**Cab Fare Prediction**

**Contents**

* 1 Introduction . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .2-3
* 2 Exploration . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .3-4
* 3 Predictive Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4
* 4 Missing Value Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 4-6
* 4.1 Missing Value Imputation . . . . . . . . . .. . . . . . . . . . . . . . . . .5-6
* 5 Outlier Analysis . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .6-9
* 6 Feature Selection . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .10-15
* 7 Feature Scaling. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 15-17
* 8 Sampling . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17
* 9 Error Metrics. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 17-19
* 10 Model Development. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .19-23
* 10.1 Linear Regression. . . . . . . . . . . . . . . . . . . . . . . . . . . . . .19-21
* 10.2 Decision Tree Regression. . . . . . . . . . . . . . . . . . . . . . . . . 21
* 10.3 KNN implementation (Regression). . . . . . . . . . . . . . . . 22-24
* 10.4 Random Forest Regression . . . . . . . . . . . . . . . . . . . . 24-25
* 11 Model Selection. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .25
* 12 Predictions . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .26
* 13 Conclusion . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .27
* 14 Advantages. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .27
* 15 Appendices. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 28-97
* 15.1 Appendix 1 figure list . . . . . . . . . . . . . . . . . . . . . . . . . . . 28
* 15.2 Appendix 2 features list. . . . . . . . . . . . . . . . . . . . . . . . 29

**1 Introduction:** In this project we are a cab rental start-up company. We have successfully run the pilot project and now want to launch our cab service across the country. We have collected the historical data from our pilot project and now have a requirement to apply analytics for fare prediction. We need to design a system that predicts the fare amount for a cab ride in the city.

The historical data which we have from our pilot project are below.

1-train\_cab.csv

2-test.csv

We will trained our model using the train data and then perform the testing on test data.

The dataset contains seven variables. Six variables are independent and one is dependent

|  |  |
| --- | --- |
| **Variable name** | **Data type** |
| pickup\_datetime | Timestamp value indicating when the cab ride started. |
| pickup\_longitude | Float for longitude coordinate of where the cab ride started. · |
| pickup\_latitude | Float for latitude coordinate of where the cab ride started. · |
| dropoff\_longitude - | Float for longitude coordinate of where the cab ride ended. |
| dropoff\_latitude | Float for latitude coordinate of where the cab ride ended |
| passenger\_count | An integer indicating the number of passengers in the cab ride |
| fare\_amount (dependent variable) | Float the amount of cab ride fare |

**Purpose**

The purpose of this document is to specify requirements and to give guidelines for the development of above said project. In particular it gives guidelines on how to prepare the above said project.

**Scope**

At Santander, mission is to help people and businesses prosper for ways to help them understand their financial health and identify which

Products and services might help them achieve their monetary goals

**Hardware Requirements**

Processor : Intel® Core™ 2 Duo Processor, and Above

RAM : 2GB RAM

**Software Requirements:**

Python 3 or above

R 3 or above

Anaconda

R Studio

Jupyter notebook

**Instruction to rum the project:**

: install Anaconda with python

: install R

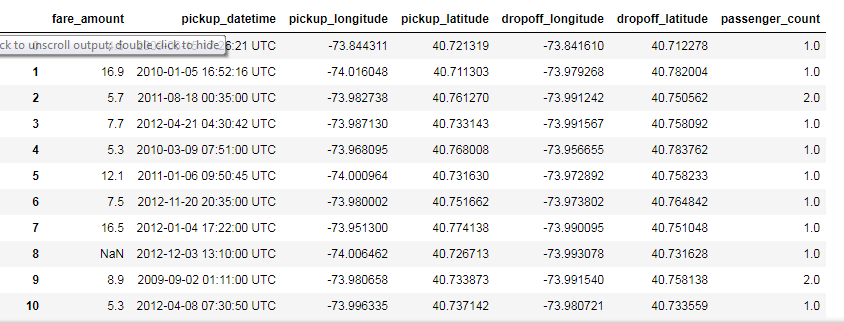
: install R studio

: open the python file with Jupyter notebook to run with python

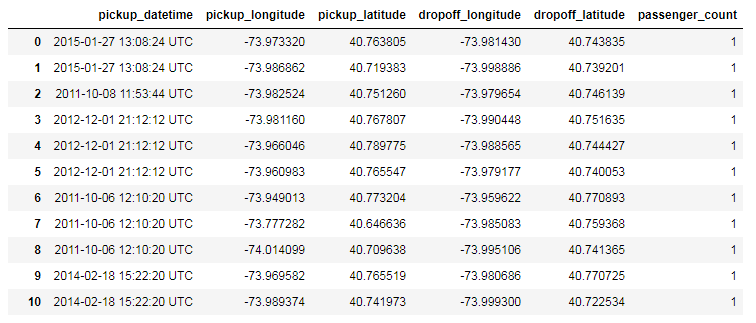
: open the .R file to run with R studio to run with python

**2 Exploration:**

The data is already given in two sets train and test ,the train data have 7 variables including the numerical dependent variable “fare\_amount” ,test data have 6 variable all the variable are same as in train data except the test data does not have the numeric dependent variable. Below are some snapshot of the datasets.

****

Training data



Test data

**3 Predictive Analysis:** Read the train data set and start the predictive analysis the data types of the variable checking the variable have the data types which required if not then convert.

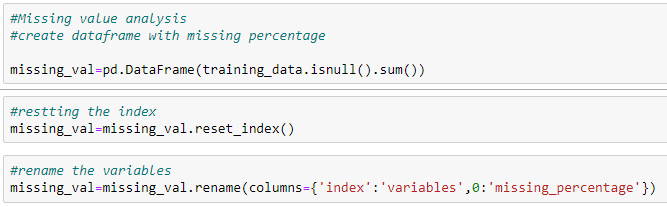
The dataset have one date time variable :pickup\_datetime

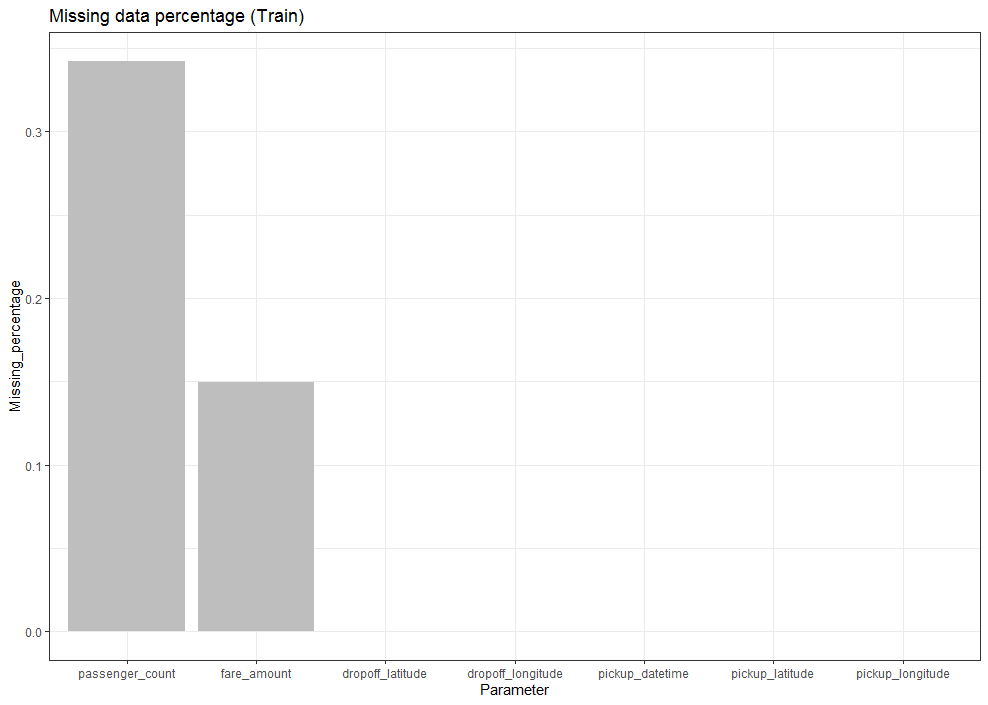
Need to convert it into numeric

Rest of the variables have their desired data types

**4 Missing value Analysis:** After the predictive analysis the most important part is missing value analysis to check the data have any missing values.

The snapshot of the code for missing value analysis is mentioned below.

****



**Missing value bar graph**

From the above predictions we concluded that the data have missing values.

Two variables have missing values **passenger\_count** and **fare\_amount.**

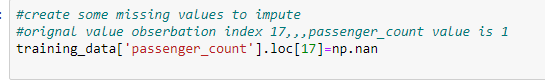
**So the data have missing values we will impute the missing values.**

**4.1 Missing Value imputation**

For missing value imputation process we need to create some missing values and we will compare the imputed values which will imputed by different methods .

The value imputed by the methods will be compared with the original value and select that method which imputation is closest**.**

Below are some screen shots of codes for the process







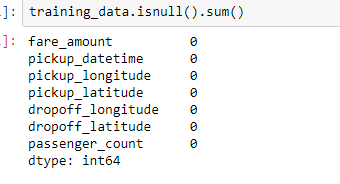




**Missing value imputation**

From the above analysis we can say that the value imputed by the median method is closest to the original so we will selected the median method for imputation.

Below is the screenshot of the code after imputation.



**5: Outlier Analysis:**

**What is outlier?**

-Observations inconsistent with rest of the dataset Global outliers.

-Couse of outliers

-Poor dataquality

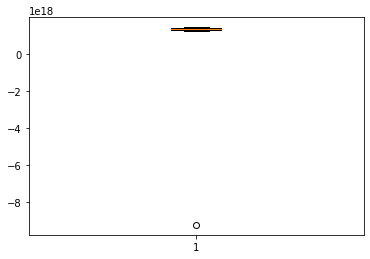
-Low Quality measurements, malfunctioning equipment, manual error -Correct but exceptional data

**How to detect the outlier?**

We can use some graphical tools.

In this model we will use Boxplot

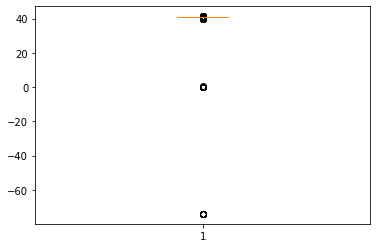
The snapshot of the boxplots are below



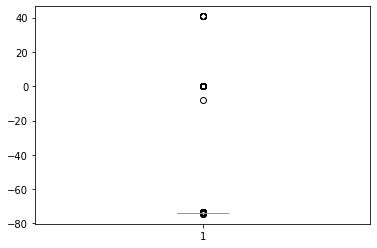
**Pickup\_datetime(boxplot)**

The pickup\_datetime have very less number of outlier

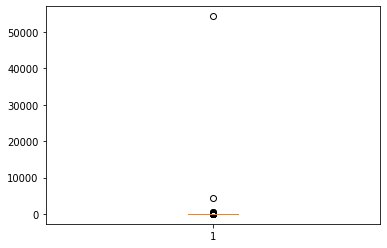
Below is the boxplot of some variables



**Dropoff\_latitude(boxplot)**

****

**Dropoff\_longitude(boxplot)**



**Fare-amount(boxplot)**

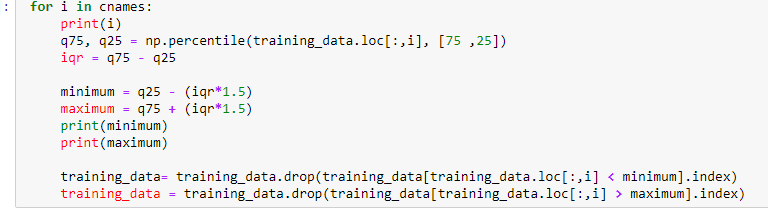
From the above analysis we can conclude the data have very less outliers.

We have to many approaches to deal with that for example

Detect and delete the outliers and replace the outlier with NA and impute them

But in this project we will **Detect and delete the outlier.**

**Below is the snapshot of code to detect and delete the outlier**

****

**6 Feature Selection:**

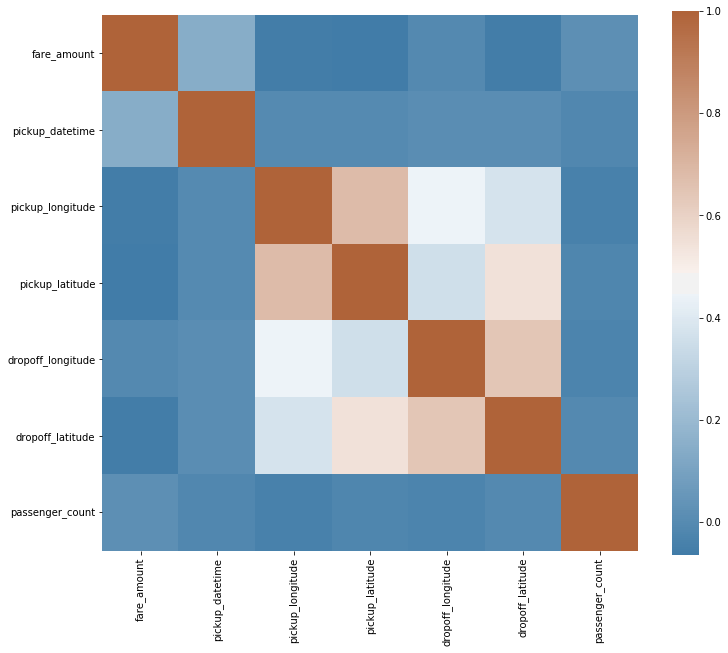
After the outlier analysis we have to select the important features of the dataset.

Selecting a subset of relevant features for use in model construction

Subset of a learning algorithms input variables upon which it should focus attention, while ignoring the rest. Delete the data that do not contain much information.

It reduces the complexity of the model. Correlation Analysis: applies only on the numerical data. Correlation tells you the association between two continuous variables. We will use the correlation plot to see the dependencies of the variables

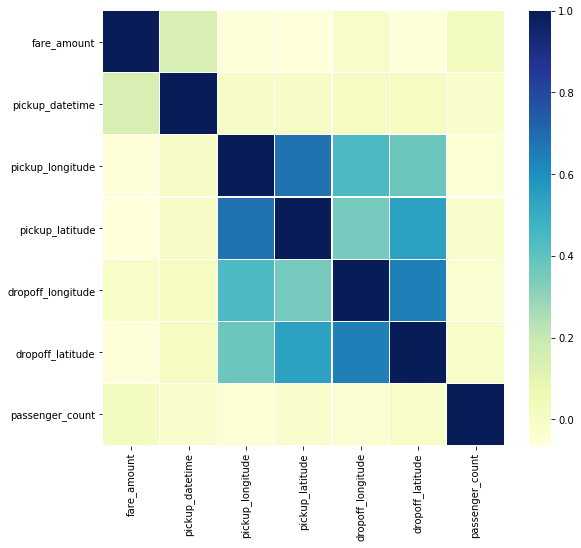
From the below mentioned correlation visualizations we will conclude the relevant features.



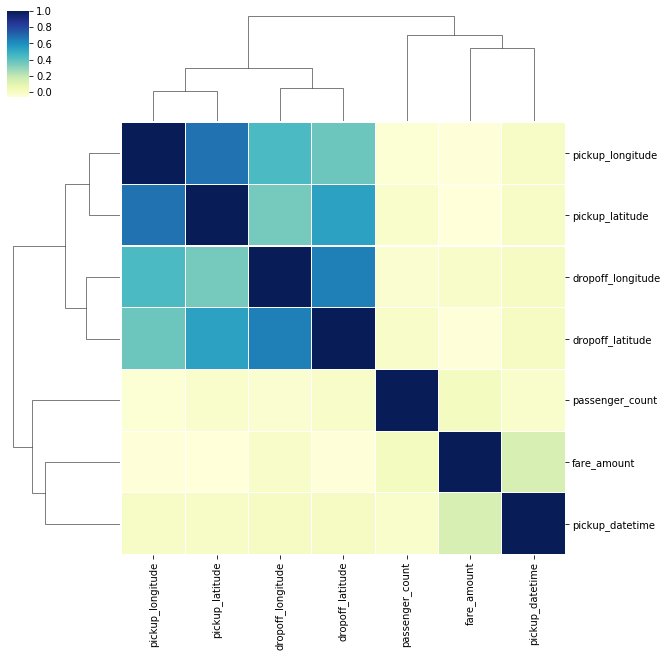
**Correlation plot 1**



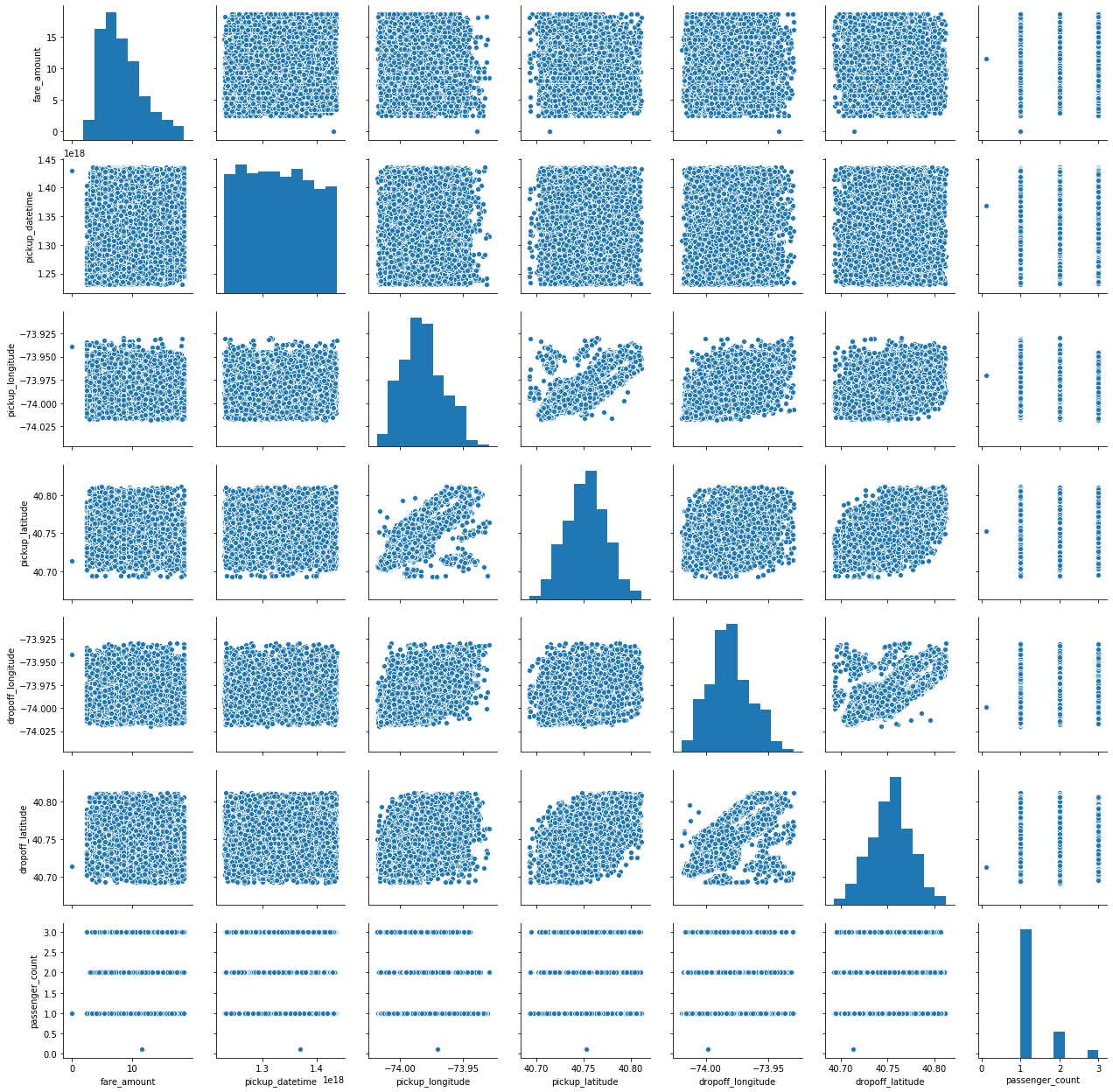
**Correlation plot 2**

****

**Correlation plot 3**

****

**Cluster plot**

****

**Pair plot**

From the above visualizations we can conclude that all the variables have the importance.

So we will proceed further with all the variables.

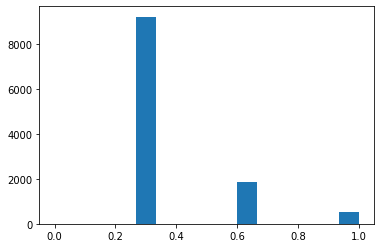
**7 Feature Scaling:**

Feature scaling is a method used to normalize the range of independent variables or features of data. In data processing.

In the analysis of the train data we choose histogram to visualize to check the data have uniform distribution or not.

In this specific dataset there is a latitude and longitude data so will not use the scaling techniques on that variables we will only perform scaling on pickup\_datetime , passenger\_count and fare\_amount , we have selected the fare\_amount variable because if we will not do a proper scaling on that the error will be so higher so to reduce the error rate we will perform the scaling on fare\_amount also.

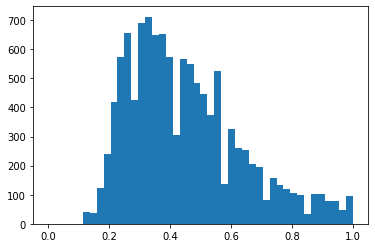
Below are histogram of variables .



**Passenger\_count(histogram)**

****

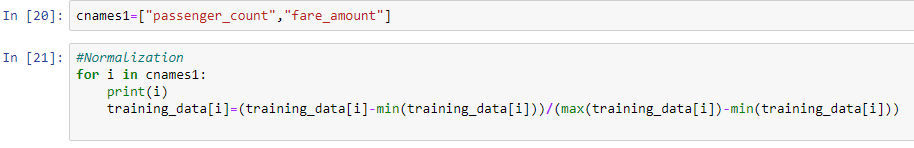
**Pickup\_datetime(histogram)**

****

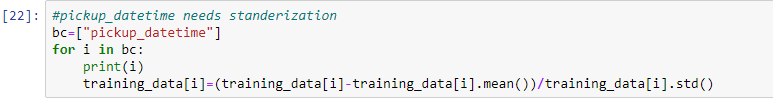
**Fare\_amount(histogram)**

The above histogram says that the passenger\_count and fare\_amount variable needs Normalization. And the pickup\_datetime variable need Standardization.

Below are the snapshot of code for Normalization. And Standardization.



Normalization



Standardization

**8 Sampling:**

The aim of any sample is to represent the characteristics of the whole data

There are different methods to generate sample

As a researcher we have to select the most appropriate method meet the requirements of our research.

**Note: we will not use the sampling**

**9 Error Metrics:**

The type of matrices will be depend on the type of model and implementation plan of model

There are to many available

But we will compare only two and choose only one from them.

Regrassion Matrix:

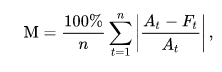
MAPE

RMSE

Mean Absolute Error (MAE) and Root mean squared error (RMSE) are two of the most common metrics used to measure accuracy for continuous variables.

**MAPE::**

The mean absolute percentage error (MAPE), also known as mean absolute percentage deviation (MAPD), is a measure of prediction accuracy of a forecasting method in statistics, for example in trend estimation, also used as a loss function for regression problems in machine learning. It usually expresses accuracy as a percentage, and is defined by the formula:

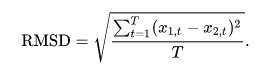


Where At is the actual value and Ft is the forecast value. The difference between At and Ft is divided by the actual value At again. The absolute value in this calculation is summed for every forecasted point in time and divided by the number of fitted points n. multiplying by 100% makes it a percentage error.

**RMSE/RMSD::**

The root-mean-square deviation (RMSD) or root-mean-square error (RMSE) (or sometimes root-mean-squared error) is a frequently used measure of the differences between values (sample or population values) predicted by a model or an estimator and the values observed. The RMSD represents the square root of the second sample moment of the differences between predicted values and observed values or the quadratic mean of these differences. These deviations are called residuals when the calculations are performed over the data sample that was used for estimation and are called errors (or prediction errors) when computed out-of-sample.

RMSE/RMSD is defined by the formula below



MAE is more robust (less sensitive to outliers) than MSE but this doesn’t mean it is always better to use MAE. The following questions help to decide:

1-Do you have outliers in the data?

If yes use MAE

2-are you sure they are outliers?

If yes use MAE

3-if they are just unexpected values we should still care about

If yes use MSE

So in this model we will use MAPE method to measure the error rate.

**10 Model Development:**

Now we will develop the machine learning model on top of our data set so we will use

Linear Regression

Decision Tree Regression

KNN Regression

And select the best one from them based on the error metric.

**10.1 Linear Regression :**

A supervised machine learning algorithm

Not based on historical data it will save the numbers in terms of co-efficient

Prediction model-

Simple linear regression

Multiple linear regression

Describe relationship among variables

The one simple case is where a dependent variable may be related to independent or explanatory variable.

Y=b0+b1x

It’s based on some assumptions.

Linear relationship

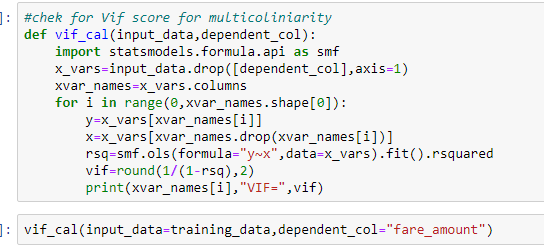
Multivariate normality

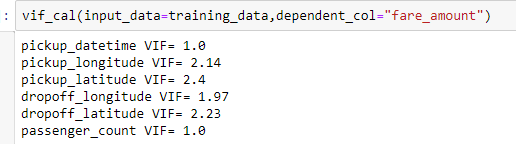
No or little multicolinearity

No auto correlation

Below is the snapshot of the code for Linear Regression –

Check VIF score for multicolinearity

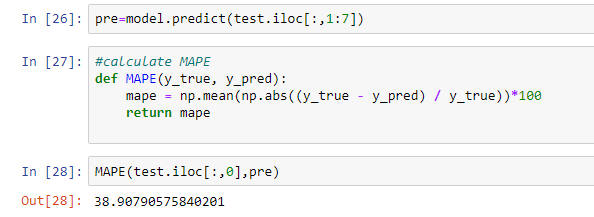






**Linear Regression**

From the Linear Regression we can now calculate the Error matrices

****

**The MAPE of the model is 38.90%**

**10.2 Decision Tree Regression:**

A predictive model based on a branching series of Boolean tests.

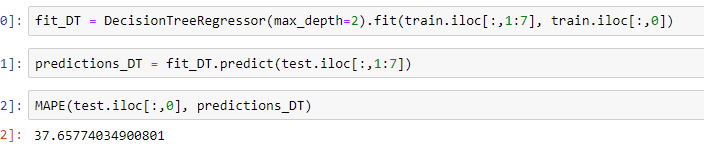
Can be used for classification and regression

There are number of different types of decision trees that can be used in machine learning algorithms.

Decision tree is a rule each branch connects node with “and” ,multiple branches are connected by “or”

Extemely easy to understand by the business users’

The snapshot of the code for Decision Tree Regression is below



**Decision Tree Regression**

**The MAPE of the model is 37.65%**

**10.3 KNN implementation (KNN Regression**)

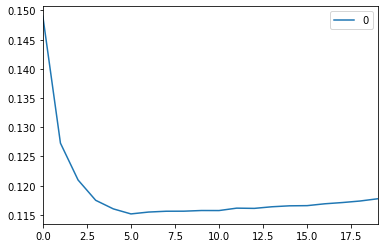
Stands for K-Nearest Neighbor

Knn is simple algorithm that stores all variable cases and classifies new cases based on a similarity measure

It’s a supervise machine learning algorithm

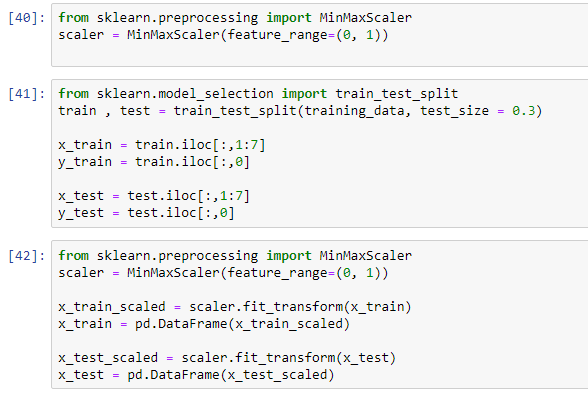
Lazy learning

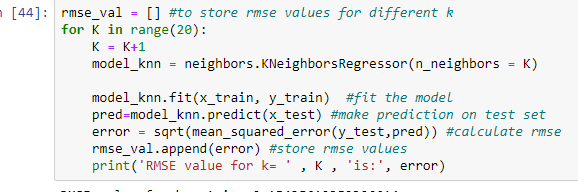
Local heuristic

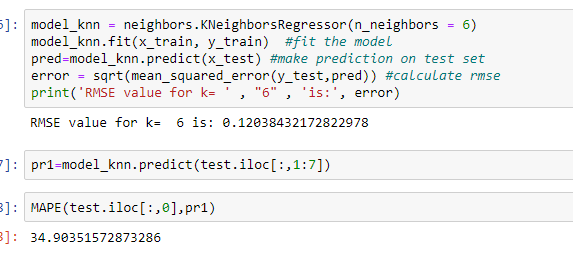


**Pot to select the number of K**

Below is the snapshot of the code





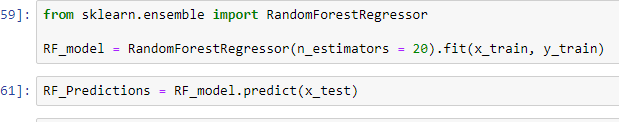


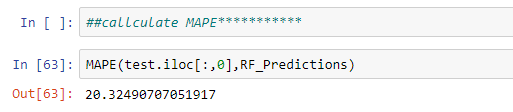
**The MAPE of the model is 34.90%**

**10.4 Random Forest Regression:**

Random forest is a Supervised Learning algorithm which uses ensemble learning method for classification and regression. Random forest is a bagging technique and not a boosting technique. The trees in random forests are run in parallel. There is no interaction between these trees while building the trees.

Below is the snapshot of the code





**The MAPE of the model is 20.32%**

**11 Model Selection:**

**From the above prediction we will select the best model for our data set**

|  |  |
| --- | --- |
| **ML Model** | **MAPE** |
| Linear Regression | 38.90 |
| Decision Tree Regression | 37.65 |
| KNN Implementation | 33.04 |
| Random Forest Regression | 20.32 |

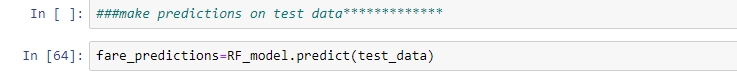
**The criteria is**

**Lower the MAPE value**

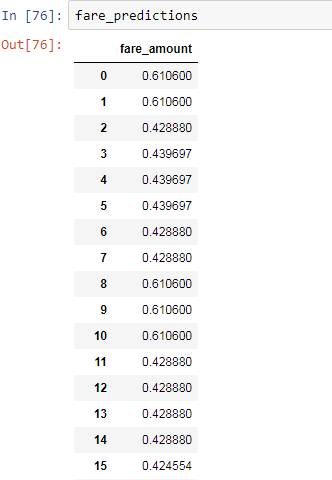
**So the best model from the above table (train and tested models)**

**Random Forest Regression is the best model we will predict with the same.**

**12 Prediction using Random Forest Regression:**

****

**Predicted Values:**

****

**Fare\_predictions**

**13 Conclusion:**The best model is Random Forest Regression for this problem from the above tested model

**14 Advantages of the project:**

The advantage to use the machine learning model in the business is that easy to estimate the genuine fare of their cab ride so this project helps to keep the customers with the client because they will provide affordable cab rental service in the city.

**15 Appendices:**

**15.1 Appendix 1 figure list**

|  |  |
| --- | --- |
| **Figure name** | **Page number** |
| Missing value bar graph | 5 |
| Pickup\_datetime(boxplot) | 7 |
| Pickup\_datetime(boxplot) | 8 |
| Dropoff\_longitude(boxplot) | 8 |
| Fare-amount(boxplot) | 9 |
| Correlation plot 1 | 10 |
| Correlation plot 2 | 11 |
| Correlation plot 3 | 12 |
| Cluster plot | 13 |
| Pair plot | 14 |
| Passenger\_count(histogram) | 15 |
| Pickup\_datetime(histogram) | 16 |
| Fare\_amount(histogram) | 16 |
| Pot to select the number of K | 22 |

**15.2 Appendix 2 features list**

|  |  |
| --- | --- |
| **Variable name** | **Data type** |
| pickup\_datetime | Timestamp value indicating when the cab ride started. |
| pickup\_longitude | Float for longitude coordinate of where the cab ride started. · |
| pickup\_latitude | Float for latitude coordinate of where the cab ride started. · |
| dropoff\_longitude - | Float for longitude coordinate of where the cab ride ended. |
| dropoff\_latitude | Float for latitude coordinate of where the cab ride ended |
| passenger\_count | An integer indicating the number of passengers in the cab ride |
| fare\_amount (dependent variable) | Float the amount of cab ride fare |