

**MGT7179 Advanced Analytics & Machine Learning**

Semester 2 – 2020/21

Assignment 1 - Regression

Vasileios Gounaris Bampaletsos

40314803

word-count

3395

Contents

[Abstract 3](#_Toc67701049)

[Indroduction 4](#_Toc67701050)

[Methodology 6](#_Toc67701051)

[Findings 21](#_Toc67701052)

[Conclusions 31](#_Toc67701053)

[References 32](#_Toc67701054)

[Appendix 33](#_Toc67701055)

# Abstract

Cycling improves movement problems in large cities and helps people exercise daily. Many cities around the world use the method of rented bicycles for the citizens of the city but also for the tourists. It is important that these bikes are available and affordable throughout the day so that there are no shortages of bicycles and waiting for customers. Proper forecasting of needs per hour will solve any problems that arise in the supply chain. The forecast will be implemented by utilizing and analyzing the data provided. This work studies some models for predicting the hourly need for bicycles. The data used relate to the weather (temperature, humidity, wind speed, visibility, solar radiation, snowfall, rainfall), calendar features such as time, seasons, holidays, months and functioning day and finally the number of rented bicycles per hour. The relationships between the variables were also studied as well as any problems and errors in the data. Five statistical regression models were used to find the most accurate prediction (Linear Regression, Support Vector Machine, Gradient Boosted Machine, Lasso, Ridge, Boosted Trees (Xgboost) .The best implementation of all was the Boosted Trees(XGBoost) which gives the better and higher r2 and lower RMSE & MAE .Many different implementations were made on the models in order to have a variety of effects, in order to find the best one for the problem.

**Keywords:** rented bikes, regression, bike sharing, predict

# Indroduction

The work focuses on the idea of ​​bike-sharing. Bike-sharing is a free (most of the time) service, in which users have the opportunity to use a bicycle for a certain period of time and leave it at a certain point for the next use by another user. It has existed for many years in major cities and not only in the world. It helps the citizens to move ecologically and quickly while doing exercise. The traffic problem in big cities has pushed governments and local communities to find alternative ways to get around on a daily basis. This feature also helps tourists to discover a larger part of the city they are visiting, in a shorter period of time in relation to the feet. Bike-sharing has a significant impact on the daily lives of citizens but mainly in the microclimate of each city as it improves many areas, from traffic to the psychology of citizens. It further strengthens the environmental idea and adaptation of each city, significantly reducing toxic gases from the use of the car.

The idea of ​​using the bicycle in the daily movement of citizens started in Amsterdam, the Netherlands. The first idea of ​​the common bicycle was born there, something that developed a few years later in the Nordic countries and specifically in Denmark, to which the first ways of sharing bicycles were applied ( (DeMaio, 2009)). Over the years this use of bicycles has increased. As reported (Pucher, et al., 1999) in the United States, and in many European countries including Denmark, Germany and the Netherlands, bicycle routes have increased by almost 20%. This enables users to have safe crossings without the risk of accident and collision with self-propelled vehicles. It also enhances the use of the bicycle even more, which brings many positives for each city or country.

So it seems that more and more countries are strengthening the fleet of shared bicycles, as well as the infrastructure for safer routes with them. Support and development in this area demonstrates the importance of this means of transportation. For the smooth operation of the whole project, analysis, monitoring and forecasting of the needs of a city or country on shared bicycles is required. So there must be a coordinated plan so that each city is ready to serve both the needs of its citizens for transportation and its tourists. Analyzing past data helps to understand and predict the demand for bicycles in the future.

This work analyzes data on communal bicycles in the city of Seoul in South Korea. Specifically, section 2 lists other scientific efforts that have been made to analyze bike-sharing data in various cities around the world. Section 3 presents the methodology used for the implementation of this study-study. Section 4 details the research findings and the explanation of the statistical machine learning models used. Finally, section 5 gives the results and important data of the research to be used in practice in the city of Seoul, while section 6 summarizes and completes the article with general comments and future improvements and uses of the study.

similar tasks

The analysis and forecasting of data is very important and useful in important areas of everyday life and science. Descriptive analysis of data using statistical methods gives researchers a general picture of the object they are studying. From it they learn how an object behaves in time. Then with the use of algorithms and engineering learning scientists can predict significant problems. For example, (Graham, et al., 2018) used machine learning and data mining to predict the situation in hospitals, the rate of admissions and visits to them. This is important because in addition to understanding and anticipation, the health system becomes better and more prepared to the needs of citizens. Also the science of analytics and data is used in the science of chemicals in order to predict the behavior and use of various substances in order to use them or not (Strempel, et al., 2013).

As far as bike-sharing is concerned, many research and analysis efforts have been made as a result of the importance of this bicycle function in the cities. One interesting study used city traffic data and bicycle frequency data in order to improve the organization of both bike-sharing and rent-bike businesses (Yang, et al., 2016). A study by Greek scientists (Tomaras, et al., 2018) focused on forecasting bicycle demand in a large city during a major event, such as parades, national holidays, concerts, where people and bicycle demand peak. The hourly demand per bicycle station has also been studied so that they are always functional and have a sufficient quantity of bicycles (Lin, et al., 2018). The studies each time use different statistical models of analysis and forecasting depending on the data and the goal they want to achieve. Like the (Feng & Wang, 2017) who made a prediction for the demand for rented bikes using multiple linear regression and random forest methods. It seems that each research has a different mix of methods and models depending on the desire of the researchers but also with the data they have. For example, (Pan, et al., 2019) used a more advanced method, that of neural networks to understand in detail all the relationships between variables that affect the demand for bike-sharing.

# Methodology

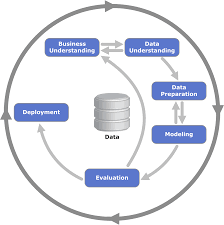
The CRISP-DM method was used to implement this project(Figure 1). This means that first the business problem that had to be solved was understood and then the goal that this project wants to achieve. The aim is to understand the data that affect the demand for bicycles in a city and to predict this demand in the future so that businesses and cities are ready for the cyclists concerned.

Figure 1: Crisp-Dm process diagram (https://www.google.com/search?q=crisp+dm&sxsrf=ALeKk01H22jQ7VZGaQejpMv1B0HosOGusA:1616777415908&source=lnms&tbm=isch&sa=X&ved=2ahUKEwiag7mctc7vAhXM4IUKHRGrAgAQ\_AUoAXoECAEQAw&biw=1920&bih=962#imgrc=m1Hh09d2zPNgBM)

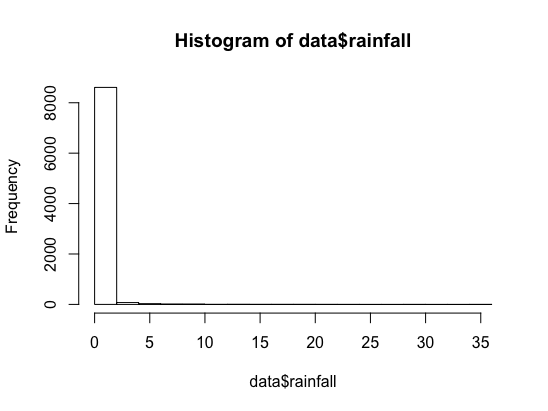
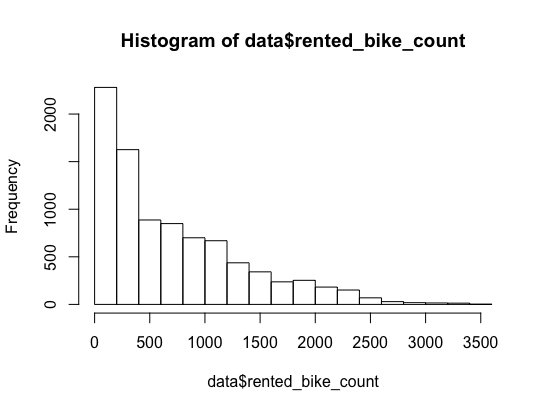
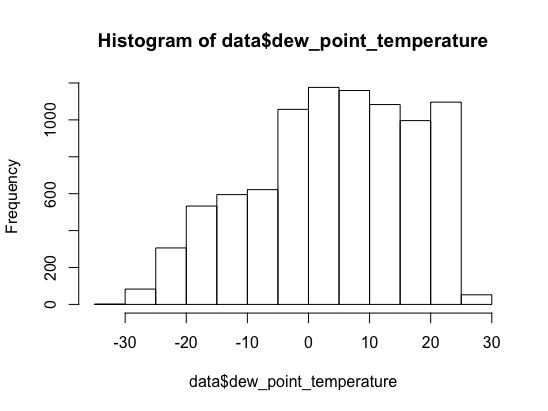
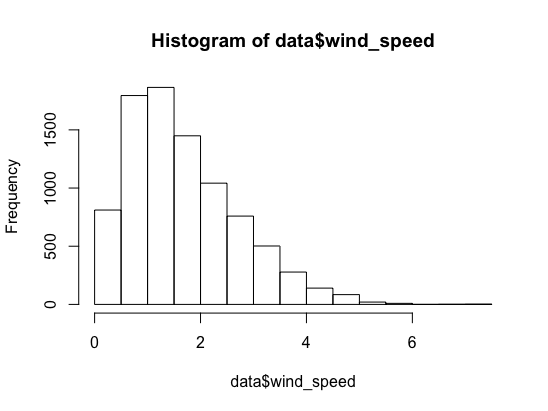
The dataset from Seoul has 8760 observations and 14 variables and the target variable is the “rented\_bike\_count”. It has 10 continues variables and 4 categorical variables.

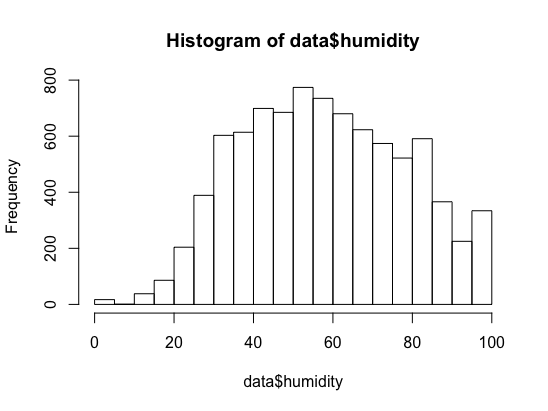
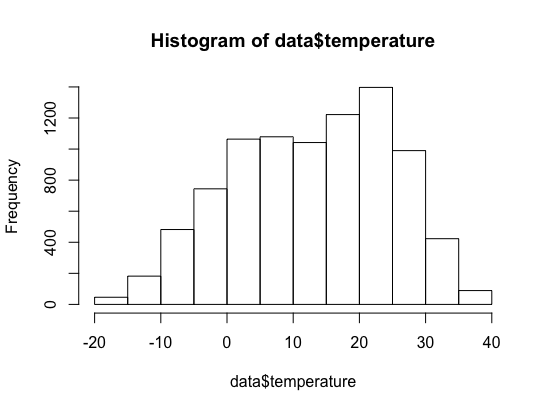
|  |  |  |
| --- | --- | --- |
| **Variables** | **Type** | **Measurement** |
| Date | Day/month/year | - |
| Rented\_bike\_count | Integer-count | 0-3556 |
| Hour | Continuous | 0-23 |
| Temperature | Continuous | Celcious |
| Humidity | Continuous | % |
| Wind\_speed | Continuous | m/s |
| Visibility | Continous | 10m |
| Dew\_point\_temperature | Continuous | Celcious |
| Solar\_radiation | Continuous | MJ/m2 |
| Rainfall | Continuous | Mm |
| Snowfall | Continuous | Cm |
| Seasons | Categorical | Winter, spring, summer, autumn |
| Holiday | Categorical | Holiday, no holiday |
| Functioning\_day | Categorical | No, Yes |

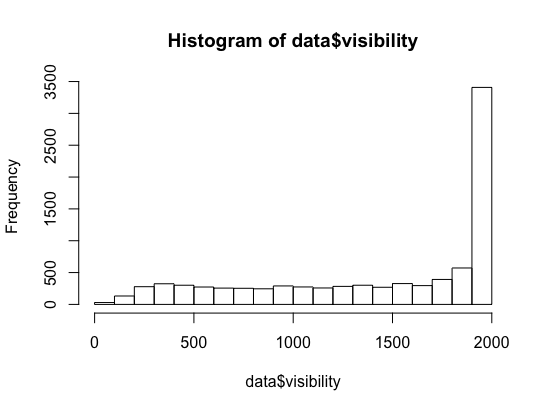
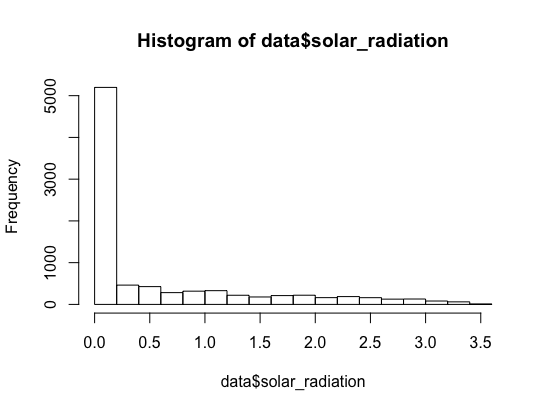
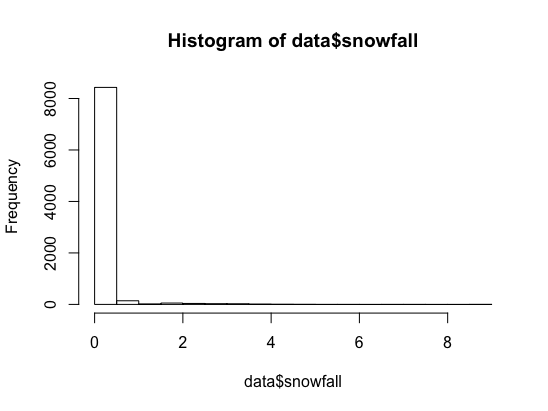
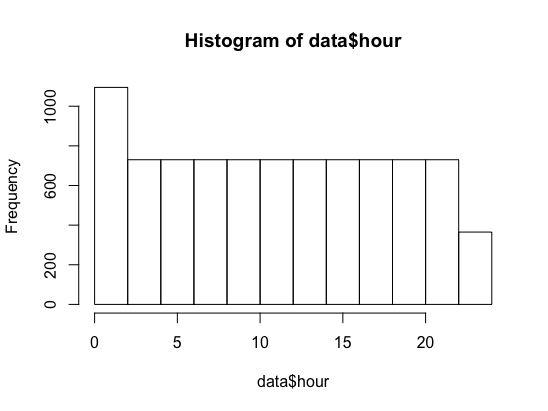
For this reason, the data from the city of Seoul were initially checked to find any errors and missing values. First are checked the continuous variables with the function summary() in R and also made histograms to visualise the raw continuous data.

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **Continuous Variable** | **Count** | **Missing Values** | **Min.** | **1st Quartile** | **Mean** | **Median** | **3rd Quartile** | **Max.** |
| Rented\_bike\_count | 8760 | 0 | 0 | 191 | 704.6 | 504.5 | 1065.2 | 3556 |
| Hour | 8760 | 0 | 0 | - | - | - | - | 23 |
| Temperature | 8760 | 0 | -17.8 | 3.5 | 12.88 | 13.7 | 22.5 | 39.4 |
| Humidity | 8760 | 0 | 0 | 42 | 58.23 | 57 | 74 | 98 |
| Wind\_speed | 8760 | 0 | 0 | 0.9 | 1.725 | 1.5 | 2.3 | 7.4 |
| Visibility | 8760 | 0 | 27 | 940 | 1437 | 1698 | 2000 | 2000 |
| Dew\_point\_temperature | 8760 | 0 | -30.6 | -4.7 | 4.074 | 5.1 | 14.8 | 27.2 |
| Solar\_radiation | 8760 | 0 | 0 | 0 | 0.5691 | 0.01 | 0.93 | 3.52 |
| Rainfall | 8760 | 0 | 0 | 0 | 0.1487 | 0 | 0 | 35 |
| Snowfall | 8760 | 0 | 0 | 0 | 0.0750 | 0 | 0 | 8.8 |

The histograms are very important to see the continuous variables for the first time the data and understand them better.

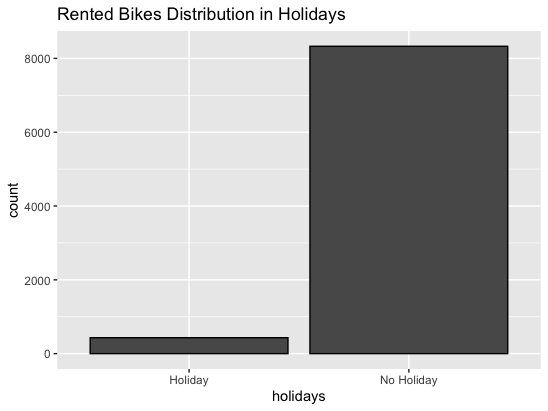


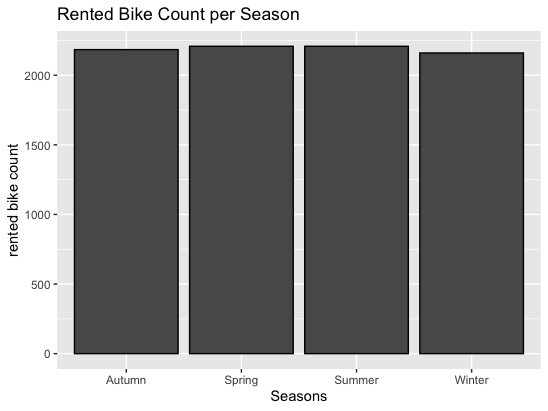
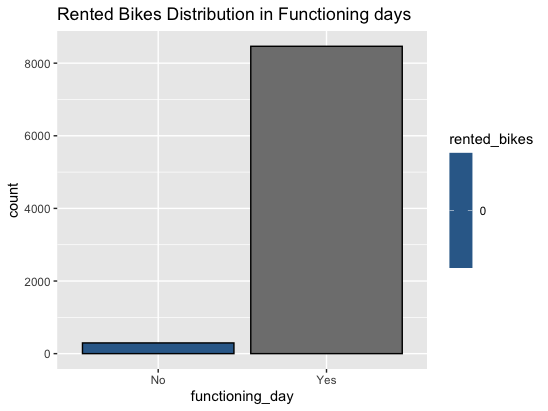




Next step is to see and visualize the categorical variables.

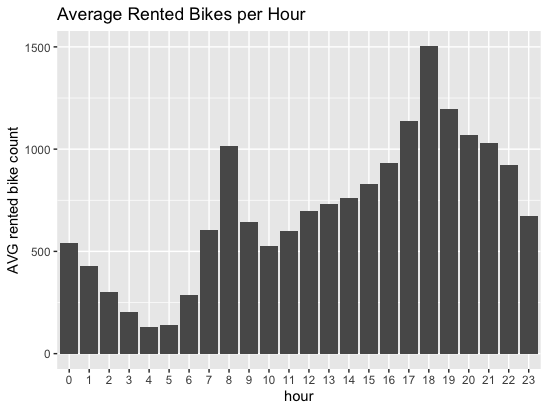
|  |  |  |  |
| --- | --- | --- | --- |
| **Categorical Variables** | **Count** | **Missing Values** | **Summary** |
| Date | 8760 | 0 | 01/12/2017 – 30/11/2018 |
| Seasons | 8760 | 0 | Autumn: 2184  Spring: 2208  Summer:2208  Winter:2160 |
| Holiday | 8760 | 0 | Holiday:432  No Holiday:8328 |
| Functioning)day | 8760 | 0 | No:295  Yes:8465 |

After the summary of the categorical variables it is useful to visualise them.

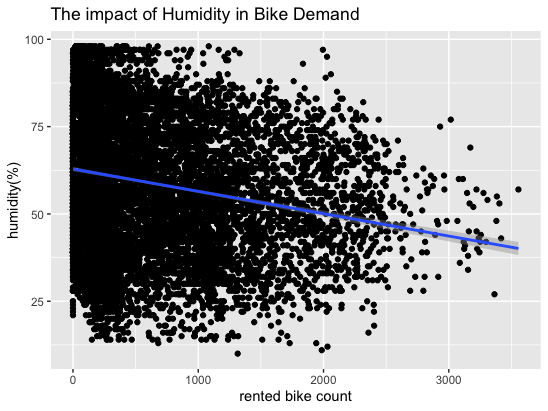


Once the data was presented and visualized for better understanding, no errors were observed. The only intervention in the data is that from the variable date, the variable month was created. The reason for this was because the month is considered a very important factor for our model, in order for the project to be able to provide data per month for the preparation of businesses in the future. Also the variable date was deleted from the dataset.

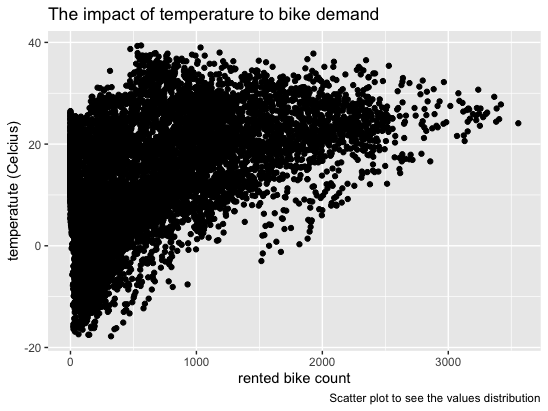
The next step of the CRISP-DM method is to understand the past of the data so that the customer (business, city) understands the demand for bicycles and what variables affect it. The data contain weather events as well as calendars. Thus 9 graphs were created which will help in understanding the data before using the machine learning models for the predictions in the future.



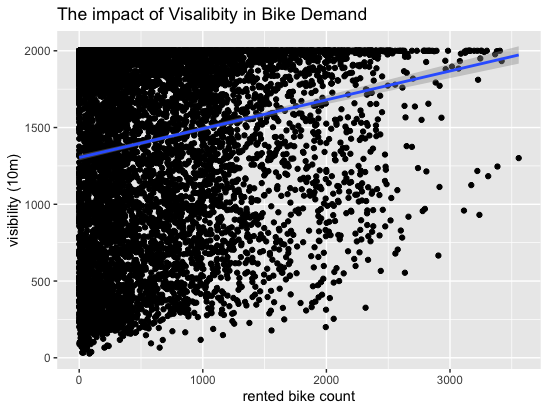
Graph 1: Average Rented BikeS per Hour (source: my Rcode)



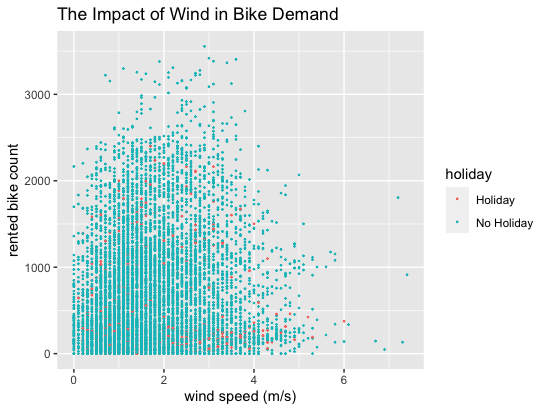
Graph 2: The Impact of Humidity in Bike Rentals (source:my Rcode)



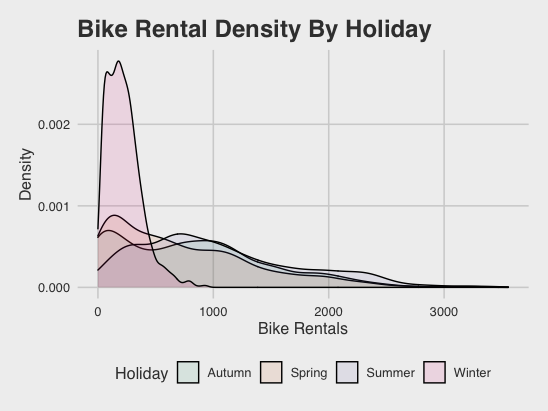
Graph 3: The Impact of Temperature in Bike Demand(source:my Rcode)



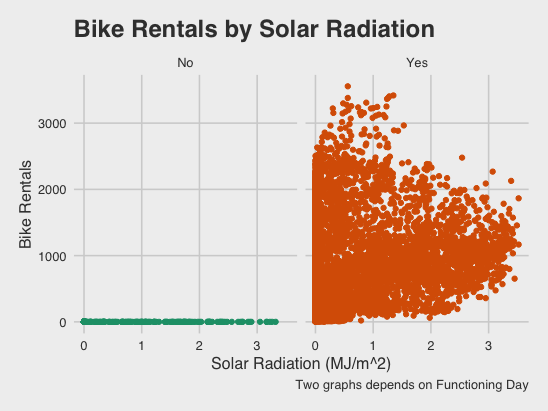
Graph 4: How visibility affects bike demand (source: my Rcode)



Graph 5: The Impact of Wind in Bike Rentals (with Holiday values) (source: my Rcode)

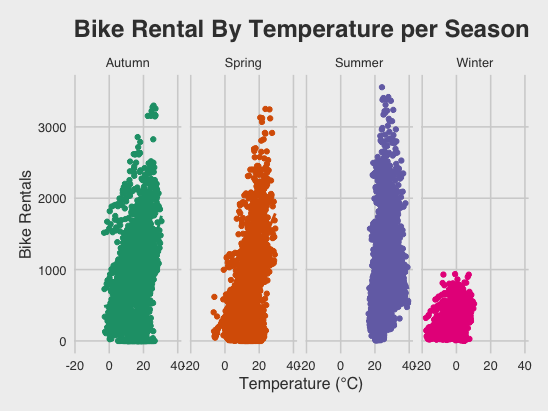


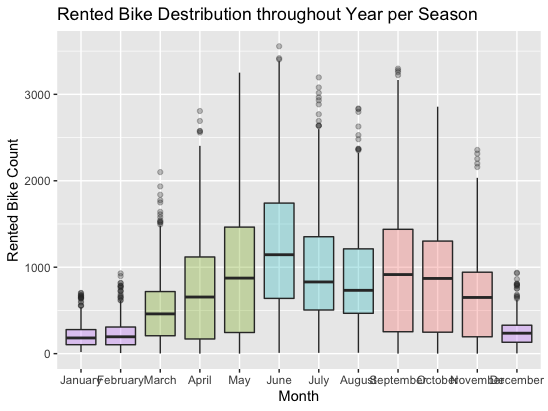
Graph 6: The Density plot of Seasons (source: my R code)



Graph 8: Bike Demand per Temperature by Season (source: my Rcode)

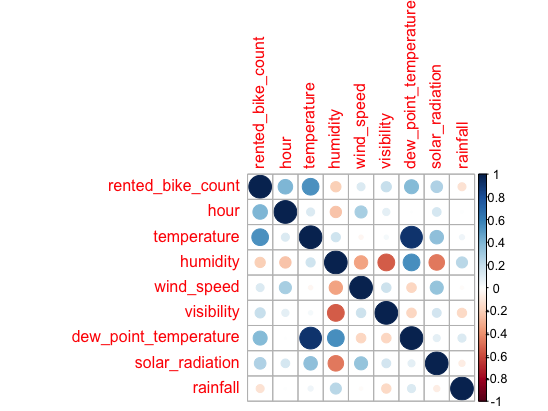
Graph 7: The Impact of the Sun in Bike Rentals (source:my R code)



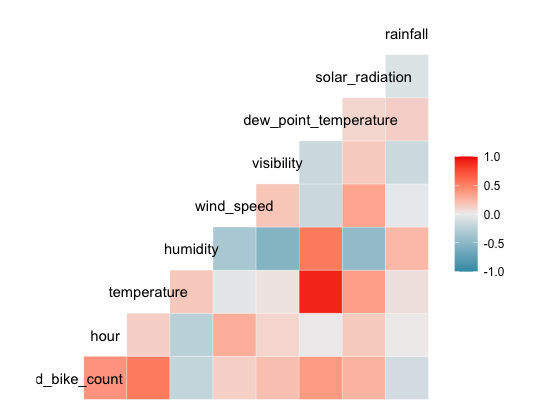


Graph 9: Rented Bike Distribution throughout Year per Season (source: my Rcode)

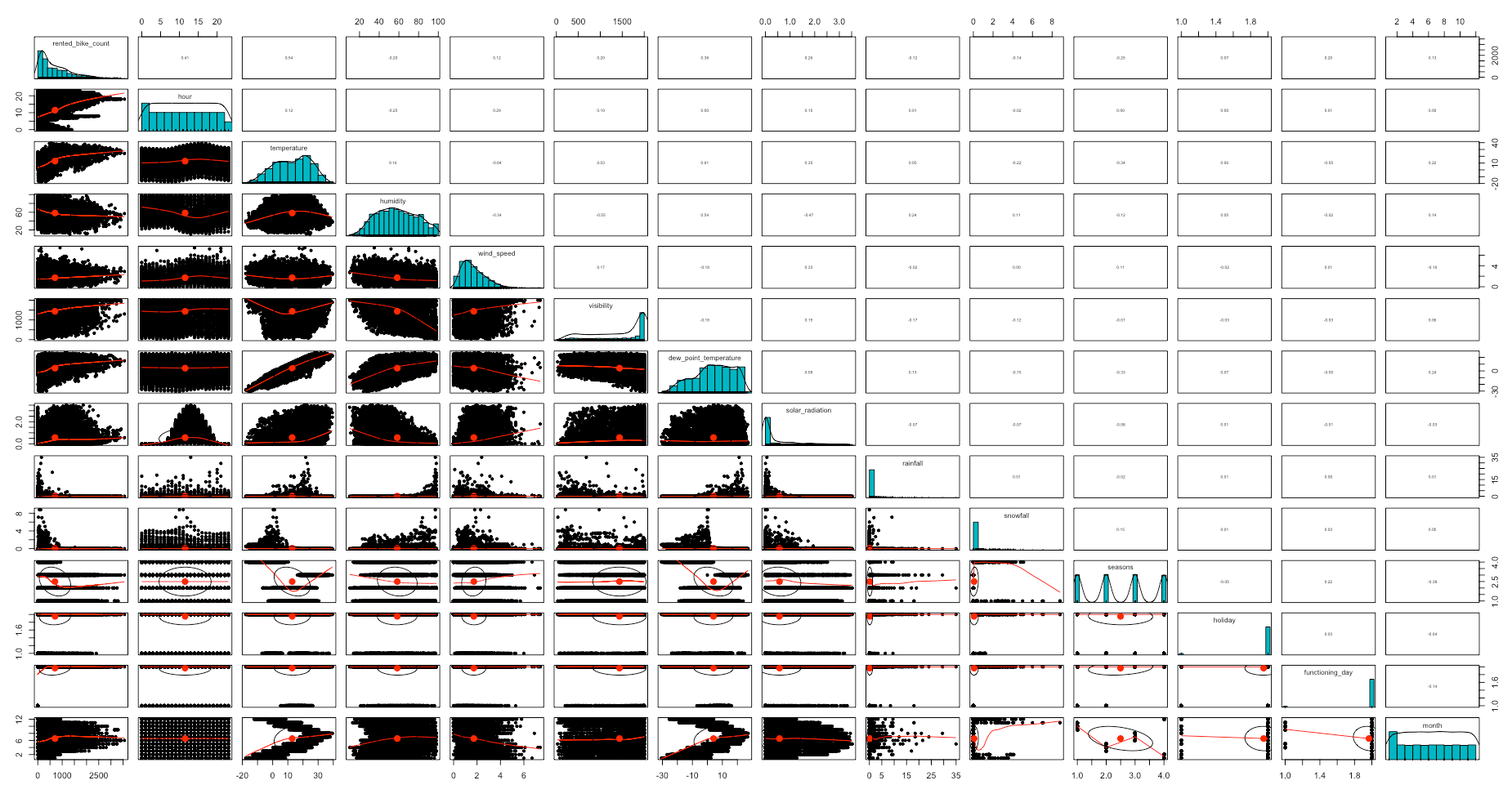
Completing the visualization of the data, the series has the deepest understanding. Correlation matrices help in this step because they give us a correct picture of the correlations between the variables so that later correct machine learning models can be built. For this reason 4 tools were used to find the pearson correlations. A complete correlation table (Correlation 1), a half correlation table (Correlation 2) and a pair plot (Correlation 3), to find the correlations between the continuous variables.



Correlation 1: Full Correlation Matrix (source: my R code)



Correlation 2: Half Correlation Matrix (source: my R code)



Correlation 3: Pair Plots (Source: my R code)

Pearson's chi-squared test for the correlations of the categorical variables with the target variable (rented\_bike\_count).

|  |  |  |  |
| --- | --- | --- | --- |
| Variables | X-squared | Df | p-value |
| Seasons | 9625 | 6495 | <2.2e-16 |
| Holiday | 1838 | 2165 | 1 |
| Functioning\_day | 8760 | 2165 | <2.2e-16 |
| Month | 27796 | 23815 | <2.2e-16 |

The tools for finding the correlations between the variables were very useful and the results will be discussed in the next section.

The data was then split into train\_set and test\_set. The ratio used was 75-25, with 75% for train\_set and 25% for the test\_set. This separation was made because our data was enough and the ratios 70-30, 75-25, 80-20 fit them better, so that the models that will be created give reasonable predictions with as small errors as possible. By giving a large amount of data to train\_set we helped to properly train the model predictions from the old values ​​of the target variables that contain our data.

|  |  |  |
| --- | --- | --- |
| Dataset | No. Observations | No. Variables |
| Train\_set | 6830 | 14 |
| Test\_set | 1930 | 14 |

Create the Models

First you had to choose the model of variables to be used. The final model was rented\_bike\_count with all variables except the dew\_point\_temperature variable (more details in the next section).

Then 6 engineering learning models were selected which will help to deliver good forecasts for the demand of bicycles per hour. The choice of models was made so that models from different techniques and algorithms could be used. As a result, there is the possibility of more accurate evaluation of the models in order to find the most appropriate for the data used. The models used are Linear Regression (LM), Support Vector Machine (SVM), Ridge, Lasso, Gradient Boosted Machine (GBM), Boosted Trees (BT).

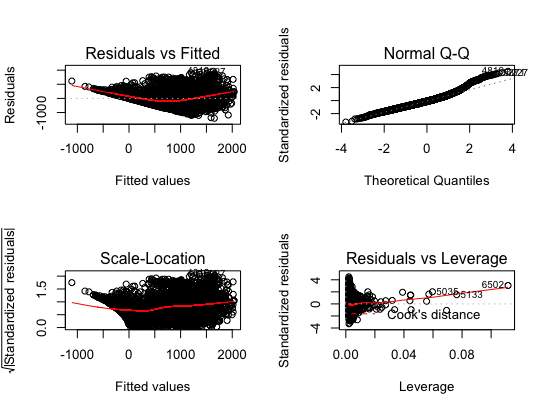
In the models are used some evaluation and tuning methods to help the models to make better predictions. Cross-Validation method used in some of the models(LM,Lasso, GBM, BT). This method used in machine learning models and split the training set as many time we want(k). This helps the whole model to train and understand better the variables from training set. Also provides better results.

|  |  |  |
| --- | --- | --- |
| Model | Formula | Tuning Method |
| Linear Regression | lm(Rented\_bike\_count ~ . -dew\_point\_temperature) | Cross validation(k =10) |
| Support Vector Machine | svm(Rented\_bike\_count ~ . -dew\_point\_temperature) | - |
| Ridge | (Rented\_bike\_count ~ . -dew\_point\_temperature) | Grid Search for Lambda |
| Lasso | (Rented\_bike\_count ~ . -dew\_point\_temperature) | Cross Validation |
| Gradient Boosted Trees | gbm(  formula = rented\_bike\_count ~ . -dew\_point\_temperature) | GBM1: shrinkage 0.001, 10000 trees, Cross validation  GBM2:shrinkage 0.1, 10000 trees, cross validation |
| Boosted Trees | (Rented\_bike\_count ~ . -dew\_point\_temperature) | XGBoost method  Cross Validation |

Also in Ridge was used the grid search method. This method is using to search the best value of a hyperparameter in a subset. In the Ridge model was used the lambda value. The grid search was used to find the best value of lambda and as result the model performed better and with very good results.

The assumptions for the linear model were also checked in order to better understand the data and see if the assumptions are violated or not.

|  |  |  |
| --- | --- | --- |
| Assumptions | Checking method | Results |
| Mean of residuals | Mean(model$residuals) | -1.333707e-14  No violation of assumption |
| Homoscedastictity of residuals | Graph below | Residuals fit is very good(top left graph) |
| Normality of residuals | Normal q-q | The data fit good to the line(Graph top right) |
| Infuential values | Influential.measures(model)  Cook’s distance | All variables are good,  Cook’s distances(bottom right graph)- no variable to violate this distance |
| multicolinearity | Variance inflication factor  (vif) | The continuous variables violate the assumpition(gvif>2) |



*Evaluation*

Having implemented and executed all the engineering learning models in order to predict the demand of bicycles per hour, at this point you have to evaluate the models and find the best one for the data used. The evaluation of the models was done using 3 measurements, Root Mean Squared Error (RMSE), Mean Absolute Error (MAE), Rsquared (R ^ 2).

RMSE is the standard deviation between the current values ​​of the target variable and the values ​​that will be predicted for this variable. This value connects the values ​​that the model has in relation to the predicted ones and finds the error. A good value of RMSE is considered to be that which tends to zero and this is because it represents the error in the predicted price.

R ^ 2 is the value that determines the fit of the model in the data and is presented from 0 to 1. So it shows how good the model is for the data given to it as input. The larger (tends to 1) the R ^ 2 the better the model fits the data.

MAE is almost identical to the RMSE, but differs from it in that it aims to find the error values ​​of its predictions by avoiding the hedge between the positive and negative values ​​of the errors. Like the RMSE, the MAE must remain as small as possible.

Therefore, the best model after being evaluated, should have as little MAE and RMSE as possible and as much R^ 2 as possible.

# Findings

In this section, the data from the analysis of the data used as well as the graphs will be analyzed in order. Then the models that were created will be analyzed and the best will be found in order to answer the question and the goal of this project. The question is what characteristics affect the demand for bicycles in a city and specifically in Seoul and the goal is to give the city an important tool for forecasting bicycle demand so that it is ready for any demand on the part of bicycle consumers.

*Exploratory data analysis*

*Visualizations*

As shown in the previous section, the data is in excellent condition, regarding missing values ​​and outliers. In all the variables there are correct limits which do not need any processing as it appeared from the initial visualizations using the histograms but also the basic statistical data given. The 9 graphs that were implemented give a very clear picture of the data and in combination with the correlation tables complete the understanding of the data.

As for the graphs, a table with their explanation follows.

|  |  |
| --- | --- |
| Visualizations | Explanation |
| Graph 1: Average Rented BikeS per Hour (source: my Rcode) | This graph shows the average number of bicycles used per hour. What is immediately noticeable is that the largest volume of demand is concentrated from the noon hours, culminating in the afternoon and at 6 o'clock. This is justified because in those hours people were returning from work and tourists were exploring the city. There is also a significant demand at 8 in the morning, which means that many people use bicycles to get to work. |
| Graph 2: The Impact of Humidity in Bike Rentals (source:my Rcode) | This graph shows the strong influence of humidity on bicycle demand. As long as the humidity is at low to normal levels (25% -75%) the demand is high. Expected result as the increased humidity is accompanied by either rain or excessive heat in places close to the sea such as Seoul, conditions that make cycling impossible. |
| Graph 3: The Impact of Temperature in Bike Demand(source:my Rcode) | It is observed that high temperatures prevent the use of bicycles, just like very low temperatures. It seems that the greatest demand is between 10 and 30 points. |
| Graph 4: How visibility affects bike demand (source: my Rcode) | Visibility is another important factor that affects bicycle use. It seems, as expected, that the better the visibility, the greater the demand for bicycles. The highest concentration is observed above 10 km of visibility. |
| Graph 5: The Impact of Wind in Bike Rentals (with Holiday values) (source: my Rcode) | Wind speed is also important for bicycle use. In cities with high wind speeds the use of a bicycle will not be possible as shown in the graph. And this is because the greatest demand for bicycles is observed when the air speed is from 3 meters per second and below. |
| Graph 6: The Density plot of Seasons (source: my R code) | An interesting and very detailed graph, which shows the area of the most frequent occurrences of each variable. It seems that the demand for bicycles in winter is small as the prices of the variable are between 0 and 500 bicycles. The other prices (spring, summer, autumn) have a wider range of prices. Which is justified by the weather conditions. |
| Graph 7: The Impact of the Sun in Bike Rentals (source:my R code) | Sunshine is an important factor in bicycle use and this is reflected in the graph. However, the very sunshine does not seem to favor the use of a bicycle and this is because summer is combined with high temperatures. So between 0-1 (MJ / M ^ 2) there is the greatest demand for bicycles. On the other hand, it does not seem to affect whether it is a functioning day or not as there is not much data. |
| Graph 8: Bike Demand per Temperature by Season (source: my Rcode) | This graph shows the demand for bicycles by season and temperature. There is a small demand in winter, as mentioned before, and the upward and frequent use of bicycles in the rest of the seasons when the temperature is tolerable. From this we understand that in Seoul the climate is good these 3 seasons and allows activities outside. |
| Graph 9: Rented Bike Distribution throughout Year per Season (source: my Rcode) | As in the previous graph, so in this it seems that the demand during the 3 months is quite high compared to the winter. June is characterized as the best month, followed by September and May. It is therefore observed that in summer (July, August) in Seoul there is no huge demand for bicycles. |

*Correlations*

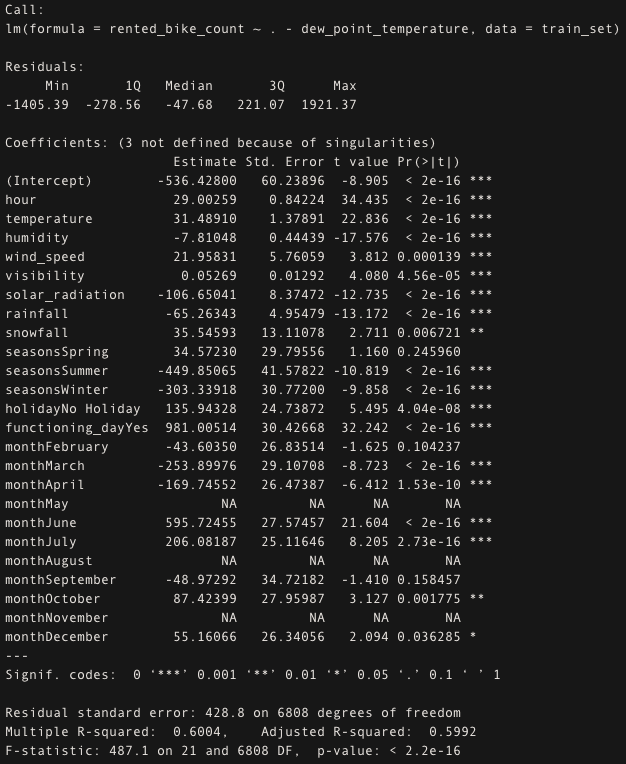
The two correlation tables (Correlation 1,Correlation 2) show the correlations between the continuous variables. The values ​​that the correlations get are between -1 and 1 (negative correlation, positive correlation). Also from the pair plot (Correlation 3) are the histograms in the diagonal, above it are the values ​​of the correlations between the variables, while below the diagonal are the bivariate scatter plots with the red line being the linear regression between the variables corresponding to each box. The most important correlation (positive) is between the number of bicycles and the temperature, which means that as the temperature rises so does the demand for bicycles. There is also a significant positive correlation between time and bicycle use. Significant positives are also the correlation of the number of bicycles with the air, visibility, dew point and sunshine. On the other hand, the important negative correlations have the humidity, the rain, the snow with the number of bicycles. This means that as they fall, the demand for bicycles increases. So we see that the weather conditions directly affect the demand for bicycles, so a first conclusion is that in cities where the weather conditions are not good, the use of bicycles is not indicated. Finally the temperatures with the dew point are almost the same variables as they measured the same thing. therefore the dew point will be excluded from the models that will be implemented because it will affect the performance and consequently the forecasts.

On the other hand the categorical variables, seasons, functioning day, month had very strong correlation with the rented bike count.

*Machine Learning Models*

Linear Regression

For the linear regression model, tests were performed so that the significance of all variables is large and R ^ 2 is relatively high (from 0.6 and above). At the same time, the dew temperature variable was exclusive. Thus the elements of the first model, as shown below, show a good model in terms of significance but with a weak R ^ 2. Significance means that almost all of our variables contribute to the model.



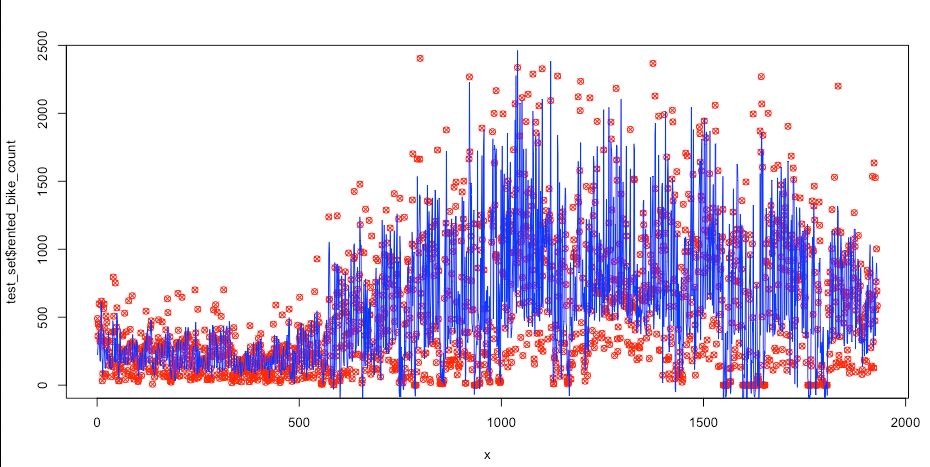
As for R ^ 2 this will be improved by using cross validation. So after using it we have the following data.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | RMSE | Rsquared | MAE |
| Train\_set | 429.1461 | 0.5987 | 324.5985 |
| Test\_set | 364.2157 | 0.5551 | 286.9954 |

The RMSE and MAE were better than the train\_set but the Rsquared was lower. The linear regression it was not so good for this dataset.

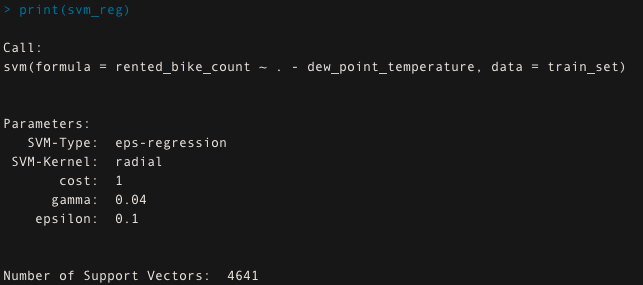
*Support Vector Machine*

This model was better than simple linear regression, offering reliable predictions and significant advantages in its use. The predictions in relation to the existing values shown in the graph (Graph 10), show how close the model falls to the actual values, thus reducing the prediction error.



Graph 10: Predictions vs Test\_ set values

The stats for SVM model is:



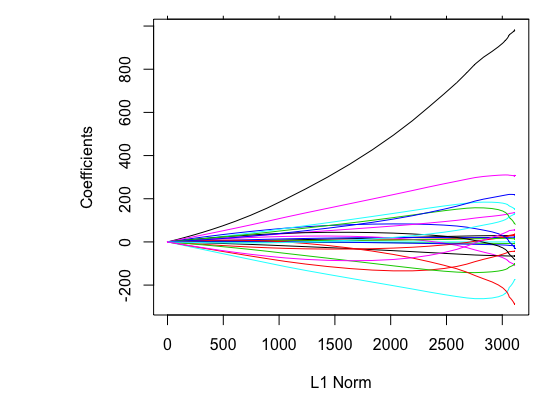
The results for SVM was:

|  |  |  |
| --- | --- | --- |
| RMSE | Rsquared | MAE |
| 255.8244 | 0.7326 | 172.4006 |

*Ridge*

Grid search was used to tune this model and found the best lambda value for this model.

Also from the graph of the model it is observed that the larger the lambda the farther the correlations of the values from zero, either to the positives or to the negatives.



Our model offers mediocre results compared to the rest that follow.

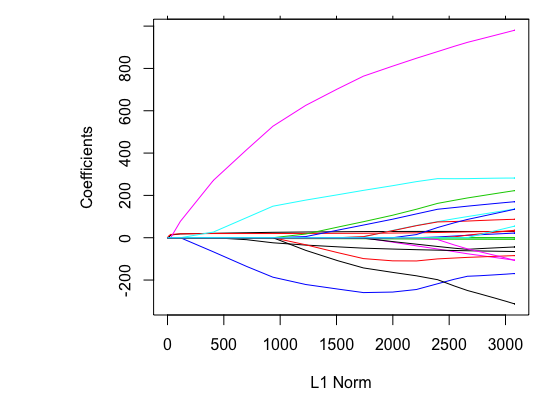
|  |  |  |
| --- | --- | --- |
| RMSE | Rsquared | MAE |
| 363.3847 | 0.554 | 286.4747 |

*Lasso*

The Lasso model used the cross validation to get better results. It was same as ridge in part of the results.

Lasso using some variables from ridge(in R) like x\_train and y\_train.

The coeffients plot was also with the same directions.

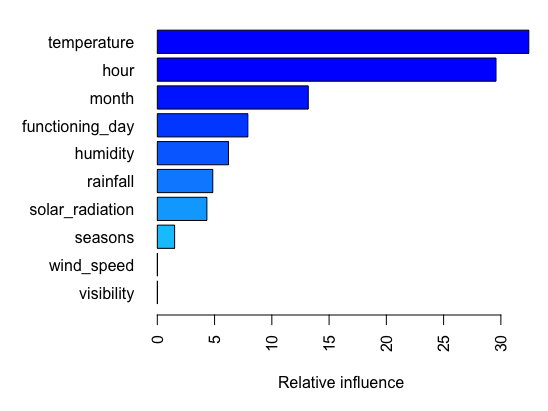


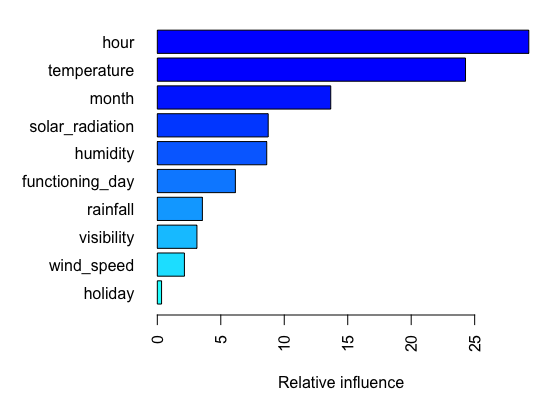
|  |  |  |
| --- | --- | --- |
| RMSE | Rsquared | MAE |
| 363.6859 | 0.5552 | 286.6032 |

*Gradient Boosted Machine*

For this model 2 sub-models were used (GBM1, GBM2) with changed parameters as explained in the previous chapter. This was done to determine the best performance and consequently the best implementation for this particular type of machine learning model.

These models provide the user with visualization of the significance-influence of the variables in the model. So below, it seems that in GBM1 the temperature and the time are with different variables that affect more the number of rented bicycles, while in GBM2 this series changes with the first time and the second the temperature. In the other variables the changes are small.





On the other hand the results were different because the tuning did good job and improved the model.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | RMSE | Rsquared | MAE |
| GBM1 | 250.2668 | 0.7389 | 195.1579 |
| GMB2 | 209.3698 | 0.8260 | 133.9270 |

*Boosted Trees (XGBoost)*

This was the best model in the project. The model seems to fit the Seoul data well and its predictions were very good with very few errors.

|  |  |  |
| --- | --- | --- |
| RMSE | Rsquared | MAE |
| 205.4325 | 0.8332 | 134.9839 |

Model Selection

Apparently the Boosted Tree (XGboost) model was the best of all. Large R ^ 2 and small the other 2 sizes of the evaluation. As a result, it is possible to predict the demand for bicycles per hour in Seoul city in the future, with very little forecast error. It is also important to mention that from the graphs until now the end of the implementation of the model, the variables of time, temperature and month played an important role from the choice of the model until its creation. It seems that these are the main variables that affect the demand for bicycle rental.

|  |  |  |  |
| --- | --- | --- | --- |
| Model | RMSE | Rsquared | MAE |
| LM | 364.2157 | 0.5551 | 286.9954 |
| SVM | 255.8244 | 0.7326 | 172.4006 |
| RIDGE | 363.3847 | 0.554 | 286.4747 |
| LASSO | 363.6859 | 0.5552 | 286.6032 |
| GBM | 209.3698 | 0.8260 | 133.9270 |
| BT(XGBoost) | 205.4315 | 0.8332 | 134.9839 |

# Conclusions

In summary, it would be useful to highlight the importance of data analysis and forecasting in this day and age. The problem of work is crucial, because through bicycle rental the traffic on the streets is reduced, people become active and play sports, the emissions of bicycles are reduced and the city that has bicycles becomes attractive to travelers to discover it. Thus the analysis of the data is important in order to predict the demand in the future. In this way the cities will have their bicycle fleet ready for use, at any time, with the accuracy provided by the forecast models. These models can also be used for financial factors improving the profits of companies. If there is enough demand for bicycles in a city, then a gap is created in the available quantity of bicycles, as a result of which a company invests in it. So it seems that the problem is multidimensional. Also a government can see the demand for bicycles, cover it and thus reduce traffic on the roads.

All of the above presupposes an evaluation of the main variables that affect the demand-use of the bicycle. The weather conditions with number one the temperature are very important and follows the time of use of the bicycle, with the afternoon hours being the most popular and the month also playing an important role. So cities-companies that want to invest in this sector should take the above variables seriously.

Finally, as far as any future analysis is concerned, there is room for the models to be further enhanced or applied to the data of another city in order to better evaluate their performance. An important addition to a future analysis would be some data on the routes that a city has on a bike path so that it can check along with the weather.

# References

DeMaio, P., 2009. Bike-Sharing: History, Impacts, Models of Provision and Future. *Journal of Public Transportation,* 4(12), pp. 41-56.

Feng, Y. & Wang, S., 2017. A forecast for bicycle rental demand based on random forests and multiple linear regression. *2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS),* pp. 101-105.

Graham, B., Bond, R., Quinn, M. & Mulvenna, M., 2018. Using Data Mining to Predict Hospital Admissions From the Emergency Department. *IEEE Access,* Issue 6, pp. 10458-10469.

Lin, L., He, Z. & Peeta, S., 2018. Predicting station-level hourly demand in a large-scale bike-sharing network: A graph convolutional neural network approach. *Transportation Research ,* 3(97), pp. 258-276.

Pan, Y., Zheng, R. C., Zhang, J. & Yao, X., 2019. Predicting bike sharing demand using recurrent neural networks. *Procedia Computer Science,* Issue 147, pp. 562-566.

Pucher, J., Komanoff, C. & Schimek, P., 1999. Bicycling renaissance in North America? Recent trends and alternative policies to promote bicycling. *Transportation Research,* 1(33), pp. 625-654.

Strempel, S., Nendza, M., Scheringer, M. & Hungerbuhler, K., 2013. USING CONDITIONAL INFERENCE TREES AND RANDOM FORESTS TO PREDICT THE BIOACCUMULATION POTENTIAL OF ORGANIC CHEMICALS. *Environmental Toxicology and Chemistry,* 5(32), pp. 1187-1195.

Tomaras, D., Boutsis, I. & Kalogeraki, V., 2018. Modeling and Predicting Bike Demand in Large City Situations. *2018 IEEE International Conference on Pervasive Computing and Communications (PerCom),* pp. 1-10.

Yang, Z. et al., 2016. Mobility Modeling and Prediction in Bike-Sharing Systems. *In Proceedings of the 14th Annual International Conference on Mobile Systems, Applications and Services (MobiSys '16),* pp. 165-178.

# Appendix

R-Code

#################################################

#Advanced Analytics & Machine Learning

#MGT7179

#Assignment 1 - Regression

#Vasileios Gounaris Bampaletsos-40314803

#regression problem

#bike-sharing dataset

#seoul's data for bike-sharing

#analyze and predict the demand of bikes per hour

#methods which used:

#linear regression (with cross validation)

#support vector machine (for regression)

#ridge (with grid search)

#lasso (with cross validation)

#gradient boosted machine x2 (1normal and 1tuned)

#boosted tress (xgboost method & cross validation)

###################################################

#set the working directory

setwd("/Users/basilisgounarismpampaletsos/Desktop/PROJECTS 2/26:03 analytics")

options(scipen = 9)

#load the libraries

library(readxl)

library(psych)

library(ggplot2)

library(caTools)

library(statsr)

library(dplyr)

library(BAS)

library(car)

library(tidyr)

library(purrr)

library(gridExtra)

library(forcats)

library(corrplot)

library(magrittr)

library(caret)

library(Hmisc)

library(tidyverse)

library(ggpubr)

library(ROCR)

library(broom)

library(lubridate)

library(GGally)

library(ISLR)

library(hrbrthemes)

library(viridis)

library(e1071)

library(plyr)

library(readr)

library(repr)

library(glmnet)

library(ggthemes)

library(scales)

library(wesanderson)

library(styler)

library(xgboost)

library(randomForest)

library(rsample)

library(gbm)

library(h2o)

library(pdp)

library(lime)

#load the data

#dataset from Seoul's bike-sharing

data <- read.csv("SeoulBikeData.csv")

###########################################################

#summarize the data for the first time

#check the data's distribution and descriptive measures

summary(data)

#check some basic statistics for seoul's weather data

#looking for the minimum, median and maximum values

#understand the data's distribution

#looking for outliers

summary(data$rented\_bike\_count) #stats for number of the bikes

summary(data$temperature) #stats for temperature (Celcius)

summary(data$wind\_speed) #stats for wind speed (miles per second (m/s))

summary(data$rainfall) #stats for rainfall (milimeters (mm))

summary(data$visibility) #stats for visibility (multipled by 10 meters)

summary(data$humidity) #stats for humidity (%)

summary(data$solar\_radiation) #stats for solar radiation (MJ/m^2)

summary(data$snowfall) #stats for snowfall (cm)

aggregate(temperature ~ hour, data = data, FUN = mean)

aggregate(solar\_radiation ~ hour, data = data, FUN = mean)

describeBy(data$rented\_bike\_count,data$hour)

describeBy(data$temperature, data$hour)

#make basic visualisations for better understanding

#make simple histograms for numerice variables

#histograms help to check the data quality

#NUMERIC VARIABLES

hist(data$rented\_bike\_count)

hist(data$rainfall)

hist(data$humidity)

hist(data$temperature)

hist(data$dew\_point\_temperature)

hist(data$wind\_speed)

hist(data$snowfall)

hist(data$hour)

hist(data$visibility)

hist(data$solar\_radiation)

#CATEGORICAL VARIABLES

#barplot with seasons and rented bikes

#see how many bikes are rented per season

ggplot(data, aes(seasons)) +

geom\_bar(colour="black", mapping = aes(fill = rented\_bike\_count)) +

labs(fill="rented\_bike\_count", x="Seasons", y= "rented bike count", title="Rented Bike Count per Season")

#barplot with functioning day and rented bikes

#see how many bikes are rented if the day was functioning or not

ggplot(data, aes(functioning\_day)) +

geom\_bar(colour="black", mapping = aes(fill = rented\_bike\_count)) +

labs(fill="rented\_bikes", x="functioning\_day", y= "count",

title="Rented Bikes Distribution in Functioning days")

#barplot with holidy and rented bikes

#see how many bikes are rented if was holiday or not

ggplot(data, aes(holiday)) +

geom\_bar(colour="black", mapping = aes(fill = rented\_bike\_count)) +

labs(fill="rented\_bikes", x="holidays", y= "count",

title="Rented Bikes Distribution in Holidays")

##################################################################################################

#FIX THE DATA

##################################################################################################

#transform the type of the date to extract month

data$date <- as.Date(data$date, format="%d/%m/%Y")

#extract month from date variable

#it is a very nice variable(month) for the models

data$month <- months(data$date)

#my code was in greek language

#it gave me the months in greek language

#i transform the greek to english

data$month[data$month == "Ιανουαρίου"] <- "January"

data$month[data$month == "Φεβρουαρίου"] <- "February"

data$month[data$month == "Μαρτίου"] <- "March"

data$month[data$month == "Απριλίου"] <- "April"

data$month[data$month == "Μαΐου"] <- "May"

data$month[data$month == "Ιουνίου"] <- "June"

data$month[data$month == "Ιουλίου"] <- "July"

data$month[data$month == "Αυγούστου"] <- "August"

data$month[data$month == "Σεπτεμβρίου"] <- "September"

data$month[data$month == "Οκτωβρίου"] <- "October"

data$month[data$month == "Νοεμβρίου"] <- "November"

data$month[data$month == "Δεκεμβρίου"] <- "December"

#put the months to chronological serie

data$month <- factor(data$month, levels = c("January", "February", "March", "April", "May",

"June", "July", "August",

"September", "October", "November", "December"))

#delete the date valiable

data$date <- NULL

#data$humidity[data$humidity <10] <- NA

#data <- na.omit(data)

##################################################################################################

#FINAL VISUALISATIONS

#VISUALISE THE DATA FOR BETTER UNDERSTANDING

##################################################################################################

#vis 1

#check the impact of wind speed in bike rental

ggplot(data, aes(x=wind\_speed, y=rented\_bike\_count, color=holiday)) +

geom\_point(size=0.2) +

labs(title = "The Impact of Wind in Bike Demand", x = "wind speed (m/s)", y = "rented bike count")

#vis 2

#check the distribution of bike rental from the eyes of visibility

ggplot(data, aes(x = visibility)) +

geom\_density(aes(y = ..count..), fill = "lightgray") +

geom\_vline(aes(xintercept = mean(visibility)), linetype = "dashed", size = 0.6, color = "#FC4E07") +

labs(fill="visibility", x="duration", y= "Visibility (10m)", title="visibility distribution",

caption="With the mean line in red")

#check the visibility's impact in bike rental

ggplot(data = data, aes(x = rented\_bike\_count, y = visibility)) +

geom\_point() +

geom\_smooth(method = "lm") +

labs(title = "The impact of Visalibity in Bike Demand", x = "rented bike count", y = "visibility (10m)")

#vis 3

#check the humidity's impact in bike rental

ggplot(data = data, aes(x = rented\_bike\_count, y = humidity)) +

geom\_point() +

geom\_smooth(method = "lm") +

labs(title = "The impact of Humidity in Bike Demand", x = "rented bike count", y = "humidity(%)")

#vis 4

#check how temperature impacts the bike rental

ggplot(data) +

geom\_point(mapping = aes(x = rented\_bike\_count, y = temperature)) +

labs(x="rented bike count", y= "temperatute (Celcius)", title="The impact of temperature to bike demand",

caption="Scatter plot to see the values distribution")

#vis 5

#create a boxplot with the distribution of rented bike throughout year

ggplot(data, aes(x=month, y=rented\_bike\_count, fill=seasons)) +

geom\_boxplot(alpha=0.3) +

theme(legend.position="none") +

labs(x="Month", y= "Rented Bike Count", title="Rented Bike Destribution throughout Year per Season")

#vis 6

#bar plot with the average bike rental per hour

ggplot(data = data) +

geom\_bar(mapping = aes(x=as.factor(hour), y = rented\_bike\_count), stat = "summary", y.mean = "mean") +

labs(title = "Average Rented Bikes per Hour", x="hour", y="AVG rented bike count")

#vis 7

#bike rental by holiday

ggplot(data) +

geom\_density(aes(x = rented\_bike\_count,

fill = seasons),

alpha = 0.1) +

scale\_fill\_brewer(palette = "Dark2") +

theme\_fivethirtyeight() +

theme(axis.title = element\_text()) +

labs(title = "Bike Rental Density By Holiday",

fill = "Holiday",

x = "Bike Rentals",

y = "Density")

#vis 8

#bike rental by temperature and season

ggplot(data, aes(y = rented\_bike\_count,

x = temperature,

color = seasons)) +

geom\_point(show.legend = FALSE) +

geom\_smooth(se = FALSE,

show.legend = FALSE) +

facet\_grid(~seasons) +

scale\_color\_brewer(palette = "Dark2") +

theme\_fivethirtyeight() +

theme(axis.title = element\_text()) +

ylab("Bike Rentals") +

xlab("Temperature (°C)") +

ggtitle("Bike Rental By Temperature per Season")

#vis 9

#rented bikes by solar radiation

ggplot(data) +

geom\_point(aes(y = rented\_bike\_count,

x = solar\_radiation,

color = functioning\_day),

show.legend = FALSE) +

facet\_grid(~functioning\_day) +

scale\_color\_brewer(palette = "Dark2") +

theme\_fivethirtyeight() +

theme(axis.title = element\_text()) +

labs(x="Solar Radiation (MJ/m^2)", y= "Bike Rentals", title="Bike Rentals by Solar Radiation",

caption="Two graphs depends on Functioning Day")

#data$seasons=as.factor(data$seasons)

#data$month=as.factor(data$month)

#data$holiday=as.factor(data$holiday)

#data$functioning\_day=as.factor(data$functioning\_day)

#split the data

#train\_set 75%

#test\_set 25%

set.seed(123)

split = sample.split(data$rented\_bike\_count, SplitRatio = 0.75)

train\_set = subset(data, split == TRUE)

test\_set = subset(data, split == FALSE)

##################################################################################################

#CORRELATIONS

##################################################################################################

#at this point we want to see the correlations how the variables connect each other

#check this connections/associations because they are important for models' building

#correlations for numeric variables

#select the numeric variables to build the cor matrix

continuous\_var <- select(data, rented\_bike\_count,hour,temperature,humidity,wind\_speed,

visibility,dew\_point\_temperature,solar\_radiation,rainfall)

#use 3 different ways to check the correlation

#very useful graphs

#full cor matrix, half cor matrix, pair plots

#correlation matrix

#using pearson method

continuous\_var.cor = cor(na.omit(continuous\_var), method = "pearson")

corrplot(continuous\_var.cor)

#half correlation matrix

#using pearson method

ggcorr(na.omit(continuous\_var), method = c("everything", "pearson"))

#pair plots

#using pearson method

#check the relation between the intependent and depends variables

pairs.panels(data,

method = "pearson", # correlation method

hist.col = "#00AFBB",

density = TRUE, # show density plots

ellipses = TRUE # show correlation ellipses

)

#correlations for categorical variables

#using pearson's chi-squared test

chisq.test(data$seasons, data$rented\_bike\_count)

chisq.test(data$holiday, data$rented\_bike\_count)

chisq.test(data$functioning\_day, data$rented\_bike\_count)

chisq.test(data$month, data$rented\_bike\_count)

##################################################################################################

#create the model

#testing the variables and choose the best model

model = lm(formula = rented\_bike\_count ~.-dew\_point\_temperature,

data = train\_set)

summary(model)

#linear regression

#making some examples before final selection

lin\_reg = lm(formula = model1,

data = train\_set)

summary(lin\_reg)

##################################################################################################

#CHECK ASSUMPTIONS

##################################################################################################

#assumption 1

#the mean of residuals is zero

mean(model$residuals)

#assumption 2

#we check 2 assumpitions with this plot

#1.Homoscedasticity of residuals or equal variance

#2.Normality of residuals

par(mfrow=c(2,2)) # set 2 rows and 2 column plot layout

plot(model)

#assumption 3

#multicollinearity

vif(model)

#assumption 4

#influential cases

influence.measures(model)

#assumption 5

#independent residuals

durbinWatsonTest(model)

######################################################################################################

#LINEAR REGRESSION

######################################################################################################

#with cross validation

set.seed(123)

train.control <- trainControl(method = "cv", number = 10)

# Train the model

model1 <- train(rented\_bike\_count ~.-dew\_point\_temperature, data = train\_set, method = "lm",

trControl = train.control)

# Summarize the results

print(model1)

# Make predictions on the test data

lin\_pred <- predict(model1, newdata = test\_set)

lin\_pred

#Compute RMSE, MAE and R^2

postResample(lin\_pred, test\_set$rented\_bike\_count)

######################################################################################################

#SVM REGRESSION

######################################################################################################

#create the svm regression model

svm\_reg = svm(rented\_bike\_count ~.-dew\_point\_temperature, data=train\_set)

print(svm\_reg) #print the

# Make predictions on the test data

svm\_pred = predict(svm\_reg, test\_set)

#plot to compare the predictions with actual numbers

x=1:length(test\_set$rented\_bike\_count)

plot(x, test\_set$rented\_bike\_count, pch=13, col="red")

lines(x, svm\_pred, lwd="1", col="blue")

#Compute RMSE, MAE and R^2

postResample(svm\_pred, test\_set$rented\_bike\_count)

#plot the stats and results from svm regression model

plot(data)

points(data$rented\_bike\_count, svm\_pred, col="red", pch=13)

######################################################################################################

#RIDGE

######################################################################################################

data = na.omit(data)

#create a matrix to run the ridge

x = model.matrix(rented\_bike\_count ~.-dew\_point\_temperature, data)

#select the indepented variable

y = data %>%

select(rented\_bike\_count)%>%

unlist()%>%

as.numeric()

#create the ridge regression using grid search

grid = 10^seq(10, -2, length = 100) #get the lambda sequence

ridge\_reg = glmnet(x, y, alpha = 0, lambda = grid)

ridge\_reg

#find and plot the coef

dim(coef(ridge\_reg))

plot(ridge\_reg)

#find the lambda, regression coef. and error

ridge\_reg$lambda[50]

coef(ridge\_reg)[,50]

sqrt(sum(coef(ridge\_reg)[-1,50]^2))

predict(ridge\_reg, s=50, type="coefficients")[1:20,]

x\_train = model.matrix(rented\_bike\_count~.-dew\_point\_temperature, train\_set)

x\_test = model.matrix(rented\_bike\_count~.-dew\_point\_temperature, test\_set)

y\_train = train\_set %>%

select(rented\_bike\_count) %>%

unlist() %>%

as.numeric()

y\_test = test\_set %>%

select(rented\_bike\_count) %>%

unlist() %>%

as.numeric()

#tune the model using the best values

ridge\_reg = glmnet(x\_train, y\_train, alpha=0, lambda = grid, thresh = 1e-12)

# Make predictions on the test data

ridge\_pred = predict(ridge\_reg, s = 4, newx = x\_test)

#Compute RMSE, MAE and R^2

postResample(ridge\_pred, test\_set$rented\_bike\_count)

######################################################################################################

#LASSO

######################################################################################################

#create the lasso regression with alpha 1

#using cross validation to build better model

lasso\_reg = glmnet(x\_train,

y\_train,

alpha = 1,

lambda = grid)

lasso\_reg

plot(lasso\_reg)

set.seed(1)

cv\_output = cv.glmnet(x\_train, y\_train, alpha = 1) # Fit lasso model on training data

plot(cv\_output) # Draw plot of training MSE as a function of lambda

best\_lambda = cv\_output$lambda.min # Select lamda that minimizes training MSE

# Make predictions on the test data

lasso\_pred = predict(lasso\_reg, s = best\_lambda , newx = x\_test) # Use best lambda to predict test data

#Compute RMSE, MAE and R^2

postResample(lasso\_pred, test\_set$rented\_bike\_count)

mean((lasso\_pred - y\_test)^2) # Calculate test MSE

out = glmnet(x, y, alpha = 1, lambda = grid) # Fit lasso model on full dataset

lasso\_coef = predict(out, type = "coefficients", s = best\_lambda )# Display coefficients using lambda chosen by CV

lasso\_coef

# Display only non-zero coefficients

lasso\_coef[lasso\_coef != 0]

######################################################################################################

#GRADIENT BOOSTED MACHINE

######################################################################################################

#GBM1

#create the gbm regression model

#cross validation using 5 steps

gbm1 <- gbm(

formula = rented\_bike\_count ~ . -dew\_point\_temperature,

distribution = "gaussian",

data = train\_set,

n.trees = 10000,

interaction.depth = 1,

shrinkage = 0.001,

cv.folds = 5,

n.cores = NULL, # will use all cores by default

verbose = FALSE

)

# print results

print(gbm1)

# plot loss function as a result of n trees added to the ensemble

gbm.perf(gbm1, method = "cv")

#influence bar chart

par(mar = c(5, 8, 1, 1))

summary(

gbm1,

cBars = 10,

method = relative.influence, # also can use permutation.test.gbm

las = 2

)

#Make predictions on the test data

gbm1\_pred <- predict(gbm1, n.trees = gbm.fit$n.trees, test\_set)

#Compute RMSE, MAE and R^2

postResample(gbm1\_pred, test\_set$rented\_bike\_count)

#GBM2

#tuning the GBM1

#making a second gbm to see the difference

#the target is to boost the gbm for better results

# train GBM model

gbm2 <- gbm(

formula = rented\_bike\_count~.-dew\_point\_temperature,

distribution = "gaussian",

data = train\_set,

n.trees = 10000,

interaction.depth = 3,

shrinkage = 0.1,

cv.folds = 5,

n.cores = NULL, # will use all cores by default

verbose = FALSE

)

# print results

print(gbm2)

# find index for n trees with minimum CV error

min\_MSE <- which.min(gbm2$cv.error)

summary(gbm2$cv.error)

# plot loss function as a result of n trees added to the ensemble

gbm.perf(gbm2, method = "cv")

#influence bar chart

par(mar = c(5, 8, 1, 1))

summary(

gbm2,

cBars = 10,

method = relative.influence, # also can use permutation.test.gbm

las = 2

)

# Make predictions on the test data

gbm2\_pred <- predict(gbm2, n.trees = gbm.fit$n.trees, test\_set)

#Compute RMSE, MAE and R^2

postResample(gbm2\_pred, test\_set$rented\_bike\_count)

######################################################################################################

#BOOSTED TREES (XGBoost)

######################################################################################################

# Fit the model on the training set

#using the XGBoost method and cross validation

set.seed(123)

model <- train(

rented\_bike\_count ~.-dew\_point\_temperature, data = train\_set, method = "xgbTree",

trControl = trainControl("cv", number = 10)

)

# Best tuning parameter mtry

model$bestTune

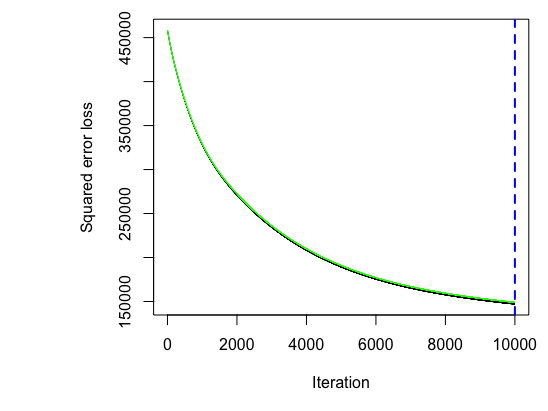
# Make predictions on the test data

predictions <- model %>% predict(test\_set)

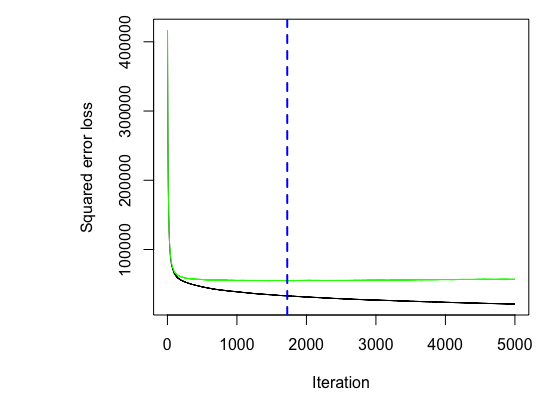
head(predictions)

#Compute RMSE, MAE and R^2

postResample(predictions, test\_set$rented\_bike\_count)

Graphs from Models

Graph 11: Loss function of GBM1 (source: my R code)



Graph 12: Loss function from BGM2 (source: my R code)