


```

> print(summary(lin_reg_lag28))
Call:
lm(formula = lag_28day__data1$E_Trđ_val_cr ~ lag_28day__data1$CB_trđ_val_cr +
    lag_28day__data1$E_Trđ_val_cr_lag)

Residuals:
    Min       1Q   Median       3Q      Max
-58651  -4927   -941    4207   98512

Coefficients:
                Estimate Std. Error t value Pr(>|t|)
(Intercept)      3.371e+03  6.864e+02   4.911 9.94e-07 ***
lag_28day__data1$CB_trđ_val_cr  3.307e-01  1.255e-01   2.634  0.00851 **
lag_28day__data1$E_Trđ_val_cr_lag  8.723e-01  1.452e-02  60.078 < 2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 9446 on 1692 degrees of freedom
Multiple R-squared:  0.7028,    Adjusted R-squared:  0.7024
F-statistic: 2000 on 2 and 1692 DF,  p-value: < 2.2e-16

```

Figure 4.49: Summary of Linear Regression Model.

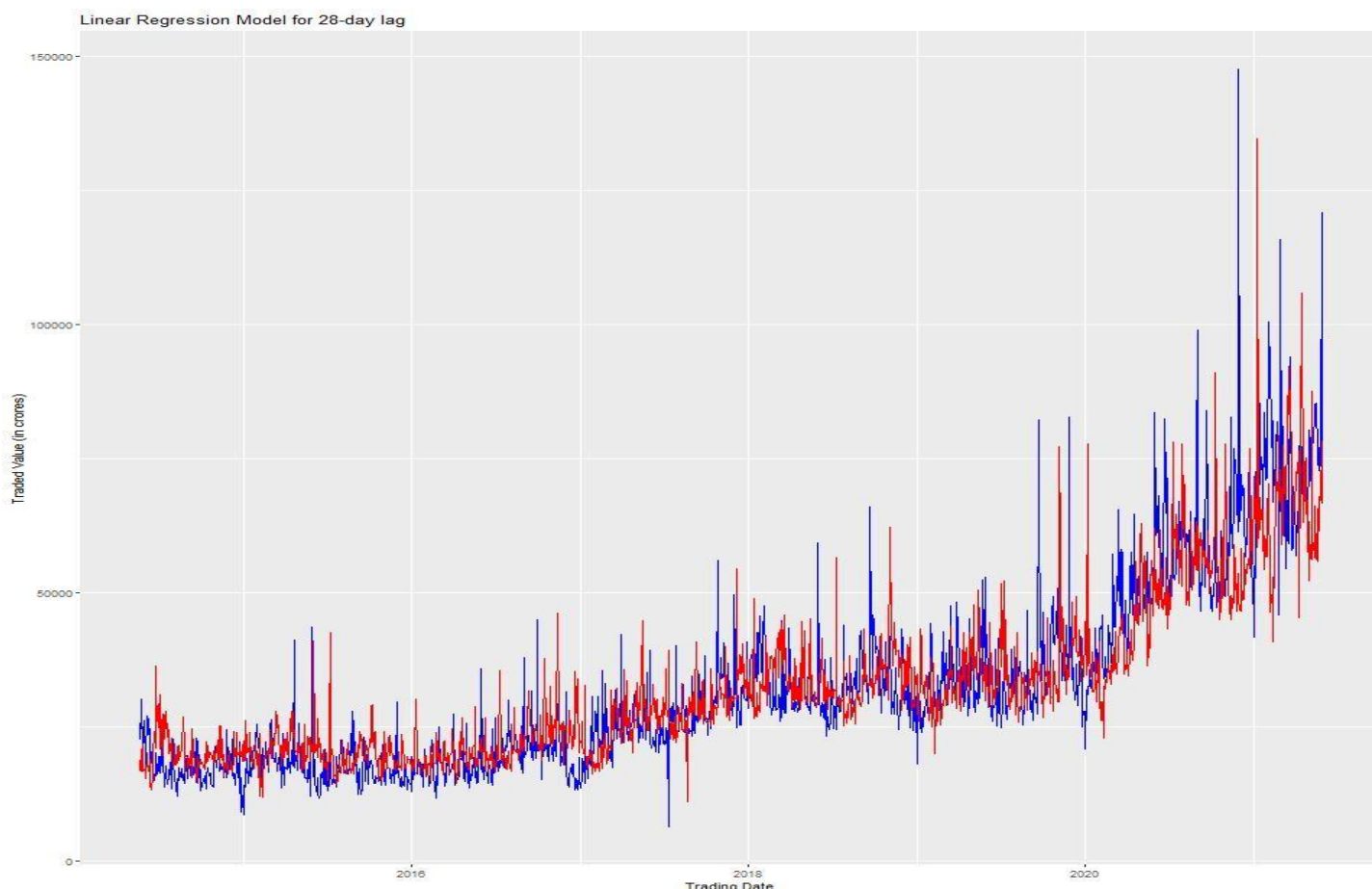


Figure 4.50: Plotting the Predicted Values obtained from the Linear Regression Model over the Original Value.

```

> print(summary(gam_cr_lag28))

Family: gaussian
Link function: identity

Formula:
lag_28day__data1$E_Trđ_val_cr ~ s(lag_28day__data1$CB_trđ_val_cr,
  bs = "cr") + s(lag_28day__data1$E_Trđ_val_cr_lag, bs = "cr")

Parametric coefficients:
              Estimate Std. Error t value Pr(>|t|)
(Intercept) 32107.2      215.4    149.1  <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Approximate significance of smooth terms:
              edf Ref.df      F p-value
s(lag_28day__data1$CB_trđ_val_cr)  4.671  5.518  2.409  0.0239 *
s(lag_28day__data1$E_Trđ_val_cr_lag) 7.645  8.404 498.133 <2e-16 ***
---
Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

R-sq.(adj) = 0.738  Deviance explained = 74%
GCV = 7.9231e+07  scale est. = 7.8608e+07  n = 1695

```

Figure 4.51: Summary of Generalized Addictive Model (GAM) with Cubic Regression Spline.

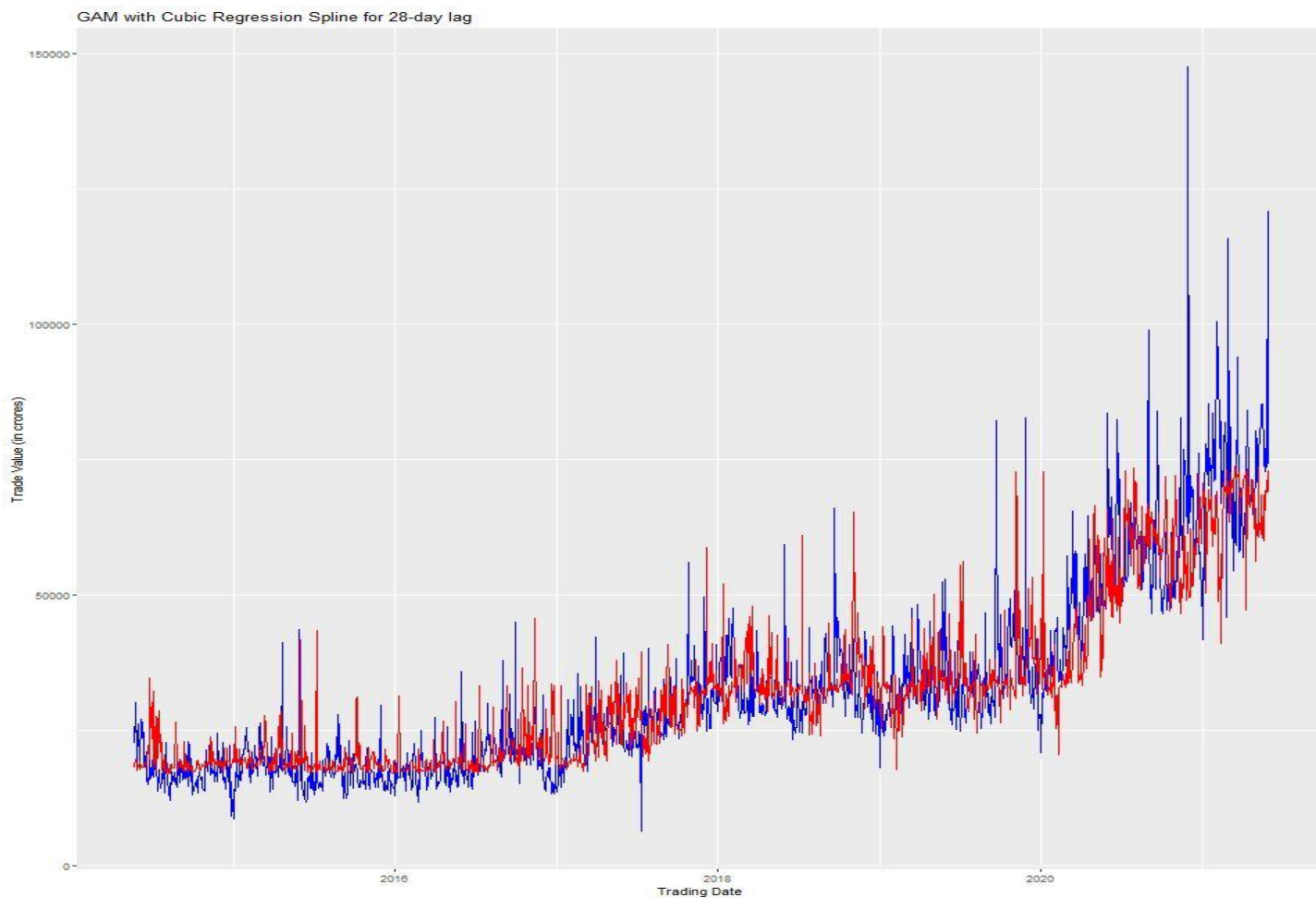


Figure 4.52: Plotting the Predicted Values obtained from the GAM over the Original Value.

Model Performance Phase:

This phase will explore the performance of the Linear Regression Model and the Generalized Addictive Model with Cubic Regression Spline for the 1-day, 3-days, 5-days, 7-days, 14-days, 21-days, 28-days lagging approach. The performance metrics used for evaluation are Root Mean Squared Error Loss (RMSE) and Mean Absolute Error Loss (MAE).

	RMSE	MAE
Linear Regression Model of 1-day lag	7181.653	4220.464
GAM with Cubic Regression Spline of 1-day lag	6680.586	4034.612
Linear Regression Model of 3-day lag	7784.799	4874.980
GAM with Cubic Regression Spline of 3-day lag	7316.394	4624.966
Linear Regression Model of 5-day lag	8180.620	5197.055
GAM with Cubic Regression Spline of 5-day lag	7690.585	4954.595
Linear Regression Model of 7-day lag	8611.524	5557.342
GAM with Cubic Regression Spline of 7-day lag	8075.015	5262.716
Linear Regression Model of 14-day lag	8602.643	5779.553
GAM with Cubic Regression Spline of 14-day lag	8165.844	5466.906
Linear Regression Model of 21-day lag	9424.986	6176.055
GAM with Cubic Regression Spline of 21-day lag	8591.006	5666.201
Linear Regression Model of 28-day lag	9437.672	6473.811
GAM with Cubic Regression Spline of 28-day lag	8831.247	5974.600

Figure 4.53: Performance Matrix of Linear Regression and Support Vector Machine Models.

SECTION 5: RESULT

The result of the model developed for the study could be summarized in the tabularized manner -

Lagged Approach	Correlation between Lagged Corporate Bond Market Variable and Equity Market Variable	Correlation between Lagged Corporate Bond Market Variable and Lagged Equity Market Variable	Correlation between Lagged Equity Market Variable and Equity Market Variable	Adjusted R-Square of Linear Regression Model	Adjusted R-Square of Generalized Addictive Model with Cubic Regression Spline
1-day	0.275	0.282	0.910	0.8275	0.85
3-day	0.253	0.278	0.893	0.7973	0.82
5-day	0.254	0.279	0.881	0.7762	0.801
7-day	0.241	0.278	0.867	0.7521	0.781
14-day	0.255	0.278	0.868	0.7525	0.776
21-day	0.261	0.278	0.838	0.7027	0.751
28-day	0.262	0.273	0.838	0.7024	0.738

Table 5.1: Summary of the Models Developed for the Study

From the Figure 4.53, it could be stated that Generalized Addictive Model with Cubic Regression Spline has performed better than the Linear Regression on each lagged approach. This statement is also evident from the Table 5.1 where The GAM with Cubic Regression Spline outperforms Linear Regression Model with respect to the deviance explained by them which is represented in the form of Adjusted R-Square values.

With the increase in the Lagged Approach, the Correlational Analysis suggests that there seems to be little deviation between the different variables of different lagged days which could be proven from the result obtained in the form of performance metrics where the deviations are way too less when compared with the original values which are in crores. The Line Graphs depicting the Predictive Model over the Original Model are also supporting the same explanation.

SECTION 6: CONCLUSION

The objective of the study was to analyze the interdependency of the Financial Markets' Instruments within the Indian Economy by using Machine Learning Algorithms. The Financial Markets' Instruments studied under the project was Equity and Corporate Bonds. The Machine Learning Algorithms used for the study were Linear Regression Analysis and Generalized Addictive Models with Cubic Regression Spline.

It was observed that to study the interdependency of the corporate bonds and equities, lagging approach was required. Hence, 1-day, 3-days, 5-days, 7-days, 14-days, 21-days, and 28-days lagging approach were utilized for the study. The performance metrics used for evaluation of Models were Root Mean Square Error Loss and Mean Absolute Error Loss.

The Lagged Corporate Bond Market variable and Lagged Equity Market variable were required to predict the present Equity Market variable because at an individual level, Lagged Corporate Bond Market variable was unable to explain the deviance with a high precision and accuracy. The Lagged Equity Market variable was able to explain the deviance but not with a high precision.

The Lagged Approach suggests that the increase in the number of lagging days could intimate an investor if a Black Swan Event is a possibility or not. There is a lot of ground to cover as they are various Machine Learning Algorithms such as Markov Chains, Hidden Markov Models etc, and Deep Learning Algorithms such as Recurrent Neural Networks, Convolutional Neural Network etc that could be deployed to better explain the deviance. The models developed in this study could be used in other countries to understand their financial market interdependencies and/or to predict one market instruments using another.

REFERENCES

Kara, Yakup., Melek Acar Boyacioglu and Omer Kaan Baykan, 2011. Predicting direction of stock price index movement using Artificial Neural Networks and Support Vector Machines: The sample of Istanbul Stock Exchange, *Expert Systems with Applications* 38, 5311-5319.

Chong, Eunsuk., Chulwoo Han and Frank C. Park, 2017. Deep Learning Networks for Stock Market Analysis and Prediction: Methodology, data representations and case studies, *Expert Systems with Applications* 83, 187-205.

Hiransha et. al., 2018. NSE Stock Market Prediction using Deep-Learning Models, *Procedia Computer Science* 132, 1351-1362.

Johnson, Robert and Philip Young, 2002. Bond Market Volatility compared with Stock Market Volatility: Evidence from the UK, *Journal of Asset Management Vol.3*, 101-111.

```
library(xlsx) #for importing Excel file
library(ggplot2) #for plotting graphs
library(MLmetrics) #for using performance metrics
library(mgcv) #for using GAM Models
library(stats) #for using complete.cases functions
library(tidyverse) #for using lag- and lead- functions
```

```
#Importing Data into the R-Studio
```

```
data <- readxl::read_xlsx("D:/Data/2019-21/MBA Business Analytics/Semester 4/Project
Dissertation/Fin_mar_inst_data.xlsx", sheet = 1)
```

```
#Viewing the Imported Data
```

```
View(data)
```

```
#Analyzing the Imported Data for NA values
```

```
table(complete.cases(data))
```

```
#Creating the subset of the Imported Data by Omitting the NA values
```

```
new_data <- na.omit(data)
```

```
#Analyzing the Newly Created Data for NA values
```

```
table(complete.cases(new_data))
```

```
#Plotting the Graphs
```

```
jpeg(file = "D:/CB-Trd-Val-cr.jpeg", width = 960, height = 960)
```

```
ggplot(data = new_data) +
```

```
  geom_line(mapping= aes(x= new_data$Date, y= new_data$CB_trd_val_cr)) +
```

```
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
```

```
  ggtitle("Total Amount of Corporate Bonds' Traded in a Particular Day")
```

```
dev.off()
```

```
jpeg(file = "D:/CB-securities.jpeg", width = 960, height = 960)
```

```
ggplot(data = new_data) +
```

```
  geom_line(mapping= aes(x= new_data$Date, y= new_data$Cb_securities)) +
```

```
  labs(x = "Trading Date", y = "No. of Securities") +
```

```
ggtitle("No. of Corporate Bonds available for Trading in a Particular Day")
dev.off()
```

```
jpeg(file = "D:/CB-trades.jpeg", width = 960, height = 960)
ggplot(data = new_data) +
  geom_line(mapping= aes(x= new_data$Date, y= new_data$CB_trades)) +
  labs(x = "Trading Date", y = "No. of Trades") +
  ggtitle("No. of Corporate Bonds' Traded in a Particular Day")
dev.off()
```

```
jpeg(file = "D:/E-Trd-val-cr.jpeg", width = 960, height = 960)
ggplot(data = new_data) +
  geom_line(mapping= aes(x= new_data$Date, y= new_data$E_Tr_d_val_cr)) +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("Total Amount of Equities' Traded in a Particular Day")
dev.off()
```

```
jpeg(file = "D:/E-securities.jpeg", width = 960, height = 960)
ggplot(data = new_data) +
  geom_line(mapping= aes(x= new_data$Date, y= new_data$E_securities)) +
  labs(x = "Trading Date", y = "No. of Securities") +
  ggtitle("No. of Equities available for Trading in a Particular Day")
dev.off()
```

```
jpeg(file = "D:/E-trades.jpeg", width = 960, height = 960)
ggplot(data = new_data) +
  geom_line(mapping= aes(x= new_data$Date, y= new_data$E_trades)) +
  labs(x = "Trading Date", y = "No. of Trades") +
  ggtitle("No. of Equities' Traded in a Particular Day")
dev.off()
```

```
jpeg(file = "D:/E-Trd-Qty-Lac.jpeg", width = 960, height = 960)
ggplot(data = new_data) +
  geom_line(mapping= aes(x= new_data$Date, y= new_data$E_trd_qty_lac)) +
```

```
labs(x = "Trading Date", y = "Traded Quantity (in Lakhs)") +  
ggtitle("Volume of Equities' Traded in a Particular Day")  
dev.off()
```

```
#Plotting the Scatter Plot Matrix
```

```
jpeg(file = "D:/Scatterplot Matrix.jpeg", width = 1440, height = 1440)  
pairs(new_data[,2:8],  
      labels=c("Corporate Bonds Securities", "Corporate Bonds Trades",  
              "Corporate Bonds Traded Value", "Equity Securities",  
              "Equity Trades", "Equity Traded Quantity", "Equity Traded Value"),  
      main = "Scatterplot Matrix")  
dev.off()
```

```
#Finding the Correlation among the dataset with no lag
```

```
cor_new_data <- round(cor(new_data[, c(2:8)], method="pearson"),3)  
View(cor_new_data)
```

```
#Compiling a new dataset with no lag value for Corporate Bonds and Equities
```

```
same_day_data <- data.frame(new_data$Date, new_data$CB_trd_val_cr, new_data$E_Trld_val_cr)  
names(same_day_data) <- c("Date", "CB_trd_val_cr", "E_Trld_val_cr")  
table(complete.cases(same_day_data))  
same_day_data1 <- na.omit(same_day_data)
```

```
#Finding the Correlation among the dataset with no lag
```

```
cor_mat_same <- round(cor(same_day_data1[, c(2:3)], method="pearson"),3)  
View(cor_mat_same)
```

```
#Compiling a new dataset with 1-day lag value for Corporate Bonds and Equities
```

```
lag_1day__data <- data.frame(new_data$Date, lead(new_data$CB_trd_val_cr,n=1),  
                            new_data$E_Trld_val_cr, lead(new_data$E_Trld_val_cr,n=1))  
names(lag_1day__data) <- c("Date", "CB_trd_val_cr", "E_Trld_val_cr", "E_Trld_val_cr_lag")  
table(complete.cases(lag_1day__data))  
lag_1day__data1 <- na.omit(lag_1day__data)
```

```

#Finding the Correlation among the dataset with 1-day lag
cor_mat_lag1 <- round(cor(lag_1day__data1[, c(2:4)], method="pearson"),3)
View(cor_mat_lag1)

#Plotting the Scatter Plot Matrix for 1-day lag
jpeg(file = "D:/Scatterplot Matrix1.jpeg", width = 960, height = 960)
pairs(lag_1day__data1[,2:4],
      labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
              "Lagged Equity Traded Value"),
      main = "Scatterplot Matrix for 1-day lag")
dev.off()

#Linear Regression Model for 1-day lag dataset
lin_reg_lag1 <- lm(lag_1day__data1$E_Tr_d_val_cr ~ lag_1day__data1$CB_tr_d_val_cr +
                  lag_1day__data1$E_Tr_d_val_cr_lag)
print(summary(lin_reg_lag1))

#Root Mean Square Error Loss for Linear Regression Model for 1-day lag dataset
RMSE_lin_reg_lag1 <- round(RMSE(lin_reg_lag1$fitted.values,lag_1day__data1$E_Tr_d_val_cr),3)

#Mean Absolute Error Loss for Linear Regression Model for 1-day lag dataset
MAE_lin_reg_lag1 <- round(MAE(lin_reg_lag1$fitted.values,lag_1day__data1$E_Tr_d_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#Linear Regression Model for 1-day lag dataset
jpeg(file = "D:/LR-1d.jpeg", width = 960, height = 960)
ggplot(data = lag_1day__data1) +
  geom_line(mapping= aes(x= lag_1day__data1$Date, y= lag_1day__data1$E_Tr_d_val_cr), colour = "blue") +
  geom_line(mapping= aes(x= lag_1day__data1$Date, y= lin_reg_lag1$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("Linear Regression Model for 1-day lag")
dev.off()

#Generative Addictive Model with Cubic Regression Spline for 1-day lag dataset

```



```
gam_cr_lag1 <- gam(lag_1day__data1$E_Tr_d_val_cr ~ s(lag_1day__data1$CB_tr_d_val_cr, bs="cr")
  + s(lag_1day__data1$E_Tr_d_val_cr_lag, bs="cr"))
print(summary(gam_cr_lag1))
```

```
#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 1-day lag dataset
RMSE_gam_cr_lag1 <- round(RMSE(gam_cr_lag1$fitted.values,lag_1day__data1$E_Tr_d_val_cr),3)
```

```
#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 1-day lag dataset
MAE_gam_cr_lag1 <- round(MAE(gam_cr_lag1$fitted.values,lag_1day__data1$E_Tr_d_val_cr),3)
```

```
#Plotting the Graph of the Predicted Value obtained from
#GAM with Cubic Regression Spline for 1-day lag dataset
jpeg(file = "D:/GAM-1d.jpeg", width = 960, height = 960)
ggplot(data = lag_1day__data1) +
  geom_line(mapping= aes(x= lag_1day__data1$Date, y= lag_1day__data1$E_Tr_d_val_cr), colour = "blue") +
  geom_line(mapping= aes(x=lag_1day__data1$Date, y= gam_cr_lag1$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("GAM with Cubic Regression Spline for 1-day lag")
dev.off()
```

```
#Compiling a new dataset with 3-day lag value for Corporate Bonds and Equities
lag_3day__data <- data.frame(new_data$Date, lead(new_data$CB_tr_d_val_cr,n=3),
  new_data$E_Tr_d_val_cr, lead(new_data$E_Tr_d_val_cr,n=3))
names(lag_3day__data) <- c("Date", "CB_tr_d_val_cr", "E_Tr_d_val_cr", "E_Tr_d_val_cr_lag")
table(complete.cases(lag_3day__data))
lag_3day__data1 <- na.omit(lag_3day__data)
```

```
#Finding the Correlation among the dataset with 3-day lag
cor_mat_lag3 <- round(cor(lag_3day__data1[, c(2:4)], method="pearson"),3)
View(cor_mat_lag3)
```

```
#Plotting the Scatter Plot Matrix for 3-day lag
jpeg(file = "D:/Scatterplot Matrix3.jpeg", width = 960, height = 960)
pairs(lag_3day__data1[,2:4],
```

```

labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
        "Lagged Equity Traded Value"),
main = "Scatterplot Matrix for 3-day lag")
dev.off()

#Linear Regression Model for 3-day lag dataset
lin_reg_lag3 <- lm(lag_3day__data1$E_Tr_d_val_cr ~ lag_3day__data1$CB_tr_d_val_cr +
                 lag_3day__data1$E_Tr_d_val_cr_lag)
print(summary(lin_reg_lag3))

#Root Mean Square Error Loss for Linear Regression Model for 3-day lag dataset
RMSE_lin_reg_lag3 <- round(RMSE(lin_reg_lag3$fitted.values, lag_3day__data1$E_Tr_d_val_cr),3)

#Mean Absolute Error Loss for Linear Regression Model for 3-day lag dataset
MAE_lin_reg_lag3 <- round(MAE(lin_reg_lag3$fitted.values, lag_3day__data1$E_Tr_d_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#Linear Regression Model for 3-day lag dataset
jpeg(file = "D:/LR-3d.jpeg", width = 960, height = 960)
ggplot(data = lag_3day__data1) +
  geom_line(mapping= aes(x= lag_3day__data1$Date, y= lag_3day__data1$E_Tr_d_val_cr), colour = "blue") +
  geom_line(mapping= aes(x= lag_3day__data1$Date, y= lin_reg_lag3$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("Linear Regression Model for 3-day lag")
dev.off()

#Generative Addictive Model with Cubic Regression Spline for 3-day lag dataset
gam_cr_lag3 <- gam(lag_3day__data1$E_Tr_d_val_cr ~ s(lag_3day__data1$CB_tr_d_val_cr, bs="cr")
                 + s(lag_3day__data1$E_Tr_d_val_cr_lag, bs="cr"))
print(summary(gam_cr_lag3))

#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 3-day lag dataset
RMSE_gam_cr_lag3 <- round(RMSE(gam_cr_lag3$fitted.values, lag_3day__data1$E_Tr_d_val_cr),3)

```

```

#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 3-day lag dataset
MAE_gam_cr_lag3 <- round(MAE(gam_cr_lag3$fitted.values,lag_3day__data1$E_Tr_d_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#GAM with Cubic Regression Spline for 3-day lag dataset
jpeg(file = "D:/GAM-3d.jpeg", width = 960, height = 960)
ggplot(data = lag_3day__data1) +
  geom_line(mapping= aes(x= lag_3day__data1$Date, y= lag_3day__data1$E_Tr_d_val_cr), colour = "blue") +
  geom_line(mapping= aes(x= lag_3day__data1$Date, y= gam_cr_lag3$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("GAM with Cubic Regression Spline for 3-day lag")
dev.off()

#Compiling a new dataset with 5-day lag value for Corporate Bonds and Equities
lag_5day__data <- data.frame(new_data$Date, lead(new_data$CB_tr_d_val_cr,n=5),
  new_data$E_Tr_d_val_cr, lead(new_data$E_Tr_d_val_cr,n=5))
names(lag_5day__data) <- c("Date", "CB_tr_d_val_cr", "E_Tr_d_val_cr", "E_Tr_d_val_cr_lag")
table(complete.cases(lag_5day__data))
lag_5day__data1 <- na.omit(lag_5day__data)

#Finding the Correlation among the dataset with 5-day lag
cor_mat_lag5 <- round(cor(lag_5day__data1[, c(2:4)], method="pearson"),3)
View(cor_mat_lag5)

#Plotting the Scatter Plot Matrix for 5-day lag
jpeg(file = "D:/Scatterplot Matrix5.jpeg", width = 960, height = 960)
pairs(lag_5day__data1[,2:4],
  labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
  "Lagged Equity Traded Value"),
  main = "Scatterplot Matrix for 5-day lag")
dev.off()

#Linear Regression Model for 5-day lag dataset
lin_reg_lag5 <- lm(lag_5day__data1$E_Tr_d_val_cr ~ lag_5day__data1$CB_tr_d_val_cr +

```

```

lag_5day__data1$E_Trđ_val_cr_lag)
print(summary(lin_reg_lag5))

#Root Mean Square Error Loss for Linear Regression Model for 5-day lag dataset
RMSE_lin_reg_lag5 <- round(RMSE(lin_reg_lag5$fitted.values, lag_5day__data1$E_Trđ_val_cr),3)

#Mean Absolute Error Loss for Linear Regression Model for 5-day lag dataset
MAE_lin_reg_lag5 <- round(MAE(lin_reg_lag5$fitted.values, lag_5day__data1$E_Trđ_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#Linear Regression Model for 5-day lag dataset
jpeg(file = "D:/LR-5d.jpeg", width = 960, height = 960)
ggplot(data = lag_5day__data1) +
  geom_line(mapping= aes(x=lag_5day__data1$Date, y= lag_5day__data1$E_Trđ_val_cr), colour = "blue") +
  geom_line(mapping= aes(x=lag_5day__data1$Date, y= lin_reg_lag5$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("Linear Regression Model for 5-day lag")
dev.off()

#Generative Addictive Model with Cubic Regression Spline for 5-day lag dataset
gam_cr_lag5 <- gam(lag_5day__data1$E_Trđ_val_cr ~ s(lag_5day__data1$CB_trđ_val_cr, bs="cr")
  + s(lag_5day__data1$E_Trđ_val_cr_lag, bs="cr"))
print(summary(gam_cr_lag5))

#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 5-day lag dataset
RMSE_gam_cr_lag5 <- round(RMSE(gam_cr_lag5$fitted.values, lag_5day__data1$E_Trđ_val_cr),3)

#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 5-day lag dataset
MAE_gam_cr_lag5 <- round(MAE(gam_cr_lag5$fitted.values, lag_5day__data1$E_Trđ_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#GAM with Cubic Regression Spline for 5-day lag dataset
jpeg(file = "D:/GAM-5d.jpeg", width = 960, height = 960)
ggplot(data = lag_5day__data1) +

```

```

geom_line(mapping= aes(x= lag_5day__data1$Date, y= lag_5day__data1$E_Tr_d_val_cr), colour = "blue") +
geom_line(mapping= aes(x= lag_5day__data1$Date, y= gam_cr_lag5$fitted.values), colour = "red") +
labs(x = "Trading Date", y = "Traded Value (in crores)") +
ggtitle("GAM with Cubic Regression Spline for 5-day lag")
dev.off()

```

```

#Compiling a new dataset with 7-day lag value for Corporate Bonds and Equities
lag_7day__data <- data.frame(new_data$Date, lead(new_data$CB_tr_d_val_cr,n=7),
                             new_data$E_Tr_d_val_cr, lead(new_data$E_Tr_d_val_cr,n=7))
names(lag_7day__data) <- c("Date", "CB_tr_d_val_cr", "E_Tr_d_val_cr", "E_Tr_d_val_cr_lag")
table(complete.cases(lag_7day__data))
lag_7day__data1 <- na.omit(lag_7day__data)

```

```

#Finding the Correlation among the dataset with 7-day lag
cor_mat_lag7 <- round(cor(lag_7day__data1[, c(2:4)], method="pearson"),3)
View(cor_mat_lag7)

```

```

#Plotting the Scatter Plot Matrix for 7-day lag
jpeg(file = "D:/Scatterplot Matrix7.jpeg", width = 960, height = 960)
pairs(lag_7day__data1[,2:4],
      labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
               "Lagged Equity Traded Value"),
      main = "Scatterplot Matrix for 7-day lag")
dev.off()

```

```

#Linear Regression Model for 7-day lag dataset
lin_reg_lag7 <- lm(lag_7day__data1$E_Tr_d_val_cr ~ lag_7day__data1$CB_tr_d_val_cr +
                  lag_7day__data1$E_Tr_d_val_cr_lag)
print(summary(lin_reg_lag7))

```

```

#Root Mean Square Error Loss for Linear Regression Model for 7-day lag dataset
RMSE_lin_reg_lag7 <- round(RMSE(lin_reg_lag7$fitted.values, lag_7day__data1$E_Tr_d_val_cr),3)

```

```

#Mean Absolute Error Loss for Linear Regression Model for 7-day lag dataset

```

```
MAE_lin_reg_lag7 <- round(MAE(lin_reg_lag7$fitted.values, lag_7day__data1$E_Tr_d_val_cr),3)
```

```
#Plotting the Graph of the Predicted Value obtained from
```

```
#Linear Regression Model for 7-day lag dataset
```

```
jpeg(file = "D:/LR-7d.jpeg", width = 960, height = 960)
```

```
ggplot(data = lag_7day__data1) +
```

```
  geom_line(mapping= aes(x= lag_7day__data1$Date, y= lag_7day__data1$E_Tr_d_val_cr), colour = "blue") +
```

```
  geom_line(mapping= aes(x= lag_7day__data1$Date, y= lin_reg_lag7$fitted.values), colour = "red") +
```

```
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
```

```
  ggtitle("Linear Regression Model for 7-day lag")
```

```
dev.off()
```

```
#Generative Addictive Model with Cubic Regression Spline for 7-day lag dataset
```

```
gam_cr_lag7 <- gam(lag_7day__data1$E_Tr_d_val_cr ~ s(lag_7day__data1$CB_tr_d_val_cr, bs="cr")
```

```
  + s(lag_7day__data1$E_Tr_d_val_cr_lag, bs="cr"))
```

```
print(summary(gam_cr_lag7))
```

```
#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 7-day lag dataset
```

```
RMSE_gam_cr_lag7 <- round(RMSE(gam_cr_lag7$fitted.values, lag_7day__data1$E_Tr_d_val_cr),3)
```

```
#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 7-day lag dataset
```

```
MAE_gam_cr_lag7 <- round(MAE(gam_cr_lag7$fitted.values, lag_7day__data1$E_Tr_d_val_cr),3)
```

```
#Plotting the Graph of the Predicted Value obtained from
```

```
#GAM with Cubic Regression Spline for 7-day lag dataset
```

```
jpeg(file = "D:/GAM-7d.jpeg", width = 960, height = 960)
```

```
ggplot(data = lag_7day__data1) +
```

```
  geom_line(mapping= aes(x= lag_7day__data1$Date, y= lag_7day__data1$E_Tr_d_val_cr), colour = "blue") +
```

```
  geom_line(mapping= aes(x= lag_7day__data1$Date, y= gam_cr_lag7$fitted.values), colour = "red") +
```

```
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
```

```
  ggtitle("GAM with Cubic Regression Spline for 7-day lag")
```

```
dev.off()
```

```
#Compiling a new dataset with 14-day lag value for Corporate Bonds and Equities
```

```

lag_14day__data <- data.frame(new_data$Date, lead(new_data$CB_trd_val_cr,n=14),
                             new_data$E_Trđ_val_cr, lead(new_data$E_Trđ_val_cr,n=14))
names(lag_14day__data) <- c("Date", "CB_trđ_val_cr", "E_Trđ_val_cr", "E_Trđ_val_cr_lag")
table(complete.cases(lag_14day__data))
lag_14day__data1 <- na.omit(lag_14day__data)

#Finding the Correlation among the dataset with 14-day lag
cor_mat_lag14 <- round(cor(lag_14day__data1[, c(2:4)], method="pearson"),3)
View(cor_mat_lag14)

#Plotting the Scatter Plot Matrix for 14-day lag
jpeg(file = "D:/Scatterplot Matrix14.jpeg", width = 960, height = 960)
pairs(lag_14day__data1[,2:4],
      labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
              "Lagged Equity Traded Value"),
      main = "Scatterplot Matrix for 14-day lag")
dev.off()

#Linear Regression Model for 14-day lag dataset
lin_reg_lag14 <- lm(lag_14day__data1$E_Trđ_val_cr ~ lag_14day__data1$CB_trđ_val_cr +
                  lag_14day__data1$E_Trđ_val_cr_lag)
print(summary(lin_reg_lag14))

#Root Mean Square Error Loss for Linear Regression Model for 14-day lag dataset
RMSE_lin_reg_lag14 <- round(RMSE(lin_reg_lag14$fitted.values, lag_14day__data1$E_Trđ_val_cr),3)

#Mean Absolute Error Loss for Linear Regression Model for 14-day lag dataset
MAE_lin_reg_lag14 <- round(MAE(lin_reg_lag14$fitted.values, lag_14day__data1$E_Trđ_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#Linear Regression Model for 14-day lag dataset
jpeg(file = "D:/LR-14d.jpeg", width = 960, height = 960)
ggplot(data = lag_14day__data1) +

```



```

geom_line(mapping= aes(x= lag_14day__data1$Date, y= lag_14day__data1$E_Tr_d_val_cr), colour = "blue")
+
geom_line(mapping= aes(x= lag_14day__data1$Date, y= lin_reg_lag14$fitted.values), colour = "red") +
labs(x = "Trading Date", y = "Traded Value (in crores)") +
ggtitle("Linear Regression Model for 14-day lag")
dev.off()

#Generative Addictive Model with Cubic Regression Spline for 14-day lag dataset
gam_cr_lag14 <- gam(lag_14day__data1$E_Tr_d_val_cr ~ s(lag_14day__data1$CB_tr_d_val_cr, bs="cr")
+ s(lag_14day__data1$E_Tr_d_val_cr_lag, bs="cr"))
print(summary(gam_cr_lag14))

#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 14-day lag dataset
RMSE_gam_cr_lag14 <- round(RMSE(gam_cr_lag14$fitted.values, lag_14day__data1$E_Tr_d_val_cr),3)

#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 14-day lag dataset
MAE_gam_cr_lag14 <- round(MAE(gam_cr_lag14$fitted.values, lag_14day__data1$E_Tr_d_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#GAM with Cubic Regression Spline for 14-day lag dataset
jpeg(file = "D:/GAM-14d.jpeg", width = 960, height = 960)
ggplot(data = lag_14day__data1) +
geom_line(mapping= aes(x= lag_14day__data1$Date, y= lag_14day__data1$E_Tr_d_val_cr), colour = "blue")
+
geom_line(mapping= aes(x= lag_14day__data1$Date, y= gam_cr_lag14$fitted.values), colour = "red") +
labs(x = "Trading Date", y = "Traded Value (in crores)") +
ggtitle("GAM with Cubic Regression Spline for 14-day lag")
dev.off()

#Compiling a new dataset with 21-day lag value for Corporate Bonds and Equities
lag_21day__data <- data.frame(new_data$Date, lead(new_data$CB_tr_d_val_cr,n=21),
new_data$E_Tr_d_val_cr, lead(new_data$E_Tr_d_val_cr,n=21))
names(lag_21day__data) <- c("Date", "CB_tr_d_val_cr","E_Tr_d_val_cr","E_Tr_d_val_cr_lag")
table(complete.cases(lag_21day__data))

```

```

lag_21day__data1 <- na.omit(lag_21day__data)

#Finding the Correlation among the dataset with 21-day lag
cor_mat_lag21 <- round(cor(lag_21day__data1[, c(2:4)], method="pearson"),3)
View(cor_mat_lag21)

#Plotting the Scatter Plot Matrix for 21-day lag
jpeg(file = "D:/Scatterplot Matrix21.jpeg", width = 960, height = 960)
pairs(lag_21day__data1[,2:4],
      labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
              "Lagged Equity Traded Value"),
      main = "Scatterplot Matrix for 21-day lag")
dev.off()

#Linear Regression Model for 21-day lag dataset
lin_reg_lag21 <- lm(lag_21day__data1$E_Trd_val_cr ~ lag_21day__data1$CB_trd_val_cr +
                  lag_21day__data1$E_Trd_val_cr_lag)
print(summary(lin_reg_lag21))

#Root Mean Square Error Loss for Linear Regression Model for 21-day lag dataset
RMSE_lin_reg_lag21 <- round(RMSE(lin_reg_lag21$fitted.values, lag_21day__data1$E_Trd_val_cr),3)

#Mean Absolute Error Loss for Linear Regression Model for 21-day lag dataset
MAE_lin_reg_lag21 <- round(MAE(lin_reg_lag21$fitted.values, lag_21day__data1$E_Trd_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#Linear Regression Model for 21-day lag dataset
jpeg(file = "D:/LR-21d.jpeg", width = 960, height = 960)
ggplot(data = lag_21day__data1) +
  geom_line(mapping= aes(x= lag_21day__data1$Date, y= lag_21day__data1$E_Trd_val_cr), colour = "blue")
+
  geom_line(mapping= aes(x= lag_21day__data1$Date, y= lin_reg_lag21$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("Linear Regression Model for 21-day lag")

```

```
dev.off()
```

```
#Generative Addictive Model with Cubic Regression Spline for 21-day lag dataset
```

```
gam_cr_lag21 <- gam(lag_21day__data1$E_Trld_val_cr ~ s(lag_21day__data1$CB_trld_val_cr, bs="cr")  
  + s(lag_21day__data1$E_Trld_val_cr_lag, bs="cr"))  
print(summary(gam_cr_lag21))
```

```
#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 21-day lag dataset
```

```
RMSE_gam_cr_lag21 <- round(RMSE(gam_cr_lag21$fitted.values, lag_21day__data1$E_Trld_val_cr),3)
```

```
#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 21-day lag dataset
```

```
MAE_gam_cr_lag21 <- round(MAE(gam_cr_lag21$fitted.values, lag_21day__data1$E_Trld_val_cr),3)
```

```
#Plotting the Graph of the Predicted Value obtained from
```

```
#GAM with Cubic Regression Spline for 21-day lag dataset
```

```
jpeg(file = "D:/GAM-21d.jpeg", width = 960, height = 960)
```

```
ggplot(data = lag_21day__data1) +  
  geom_line(mapping= aes(x= lag_21day__data1$Date, y= lag_21day__data1$E_Trld_val_cr), colour = "blue")  
+  
  geom_line(mapping= aes(x= lag_21day__data1$Date, y= gam_cr_lag21$fitted.values), colour = "red") +  
  labs(x = "Trading Date", y = "Traded Value (in crores)") +  
  ggtitle("GAM with Cubic Regression Spline for 21-day lag")  
dev.off()
```

```
#Compiling a new dataset with 28-day lag value for Corporate Bonds and Equities
```

```
lag_28day__data <- data.frame(new_data$Date, lead(new_data$CB_trld_val_cr,n=28),  
  new_data$E_Trld_val_cr, lead(new_data$E_Trld_val_cr,n=28))  
names(lag_28day__data) <- c("Date", "CB_trld_val_cr", "E_Trld_val_cr", "E_Trld_val_cr_lag")  
table(complete.cases(lag_28day__data))  
lag_28day__data1 <- na.omit(lag_28day__data)
```

```
#Finding the Correlation among the dataset with 28-day lag
```

```
cor_mat_lag28 <- round(cor(lag_28day__data1[, c(2:4)], method="pearson"),3)  
View(cor_mat_lag28)
```

```

#Plotting the Scatter Plot Matrix for 28-day lag
jpeg(file = "D:/Scatterplot Matrix28.jpeg", width = 960, height = 960)
pairs(lag_28day__data1[,2:4],
      labels=c("Lagged Corporate Bonds Traded Value", "Equity Traded Value",
              "Lagged Equity Traded Value"),
      main = "Scatterplot Matrix for 28-day lag")
dev.off()

#Linear Regression Model for 28-day lag dataset
lin_reg_lag28 <- lm(lag_28day__data1$E_Tr_d_val_cr ~ lag_28day__data1$CB_tr_d_val_cr +
                  lag_28day__data1$E_Tr_d_val_cr_lag)
print(summary(lin_reg_lag28))

#Root Mean Square Error Loss for Linear Regression Model for 28-day lag dataset
RMSE_lin_reg_lag28 <- round(RMSE(lin_reg_lag28$fitted.values, lag_28day__data1$E_Tr_d_val_cr),3)

#Mean Absolute Error Loss for Linear Regression Model for 28-day lag dataset
MAE_lin_reg_lag28 <- round(MAE(lin_reg_lag28$fitted.values, lag_28day__data1$E_Tr_d_val_cr),3)

#Plotting the Graph of the Predicted Value obtained from
#Linear Regression Model for 28-day lag dataset
jpeg(file = "D:/LR-28d.jpeg", width = 960, height = 960)
ggplot(data = lag_28day__data1) +
  geom_line(mapping= aes(x= lag_28day__data1$Date, y= lag_28day__data1$E_Tr_d_val_cr), colour = "blue")
+
  geom_line(mapping= aes(x= lag_28day__data1$Date, y= lin_reg_lag28$fitted.values), colour = "red") +
  labs(x = "Trading Date", y = "Traded Value (in crores)") +
  ggtitle("Linear Regression Model for 28-day lag")
dev.off()

#Generative Addictive Model with Cubic Regression Spline for 28-day lag dataset
gam_cr_lag28 <- gam(lag_28day__data1$E_Tr_d_val_cr ~ s(lag_28day__data1$CB_tr_d_val_cr, bs="cr")
                  + s(lag_28day__data1$E_Tr_d_val_cr_lag, bs="cr"))

```

```
print(summary(gam_cr_lag28))
```

```
#Root Mean Square Error Loss for GAM with Cubic Regression Spline for 28-day lag dataset
```

```
RMSE_gam_cr_lag28 <- round(RMSE(gam_cr_lag28$fitted.values, lag_28day__data1$E_Trd_val_cr),3)
```

```
#Mean Absolute Error Loss for GAM with Cubic Regression Spline for 28-day lag dataset
```

```
MAE_gam_cr_lag28 <- round(MAE(gam_cr_lag28$fitted.values, lag_28day__data1$E_Trd_val_cr),3)
```

```
#Plotting the Graph of the Predicted Value obtained from
```

```
#GAM with Cubic Regression Spline for 28-day lag dataset
```

```
jpeg(file = "D:/GAM-28d.jpeg", width = 960, height = 960)
```

```
ggplot(data = lag_28day__data1) +
```

```
  geom_line(mapping= aes(x= lag_28day__data1$Date, y= lag_28day__data1$E_Trd_val_cr), colour = "blue")
```

```
+
```

```
  geom_line(mapping= aes(x= lag_28day__data1$Date, y= gam_cr_lag28$fitted.values), colour = "red") +
```

```
  labs(x = "Trading Date", y = "Trade Value (in crores)") +
```

```
  ggtitle("GAM with Cubic Regression Spline for 28-day lag")
```

```
dev.off()
```

```
#Collaborating all RMSE Metrics
```

```
rmse_model <- c(RMSE_lin_reg_lag1, RMSE_gam_cr_lag1,
```

```
  RMSE_lin_reg_lag3, RMSE_gam_cr_lag3,
```

```
  RMSE_lin_reg_lag5, RMSE_gam_cr_lag5,
```

```
  RMSE_lin_reg_lag7, RMSE_gam_cr_lag7,
```

```
  RMSE_lin_reg_lag14, RMSE_gam_cr_lag14,
```

```
  RMSE_lin_reg_lag21, RMSE_gam_cr_lag21,
```

```
  RMSE_lin_reg_lag28, RMSE_gam_cr_lag28)
```

```
#Collaborating all MAE Metrics
```

```
mae_model <- c(MAE_lin_reg_lag1, MAE_gam_cr_lag1,
```

```
  MAE_lin_reg_lag3, MAE_gam_cr_lag3,
```

```
  MAE_lin_reg_lag5, MAE_gam_cr_lag5,
```

```
  MAE_lin_reg_lag7, MAE_gam_cr_lag7,
```

```
  MAE_lin_reg_lag14, MAE_gam_cr_lag14,
```

```
MAE_lin_reg_lag21, MAE_gam_cr_lag21,  
MAE_lin_reg_lag28, MAE_gam_cr_lag28)
```

```
#Building a Performance Evaluation Model
```

```
performance_model <- data.frame(rmse_model, mae_model)
```

```
colnames(performance_model) <- c("RMSE", "MAE")
```

```
rownames(performance_model) <- c("Linear Regression Model of 1-day lag", "GAM with Cubic Regression  
Spline of 1-day lag",
```

```
      "Linear Regression Model of 3-day lag", "GAM with Cubic Regression Spline of 3-day  
lag",
```

```
      "Linear Regression Model of 5-day lag", "GAM with Cubic Regression Spline of 5-day  
lag",
```

```
      "Linear Regression Model of 7-day lag", "GAM with Cubic Regression Spline of 7-day  
lag",
```

```
      "Linear Regression Model of 14-day lag", "GAM with Cubic Regression Spline of 14-day  
lag",
```

```
      "Linear Regression Model of 21-day lag", "GAM with Cubic Regression Spline of 21-day  
lag",
```

```
      "Linear Regression Model of 28-day lag", "GAM with Cubic Regression Spline of 28-day  
lag")
```

```
View(performance_model)
```