Cab fare prediction

Anisha Shaj

**CHAPTER 1**

**Introduction**

**1.1 Problem Statement**

You are a cab rental start-up company. You have successfully run the pilot project and now want to launch your cab service across the country. You have collected the historical data from your pilot project and now have a requirement to apply analytics for fare prediction. You need to design a system that predicts the fare amount for a cab ride in the city.

Number of attributes:

**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.

**passenger\_count -** an integer indicating the number of passengers in the cab ride.

**1.2 DATA**

**Task** : to build regression to predict count based on the given data from train\_cab.csv and predict fare\_amounts in test.csv

**Target variable**

**fare\_amount:** fare charged to passengers

**Predictor variables**

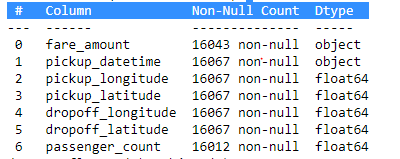
**pickup\_datetime** - timestamp value indicating when the cab ride started.

**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.

**passenger\_count -** an integer indicating the number of passengers in the cab ride.



**Chapter 2**

**Methodology**

**2.1 Pre Processing**

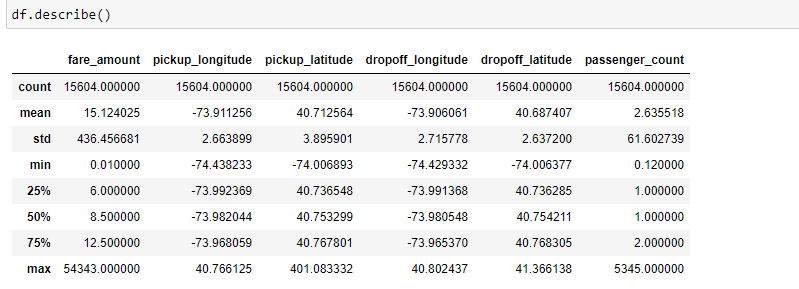
Any ML modeling requires that we look at the data before we start modelling. Looking at data refers to exploring the data, cleaning the data as well as visualizing the data through graphs and plots. This is often called as Exploratory Data Analysis.

To start this process we will first try and look at all the probability distributions of the variables. Most analysis like regression, require the data to be normally distributed. We can visualize that in a glance by looking at the probability distributions or probability density functions of the variable.

**Data cleaning** is an important step in any machine learning project and we need to remove variables that are not needed to create an ML model. Here we have 4 columns that has latitude and longitude values ,that tells us that data is for New York cab rental data. From latitude and longitude data, we can get distance.

This dataset needs to be cleaned. It has null values and other values which will prevent data from being predicted accurately.

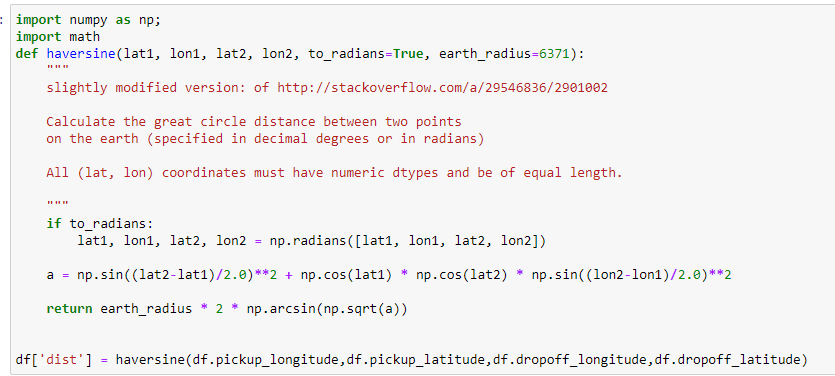
After removing null values

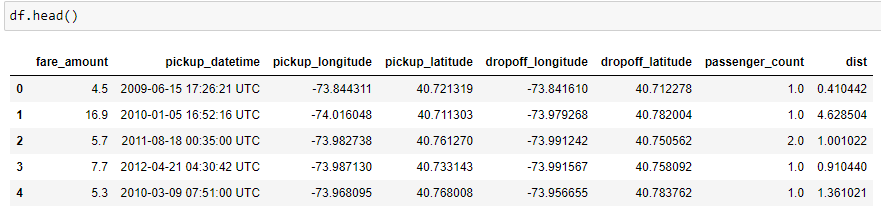


From latitude and longitude, we can calculate distance using haversine formula. The details of which are available in below link.

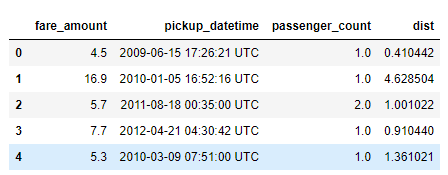
<https://en.wikipedia.org/wiki/Haversine_formula#:~:text=The%20haversine%20formula%20determines%20the,and%20angles%20of%20spherical%20triangles>.

A new column is made called ‘dist’ to predict fare charges.

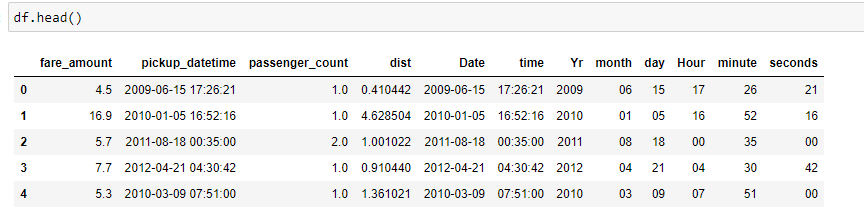


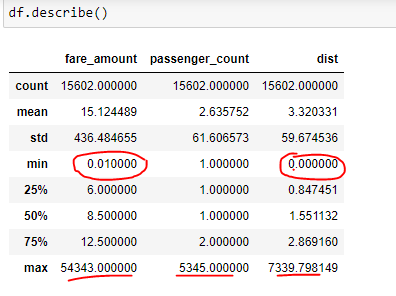


After calculating distance , we can drop the latitude and longtitude columns.

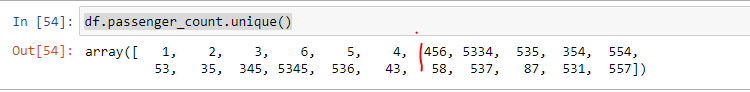


Now we need to strip the year, month, hour from pickup\_datetime . Because timing also affects fare charges.



****

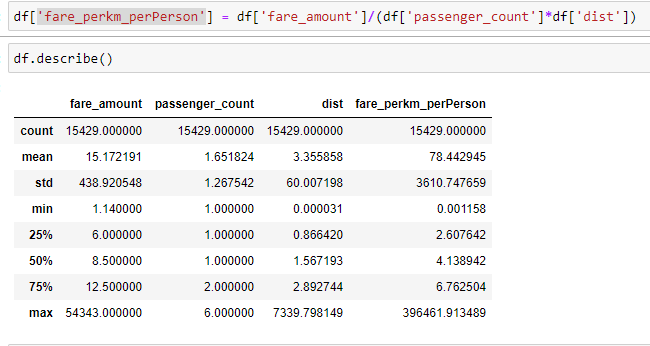
After further exploration, we see min fare\_amount = 0.01. Nobody would pay 1 cent to travel anywhere. These are outliers that need to be removed. Max fare\_amount = 54343 . Airline tickets are cheaper than this. These are outliers which need to be removed. The encircled values are all outliers which need to explored.



Passenger count > 6 would be considered as outliers because no cab can take more than 6 people in NY.

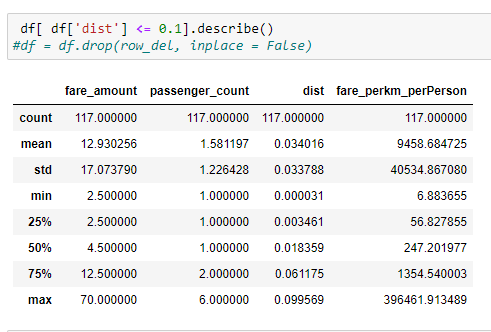
Also deleting rows where fare\_amount is less than $1.

For exploratory data analysis , we will create a new column **'fare\_perkm\_perPerson'**



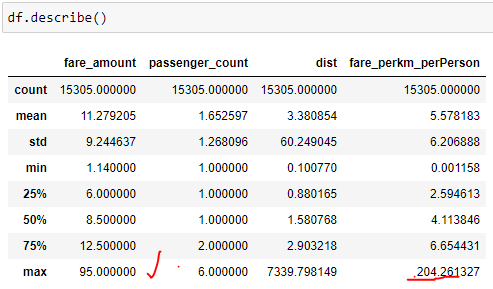
We can see min **'fare\_perkm\_perPerson'**  = 0.001 and max **'fare\_perkm\_perPerson'**  =369461.91.

Now we can see min **‘dist’ =** 0.000031 **– an impossible distance**

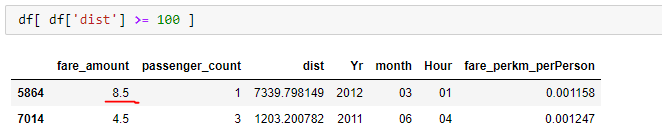
****

We can see that there are 117 rows with values less than 0.1 km.

So keep min\_dist = 0.1 , removing these values we can drastic change in max,average of the dataframe.

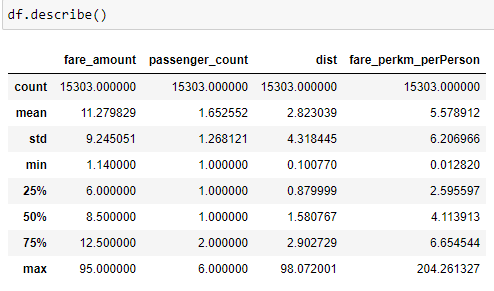


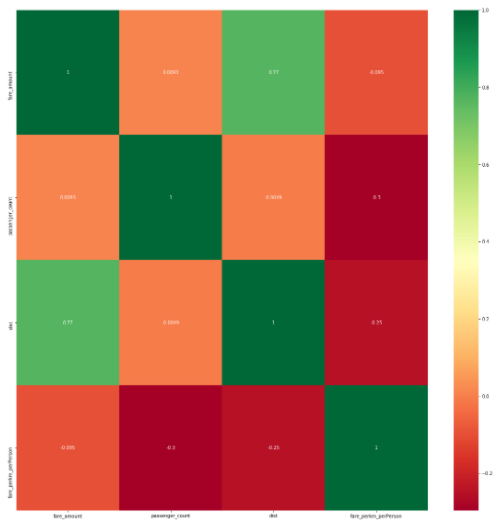
Now for further analysis , we can observe values where distance > 100

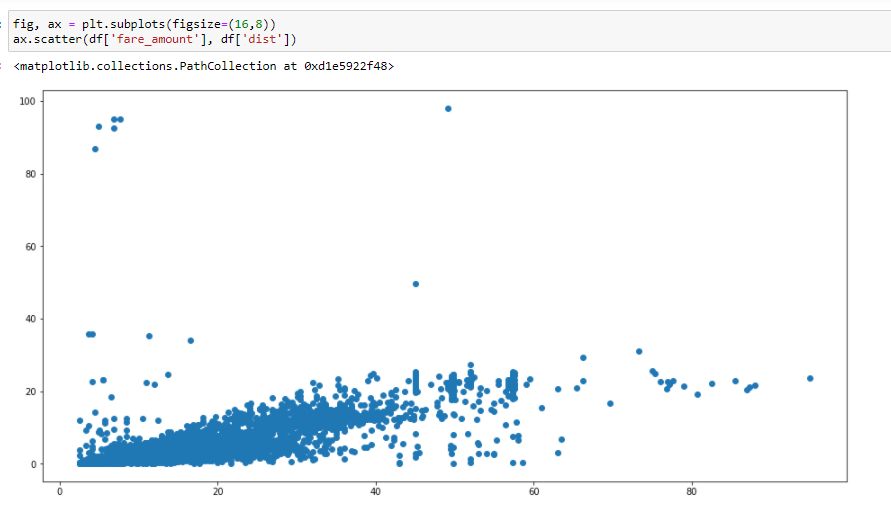


$8.5 – fare charge for 7339 kms is unreasonable .

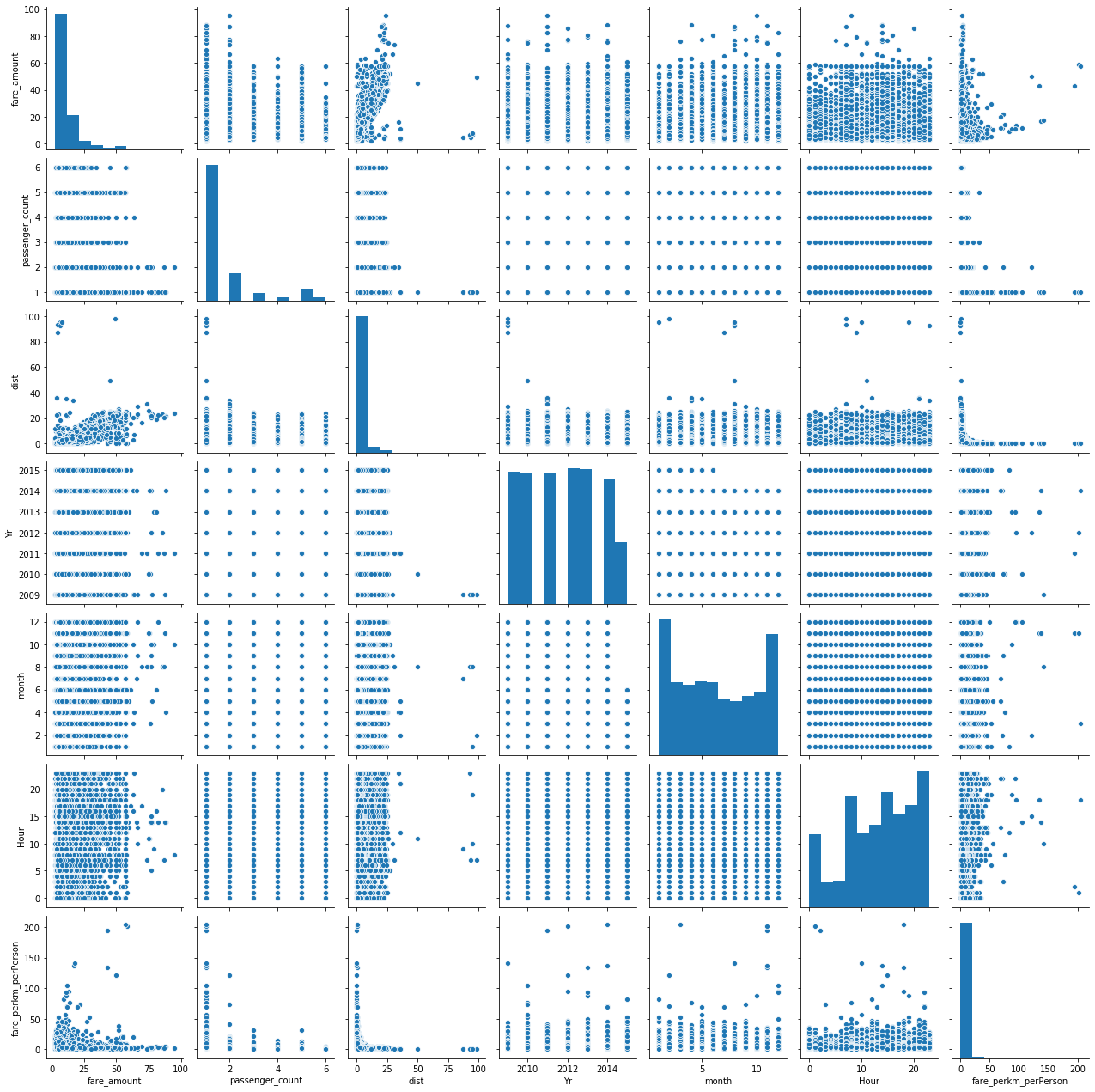
So these values need to be removed.



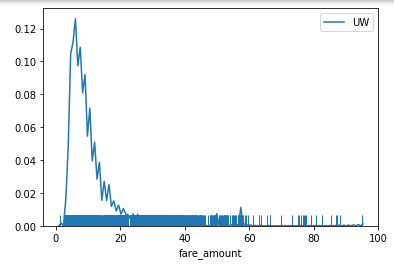


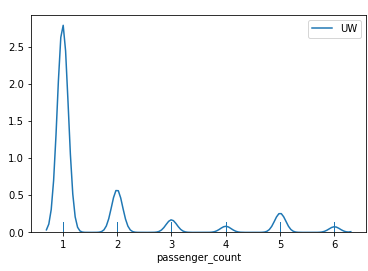


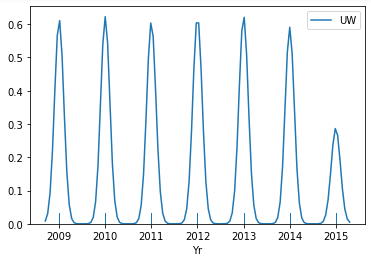
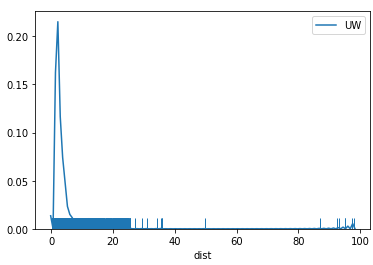
Now dist and fare\_charge have 0.77 correlation.

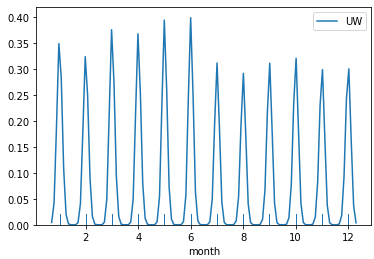


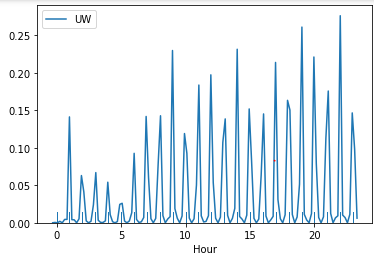
Above is the pairplot of the data frame.

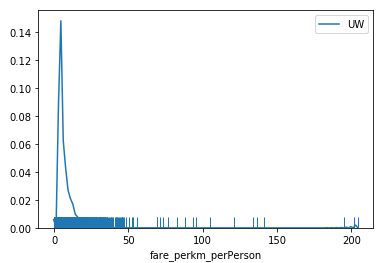










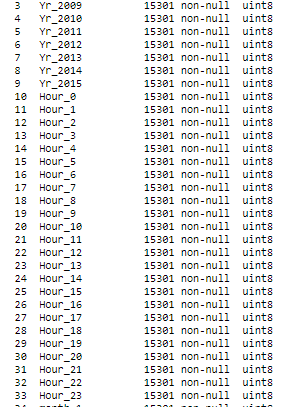


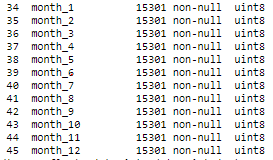
With this distribution plot and pair-plot, we can conclude that months and year and hour have importance in the fare charges.

**2.2One hot encoding**

Categorical data are variables that contain label values rather than numeric values and Categorical variables are often called nominal.

In day.csv , below given are already converted to numerical.



****

**3.Modelling**

The target variable is fare\_amount. These three columns are continuous variable. These is a regression type problem. Following algorithms can be used to predict regression problems.

* Simple Linear Regression model
* Lasso Regression
* Logistic regression
* Support Vector Machines
* Multivariate Regression algorithm
* Multiple Regression Algorithm

But our case includes predicting 3 target variables. For these cases, below mentioned methods are usually used.

* LinearRegression
* KNeighborsRegressor
* DecisionTreeRegressor
* RandomForestRegressor

For each ML algorithm, we will be finding r2\_score to predict how accurate is the model.

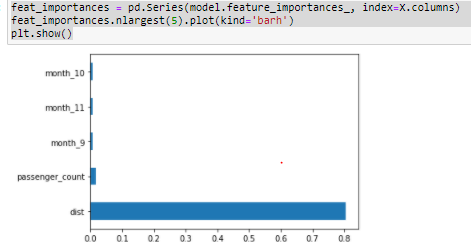
Definition is “(total variance explained by model) / total variance.” So if it is 100%, the two variables are perfectly correlated, i.e., with no variance at all. A low value would show a low level of correlation, meaning a regression model that is not valid, but not in all cases.

The best method as per r2\_score will be chosen as the best.

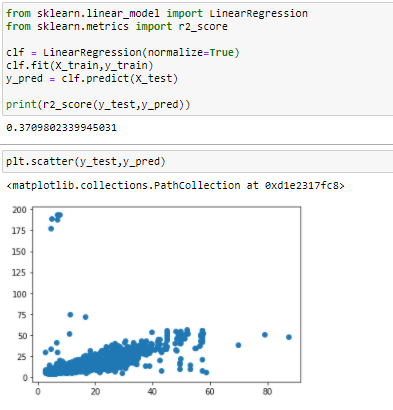
But for this model , we will also be using 3 metrics to find the best model – **MAE,MSE,RMSE**  as target variable and other predictor variables do not have Gaussisan distribution.

**Using train data , we will train models and find the appropriate moel for predicting values in test data.**

We will use ExtraTreesRegressor to find the features with highest importance.



**1)Linear regression**



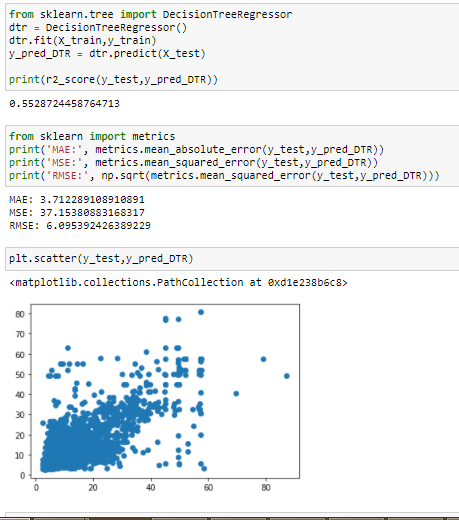
R2\_score -> 0.3709

MAE: 2.9536990563118812

MSE: 52.26803833042621

RMSE: 7.229663777135573

**2) Decision Tree Regressor**



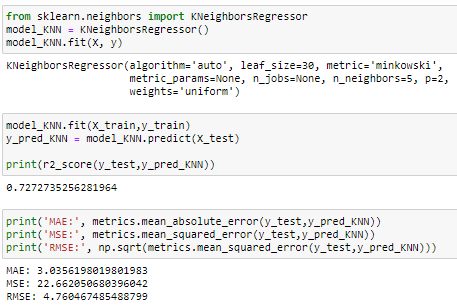
R2\_score - 0.5528724458764713

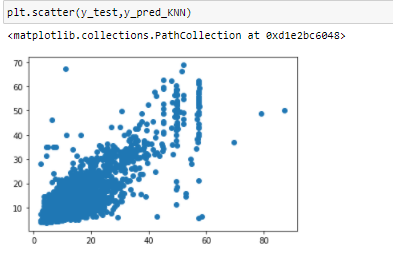
MAE: 3.712289108910891

MSE: 37.15380883168317

RMSE: 6.095392426389229

3)KNeighboursRegressor





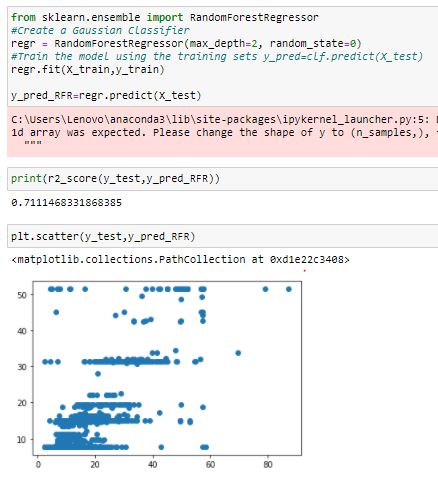
R2\_score - 0.7272735256281964

MAE: 3.0356198019801983

MSE: 22.662050680396042

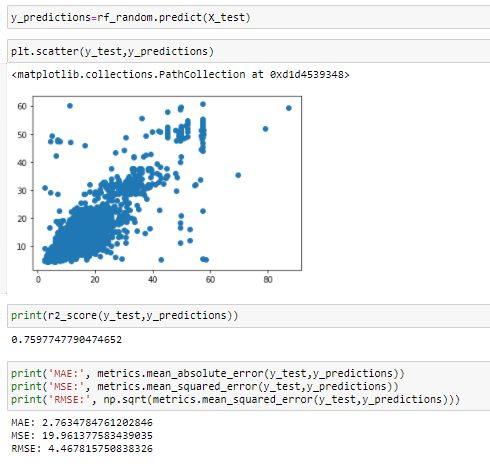
RMSE: 4.760467485488799

4) RandomForestRegressor



R2\_score -> 0.7111468331868385

5) Random Forest Regressor with randomized search CV



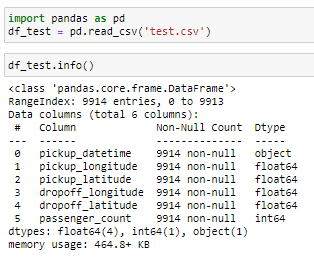
R2\_score - 0.7597747790474652

MAE: 2.7634784761202846

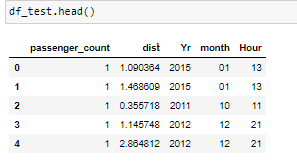
MSE: 19.961377583439035

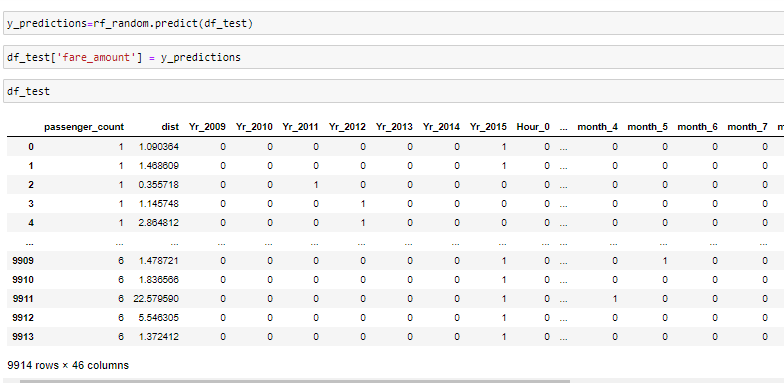
RMSE: 4.467815750838326

Hence based on this observation, we can conclude that RandomForest with randomized search cv is the best model for prediction this regression model.



After initial data wrangling as done for train\_data , this is the final dataset





df\_test.to\_csv('test\_result.csv')

The predicted variables are then stored to another csv file.

**4.Conclusion**

Out of the 5 algorithms tried, Random forest regressor with randomized search cv proves to be most efficient for this model which predicts target variable based on predictive performance and accuracy.

**5.Python CODE**

The python code is attached with zip file along with this document along with test\_result.csv.