|  |
| --- |
| **Cab-fare prediction** |
| **Project Report** |
| **Kunal Pandya** |

Table of contents

**• Introduction………………………………………………………….…..3**

**1. Understanding data…………………………………………….…….3**

**2. Data Pre processing…………………………………………….…….4**

2.1 Data Conversion & Feature Engineering….……….4

2.2 Missing Value Analysis………………………………………5

2.3 Outlier Analysis…………………………..……………………7

2.4 Feature Selection……………………..………………………9

2.5 Feature Scaling…………………………………………………12

**3. Modeling……………………………………………………………….14**

3.1 Linear Regression…………………………………………….14

3.2 Decision Trees..……………….……………………………..16

3.3 Random Forest……………….………………………………18

**4. Model Selection……………………….………………………………20**

**5. Model Fitting & Conclusion……………………………………..21**

**Introduction**

**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.

**1. Understanding data** - Two datasets are provided namely “train\_cab” & “test” which contains variables as

1. fare\_amount

2. pickup\_datetime

3. pickup\_latitude & longitude

4. dropoff\_latitude & longitude

5. passenger\_count

The test dataset excludes the ‘fare\_amount’ & the task is to predict the value of ‘fare\_amount’ column in the test dataset.

As the predicted results should be continuous, this is case of regression problem. A machine learning model has to be built on the train data and to be applied on test data to predict the ‘fare\_amount’. As this is a regression problem, models such as Linear regression, Decision tree & Random forest algorithm can be applied. To choose the best model, the train dataset can be divided into subset train & subset test data. After building models on the subset train, it can be applied to the subset test data to check the accuracy & other evaluation metrics. The best model can be selected considering these metrics and it can be applied to the main test dataset to find the ‘fare\_amount’.

2. Data Pre-processing

**2.1 Data Conversion & Feature Engineering**

To convert the features to appropriate data types, the default data type is checked first and the results are as given:

fare\_amount – factor/object

pickup\_datetime – factor/object

pickup\_latitude & longitude – numeric/float64

dropoff\_latitude & longitude – numeric/float64

passenger\_count – numeric/float64

Among these, ‘fare\_amount’ can be categorized as continuous and hence converted to numeric/float64 type.

‘passenger\_count’ can be categorized as both categorical & continuous, but on visual checking of the data in excel, it was easily found that there are many outliers present in the ‘passenger\_count’. As outlier removal can be done only on continuous data, ‘passenger\_count’ was kept as numeric/float64.

‘pickup\_datetime’ itself is difficult to comprehend but can be a very useful if valuable features are extracted. Feature engineering was carried out and new variables such as ‘date’, ‘year’, ‘months’, ‘weekdays’ & ‘time (in hours)’ were extracted. From these, ‘date’ & ‘year’ were converted to numeric and ‘months’, ‘weekdays’ & ‘time’ were converted to factors.

2.2 Missing Value Analysis

**Theory**

Missing values occur when no data is recorded for an observation; it was intended to make an observation, but because of some reason did not. Missing data are a common occurrence and can have a significant effect on the statistical analysis. If the missing values are not handled properly by the researcher, then he/she may end up drawing an inaccurate inference about the data. Due to improper handling, the result obtained by the researcher will differ from ones where the missing values are present.

If the total missing values of a feature is greater than 30% of the total observations, then the missing observations may be removed from original data. If they are lesser than 30%, then it is suggested to impute them with methods such as mean, mode, median, knn imputation methods.

**Methodology**

• Convert the negative values to zero in the variable ‘fare\_amount’ as it cannot be negative in real case scenario

• Replace all the zeroes with NA / NaN

• Create a data frame with total no of missing values for each variable

• Calculate percentage of missing values against each variable

• Sort in descending order

• In this project, missing values are decided to be imputed irrespective of their percentage, to avoid information loss

Observations & Results

After replacing the negative values in the feature ‘fare\_amount’, the total number of ‘NAs’ was found to be 85.

Zeroes were converted to ‘NAs’ and then the number was increased to 1402. After testing with mean, median & knn methods, the missing values were finally imputed with the mean method.

**2.3 Outlier Analysis**

**Theory**

An outlier is an observation that is abnormal compared to other observations in that dataset. One of the most important tasks from large data sets is to find an outlier because outliers can significantly alter the results even though they are present in small proportions.

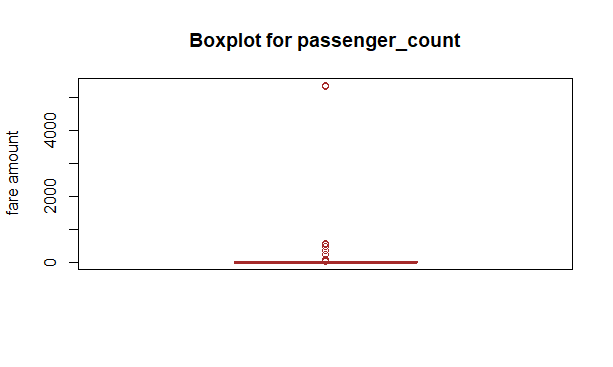
To find an outlier, inter quartile range (IQR) is found first. IQR represents the middle 50% of the data. The position of first quartile can be found using formula (N+1)/4 & third quartile can be found by 3\*(N+1)/4 where N is the total no. of observations. The difference of the values in the first & third quartiles is the IQR. If any observation falls below 1.5 times IQR from the first quartile value, or if it falls above 1.5 times IQR from the third quartile value, then the value can be qualified as an outlier.

Outliers can be found using box plot method which can be plotted in both R & Python. After finding the outliers, they can be removed from the dataset or they can be imputed by KNN method.

Observations & Results

Original train data contained a total of 16067 observations. After removing the outliers, KNN imputation was done in R, but due to some technical issue, KNN imputation could not be done in Python. Thus, the final observations in Python were 11878. So, a sum of 4189 outliers were detected from all variables and imputed through KNN in R & eliminated in Python from the train data.

**One of the box-plot is shown below:**



**It could be concluded from the box plot that most of the observations were very close to each other and only few outliers were present.**

2.4 Feature Selection

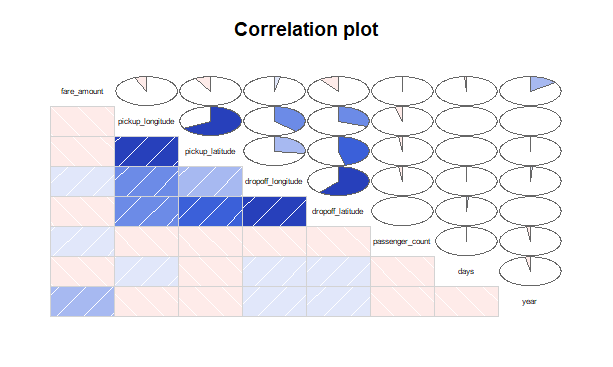
**Theory**

Feature Selection is the process of selecting those features which contribute most to the prediction variable. Having irrelevant features in data can decrease the accuracy of the models and make the model learn based on irrelevant features. Correlation analysis is used in feature selection for numerical variables and chi square test is used for categorical variables.

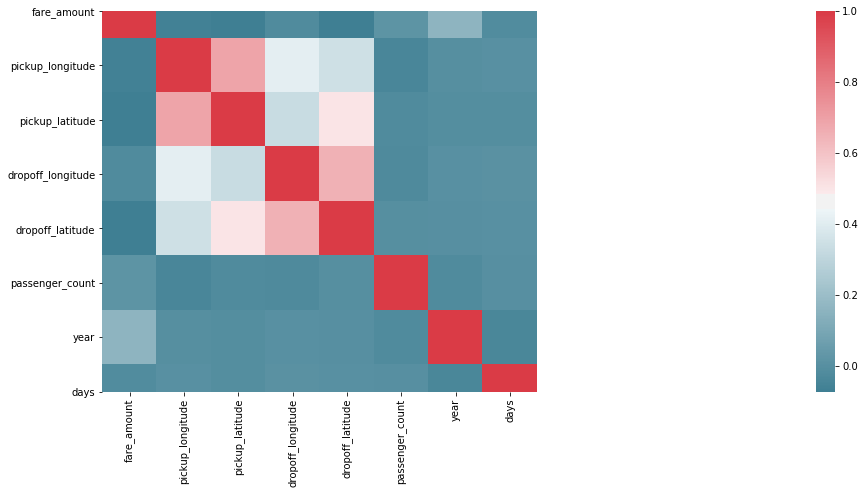
In correlation analysis, correlation coefficient is calculated between two variables which ranges from -1 to +1. Correlation coefficient approaching -1 or +1 means that both the variables are strongly correlated (negatively & positively correlated respectively). While value close to 0 implies little or no correlation. In chi square test, a null hypothesis is formulated which states that the given two variables are independent of each other and an alternate hypothesis states that the two variables are not independent. A critical value is found using chi square value and degree of freedom, which if less than chi square value, alternate hypothesis is accepted and if greater than chi square, null hypothesis is accepted.

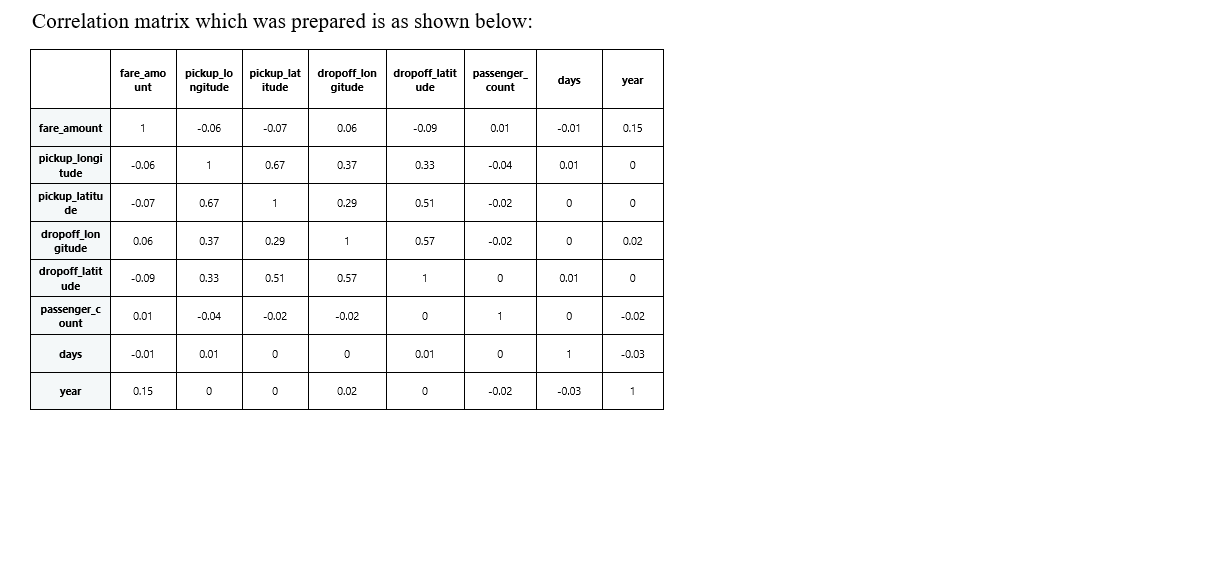
Observations & Results

In R, correlation plot was made as shown below:



**In python, heat-map of correlation plot was made and is as shown below:**

****

****If correlation value is greater than 0.8 or less than -0.8, then it can be safely assumed that the two variables in consideration are highly correlated and one of them may be dropped. But, from correlation matrix in R, all the values were between the range -0.8 & 0.8, which indicates that they are independent of each other. Similar conclusion can be drawn from the heat map & correlation plot, in which low shade of red color which indicates dependency of the variables, was found. Hence, none of the numeric features were eliminated from the train data.

**2.5 Feature Scaling**

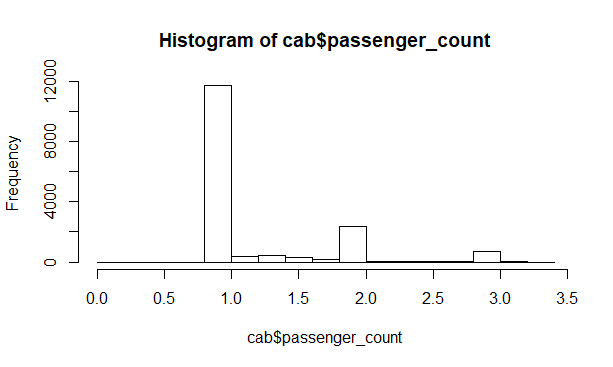
**Theory**

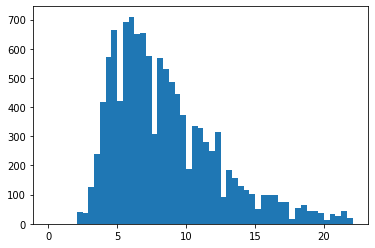
Feature scaling means adjusting data that has different scales into the same range. Feature scaling is an important technique in Machine Learning and it is one of the most important steps during the preprocessing of data before creating a machine learning model. Most of the times, the dataset contains features highly varying in magnitudes, units and range. The two most important scaling techniques is Standardization and Normalization.

Normalization is the process of rescaling the features to the range of 0 to 1. Standardization is the process of rescaling data to have a mean of 0 and a standard deviation of 1. This is usually applied to the dataset which is normally distributed.

**Observations & Results**

**Histograms were plotted to check the normality of the data and the plots of some of the variables are as follows:**

****

****

From the above histograms, it can be concluded that the data is not normally distributed. So, to scale the data, normalization was applied.

**Range of the data before normalization = ( -74.03215, 40.81788 )**

**Range of the data after normalization = ( 0, 1)**

3.Modeling

**3.1 Linear Regression**

**Theory**

Linear regression is the simplest and most widely used statistical technique for predictive modeling. It basically gives us an equation, where we have our features as independent variables, on which our target variable is dependent upon. Linear regression is a linear model, e.g. a model that assumes a linear relationship between the input variables (x) and the single output variable (y). More specifically, that y can be calculated from a linear combination of the input variables (x). When there is a single input variable (x), the method is referred to as simple linear regression. When there are multiple input variables, literature from statistics often refers to the method as multiple linear regression. The equation of regression represents a straight line and the line of best fit is chosen so as to minimize the erros

**Observations & Results**

MAPE = mean( (actual value – predicted value) / actual value ) \* 100

Accuracy = 1 – MAPE

RMSE = square root (mean ( (actual value – predicted value) ^ 2 ) )

**In R,**

**MAPE = 39.69**

**Accuracy = 60.31**

**RMSE = 0.182**

**In Python,**

**MAPE = 40.42**

**Accuracy = 59.58**

**RMSE = 0.188**

**3.2 Decision Tree**

**Theory**

Decision tree is a type of supervised learning algorithm that is mostly used in classification problems. It works for both categorical and continuous input and output variables. In this technique, the population or sample is split into two or more homogeneous sets based on most significant splitter / differentiator in input variables.

The decision of making strategic splits heavily affects a tree’s accuracy. The decision making criteria is different for classification and regression trees. Decision trees use multiple algorithms to decide to split a node in two or more sub-nodes. The creation of sub-nodes increases the homogeneity of resultant sub-nodes. Decision tree splits the nodes on all available variables and then selects the split which results in most homogeneous sub-nodes.

**Observations & Results**

MAPE = mean( (actual value – predicted value) / actual value ) \* 100

Accuracy = 1 – MAPE

RMSE = square root (mean ( (actual value – predicted value) ^ 2 ) )

**In R,**

**MAPE = 36.49**

**Accuracy = 63.51**

**RMSE = 0.169**

**In Python,**

**MAPE = 38.36**

**Accuracy = 61.64**

**RMSE = 0.162**

**3.3 Random Forest**

**Theory**

Random forest consists of a large number of individual decision trees that operate as an ensemble. Each individual tree in the random forest gives a class prediction and the class with the most votes becomes the model’s prediction. Random forest works on the principle that a large number of relatively uncorrelated models (trees) operating as a group will outperform any of the individual constituent models. While some trees may be wrong, many other trees will be right, so as a group the trees are able to move in the correct direction. The predictions (and therefore the errors) made by the individual trees need to have low correlations with each other.

Observations & Results

MAPE = mean( (actual value – predicted value) / actual value ) \* 100

Accuracy = 1 - MAPE

RMSE = square root (mean ( (actual value – predicted value) ^ 2 ) )

**In R,**

**MAPE = 27.25**

**Accuracy = 72.75**

**RMSE = 0.134**

**In Python,**

**MAPE = 0.012**

**Accuracy = 99.988**

**RMSE = 0.0004**

**4.Model Selection**

**Model selection was done on the basis of given evaluation metrics which are summarized in the tabular form as given below:**

|  |  |  |  |
| --- | --- | --- | --- |
| **MODEL-R** | **MAPE** | **ACCURACY** | **RMSE** |
| **Linear-regression** | **39.69** | **60.31** | **0.182** |
| **Decision tree** | **36.49** | **63.51** | **0.169** |
| **Random forest** | **27.25** | **72.75** | **0.134** |

|  |  |  |  |
| --- | --- | --- | --- |
| **Model- Python** | **MAPE** | **ACCURACY** | **RMSE** |
| **Linear-regression** | **40.42** | **59.58** | **0.188** |
| **Decision tree** | **38.36** | **61.64** | **0.162** |
| **Random forest** | **0.012** | **99.98** | **0.0004** |

**From the table, it can be seen that Random forest scores better than the other 2 algorithms in all aspects. Both MAPE & RMSE are low for Random forest than Linear regression & Decision tree models. Also, the accuracy is also better than both of them. So, it can be easily concluded that Random forest is the better option among the three models.**

**5.Model Fitting & Conclusion**

**After the selection of the best possible model, it was fit to the large test dataset for which the ‘fare\_amount’ was to be predicted. Data pre- processing was also done on the test data for maximum accuracy. No missing observation was found in missing value analysis. Feature scaling was also done because the original train dataset was trained on the scaled data, thus the predicted results would be accurate only if the model fitting is done on the scaled test data. After fitting the model, the results were in the range from 0 to 1 as the whole data was normalized.**