Cab fare Prediction

Jinal Patel

21 April 2019

Contents

Introduction 3

1.1 Problem Statement 3

1.2 Data 3

Methodology 4

2.1 Pre Processing 4

2.1.2 Feature Extraction **8**

2.2 Modelling 16

2.2.1 Model Selection 16

2.2.2 Linear Regression 16

2.2.2 Decision Tree 17

2.2.2 Random Forest 17

Conclusion 17

3.1 Model Evaluation 17

3.1.1 Mean Absolute Percentage Error (MAPE) 17

3.1.1 Root Mean Square Error (RMSE) 18

3.2 Model Selection 18

References 20

Chapter 1

Introduction

1.1 Problem Statement

The objective of this Case is to predict the Cab fare provided 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.

The process of building the model is moved from simple to complex. Every model is supported by reason of acceptance or rejection. Special emphasis on the reasons why the algorithm has been picked/dropped is given.

1.2 Data

Our task is to build regression models which will predict the cab fare of a start-up company. Given below is a sample of the data set that we are using to predict the fare:

Table 1.1: Cab fare Prediction Sample Data (Columns: 1-4)

fare_amount	pickup_datetime	pickup_longitude	pickup_latitude
4.5	2009-06-15 17:26:21 UTC	-73.8443	40.72132
16.9	2010-01-05 16:52:16 UTC	-74.016	40.7113
5.7	2011-08-18 00:35:00 UTC	-73.9827	40.76127
7.7	2012-04-21 04:30:42 UTC	-73.9871	40.73314
5.3	2010-03-09 07:51:00 UTC	-73.9681	40.76801
12.1	2011-01-06 09:50:45 UTC	-74.001	40.73163

Table 1.2: Cab fare Prediction Sample Data (Columns: 5-7)

dropoff_longitude	dropoff_latitude	passenger_count
-73.8416	40.71228	1
-73.9793	40.782	1
-73.9912	40.75056	2
-73.9916	40.75809	1
-73.9567	40.78376	1
-73.9729	40.75823	1

As you can see in the table below we have the following 6 variables, using which we have to correctly predict the cab fare amount:

Table 1.4: Predictor Variables

S.No.	Predictor
1	pickup_datetime
2	pickup_longitude
3	pickup_latitude
4	dropoff_longitude
5	dropoff_latitude
6	passenger_count

Chapter 2

Methodology

2.1 Pre Processing

Any predictive modelling 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.** We start this process after importing the train_cab.csv file in python jupyter notebook or R Studio.

Before importing the data, having a look on the excel file pf data can help us draw many conclusions. After examining the visual data we have decided to skip 3 rows of improper format. One where the datetime format was incorrect and other where the fare amount format.

After importing the csv file we start cleaning the data. We have also the the test.csv file along with it.

The data comprises a date time column with UTC time zone. For proper analysis we can convert it to a datetime format so that we can further use it for future extraction.

Now we check for existing missing values in the dataframe. As mentioned there are missing value.

	Variables	Percentage
0	fare_amount	0.149402
1	pickup_datetime	0.000000
2	pickup_longitude	0.000000
3	pickup_latitude	0.000000
4	dropoff_longitude	0.000000
5	dropoff_latitude	0.000000
6	passenger_count	0.342380

After checking for the percentage of missing values, we can ignore those missing values by dropping it. We choose to drop it instead of imputation because these are incorrect observation and the percentage of such values is very less.

Now let's describe all the variables to figure out the extent for cleaning the data.

	fare_amount	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	passenger_count
count	15985.000000	15985.000000	15985.000000	15985.000000	15985.000000	15985.000000
mean	15.029829	-72.468885	39.918072	-72.468444	39.901166	2.623311
std	431.227426	10.558379	6.821952	10.555028	6.178521	60.894045
min	-3.000000	-74.438233	-74.006893	-74.429332	-74.006377	0.000000
25%	6.000000	-73.992145	40.734935	-73.991182	40.734659	1.000000
50%	8.500000	-73.981693	40.752603	-73.980168	40.753557	1.000000
75%	12.500000	-73.966824	40.767353	-73.963646	40.768005	2.000000
max	54343.000000	40.766125	401.083332	40.802437	41.366138	5345.000000

We see that the minimum fare amount is negative which can't be possible. Also the range of pickup_latitude is having a maximum value of 401.083332 which is incorrect. The range of latitudes should be in the interval of -90 to +90 whereas the range of longitudes is within -180 to +180. The minimum passenger count is 0 which is not possible. Also the maximum count is 5345 which is an outlier.

Simultaneously we can check our test data.

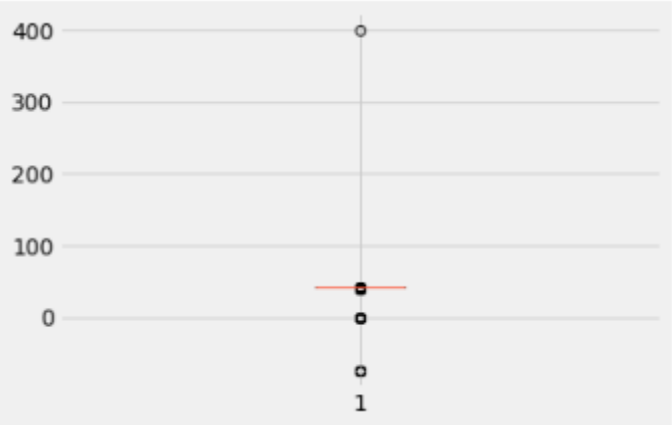
	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	passenger_count
count	9914.000000	9914.000000	9914.000000	9914.000000	9914.000000
mean	-73.974722	40.751041	-73.973657	40.751743	1.671273
std	0.042774	0.033541	0.039072	0.035435	1.278747
min	-74.252193	40.573143	-74.263242	40.568973	1.000000
25%	-73.992501	40.736125	-73.991247	40.735254	1.000000
50%	-73.982326	40.753051	-73.980015	40.754065	1.000000
75%	-73.968013	40.767113	-73.964059	40.768757	2.000000
max	-72.986532	41.709555	-72.990963	41.696683	6.000000

The test data is within proper range of latitude and longitude. Also the minimum and maximum count of passengers is within proper interval.

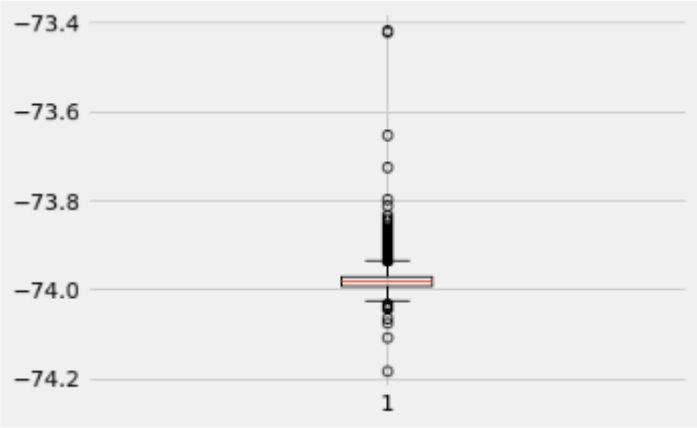
2.1.1 Outlier Analysis

We can clearly observe from the description of variables that there are numerous amount of outliers present in the data. Thus we check for the outliers in each and every variable.

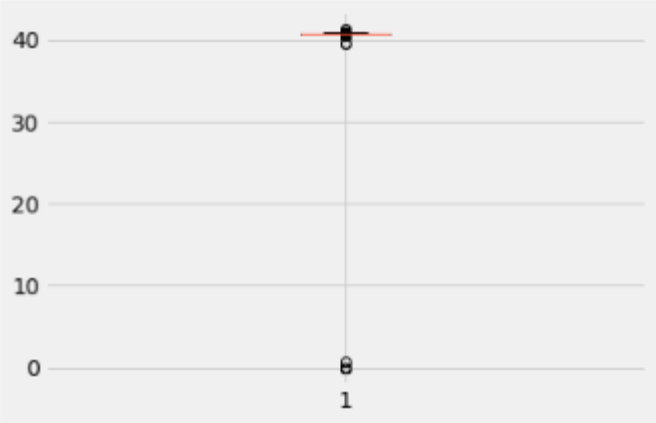
Outliers in Pickup Latitude



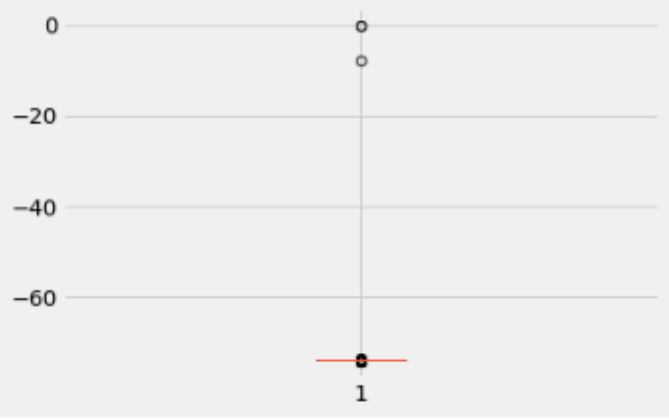
Outliers in Pickup Longitude



Outliers in Drop off Latitude



Outliers in Drop off Longitude

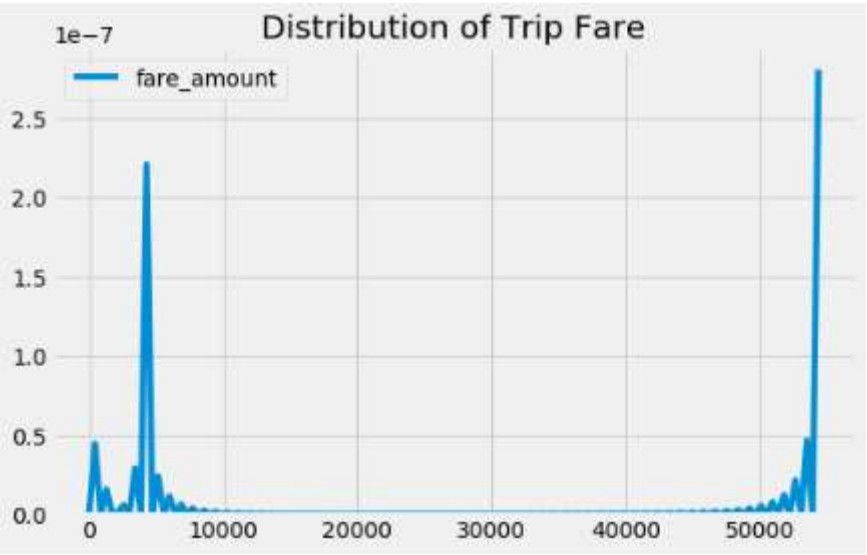


From outlier analysis we remove all the outliers present in the data.

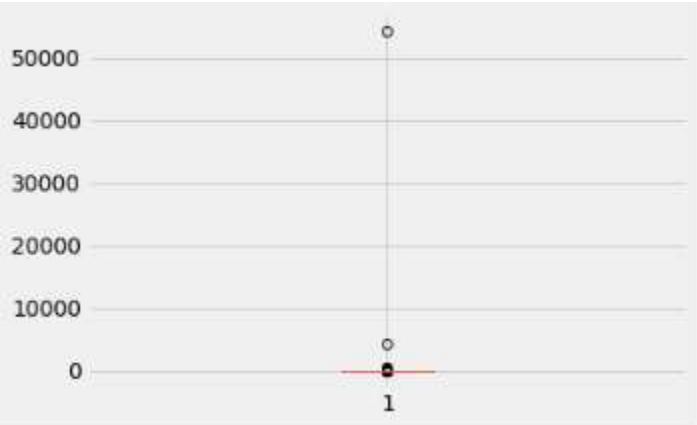
From the graph of outliers it is prominent that 0 lat and long is an outlier present in all the four variables. Thus double checking for the removal of all zeroes lat and long should be done. All such observations are removed due to outlier analysis.

After performing the outlier analysis on all latitude and longitude variables it is convenient to plot the heat map of these observations to figure out the data belongs to which city. After doing so we get to know that the data provided is of New York City.

Now we check the distribution of our target variable i.e. fare_amount.

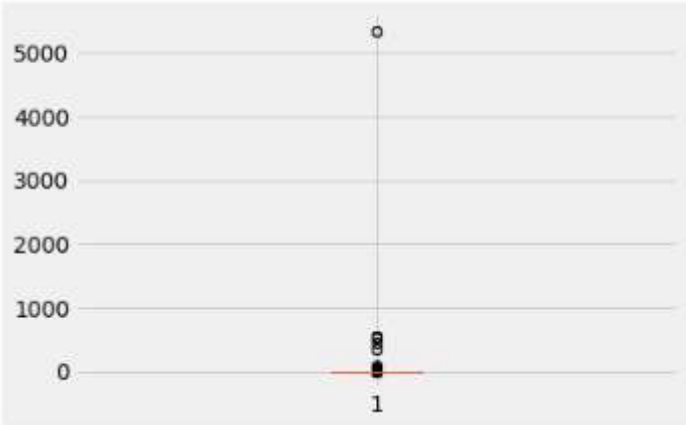


From the distribution of this variable it is obvious that the target variable has got outliers. We check its outlier box plot graph.



As confirmed, the target variable comprises of outliers and we should remove it.

Outliers in Passenger Count



There are outliers in passenger count as well. After finding the unique values of passenger count in dataframe we find the significant range for passenger count is from 1 to 6. Also the count less than 1 is found which are outliers and should be removed. The range of passenger count in test set is from 1 to 6.

2.1.2 Feature Extraction

The data set consist of datetime column from which we can derive many meaningful feature. The important characteristics while finding the cab fare is the amount of distance covered. Although we are not provided the amount of distance covered but we can find it using Haversine formula.

The haversine formula determines the great-circle distance between two points on a sphere given their longitudes and latitudes. It is important for use in navigation. The haversine can be expressed in trignometric function as:

$$haversine(\theta) = sin^2\left(\frac{\theta}{2}\right)$$

The haversine of the central angle (which is d/r) is calculated by the following formula:

$$\left(\frac{d}{r}\right) = haversine(\Phi_2 - \Phi_1) + cos(\Phi_1)cos(\Phi_2)haversine(\lambda_2 - \lambda_1)$$

where r is the radius of earth(6371 km), d is the distance between two points, \phi_1, \phi_2 is latitude of the two points and \lambda_1, \lambda_2 is longitude of the two points respectively.

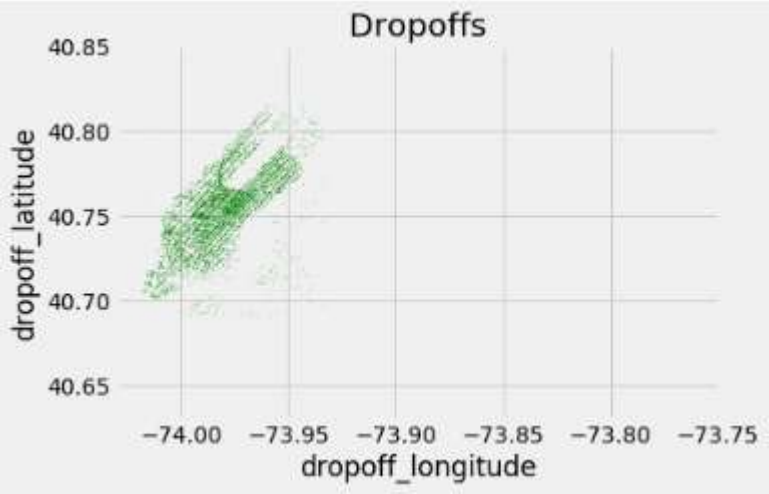
Solving d by applying the inverse haversine or by using the inverse sine function, we get:

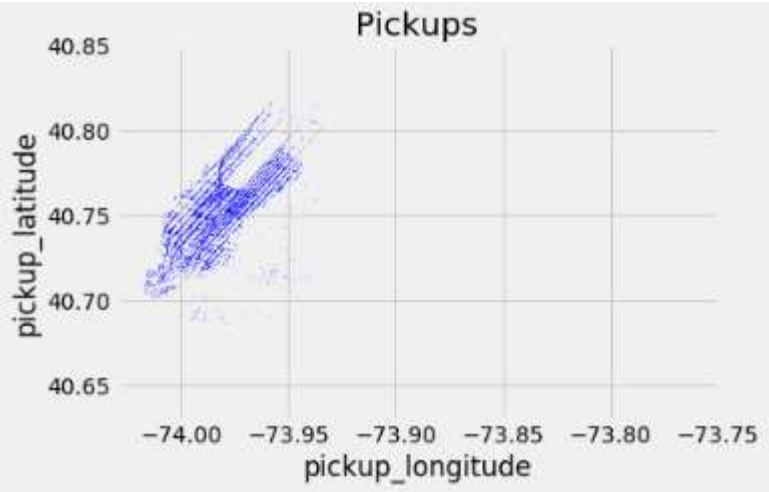
$$d = rhav^{-1}(h) = 2rsin^{-1}(\sqrt{h})$$

So a new feature is added by adding a new column to the data frame having the distance covered in kilometres.

Let's describe the dataframe we have. We see that the minimum distance covered between two points is 0. This is not a practical case and is possible when the pickup lat long are same as drop lat long. We remove such observations.

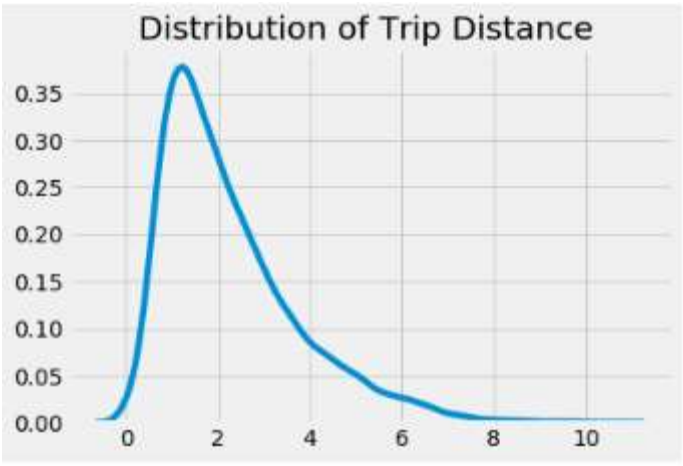
Let's plot the scatter plot of all pickup and drop off latitude longitudes.

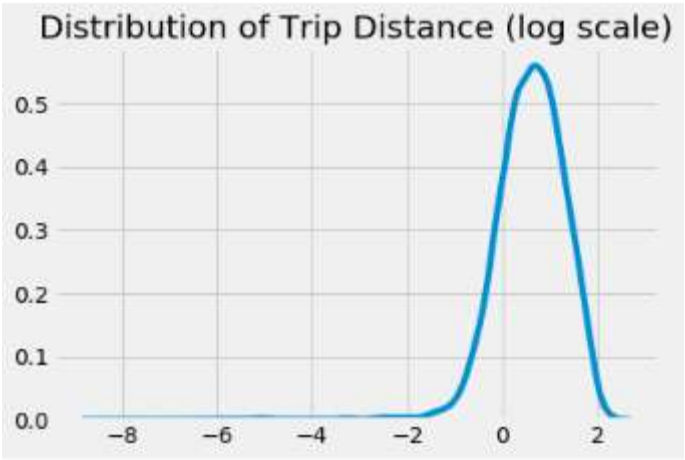




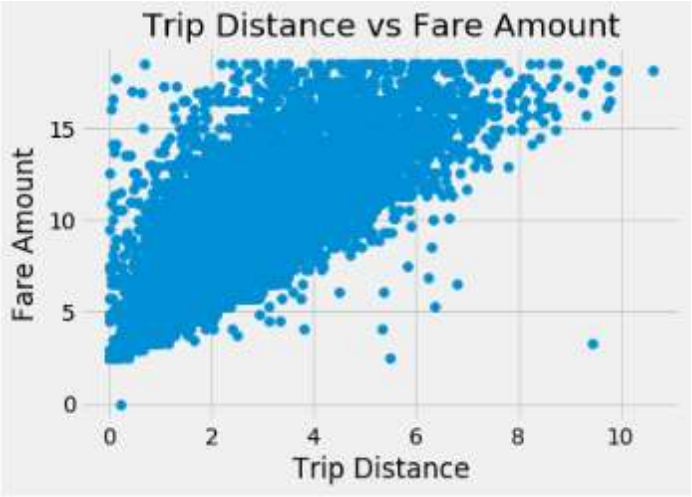
The scatter plot shows that most pickups and drops are from Manhattan in New York.

The distribution of the newly added feature distance is normally distributed.





Now let's see the scatter plot of distance covered against its fare amount.



We see a linear distribution of fare amount with the distance covered except for few cases where the fare amount is less even though the distance covered is large. There can be exceptions.

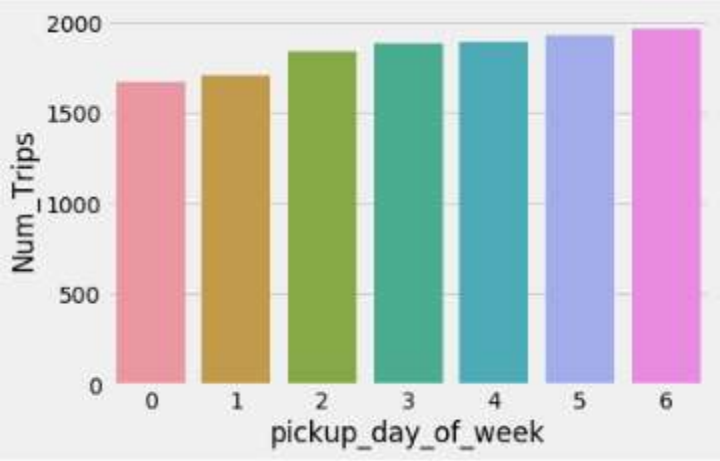
The derivation of important feature from date time column starts after its proper type conversion. We find many features from the date time column like day of pickup, hour of pickup, month of pickup, year of pickup, pickup day of week, late night pickup.

From all these features we encode pickup day of week from Sunday to Saturday I the range of 0 to 6. Also the feature late night which tells whether the pickup is done in late night or day is encoded to binary 0 and 1.

The passenger count variable is converted to type integer.

Simultaneously we extract the same features from our test dataset with the same type conversions.

The graph of number of trips per week days is shown.

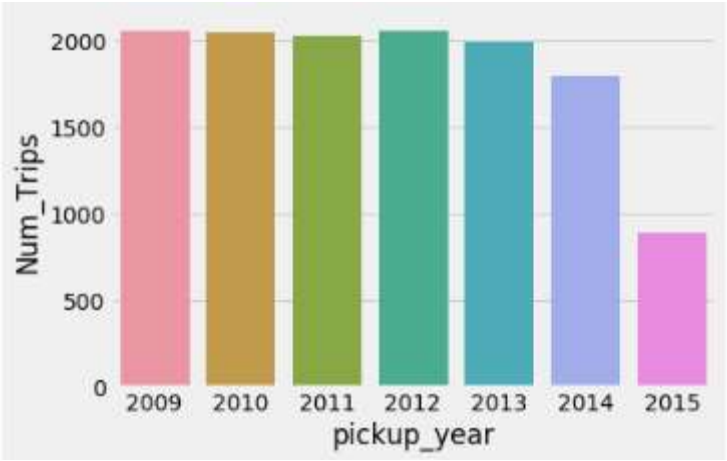


We see that there are less pickups on Sunday as its weekend.

Now the distribution of average cab fare per week day is shown.

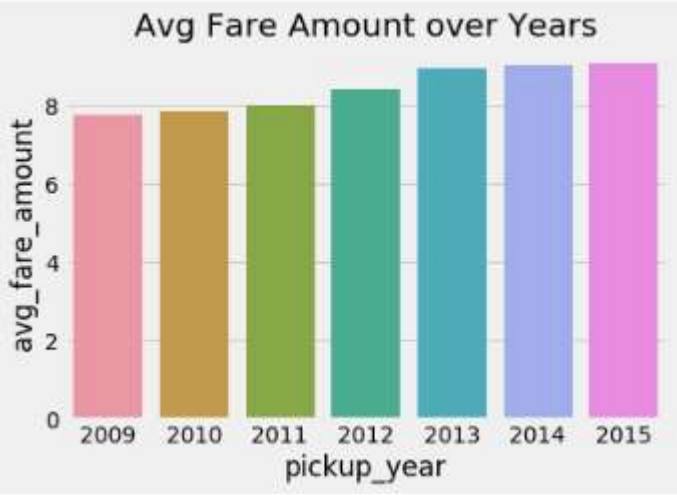
However there seems to be less variation in average fare amount in a week.

The graph of number of trips in different years is shown below.

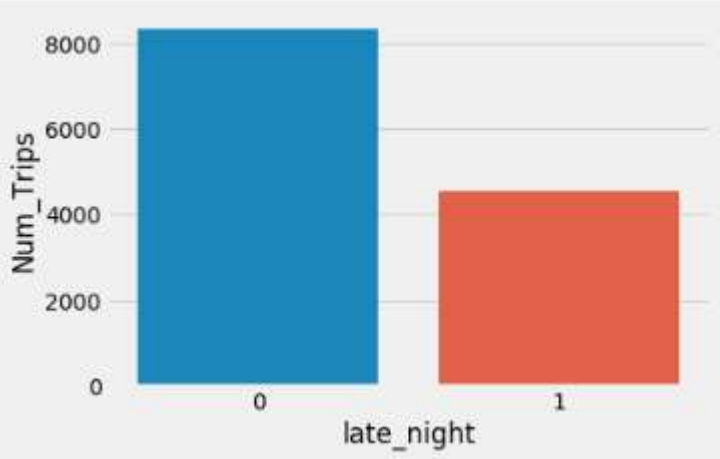


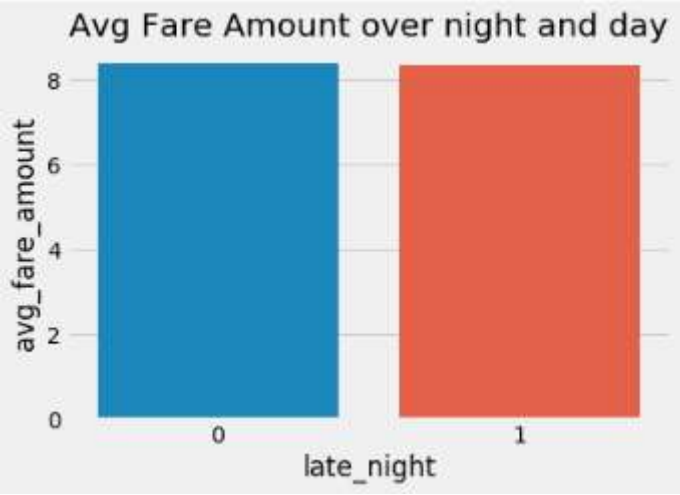
It has been decreasing maybe because people are preferring their own mode of transport.

The graph of average fare amount and different progressing years show that the fare has been increasing every year except from 2013 to 2014 where there is no change.

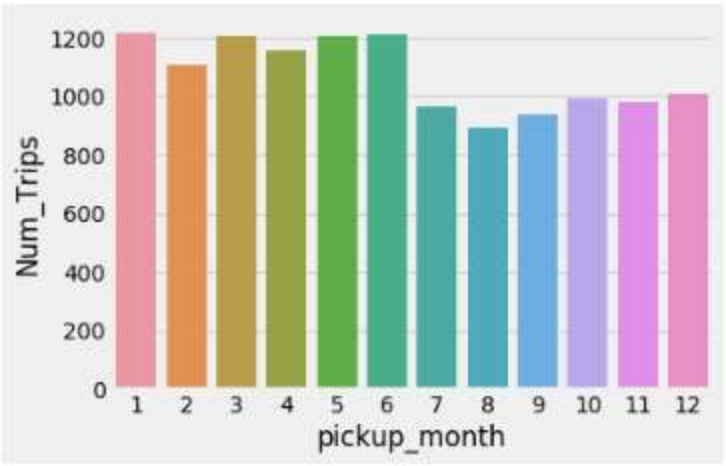


The graph of number of trips taken during day and night shows that there are more trips taken during day. Although the average fare amount is same.



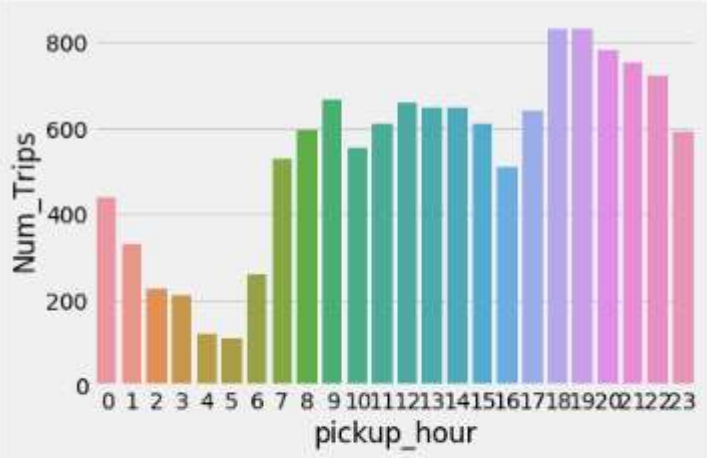


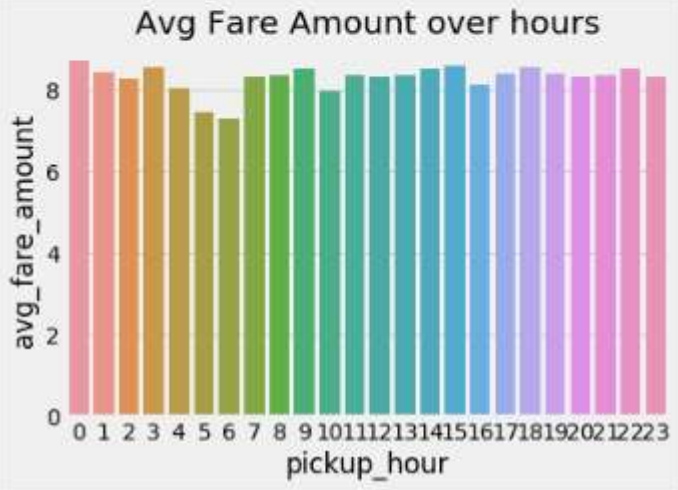
The graph of trips taken in different months is shown followed by the average fare amount per month.



There is no significant change in fare amount over different months.

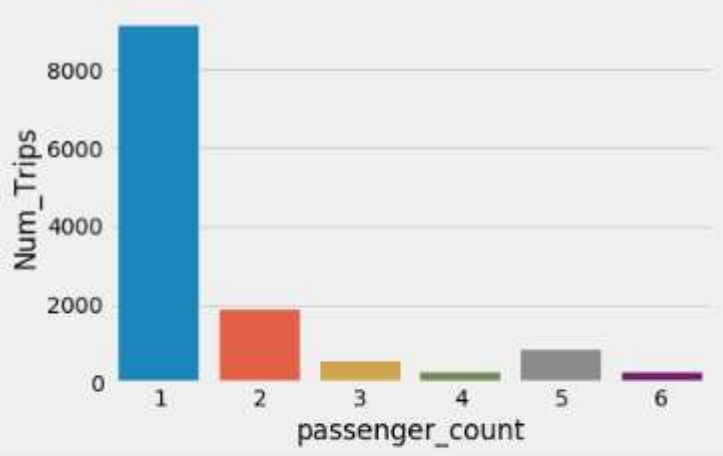
The graph of number of trips taken over different hours is shown followed by the average fare amount over different hours.





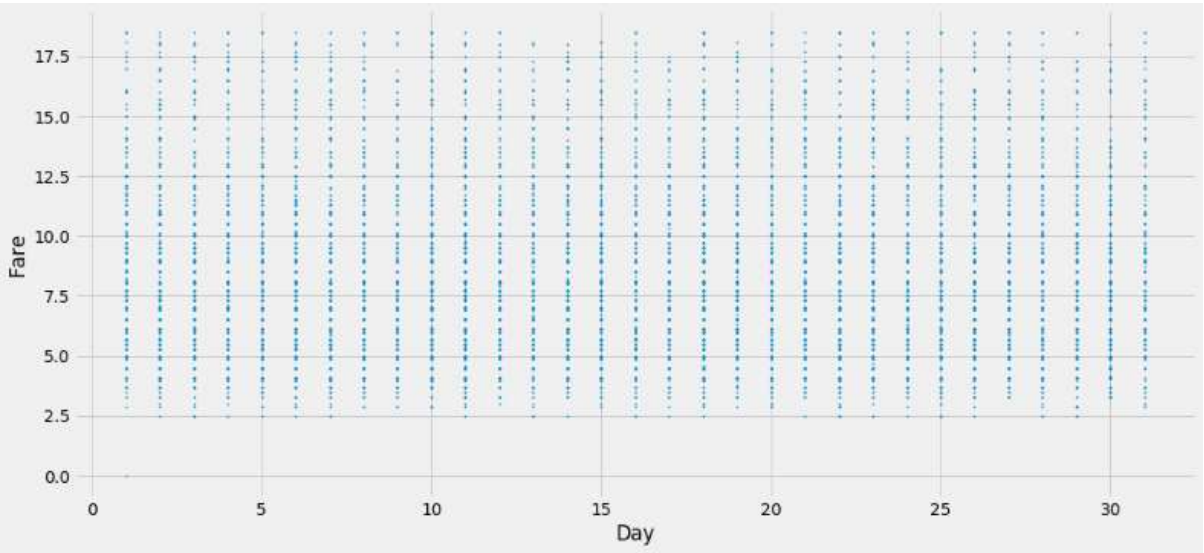
Less trips are taken in the morning although the fare amount is high.

The number of trips taken by different count of passenger implies that mostly the cabs are taken by 1 passenger.

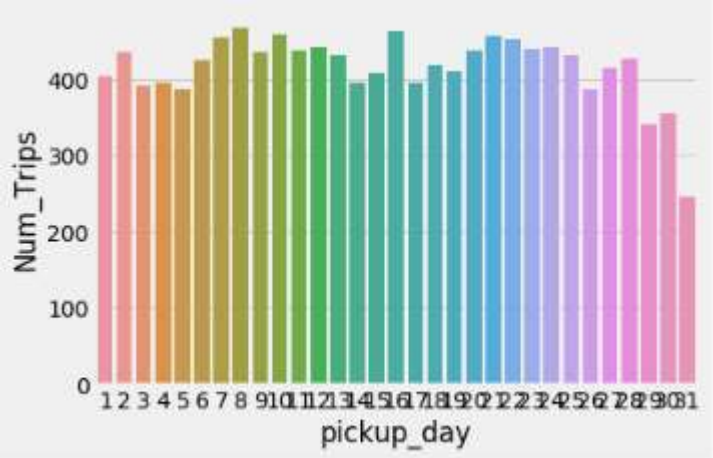


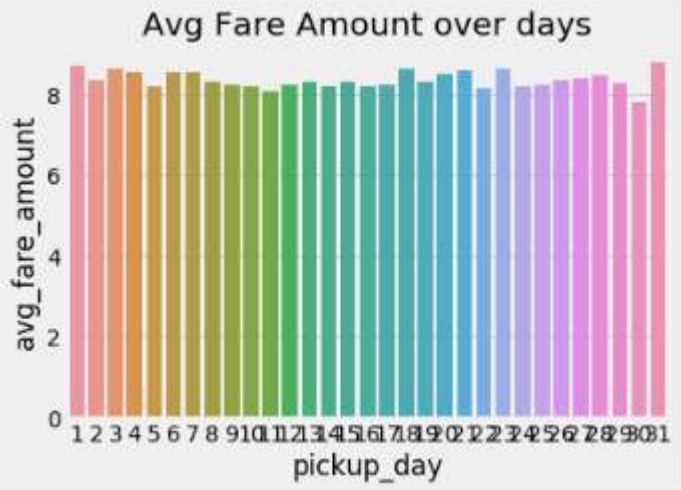
However the cab fare is highest for the count of 4 and 6 passengers.

A scatter plot of the number of trips taken per day is shown below.



Its variation against the number of trips is shown followed by the average fare amount per day.





2.2 Modelling

2.2.1 Model Selection

The important step before proceeding toward model selection is dividing the training set into training and validation sets to evaluate the model performance after model development.

We keep 75% data for training and the rest 25% for validating the model.

2.2.2 Linear Regression

The first step seeing the dependence of various features on the target variable shows a linearity in behaviour. So the simplest model we use to develop on our data is linear regression.

Before applying Linear Regression we find the variance inflation factor to find the multicollinearity in the data. After checking we found that that there is no collinearity present.

We predict the fare amount using linear regression.

2.2.2 Decision Tree

Decision tree can be used for both classification and regression. So we use Decision tree regression to predict the validation set after model development.

A decision tree is a flow-chart-like structure, where each internal (non-leaf) node denotes a test on an attribute, each branch represents the outcome of a test, and each leaf (or terminal) node holds a class label. The topmost node in a tree is the root node.

2.2.2 Random Forest

When the performance of decision tree doesn't suffice your model, give a try to random forest which aggregates numerous decision trees to give a better output. Here in case of regression, average of all the decision tree is taken to reach the conclusion.

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks that operates by constructing a multitude of decision trees at training time and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees. Random decision forests correct for decision trees' habit of overfitting to their training set.

Chapter 3

Conclusion

3.1 Model Evaluation

Now that we have a few models for predicting the target variable, we need to decide which one to choose. There are several criteria that exist for evaluating and comparing models. We can compare the models using any of the following criteria:

1. Predictive Performance
2. Interpretability
3. Computational Efficiency

In our case of Wine Data, the latter two, *Interpretability* and *Computation Efficiency*, do not hold much significance. Therefore we will use *Predictive performance* as the criteria to compare and evaluate models.

Predictive performance can be measured by comparing Predictions of the models with real values of the target variables, and calculating some average error measure.

3.1.1 Mean Absolute Percentage Error (MAPE)

The mean absolute percentage error (MAPE), 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:

$$\mathrm{M} = \frac{100\%}{n} \sum_{t=1}^{n} \left| \frac{A_t - F_t}{A_t} \right|$$

where At is the actual value and Ft is the predicted 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 predicted point in time and divided by the number of fitted points n. Multiplying by 100% makes it a percentage error.

3.1.1 Root Mean Square Error (RMSE)

Root Mean Square Error (RMSE) is the standard deviation of the residuals (prediction errors). Residuals are a measure of how far from the regression line data points are; RMSE is a measure of how spread out these residuals are. In other words, it tells you how concentrated the data is around the line of best fit. Root mean square error is commonly used in regression analysis to verify experimental results.

$$\mathrm{RMSD} = \sqrt{\frac{\sum_{t=1}^{n} (\hat{y}_t - y_t)^2}{n}}.$$

It squares the error, finds their average and then takes square root.

3.2 Model Selection

We can see the performance of all the three models evaluated on the described evaluation criteria.Apart from the model evaluation criteria described above we check the R^2 value for all the models.

R-squared is a statistical measure of how close the data are to the fitted regression line. It is also known as the coefficient of determination, or the coefficient of multiple determination for multiple regression.

The definition of R-squared is fairly straight-forward; it is the percentage of the response variable variation that is explained by a linear model. Or:

R-squared = Explained variation / Total variation

R-squared is always between 0 and 1.

It explains how much variance is explained by the trained data.

From the linear dependence of independent variables and the target variable we choose linear regression. By its nature, linear regression only looks at linear relationships between dependent and independent variables. That is, it assumes there is a straight-line relationship between them. Sometimes this is incorrect. There are exceptions where the fare amount won't increase linearly with distance covered. Outliers can have huge effects on the regression. Because of the outliers in our data, this model can't fit the test set to the extent of better predictability.

Then we chose to fit our model using decision tree regression. However it couldn't perform better than the Linear Regression Model based on the error metric. While Decision Trees are generally robust to outliers, due to their tendency to over fit, they are prone to sampling errors. At each level, tree looks for binary split such that impurity of tree is reduced by maximum amount. This is a greedy algorithm and achieves local optima.

So finally we give a try to Random Forest. Random forests overcome several problems with decision trees, including reduction in overfitting by averaging several trees, there is a significantly lower risk of overfitting. Also there is less variance as by using multiple trees, you reduce the chance of stumbling across a classifier that doesn't perform well because of the relationship between the train and test data.

As a consequence, in almost all cases, random forests are more accurate than decision trees.

The comparison of different models with their model evaluation parameters in Python and R is shown below.

Evaluation Parameters in Python

Evaluation Parameters	Linear Regression	Decision Tree	Random Forest
MAPE	17.98278254	23.62708276	16.76821218
RMSE	1.969662873	2.62207572	1.867160834
Accuracy	82.01721746	76.37291724	83.23178782

Evaluation Parameters in R

Evaluation Parameters	Linear Regression	Decision Tree	Random Forest
MAPE	17.69967	19.53188	16.35455
RMSE	1.8958751	2.0448274	1.7548357
Accuracy	82.30033	80.46812	83.64545

We see that MAPE, RMSE and Accuracy is best for Random forest, then linear regression and then decision tree. Also the variance is best explained by the Random Forest.

From this we can conclude that the best model we select amongst the three should be Random Forest.

We use the Random Forest model to predict the test set.

References

https://www.researchgate.net/publication/324706525_Taxi_Fare_Rate_Classification_Using_Deep_Networks

https://editorialexpress.com/cgi-bin/conference/download.cgi?db_name=IIOC2018&paper_id=408

http://beatbamboo.marketing/udbl/new-york-city-taxi-fare-prediction-github.html