Taxi Fare Prediction

10.11.2020

Akshay Rahate

Contents

1 Introduction....3
1.1 Problem Statement....3
1.2 Data....3
1.3 High Level Architecture....4
2 PreProcessing....5
2.1.1 Missing Value Analysis....5
2.1.2 Haversine Formula....6
2.1.3 KNN Imputation....6
2.1.4 Outlier Analysis....7
3 Exploratory Data Analysis....13
3.1.1 Correlation Analysis....14
3.1.2 Count Plots....18
3.1.3 Joint Plots....20
3.1.4 Seasonality Analysis....25
3.1.5 Bar Plots....25
4 Model Building....28
4.1.1 Log Transform....28
4.1.2 Linear Regression....29
4.1.3 Decision Tree Regressor....30
4.1.4 Random Forest Regressor....30
4.1.5 Gradient Boosting Regressor....31
5 Model Evaluation....32
6 Hyperparameter Tuning....35
7 Model Selection....36
8 Conclusion....36

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 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 objective of this project is to predict the Cab fare amount for the new test case , by analysing the given Historical Data.

1.2 Data

The given data attributes are

Train_cab.csv

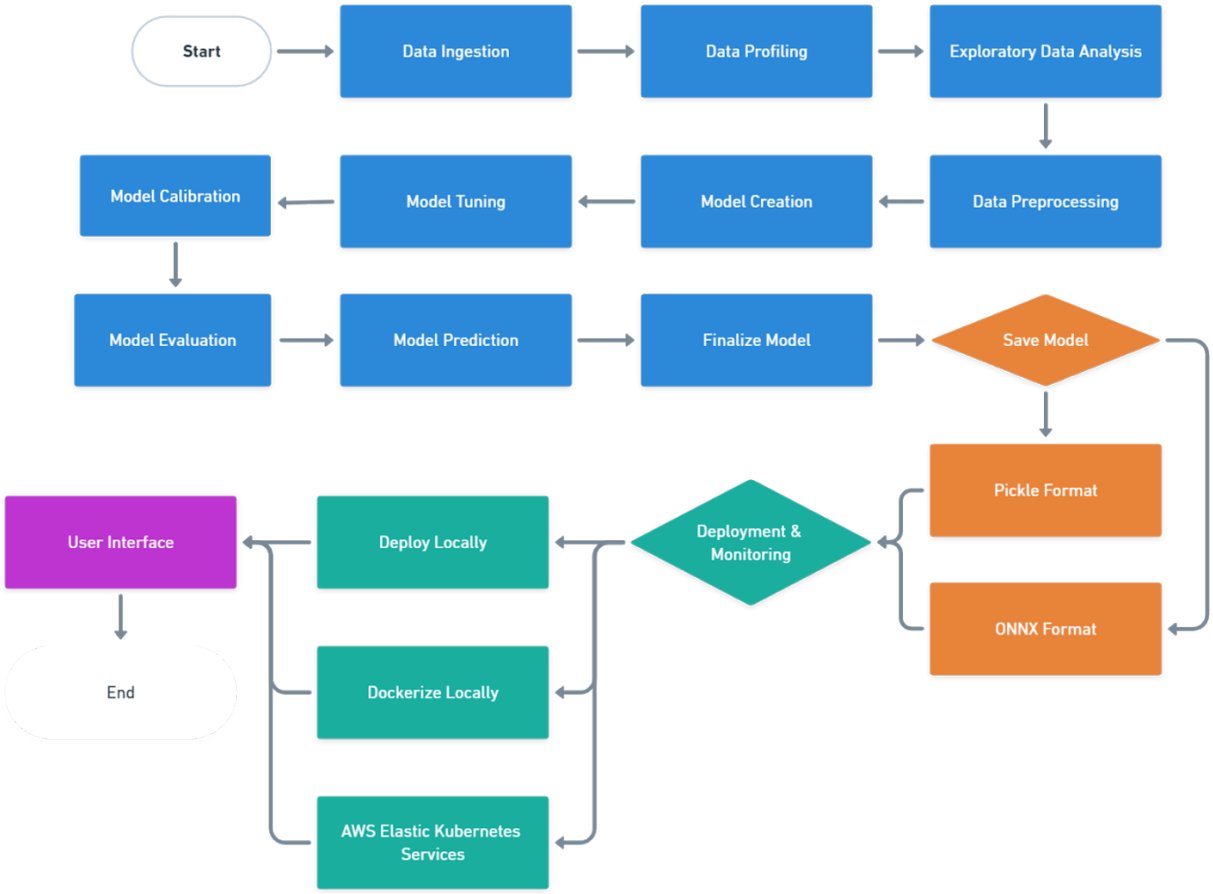
Test.csv

The datasets features consist of dependent and independent features.

Dependent Features:- fare_amont, which exists only in train dataset.

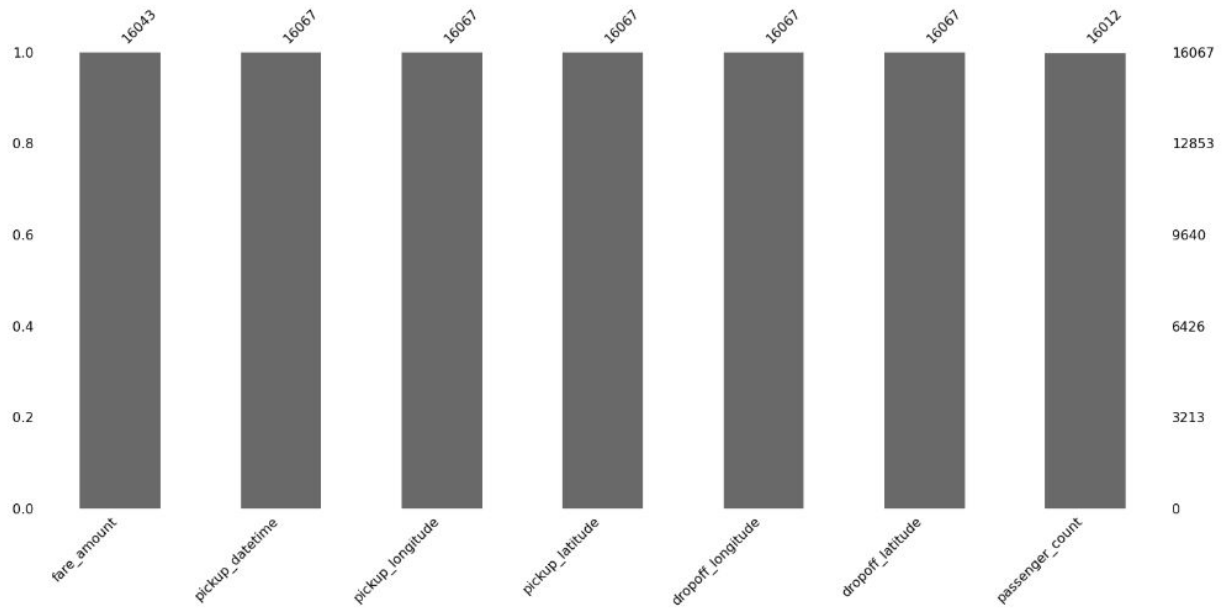
Independent Features:- pickup_datetime, pickup_longitude, pickup_latitude, dropoff_longitude, dropoff_latitude, passenger_count. And these features are common for both data sets.

1.3 High Level Architecture



2. Data Preprocessing

2.1 Missing Value Analysis



2.2 Coordinate Ranges

Latitude & Longitude Range

Latitude Range -90 to 90

Longitude Range -180 to 180

```
print("Range of Pickup Latitude is ", (min(df['pickup_latitude']),max(df['pickup_latitude'])))
```

```
Range of Pickup Latitude is  (-74.006893, 401.083332)
```

```
df[df['pickup_latitude']>90]
```

	fare_amount	pickup_datetime	pickup_longitude	pickup_latitude	dropoff_longitude	dropoff_latitude	passenger_count
5686	3.3	2011-07-30 11:15:00 UTC	-73.947235	401.083332	-73.951392	40.778927	1.0

```
print("Range of Dropoff Latitude is ", (min(df['dropoff_latitude']),max(df['dropoff_longitude'])))
```

```
Range of Dropoff Latitude is  (-74.006377, 40.802437)
```

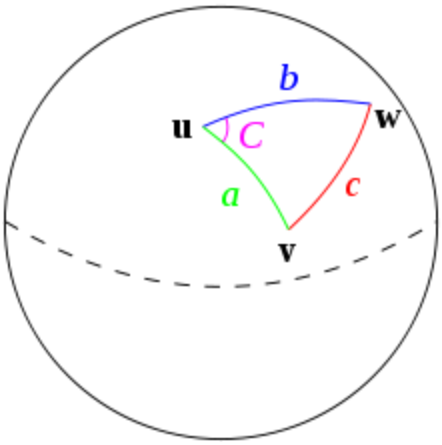
```
print("Range of Pickup Longtitude is ", (min(df['pickup_longitude']),max(df['pickup_longitude'])))
```

```
Range of Pickup Longtitude is  (-74.438233, 40.766125)
```

```
print("Range of Dropoff Longtitude is ", (min(df['dropoff_longitude']),max(df['dropoff_longitude'])))
```

```
Range of Dropoff Longtitude is  (-74.42933199999999, 40.802437)
```

2.3 Calculating distance using Haversine Formula



$$a = \sin^2\left(\frac{\Delta\varphi}{2}\right) + \cos\varphi 1 \cdot \cos\varphi 2 \cdot \sin^2\left(\frac{\Delta\lambda}{2}\right)$$

$$c = 2 \cdot \operatorname{atan2}\left(\sqrt{a}, \sqrt{(1-a)}\right)$$

$$d = R \cdot c$$

2.4 Drop Zero Values

Dropping rows whose distances is 0

Dropping rows whose any of those both latitudes and longitudes are 0

2.5 Replacing outlier values of both longitude and latitude with NaN

[-0.004093, 0.0335, 401.083332, 0, 0.016852, -7.9866399999999995, 0.01798, 0.728087]

2.6 Converting fare_amount datatype to numeric

Replacing outlier values of both longitude and latitude with NaN

[0, 5345, 5334, 557, 554, 537, 536, 535, 531.2, 456, 354, 345, 108.2, 87, 58, 55, 53, 43, 43, 35, 12]

2.7 Imputing NaN values with KNN Algorithm

2.8 Convert object to datetime

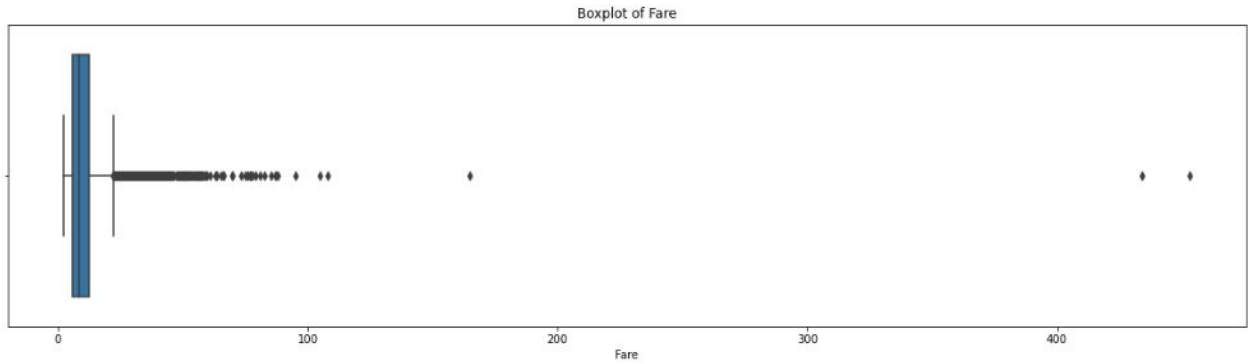
pickup_datetime
2009-06-15 17:26:21 UTC
2010-01-05 16:52:16 UTC
2011-08-18 00:35:00 UTC
2012-04-21 04:30:42 UTC
2010-03-09 07:51:00 UTC
...
2014-12-12 07:41:00 UTC
2009-07-13 07:58:00 UTC
2009-11-11 11:19:07 UTC
2010-05-11 23:53:00 UTC
2011-12-14 06:24:33 UTC

→

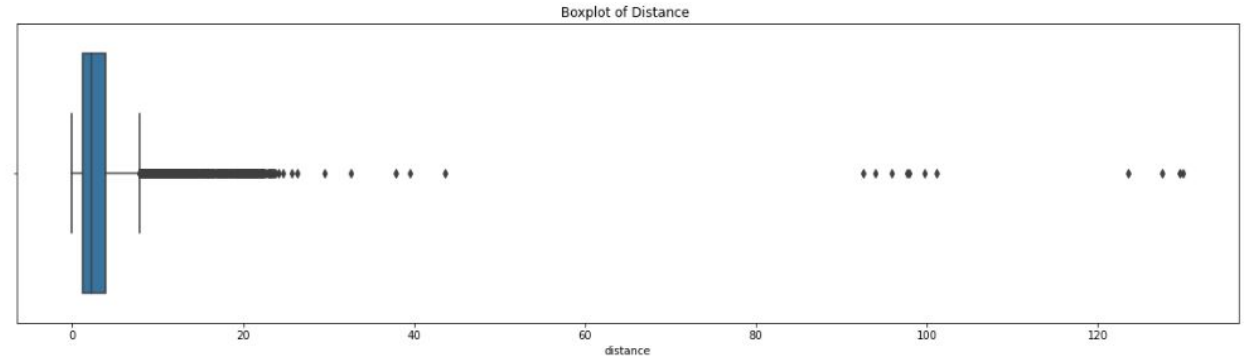
year	Month	Date	Hour	Minute	Second
2009	6	15	17	26	21
2010	1	5	16	52	16
2011	8	18	0	35	0
2012	4	21	4	30	42
2010	3	9	7	51	0
...	...	...	...	...	...
2014	12	12	7	41	0
2009	7	13	7	58	0
2009	11	11	11	19	7
2010	5	11	23	53	0
2011	12	14	6	24	33

2.9 Outlier Analysis

Boxplot for Fare

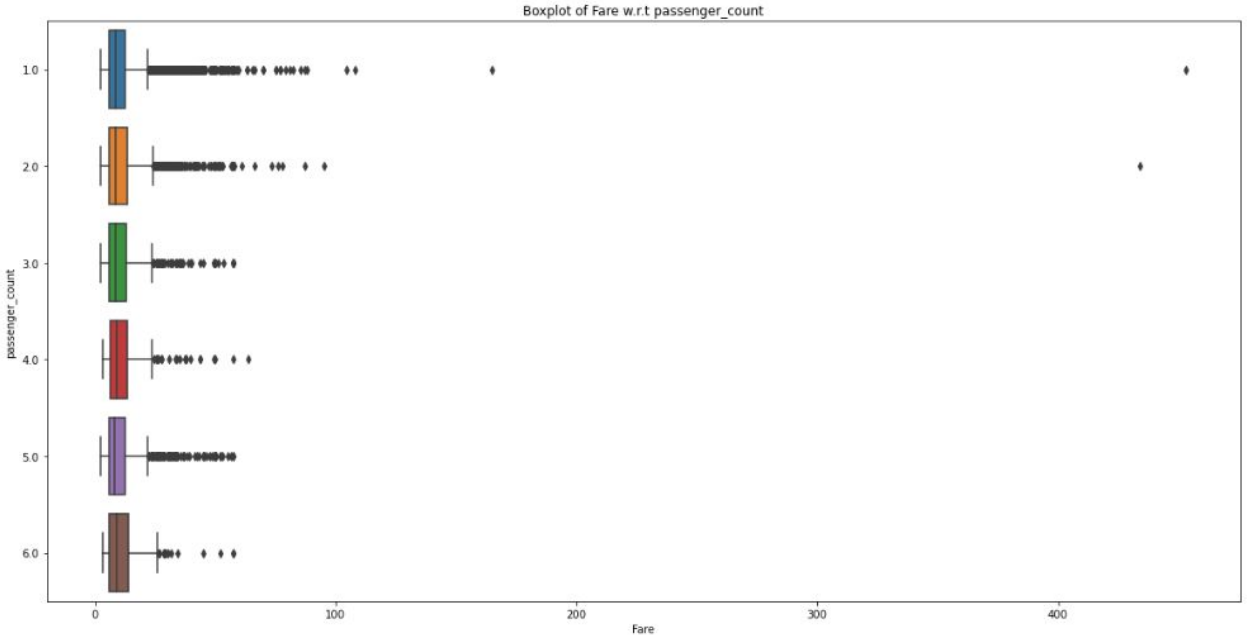


Boxplot for Distance

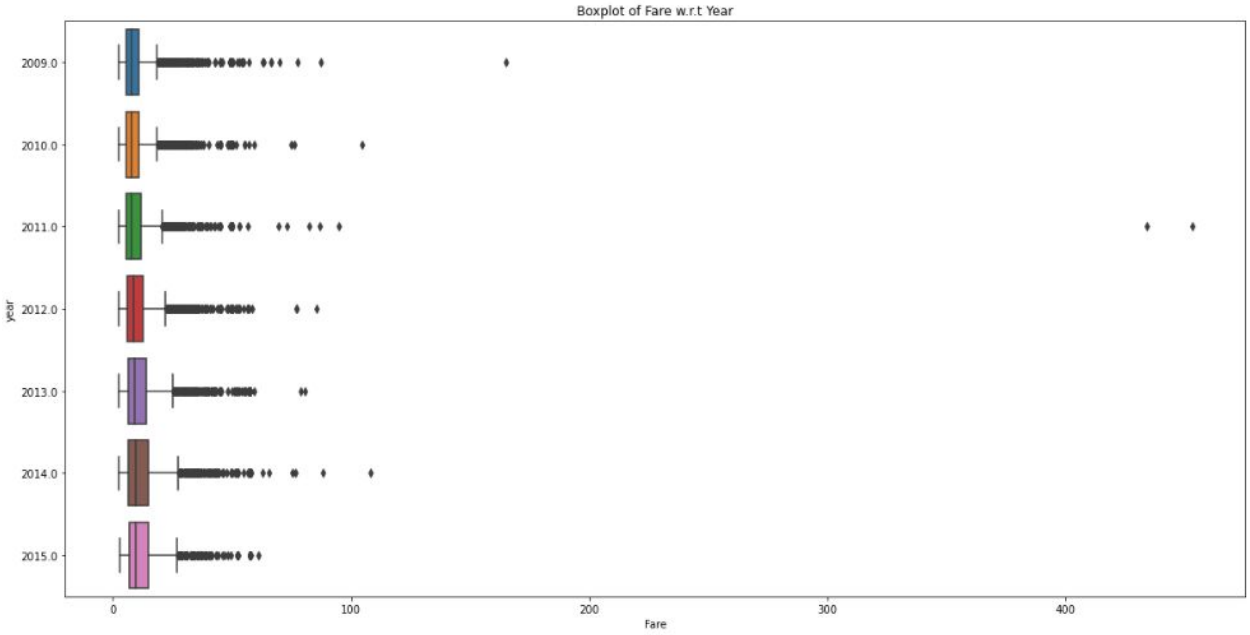


Some distances greater than 100 kms or maybe trips outside city

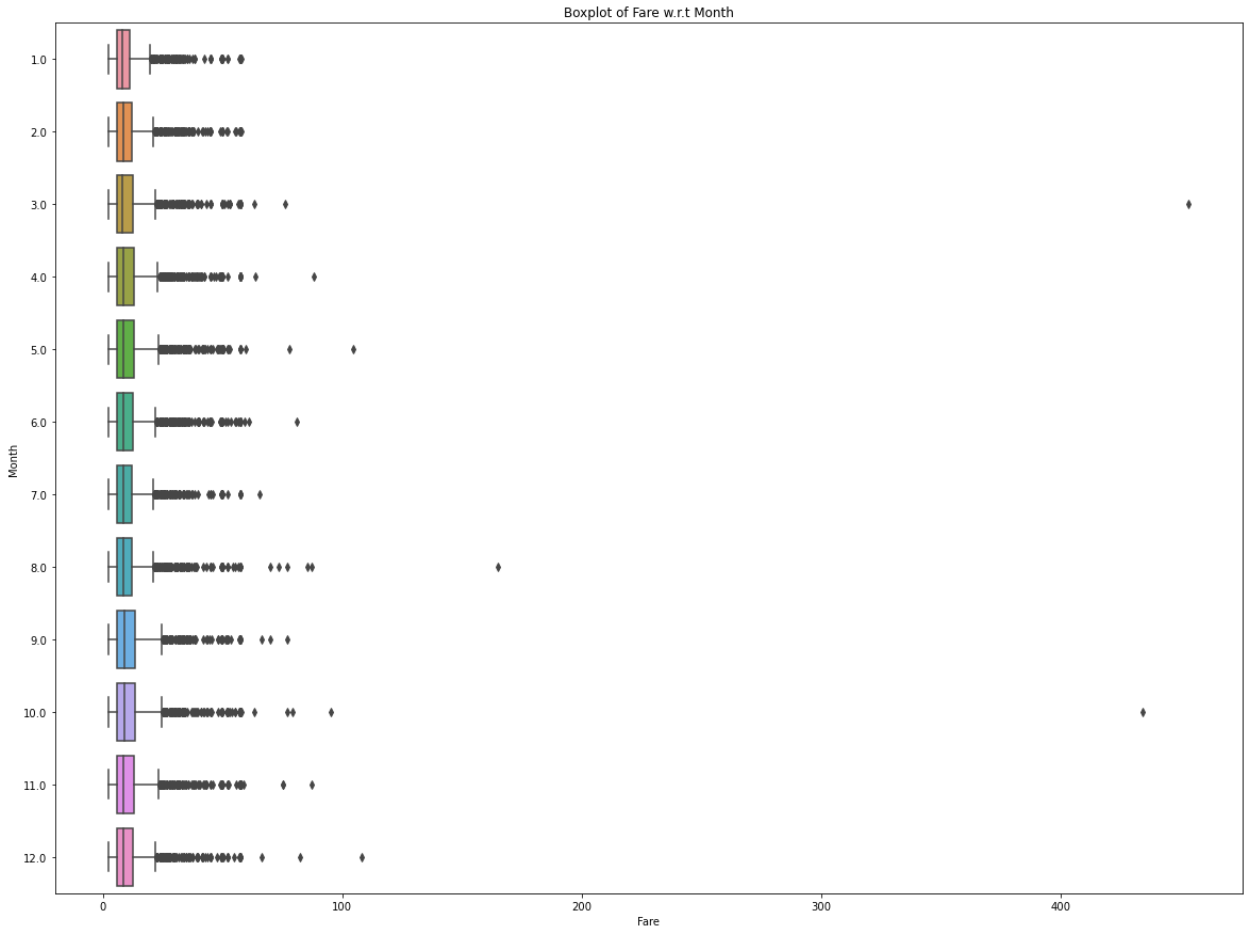
Bivariate boxplot of Fare wrt passenger_count



Bivariate boxplot of Fare wrt Year



Bivariate boxplot of Fare wrt Month




Boxplot of Fare w.r.t Hour
Hour
0.0
1.0
2.0
3.0
4.0
5.0
6.0
7.0
8.0
9.0
10.0
11.0
12.0
13.0
14.0
15.0
16.0
17.0
18.0
19.0
20.0
21.0
22.0
23.0
0
100
200
300
400
Fare

2.10 Dropping rows having outliers

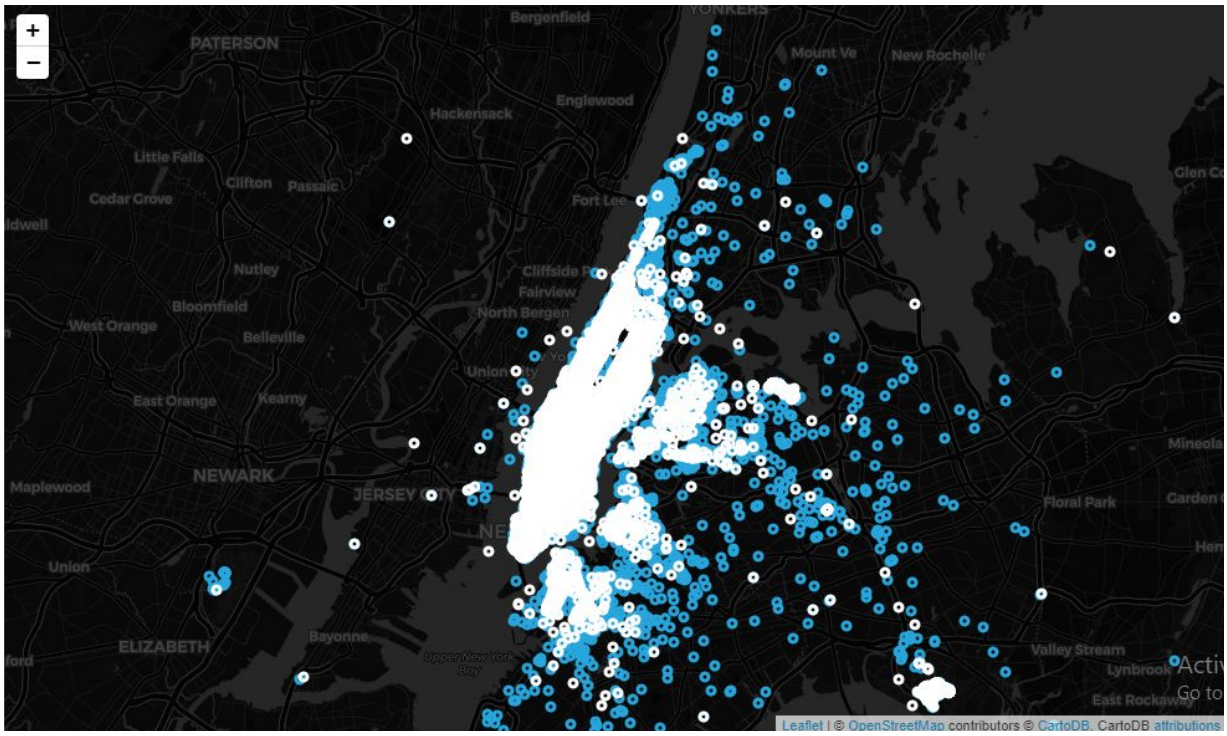
Final dataset after preprocessing

	Fare	distance	passenger_count	year	Month	Date	Hour	Minute	Second
0	4.5	1.030765	1.0	2009.0	6.0	15.0	17.0	26.0	21.0
1	16.9	8.450145	1.0	2010.0	1.0	5.0	16.0	52.0	16.0
2	5.7	1.389527	2.0	2011.0	8.0	18.0	0.0	35.0	0.0
3	7.7	2.799274	1.0	2012.0	4.0	21.0	4.0	30.0	42.0
4	5.3	1.999160	1.0	2010.0	3.0	9.0	7.0	51.0	0.0
...	...	...	...	...	...	...	...	...	...
15581	6.5	0.850046	1.0	2014.0	12.0	12.0	7.0	41.0	0.0
15582	16.1	7.867649	2.0	2009.0	7.0	13.0	7.0	58.0	0.0
15583	8.5	1.469108	1.0	2009.0	11.0	11.0	11.0	19.0	7.0
15584	8.1	2.590040	1.0	2010.0	5.0	11.0	23.0	53.0	0.0
15585	8.5	3.898118	1.0	2011.0	12.0	14.0	6.0	24.0	33.0

15586 rows × 9 columns

3. Exploratory Data Analysis

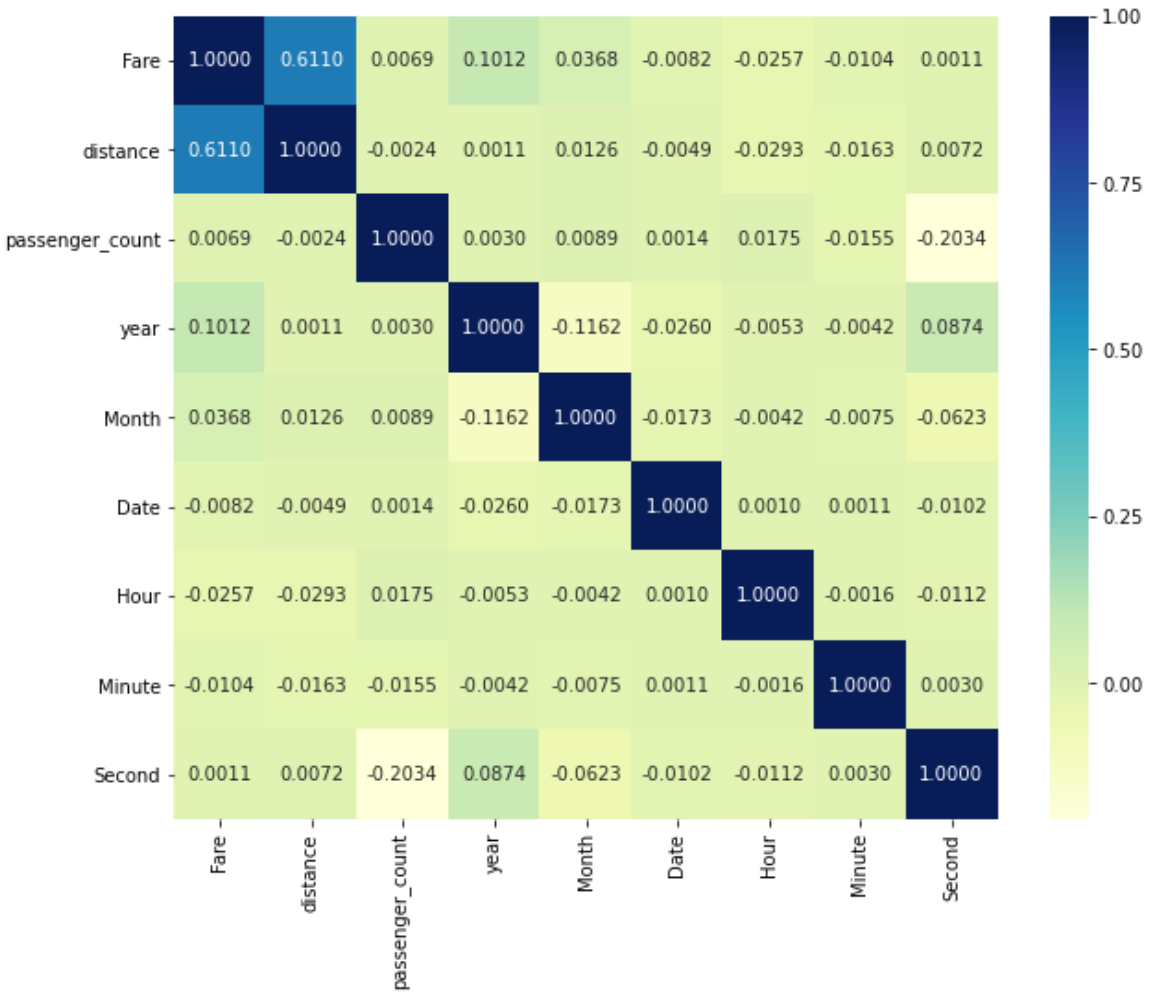
3.1 Pickup vs Dropoff coordinates visualization



White - Pickup Coordinates

Blue - Drop Off Coordinates

3.2 Correlation Plot



Distance is moderately correlated with Fare

3.3 Multicollinearity Test

Multicollinearity means occurrences of highly correlated independent variables in the regression model.

- If the degree of correlation between independent variables is high.
    - It tends to inflate the variance of at least one estimated regression coefficient.
    - Regression coefficient can swing wildly which leads to wrong interpretation of results.
- Ways of dealing with collinearity
    - Multicollinearity affects only the specific independent variable that are correlated, Ignore that specific independent variable.
    - Get rid of the redundant variables using a variable selection technique.

	variables	VIF
0	Fare	3.418065
1	distance	2.494577
2	passenger_count	2.826142
3	year	19.441151
4	Month	4.328410
5	Date	4.258776
6	Hour	5.313276
7	Minute	3.942008
8	Second	1.770528

We found that 'year' is a highly correlated variable

3.4 Analysis Of Variance (ANOVA)

- Variance is the measurement of spread of numbers from their mean.
- ANOVA compares means of two or more groups of independent (unrelated) groups.
- The one-way ANOVA compares the means between the groups you are interested in and determines whether any of those means are statistically significantly different from each other.

$$H_0: \mu_1 = \mu_2 = \mu_3 = \cdots = \mu_k$$

μ = group mean and k = number of groups

- In ANOVA we used one categorical variable and one numerical variable so that a statistical experiment was evaluated again to verify the correlations between the two variables.
- **Example**:
  - We have two independent variables, one is continuous variable and other is categorical variable having categories rich, middle class and poor.
  - Now, calculate the mean of each categorical variable i.e rich, middle class, poor.
  - Compare means of these variables.
  - Now our assumption is that the mean of all three categories are different.
  - Therefore our null hypothesis is that the mean of each categorical variable are equal and for alternate hypotheses these mean are not equal.
  - If p-value < 0.05 then we reject the null hypothesis because the means of categorical variables are not equal.

The p value obtained from ANOVA analysis is significant ($p < 0.05$), and therefore, we conclude that there are significant differences among treatments.

3.5 Pearson's Correlation Test

Correlation coefficients are used in statistics to measure how strong a relationship is between two variables. There are several types of correlation coefficient, but the most popular is Pearson's. Pearson's correlation (also called Pearson's R) is a correlation coefficient commonly used in linear regression.

- A correlation coefficient of 1 means that for every positive increase in one variable, there is a positive increase of a fixed proportion in the other. For example, shoe sizes go up in (almost) perfect correlation with foot length.
- A correlation coefficient of -1 means that for every positive increase in one variable, there is a negative decrease of a fixed proportion in the other. For example, the amount of gas in a tank decreases in (almost) perfect correlation with speed.
- Zero means that for every increase, there isn't a positive or negative increase. The two just aren't related.

	Fare	distance	passenger_count	year	Month	Date	Hour	Minute	Second
Fare	1	0.611	0.0069	0.1012	0.0368	-0.0082	-0.0257	-0.0104	0.0011
distance	0.611	1	-0.0024	0.0011	0.0126	-0.0049	-0.0293	-0.0163	0.0072
passenger_count	0.0069	-0.0024	1	0.003	0.0089	0.0014	0.0175	-0.0155	-0.2034
year	0.1012	0.0011	0.003	1	-0.1162	-0.026	-0.0053	-0.0042	0.0874
Month	0.0368	0.0126	0.0089	-0.1162	1	-0.0173	-0.0042	-0.0075	-0.0623
Date	-0.0082	-0.0049	0.0014	-0.026	-0.0173	1	0.001	0.0011	-0.0102
Hour	-0.0257	-0.0293	0.0175	-0.0053	-0.0042	0.001	1	-0.0016	-0.0112
Minute	-0.0104	-0.0163	-0.0155	-0.0042	-0.0075	0.0011	-0.0016	1	0.003
Second	0.0011	0.0072	-0.2034	0.0874	-0.0623	-0.0102	-0.0112	0.003	1

3.6 Chi-Square Test

The Pearson's Chi-Square statistical hypothesis is a test for independence between categorical variables. We start by defining the **null** hypothesis (**H0**) which states that there is no relation between the variables. An **alternate** hypothesis would state that there is a significant relation between the two.

We define a significance factor to determine whether the relation between the variables is of considerable significance. Generally a significance factor or alpha value of 0.05 is chosen. This alpha value denotes the probability of erroneously rejecting H0 when it is true. A lower alpha value is chosen in cases where we expect more precision. If the p-value for the test comes out to be strictly greater than the alpha value, then H0 holds true.

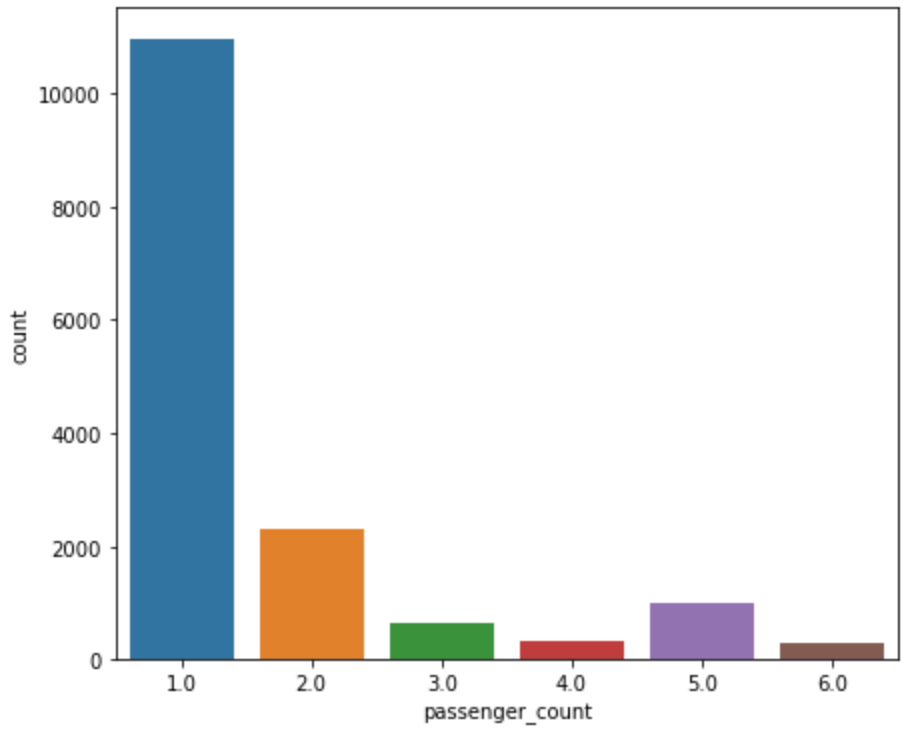
```
# defining the table
stat, p, dof, expected = chi2_contingency(df)

# interpret p-value
alpha = 0.05
print("p value is " + str(p))
if p <= alpha:
    print('Dependent (reject H0)')
else:
    print('Independent (H0 holds true)')
```

```
p value is 0.0
Dependent (reject H0)
```

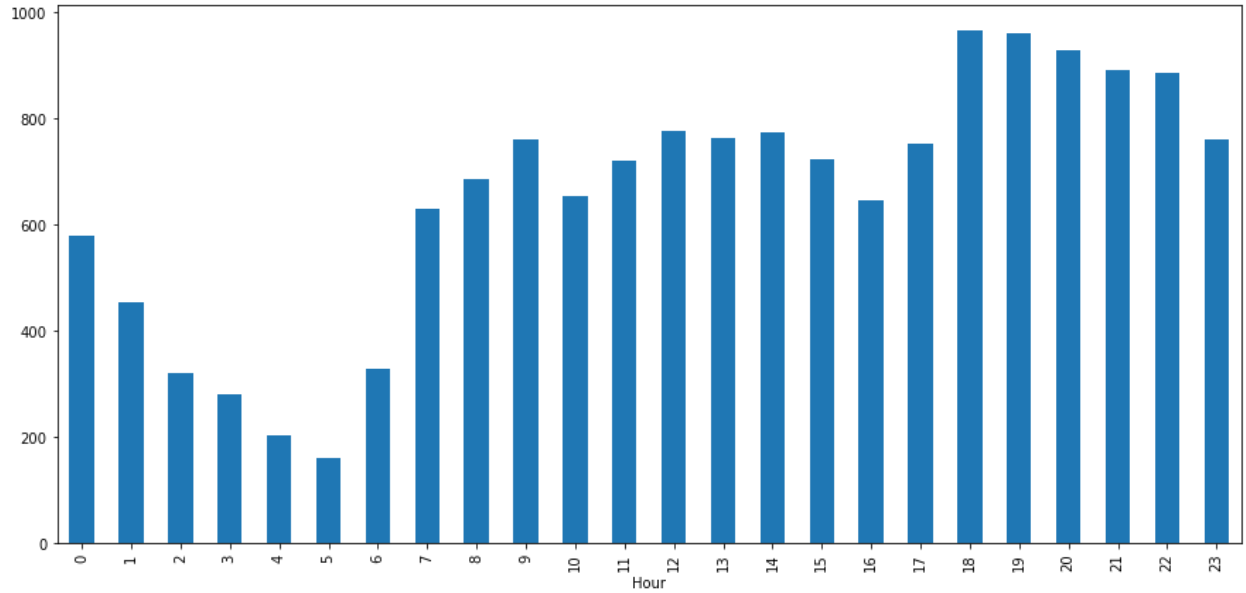
Since, p-value > alpha Therefore, we reject H0, that is, the variables do have significant relationship.

3.7 Count Plot

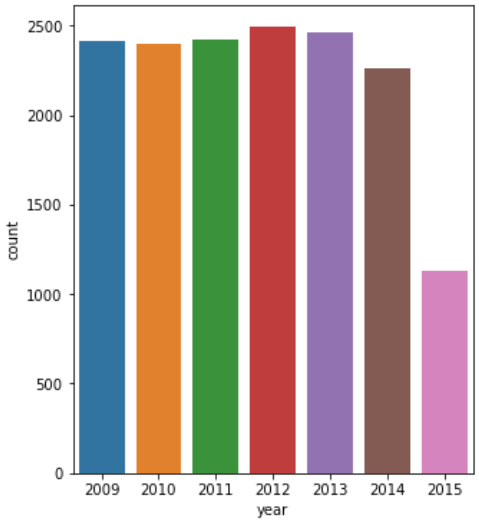
Countplot method is used to Show the counts of observations in each categorical bin using bars.

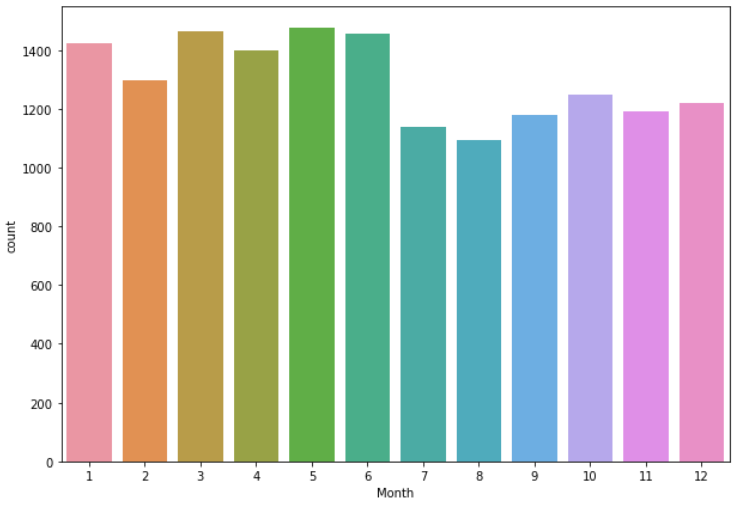


Single travelling passengers are the most frequent travellers. At the sametime we can also conclude that the highest fares are coming from single & double travelling passengers.



Lowest cabs at 5 AM and highest at and around 7 PM i.e the office rush hours

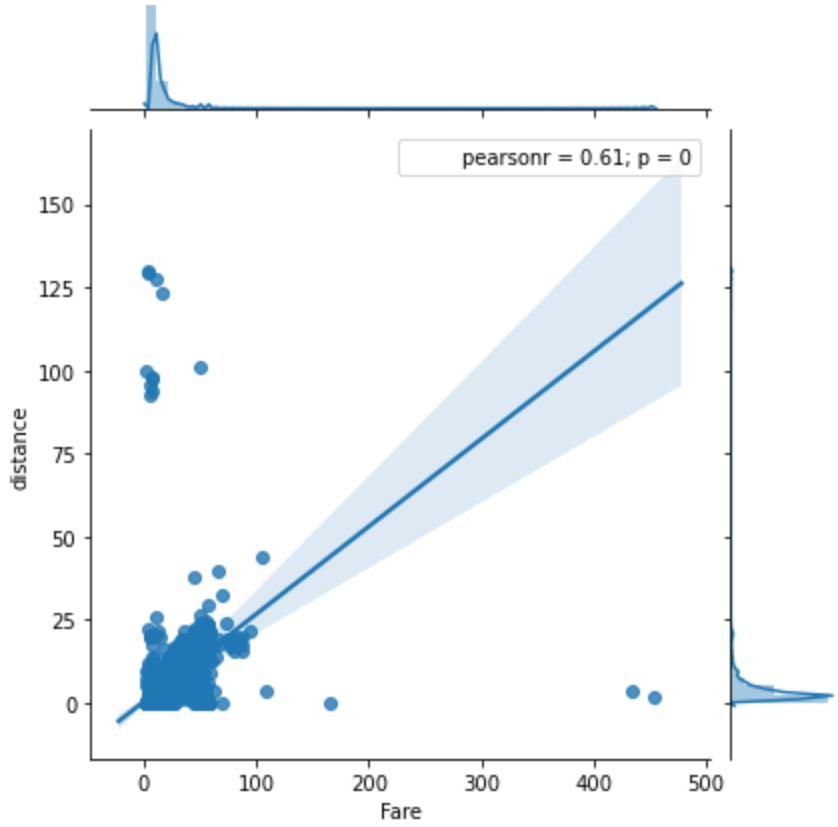




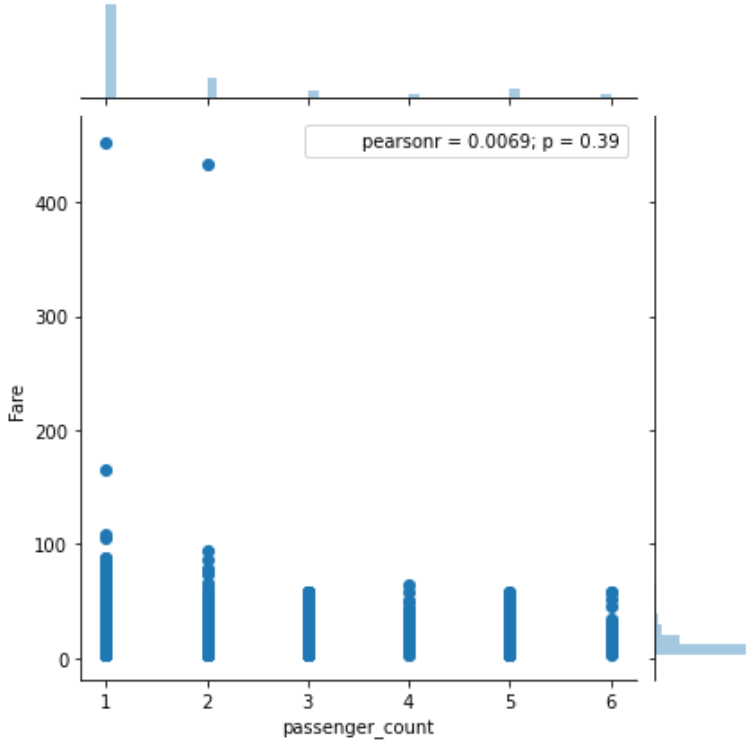
3.8 Jointplot

A Jointplot comprises three plots. Out of the three, one plot displays a bivariate graph which shows how the dependent variable(Y) varies with the independent variable(X). Another plot is placed horizontally at the top of the bivariate graph and it shows the distribution of the independent variable(X). The third plot is placed on the right margin of the bivariate graph with the orientation set to vertical and it shows the distribution of the dependent variable(Y). It is very helpful to have univariate and bivariate plots together in one figure. This is because the univariate analysis focuses on one variable, it describes, summarizes and shows any patterns in your data and the bivariate analysis explores the relationship between two variables and also describes the strength of their relationship. The function jointplot() in Seaborn library creates a scatter plot with two histograms at the top and right margins of the graph by default.

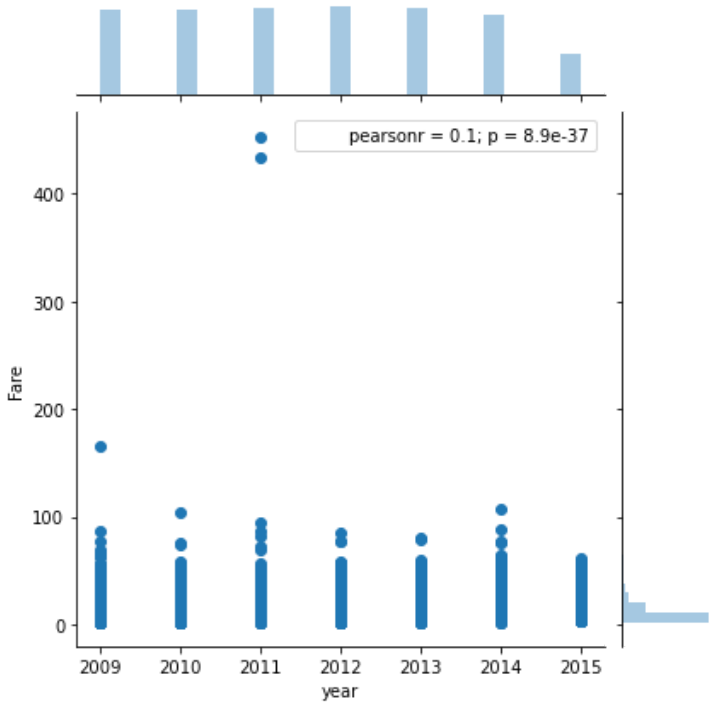
Jointplot for distance and fare



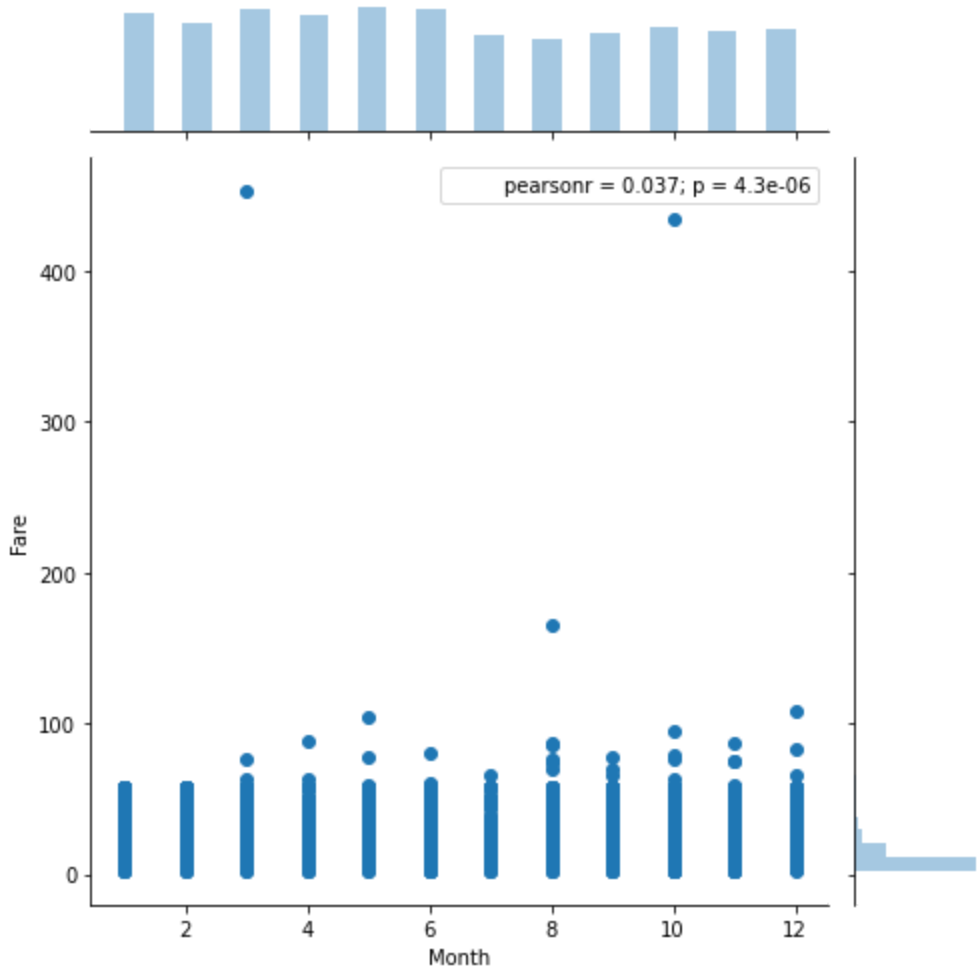
Jointplot for passenger_count and Fare



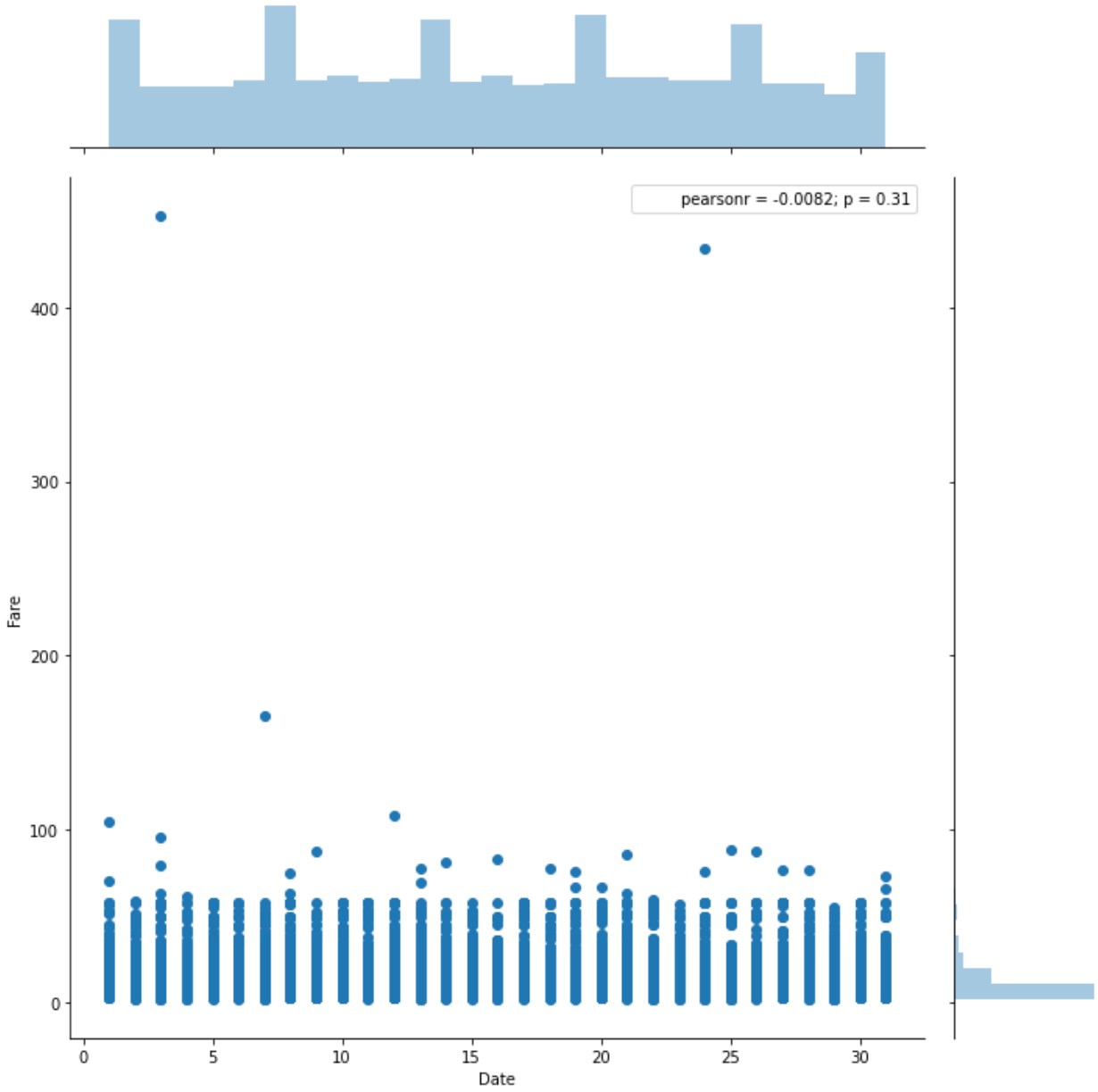
Jointplot for year and Fare



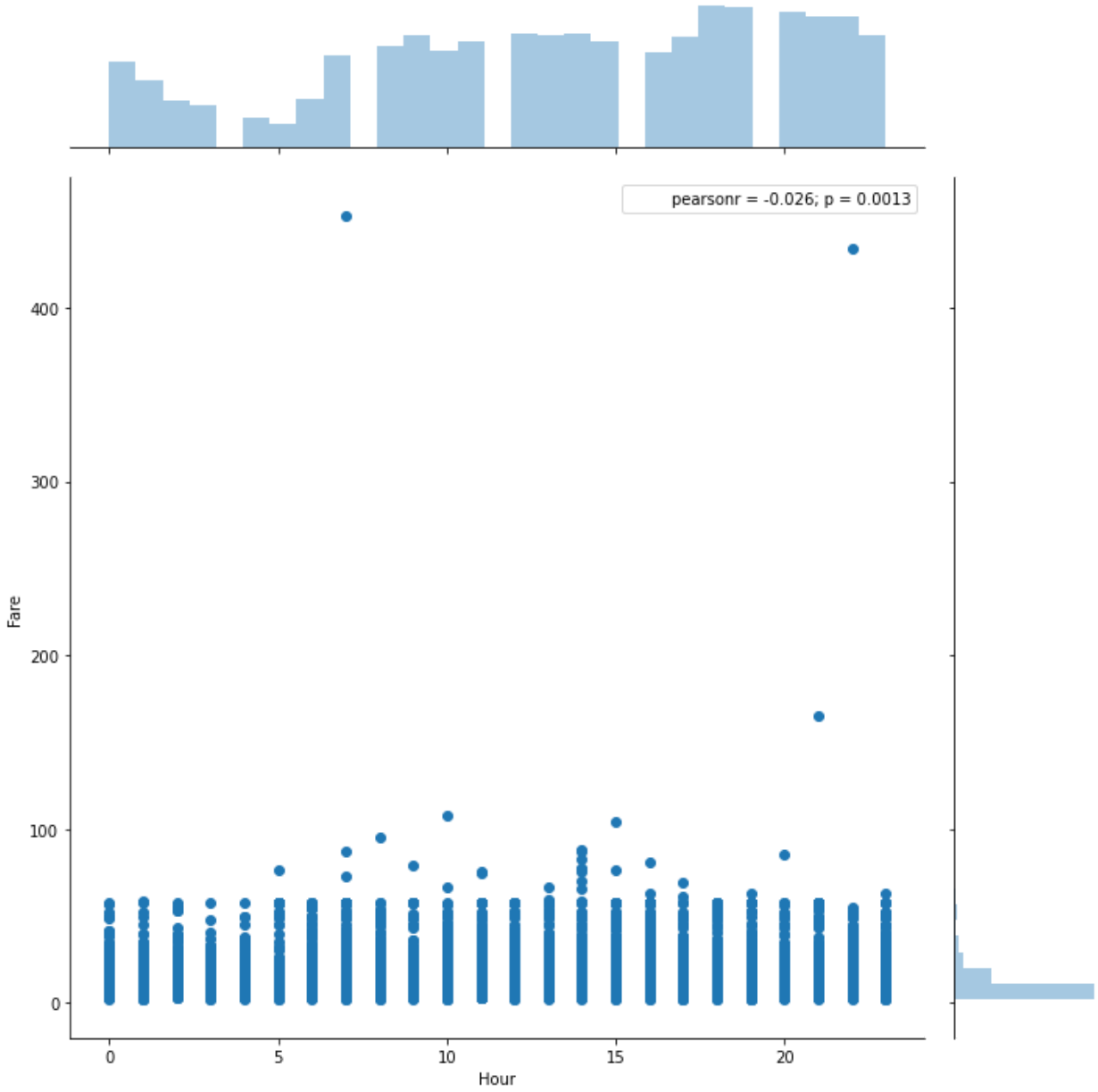
Jointplot for Month and Fare



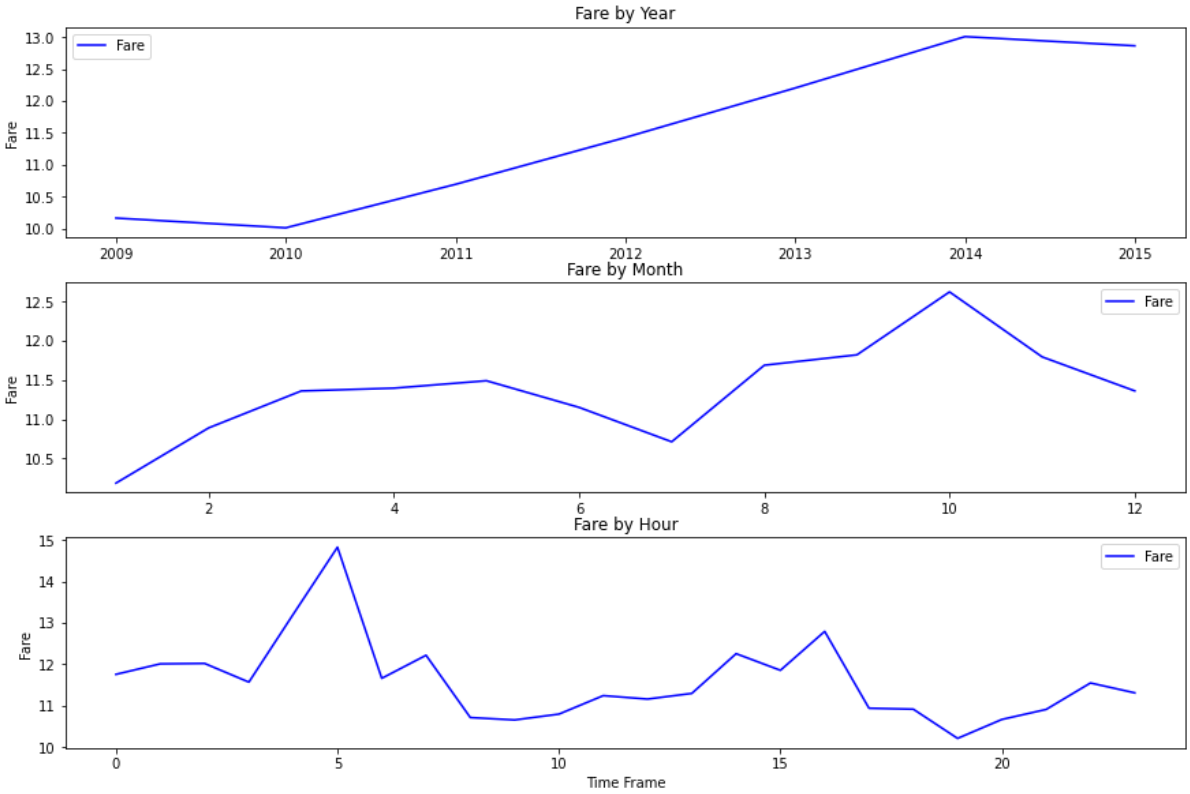
Jointplot for Date and Fare



Jointplot for Hour and Fare



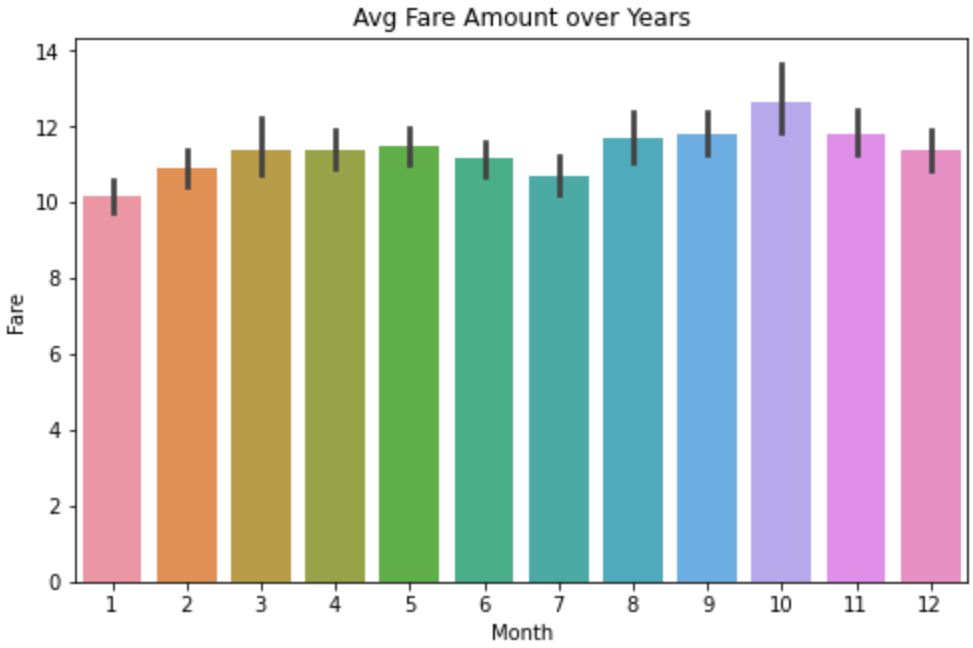
3.9 Seasonality Analysis

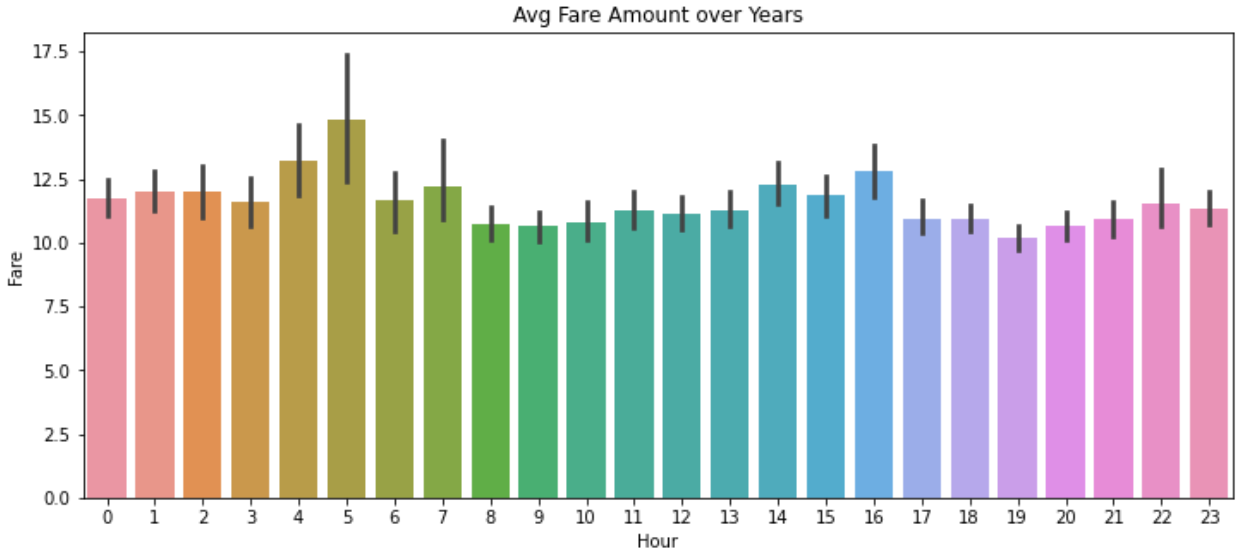


Surge Fare at 5 AM and 4-5 PM office goers

3.10 Barplot

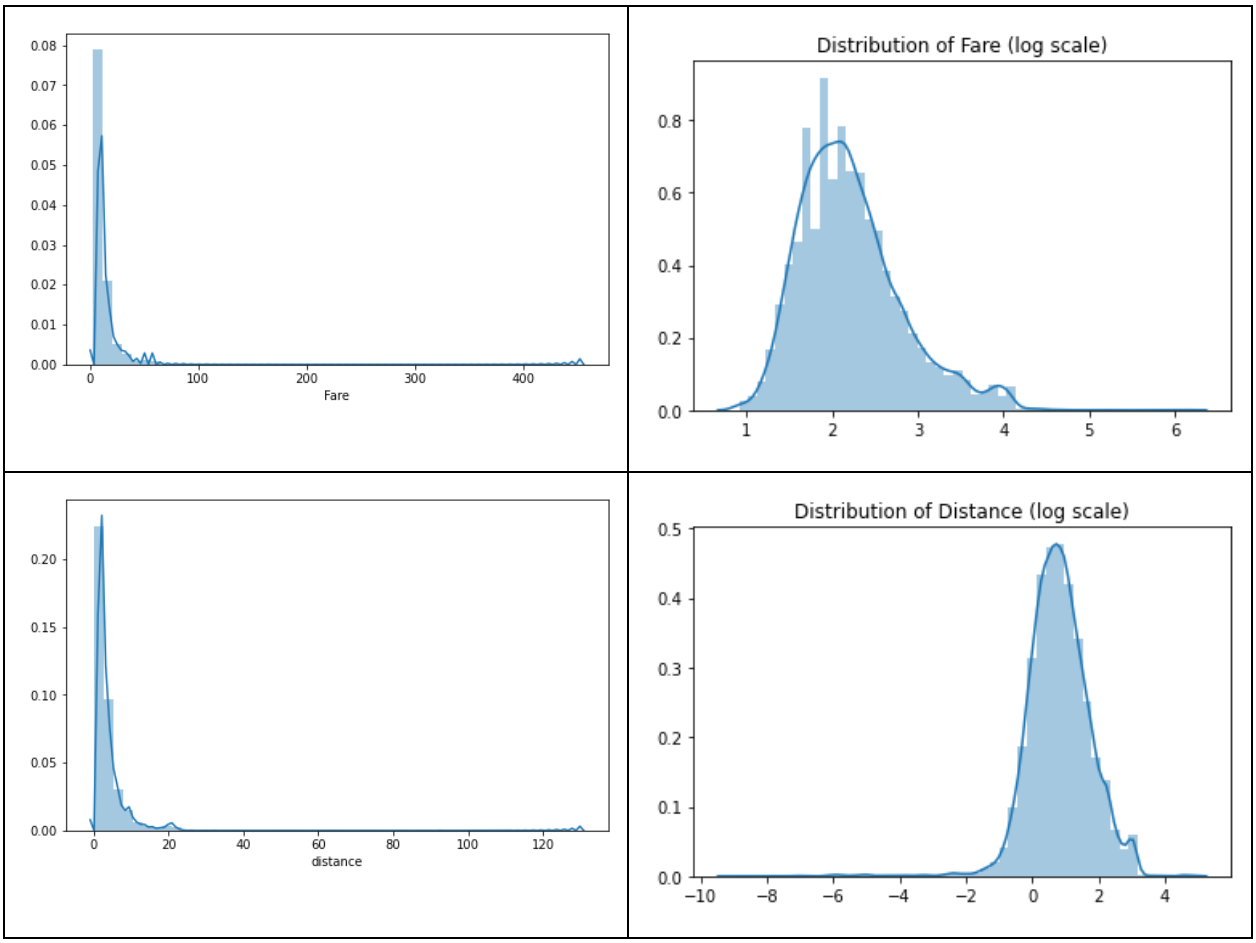
A bar plot represents an estimate of central tendency for a numeric variable with the height of each rectangle and provides some indication of the uncertainty around that estimate using error bars. Show point estimates and confidence intervals as rectangular bars.


Avg Fare Amount over Years
Fare
Month
0
2
4
6
8
10
12
14
1
2
3
4
5
6
7
8
9
10
11
12


Avg Fare Amount over Years
Fare
Hour
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
0
1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23

3.11 Distplot

Distplot lets you show a histogram with a line on it. A distplot plots a univariate distribution of observations. The distplot() function combines the matplotlib hist function with the seaborn kdeplot() and rugplot() functions.

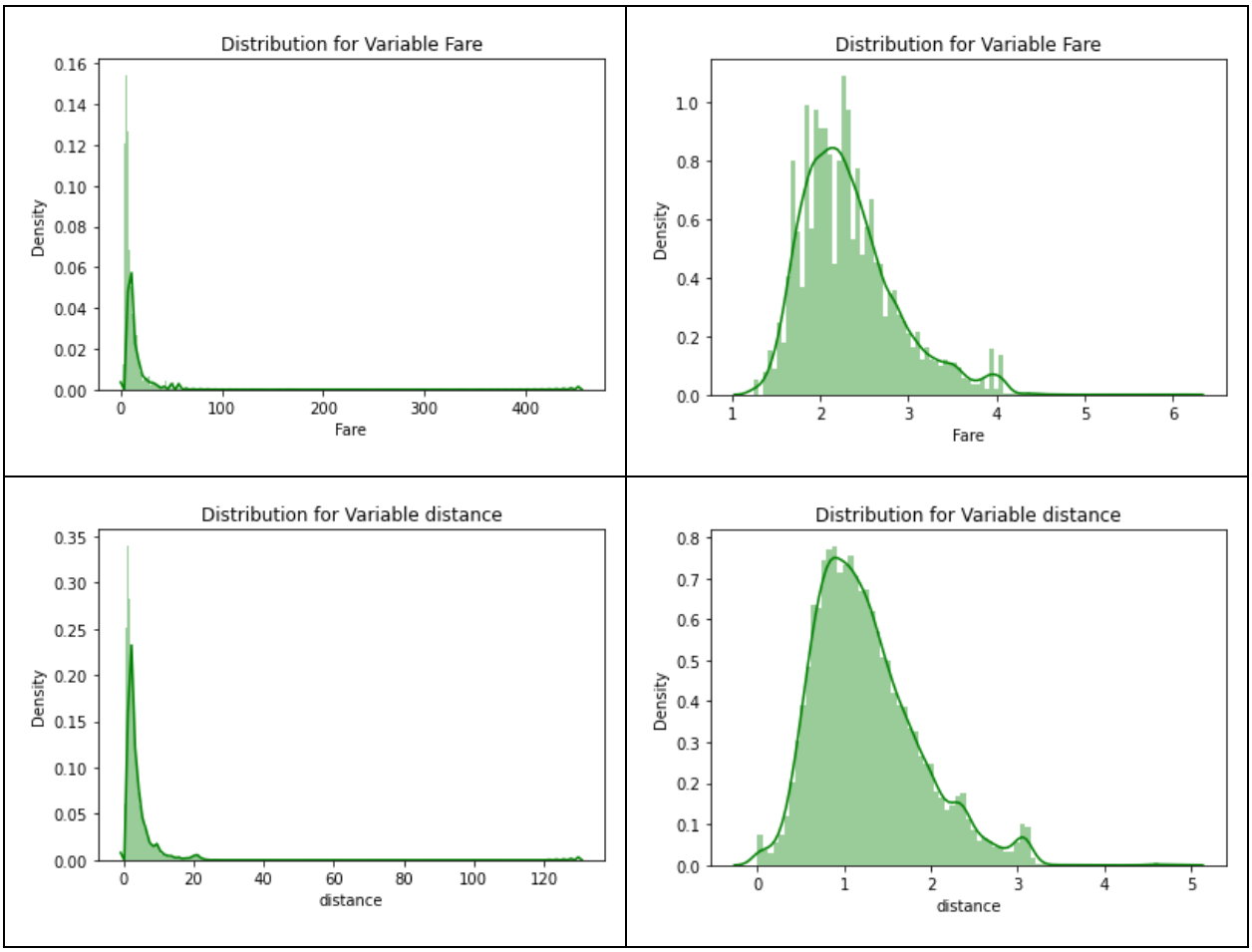


4. Model Building

4.1 Skewness

Skewness is asymmetry in a statistical distribution, in which the curve appears distorted or skewed either to the left or to the right. Skewness can be quantified to define the extent to which a distribution differs from a normal distribution.

Log Transformation:- The log transformation can be used to make highly skewed distributions less skewed. This can be valuable both for making patterns in the data more interpretable and for helping to meet the assumptions of inferential statistics.



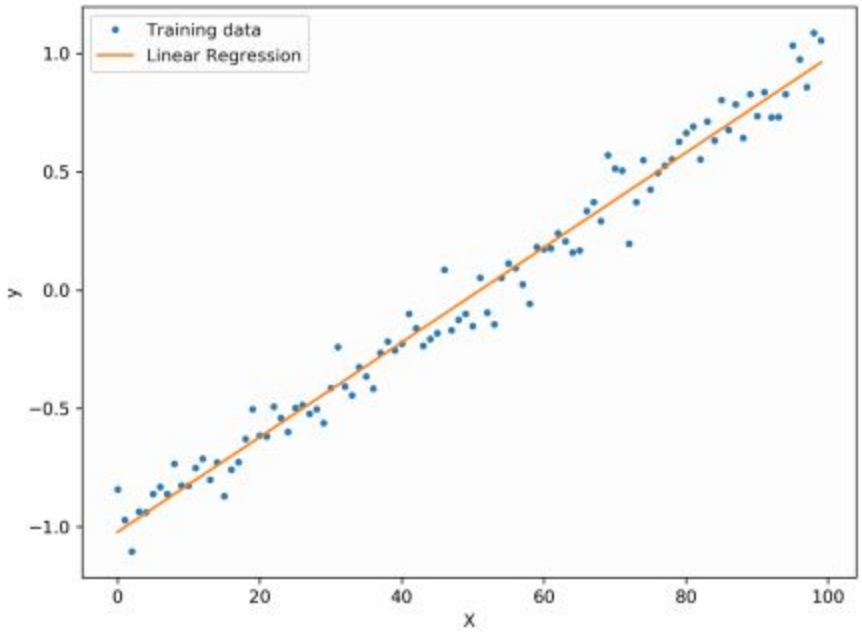
Here we can see bell shaped distribution. Hence our continuous variables are now normally distributed.

We drop 'year' because it has high collinearity among other variables which won't add any much substantial detail while building the model. Concluded by Chi-Square Test.

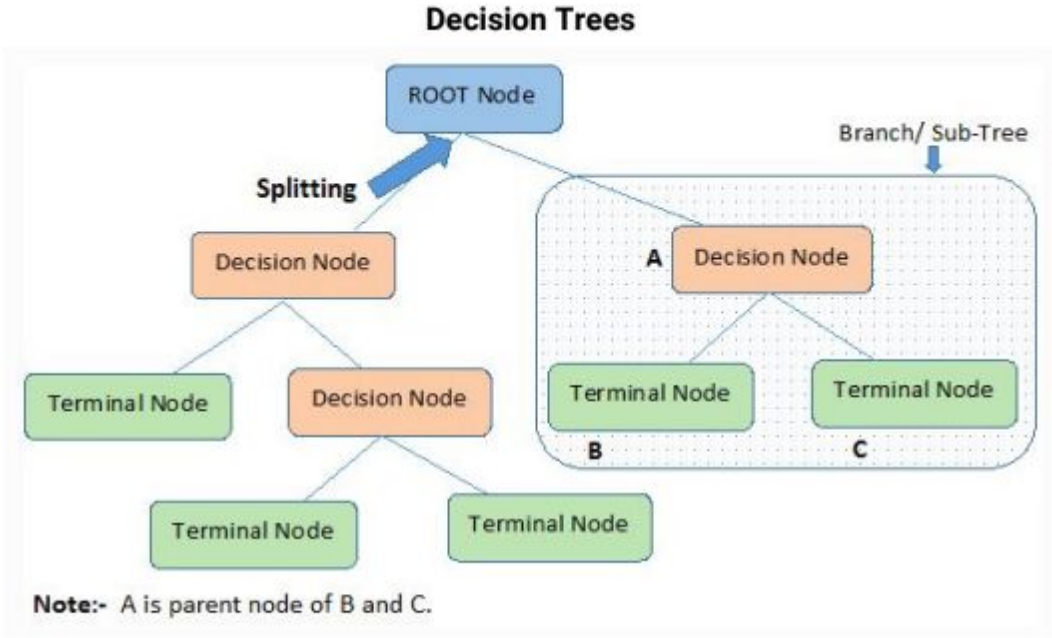
We see very less deviation of spread of Fare variable when doing bivariate analysis against columns year, date, minute, second. Therefore we won't include it in our model building.

4.2 Linear Regression model

- Regression analysis is all about determining how changes in the independent variables are associated with changes in the dependent variable.
- The coefficients describe the mathematical relationship between each independent variable and the dependent variable.
- The coefficient value signifies how much the mean of the dependent variable changes given a one-unit shift in the independent variable while holding other variables in the model constant.
- Regression coefficients represent the mean change in the response variable for one unit of change in the predictor variable while holding other predictors in the model constant.
- The key to understanding the coefficients is to think of them as slopes, and they're often called slope coefficients.



4.3 Decision Tree Regressor



Decision trees classify the examples by sorting them down the tree from the root to some leaf/terminal node, with the leaf/terminal node providing the classification of the example.

Each node in the tree acts as a test case for some attribute, and each edge descending from the node corresponds to the possible answers to the test case. This process is recursive in nature and is repeated for every subtree rooted at the new node.

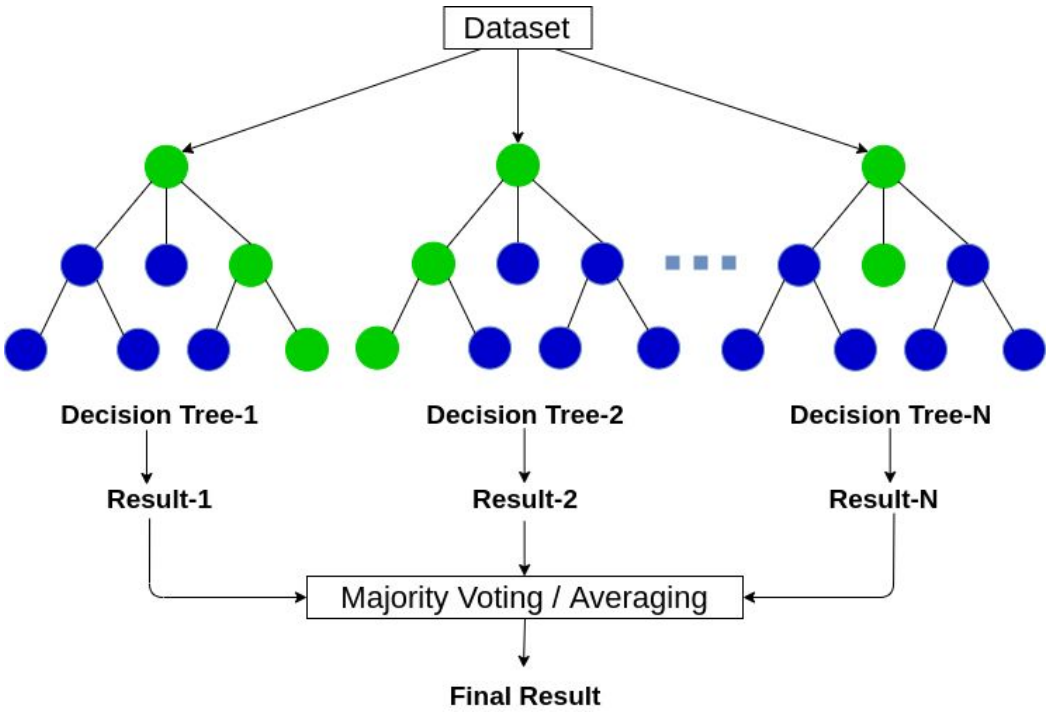
4.4 Random Forest Regressor

Random forests or random decision forests are an ensemble learning method for classification, regression and other tasks, that operate 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.

To say it in simple words: Random forest builds multiple decision trees and merges them together to get a more accurate and stable prediction.

Approach :

- Pick at random K data points from the training set.
- Build the decision tree associated with those K data points.
- Choose the number Ntree of trees you want to build and repeat step 1 & 2.
- For a new data point, make each one of your Ntree trees predict the value of Y for the data point, and assign the new data point the average across all of the predicted Y values.



4.5 Gradient Boosting Regressor

Gradient boosting is a machine learning technique for regression and classification problems, which produces a prediction model in the form of an ensemble of weak prediction models, typically decision trees. It builds the model in a stage-wise fashion like other boosting methods do, and it generalizes them by allowing optimization of an arbitrary differentiable loss function.

5.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

Evaluating a model is a core part of building an effective machine learning model There are several evaluation metrics, like **confusion matrix, cross-validation, AUC-ROC curve**, etc. Different evaluation metrics are used for different kinds of problems.

The regression evaluation statistics calculated by this function belong to two different groups of measures: absolute and relative. The former include

"mae", "mse", and "rmse" and are calculated as follows:

"mae": **mean absolute error**, which is calculated as sum(|t_i - p_i|)/N, where t's are the true values and p's are the predictions, while N is supposed to be the size of both vectors.

"mse": **mean squared error**, which is calculated as sum((t_i - p_i)^2)/N

"rmse": **root mean squared error** that is calculated as sqrt(mse)

5.1 Mean Absolute Percentage Error (MAPE)

Here in the regression model the evaluation is done by 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, by Multiplying 100 to percentage error.

For R

```
mape=function(av,pv){
  mean(abs((av-pv)/av))*100 #av=actual value and pv=
predicted value
}
```

For python

```
#mape                    #av= actual value and pv= predicted
value
def mape(av, pv):
    mape = np.mean(np.abs((av - pv) / av))*100
    return mape
```

From above we can define the mape function in both R and Python.

The internal mape operation is

$$\frac{\sum_{t=1}^{n} \left| \frac{(Y_t - \hat{Y}_t)}{Y_t} (100) \right|}{n}$$

5.2 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 climatology, forecasting, and regression analysis to verify experimental results.

The formula is:

$$RMSE = \sqrt{(f-o)^2}$$

OR

$$RMSE = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

Where:

f = forecasts (expected values or unknown results),

o = observed values (known results).

Σ = summation ("add up")

(zfi – Zoi)2 = differences, squared

N = sample size.

5.3 r2 Score

The r2 score varies between 0 and 100%. It is closely related to the MSE (see below), but not the same.

" ...the proportion of the variance in the dependent variable that is predictable from the independent variable(s)."

Another 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.

5.4 Mean Square Error

Mean square error (MSE) is the average of the square of the errors. The larger the number the larger the error. Error in this case means the difference between the observed values y1, y2, y3, ... and the predicted ones pred(y1), pred(y2), pred(y3), ... We square each difference (pred(yn) – yn)) ** 2 so that negative and positive values do not cancel each other out.

There is no correct value for MSE. Simply put, the lower the value the better and 0 means the model is perfect. Since there is no correct answer, the MSE's basic value is in selecting one prediction model over another.

Similarly, there is also no correct answer as to what R2 should be. 100% means perfect correlation. Yet, there are models with a low R2 that are still good models.

ML Algorithm	RMSE Train	RMSE Test	R2 Score Train	R2 Score Test
Linear Regression	0.28	0.28	0.74	0.73
Decision Tree	0.30	0.29	0.70	0.71
Random Forest	0.10	0.27	0.96	0.75
Gradient Boosting	0.23	0.25	0.78	0.81

6. Hyper Parameters Tunings

Model hyperparameters are set by the data scientist ahead of training and control implementation aspects of the model. The weights learned during training of a linear regression model are parameters while the number of trees in a random forest is a model hyperparameter because this is set by the data scientist. Hyperparameters can be thought of as model settings. These settings need to be tuned for each problem because the best model hyperparameters for one particular dataset will not be the best across all datasets. The process of hyperparameter tuning (also called hyperparameter optimization) means finding the combination of hyperparameter values for a machine learning model that performs.

Here we have used two hyper parameters tuning techniques

1. Random Search CV
2. Grid Search CV

1. Random Search CV: This algorithm set up a grid of hyperparameter values and select random combinations to train the model and score. The number of search iterations is set based on time/resources.

2. Grid Search CV: This algorithm set up a grid of hyperparameter values and for each combination, train a model and score on the validation data. In this approach, every single combination of hyperparameter values is tried which can be very inefficient.s the best - as measured on a validation dataset - for a problem.

Tuning Method	ML Algorithm	RMSE (Test)	R Squared (Test)
Random Search CV	Random Forest	0.24	0.79
	Gradient Boosting	0.25	0.77
Grid Search CV	Random Forest	0.23	0.80
	Gradient Boosting	0.24	0.79

7. Model Selection

On the basis RMSE and R Squared results a good model should have least RMSE and max R Squared value. So, from above tables we can see:

- From the observation of all RMSE Value and R-Squared Value we have concluded that,
- Both the models- Gradient Boosting Default and Random Forest perform comparatively well while comparing their RMSE and R-Squared value.
- After this, I chose Random Forest CV and Grid Search CV to apply cross validation technique and see changes brought about by that.
- After applying tunings, the Random forest model shows best results compared to gradient boosting.
- So finally, we can say that the Random forest model is the best method to make predictions for this project with highest explained variance of the target variables and lowest error chances with parameter tuning technique Grid Search CV.

8. Conclusion

- As we can observe that the error rate is less in models like lasso and random forest regression models , while comparing to other models.
- Each and every model used here linear , decision trees , random forest regression, gradient boosting regression models are good at predicting the numerical outcomes.
- We select the model with less errors and high accuracy.