BIKE RENTING

Sudipto Mukherjee

01st December, 2018

Contents

[Introduction 3](#_Toc531387531)

[1.1 Problem Statement 3](#_Toc531387532)

[1.2 Data 3](#_Toc531387533)

[Methodology 4](#_Toc531387534)

[2.1 Data Pre-processing 4](#_Toc531387535)

[2.1.1 Missing Value Analysis 6](#_Toc531387536)

[2.1.2 Outlier Analysis 7](#_Toc531387537)

[2.1.3 Feature Selection 9](#_Toc531387538)

[2.1.4 Check for Multicollinearity 10](#_Toc531387539)

[2.2 Modeling 11](#_Toc531387540)

[2.2.1 Model Selection 11](#_Toc531387541)

[2.2.2 Linear Regression Model 12](#_Toc531387542)

[2.2.3 Decision Tree 15](#_Toc531387543)

[Conclusion 16](#_Toc531387544)

Appendix A - R Code………………………………………………………….17

Appendix B - Python Code…………………………………………………….19

Chapter 1

# Introduction

## **Problem Statement**

Ride sharing companies like Uber and Ola provide extremely convenient, affordable and efficient transportation options for customers without the hassle of owning a vehicle. However, with increasing number of automobiles, ride sharing in cars are not efficient especially during peak hours due to heavy traffic. Therefore, bike sharing is a great idea which provides people with a smarter mode of transportation without worrying about being stuck in traffic.

In this project we will be investigating the bike rental data with an objective of determining the count of bikes on the basis of the parameters thus provided to us.

## **Data**

Our task is to build a regression model based on the data which will give the count of bikes on a daily basis based on the environmental and seasonal settings.

A sample of the dataset is given below.

Table 1.1 Data (Columns 1 - 8)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Instant | Dteday | Season | Yr | Mnth | Holiday | Weekday | Workingday |
| 1 | 01-01-2011 | 1 | 0 | 1 | 0 | 6 | 0 |
| 2 | 02-01-2011 | 1 | 0 | 1 | 0 | 0 | 0 |
| 3 | 03-01-2011 | 1 | 0 | 1 | 0 | 1 | 1 |
| 4 | 04-01-2011 | 1 | 0 | 1 | 0 | 2 | 1 |
| 5 | 05-01-2011 | 1 | 0 | 1 | 0 | 3 | 1 |
| 6 | 06-01-2011 | 1 | 0 | 1 | 0 | 4 | 1 |
| 7 | 07-01-2011 | 1 | 0 | 1 | 0 | 5 | 1 |

Table 1.2 Data (Columns 9 - 16)

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Weathersit | Temp | Atemp | Hum | Windspeed | Casual | Registered | Cnt |
| 2 | 0.344167 | 0.363625 | 0.805833 | 0.160446 | 331 | 654 | 985 |
| 2 | 0.363478 | 0.353739 | 0.696087 | 0.248539 | 131 | 670 | 801 |
| 1 | 0.196364 | 0.189405 | 0.437273 | 0.248309 | 120 | 1229 | 1349 |
| 1 | 0.2 | 0.212122 | 0.590435 | 0.160296 | 108 | 1454 | 1562 |
| 1 | 0.226957 | 0.22927 | 0.436957 | 0.1869 | 82 | 1518 | 1600 |
| 1 | 0.204348 | 0.233209 | 0.518261 | 0.0895652 | 88 | 1518 | 1606 |
| 2 | 0.196522 | 0.208839 | 0.498696 | 0.168726 | 148 | 1362 | 1510 |

As seen from the above table, below are the predictor variables using which we will have to predict the count of bikes.

Table 1.3 Predictor Variables

|  |  |  |
| --- | --- | --- |
| **S. No** | **Predictors** | **Categories (if applicable)** |
| 1 | Instant | -- |
| 2 | Dteday | -- |
| 3 | Season | 1: Spring  2: Summer  3: Fall  4: Winter |
| 4 | Yr | 0: 2011  1: 2012 |
| 5 | Mnth | -- |
| 6 | Holiday | -- |
| 7 | Weekday | -- |
| 8 | Workingday | -- |
| 9 | Weathersit | 1: Clear, Few clouds, Partly cloudy, Partly cloudy  2: Mist + Cloudy, Mist + Broken clouds, Mist + Few clouds, Mist  3: Light Snow, Light Rain + Thunderstorm + Scattered clouds, Light Rain + Scattered clouds  4: Heavy Rain + Ice Pallets + Thunderstorm + Mist, Snow + Fog |
| 10 | Temp | -- |
| 11 | Atemp | -- |
| 12 | Hum | -- |
| 13 | Windspeed | -- |
| 14 | Casual | -- |
| 15 | Registered | -- |
| 16 | Cnt | -- |

Chapter 2

## Methodology

## **2.1 Data Pre-processing**

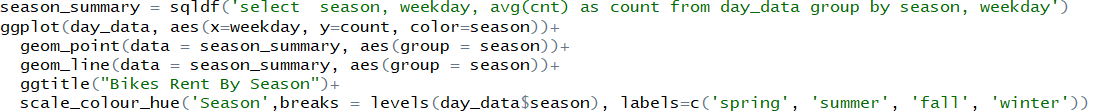
Data pre-processing is basically a data mining technique that involves transforming raw data into a format that is understandable. In here we explore the data, understand the meaning of each and every aspect of it including the variables and what each variable means and try to figure out how and in what way the variable will be influencing the target variable.

In the given dataset, as a part of data pre-processing, we remove the variables which do not contribute towards the target variable i.e. which have no role to play towards predicting the count of bikes on a daily basis based on the season. Those variables were, registered, casual, dteday and instant. Once we were done with the dimensionality reduction, we were left with 731 observations and 12 variables.

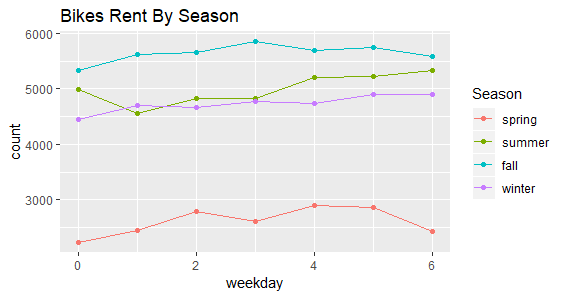
After that, we had a look the structure of the data set. It was observed that some variables like season, holiday, yr, etc. had the data categorized into several categories. So, as a part of exploratory data analysis, such variables were converted to factor and were treated as categorical variables throughout. Those variables were season, yr, holiday, workingday and weathersit.

On fetching the average count of bikes by season and weekday, it was found that the average count of bikes is highest in fall season and least in spring season.

Following is the code snippet to know the same.

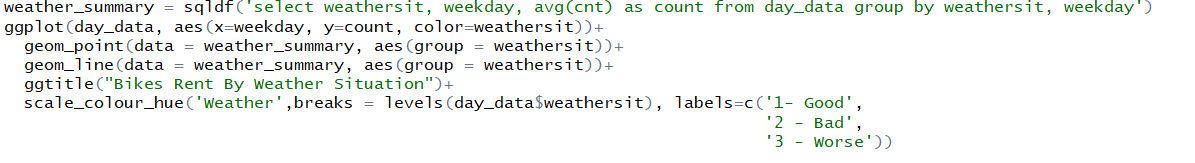


The output graph for the above snippet is the following.

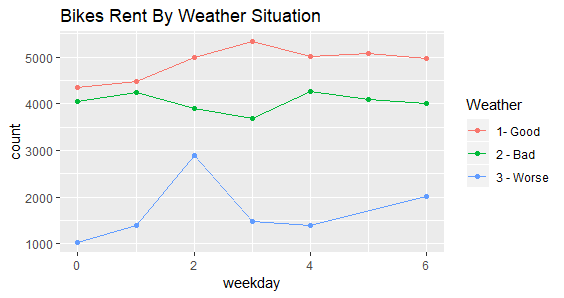


On fetching the average count of bikes by weather situation and weekday, it was found that the average count is the highest when the weather is clear with few clouds or partly cloudy.

Following is the code snippet to know the same.



The output graph of the above snippet is the following.



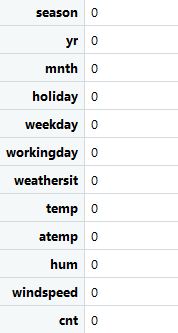
### **2.1.1 Missing Value Analysis**

The concept of missing values is important to understand in order to successfully manage the data. If the missing values are not handled properly, then inaccurate inferences may be drawn about the data. Also, due to improper handling, the result obtained will be different from the ones where missing values are present.

In the given dataset, no missing values were found. Following is the code snippet to know the same.



Following is the sample output obtained.

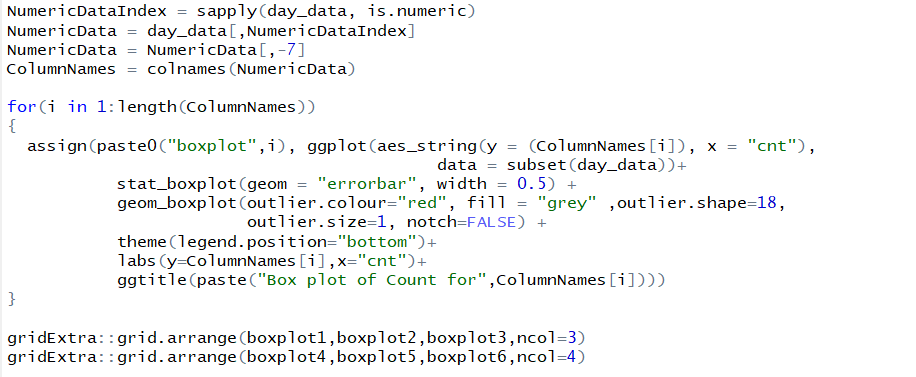


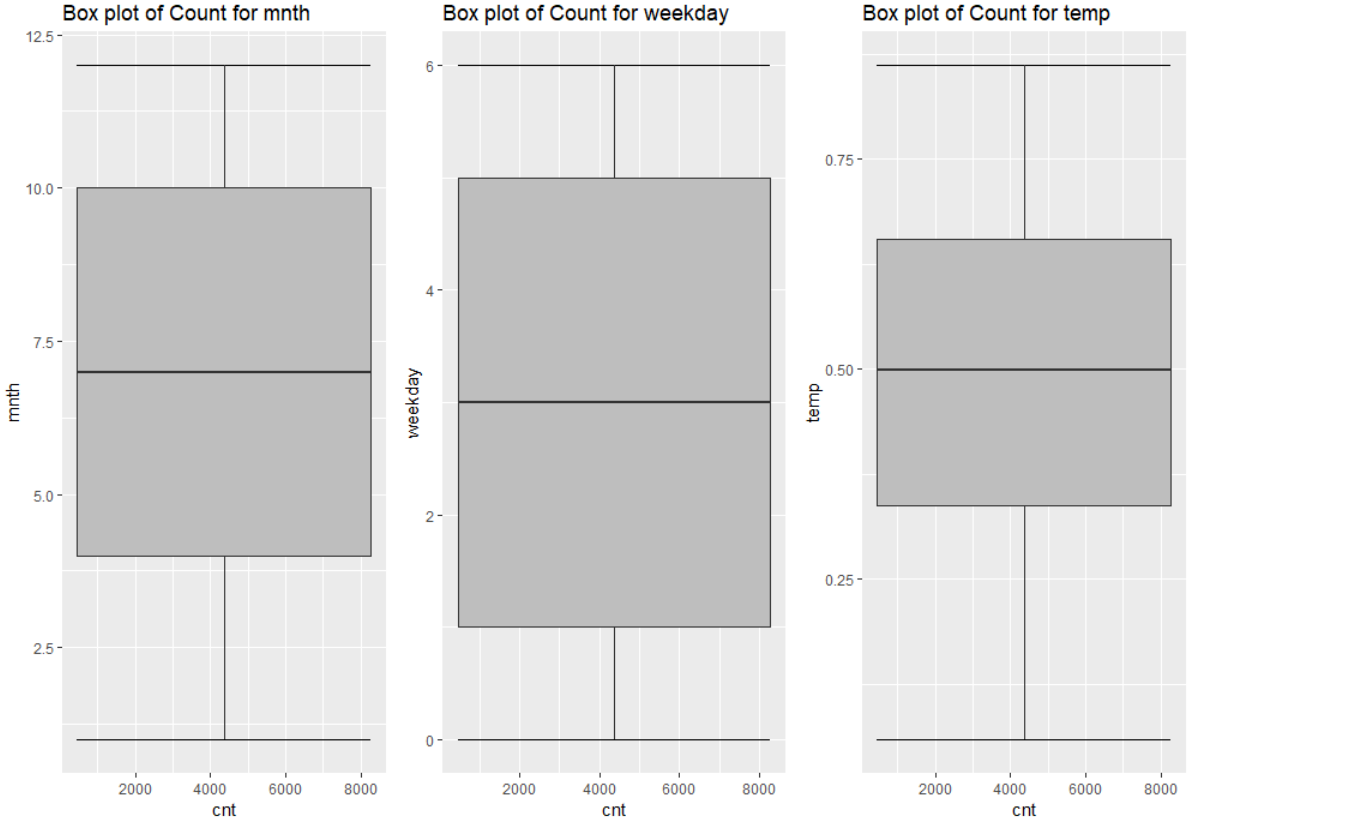
### **2.1.2 Outlier Analysis**

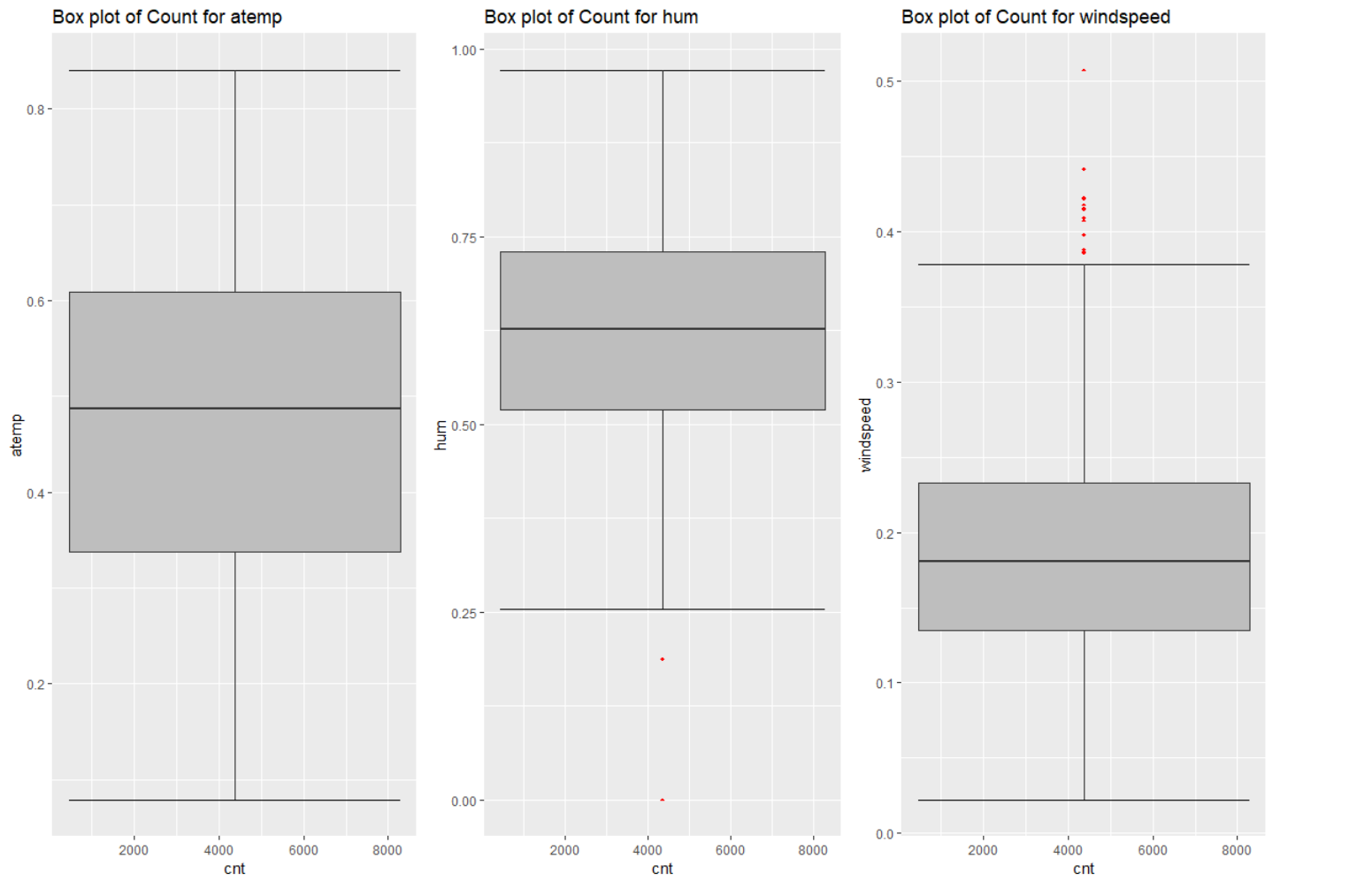
Outliers are the extreme values that fall a long way outside the other observations. Outliers in input data can skew and mislead the training process of machine learning algorithms resulting in longer training times and less accurate models. There has been a constant discussion on whether to remove the outliers or not. The outliers can be dropped in situations like if we have a lot of data and the sample won’t be hurt by dropping the questionable data or if we have an extremely good sense of within what range will the data of a particular fall, then we can drop it.

It is not recommended to drop an outlier if the results are critical like analysing the symptoms of a deadly disease. Also, in scenarios where the outliers constitute a major percent of the data, it is considered not to drop the outliers.

For the given data, we have plotted the boxplots for all the predictor variables. Following are the code fragment and corresponding box plots.







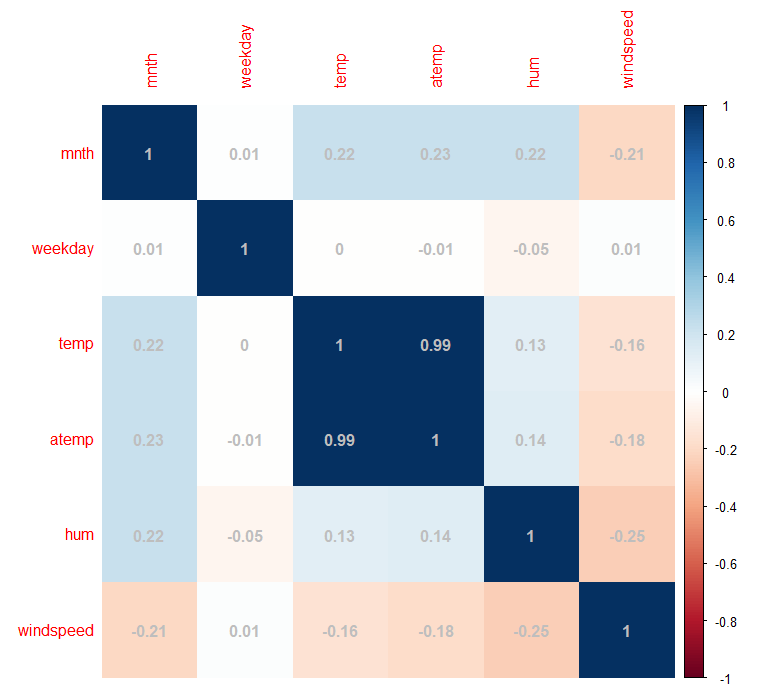
### **2.1.3 Feature Selection**

Feature selection basically refers to reducing the inputs for processing and analysis or of finding the most meaningful inputs. Data now almost always contains way more information than it is needed to build a model. And also, sometimes wrong kind of information. Not only feature selection improves the quality of the model but also makes the process of modelling more efficient. It also reduces the complexity of the model and makes it easier to interpret.

For the given data, we have used correlation plot to know about the correlation among the variables. Following is the code fragment for the same.



The correlation plot obtained for the data is the following.



From the plot is observed that temp is highly positively correlated atemp. Hence, we can remove any one of the aforementioned variables. For our analysis purpose, we have removed temp variable.

Following is the code fragment for the same.



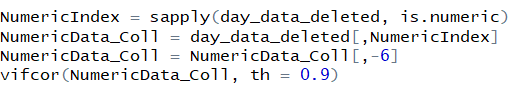
After deleting the temp variable, now are left with 731 observations and 11 variables.

### **2.1.4 Check for Multicollinearity**

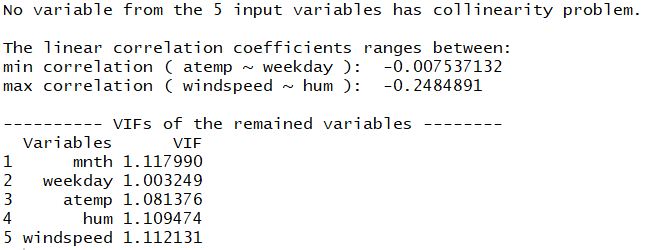
In regression, multicollinearity refers to predictors that are correlated with other predictors. It occurs when the model includes multiple factors that are correlated not just with the response variable but also with each other. So, basically it occurs when the independent variables are correlated. It is a problem since independent variables are suppose to be *independent.* If the degree of correlation is high enough, it can cause problems when we try fitting the model and interpret the results.

Since this is a regression model, we will check the model for multicollinearity. In here, we have opted the Variance Inflation Factor (VIF) to detect multicollinearity. It basically quantifies how much variance is inflated.

Following the code fragment for detecting multicollinearity.



The output of the aforementioned code fragment is the following.



As we can see from the output, there is no collinearity problem in any of the 5 input variables.

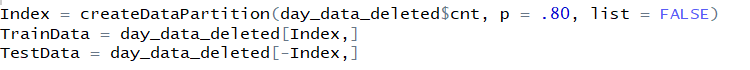
## **2.2 Modeling**

### **2.2.1 Model Selection**

From the problem statement and the dataset, we were able to deduce that the target variable is a continuous one and that it is a regression problem.

Before implementing the model, we had divided the data into train and test so that the model can be trained with the training data and the model thus trained can be implemented in the test data. In here, the training data set contains 80% of the total data and the test dataset contains the rest 20% of it.

Following is the code fragment for the same.



### **2.2.2 Linear Regression Model**

Linear regression model is a predictive model used to predict the value of a target variable on the basis of one or more input predictor variables. Its main aim is to model a continuous variable as a mathematical function of one or more predictor variables.

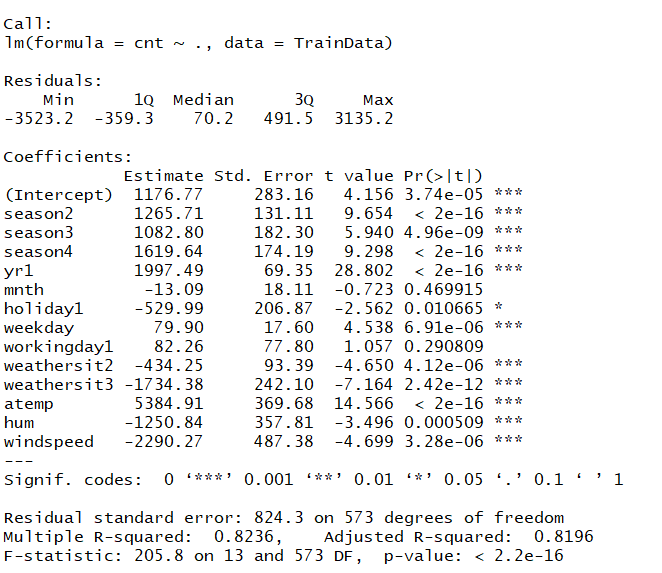
Following are the assumptions made in linear regression.

* There exists a linear and additive relationship between dependent (DV) and independent variables (IV). By linear, it means that the change in DV by 1-unit change in IV is constant. By additive, it refers to the effect of X on Y is independent of other variables.
* There must be no correlation among independent variables. Presence of correlation in independent variables lead to Multicollinearity. If variables are correlated, it becomes extremely difficult for the model to determine the true effect of IVs on DV.
* The error terms must possess constant variance. Absence of constant variance leads to heteroskedasticity.
* The error terms must be uncorrelated i.e. error at ∈t must not indicate the at error at ∈t+1. Presence of correlation in error terms is known as Autocorrelation. It drastically affects the regression coefficients and standard error values since they are based on the assumption of uncorrelated error terms.
* The dependent variable and the error terms must possess a normal distribution.

Following is the code fragment for implementing a linear regression model on the test data.

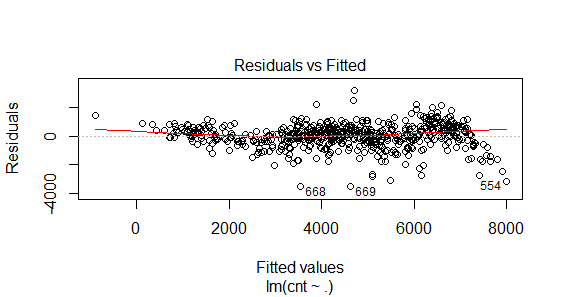


From the summary of the model we could find the following output.

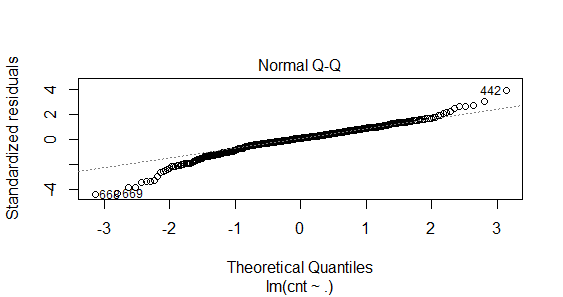


From the output we could see that the adjusted R-squared is 81.96% which says that the model explains 81.96% of the total variance in the data. The p-value obtained is also less than 0.5 which is agreement with the standard norms.

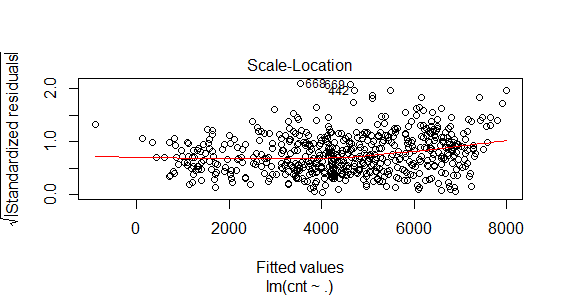
After this, we have plotted the residual plots to help us understand the pattern and derive actionable insights. Also, we can check if any of the assumptions are violated.

1. **Residual vs Fitted Plots**: Ideally, this ploy shouldn’t show any pattern. But if any particular shape is seen, it suggests non-linearity in the dataset. 

As we can see that there are no specific patterns seen in here. So, we can safely conclude that there is no non-linearity.

1. **Normal Q-Q Graph**: It is used to determine the normal distribution. Ideally, this plot should show a straight line. But if it shows a curved or distorted line, then the residuals have a non-normal distribution. 

As we can see that the graph shows a straight line and no curves. Hence, it shows normal distribution.

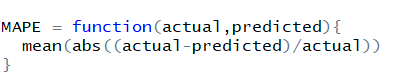
1. **Scale Location Plot:** Ideally, this should not show any pattern. 

As seen, it does not show any pattern.

Hence, from the above details, we can conclude that the dataset does not violate any of the assumptions of linear regression model.

For evaluating the performance of this model, we take the help of error metrics. Error metrics basically help us to know the evaluate the analytical model and also to know the recommendations and the business point of view. Since the problem is a regression model, we will go for Mean Absolute Percentage Error (MAPE). Also known as Mean Absolute Percentage Deviation, MAPE is the average absolute percentage error for each predicted value minus the actual value divided by the actual value.

Following is the function for the same.



Now, from the mean absolute percentage error it was found out that the error rate of our model is 19.46% and the accuracy is 80.54%.

Following is the code fragment for the same.



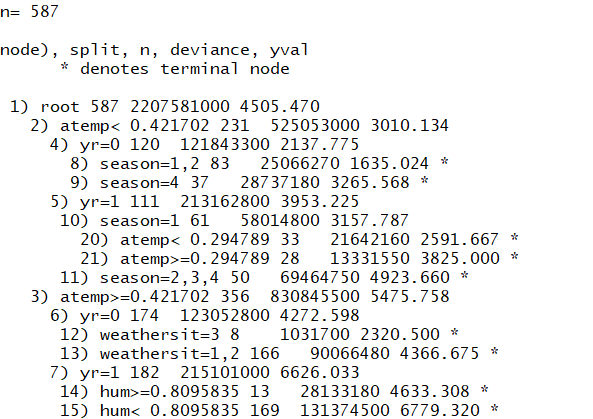
### **2.2.3 Decision Tree**

It is a predictive model based on the branching series of Boolean tests. It is a type of supervised machine learning algorithm. Its general motive is to create a training model which can be used to predict the class of the target variable by learning the distinct roots inferred from the historical data.

For the data provided, we have plotted decision for regression analysis. Following is the code fragment for the same.



Following is the sample output obtained.



On applying the model to test data and evaluating the performance, it was found that the error rate was 27.07% and the accuracy was 72.97%.

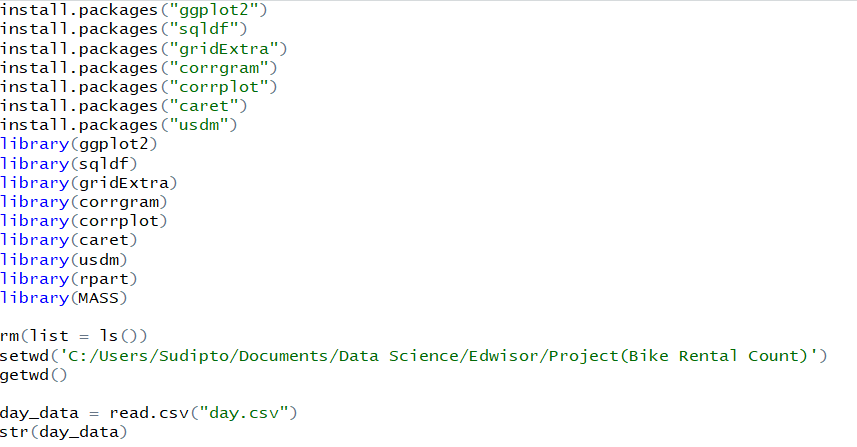
Chapter 3

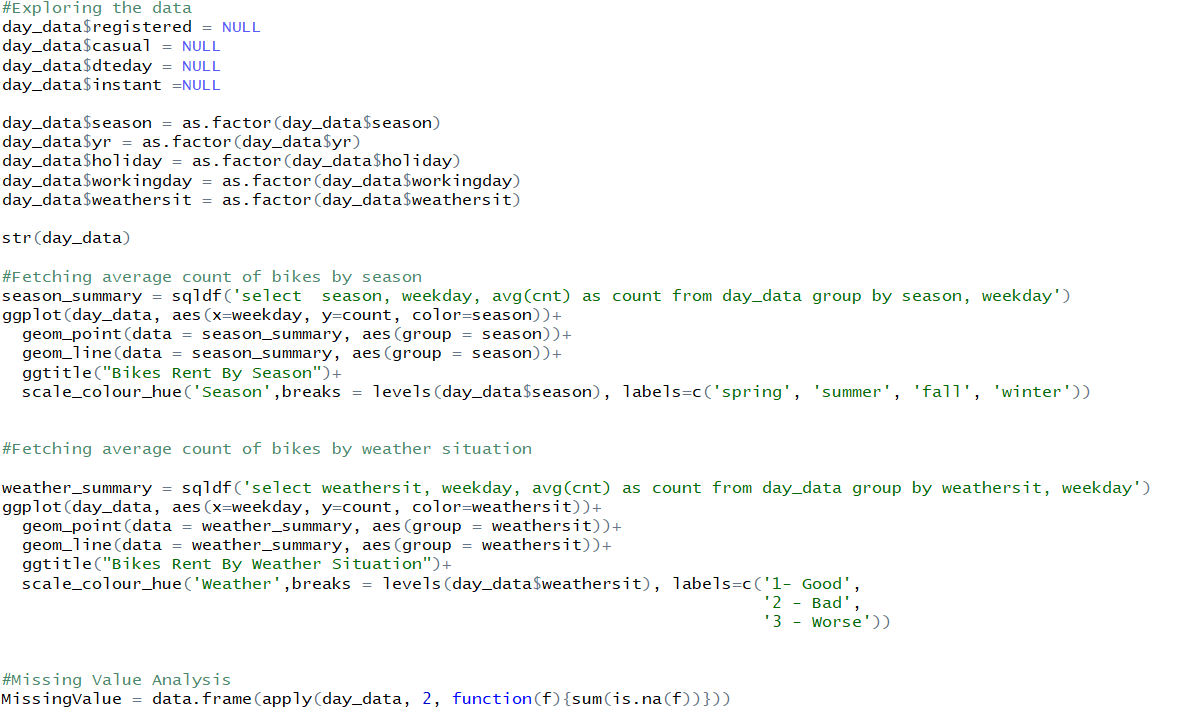
# Conclusion

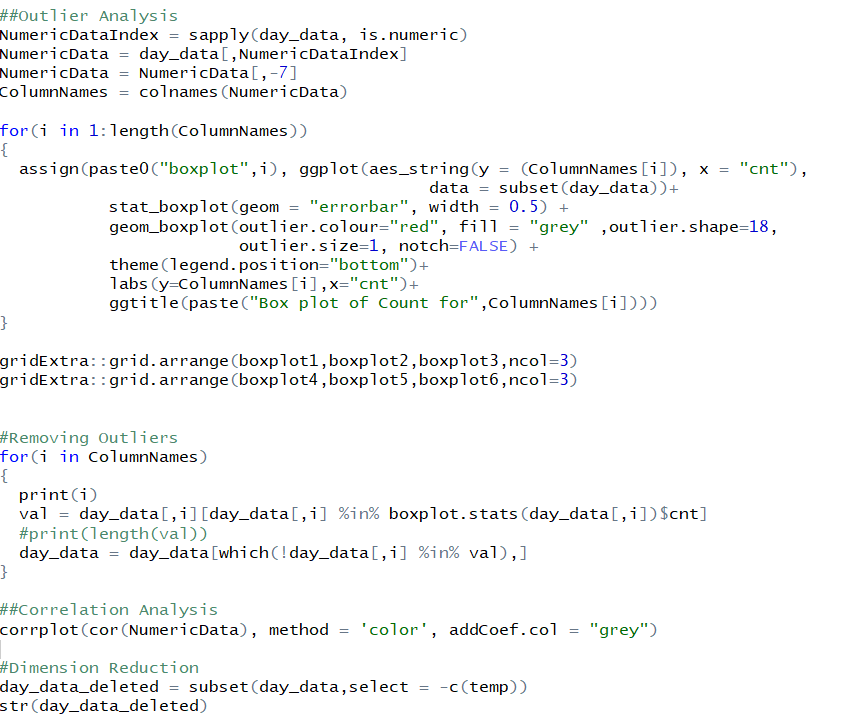
Now that we have a few models, we need to decide which one to choose. Logically, we need to select a model which has the highest accuracy rate.

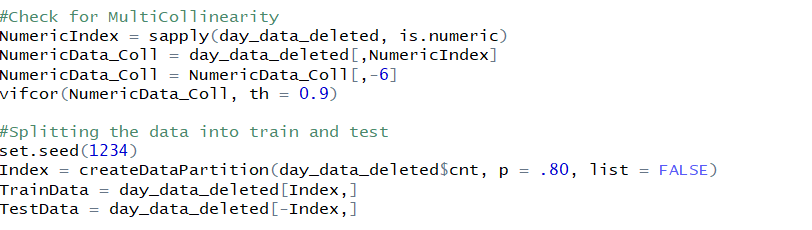
So, from the performance evaluation of the models, we can conclude that Linear Regression Model can be selected as the final model.

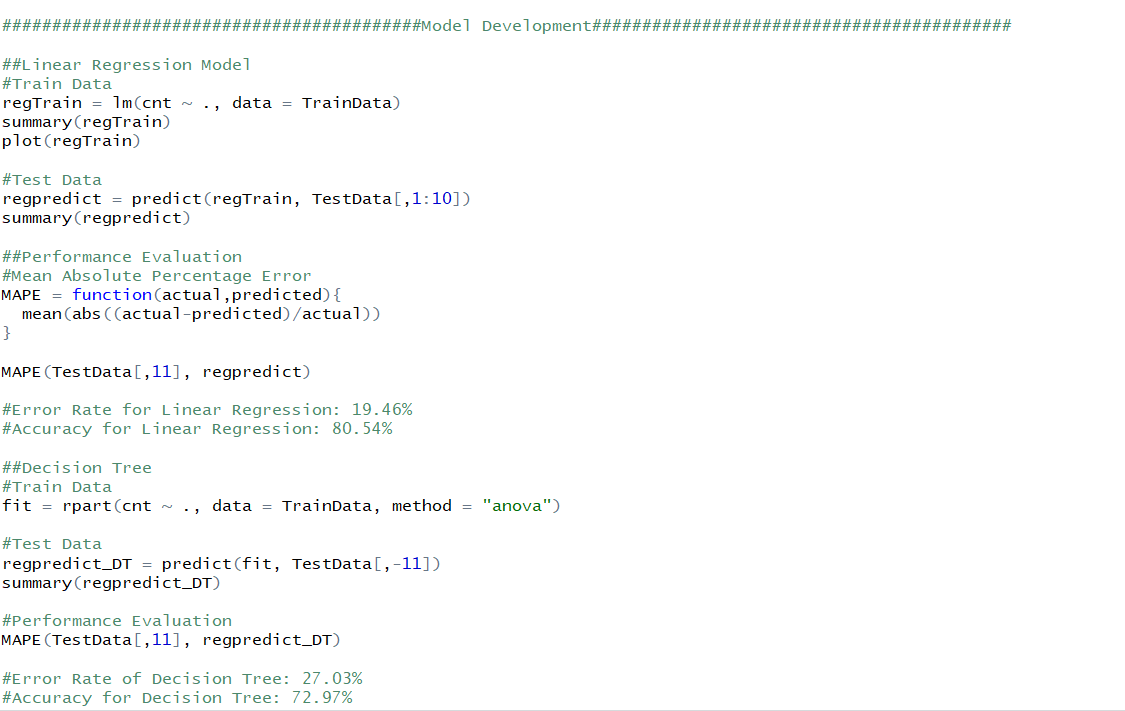
Appendix A – R Code











Appendix B – Python Code



