New York City Taxi Fare Prediction

(A Machine Learning Approach)

By

Marcel I Ngwueke

Springboard Data Science Career Track,

May 2021 Cohort

Capstone2 Project Report.

Table of Contents ii

1.1 Introduction 1  
1.2 Study Review Involving NYC Taxi Fare 1  
1.3 Problem statement 2   
1.3.1 Business definition 2  
1.3.2 Business Objective 2  
2.0 Data Source & Data Wrangling 3  
2.1 Introduction to Data wrangling 3  
2.1.1 Data wrangling 3  
2.1.2 Feature Engineering 4  
3.0 Introduction to Exploratory Data Analysis 7  
3.1 Coordinate system 7  
3.2 Pickup-DateTime system 7  
4.0 Machine Learning Algorithms and Model Assessment. 10  
4.1 GridSearch XGBoost Regressor 10  
4.2 GridSearch LightGBM Regressor 14  
4.3 Ridge Regression 18  
4.4 Random Forest Regressor 20  
4.5 Ordinary Least Square Regression 23  
5.0 Discussions and Model Assessment 26  
5.1 Conclusion 28  
5.2 Recommendations 28  
 References 29

1.1 **Introduction**  
Several systems exist that can calculate or estimate taxi fares for commuters. These are online taxi fare calculators for most cities in the United States of America How accurate the estimations are no one can say because no poly has shown opinion results. Greta & Combessie(2019) identified the number of for-hire vehicles in New York City increased from 63k to 100k, and the number of trips in app-based vehicles has risen from 6million to 17million years with taxi trips falling from 11 million to 8.5 million. Their study didn’t show that distance was the main factor impacting taxi fare.

Commuters in a 5-hour round trip pay more than $800 and a trip inside Manhattan, and another city like New York cost $400 more than an airport ride.

According to Monalisha Ojha et al (2019), a taxi fare ride is a function of the duration of the trip, which is the sum of drop charge, distance charge, and time charge. Christophoros Antoniades, et al (2016) solve a similar problem, that is estimating ride duration without real-time data, by analyzing data collected from taxis. The articles stated that one way to predict duration is by doing short-term prediction with the help of real-time data collection, which Vanajakshi, L, et al (2009), solved by using data from buses (GPS) and an algorithm based on Kalman filters. Biagioni, James, et al (2011) applied real-time data from smartphones placed inside mobile vehicles. Balika J. Chelliah, et al (2021) design model has the capability of reducing the number of private vehicles on the roads, shorter rides for the passengers, and reducing the fares for the rides.

**1.2 Study Review Involving New York City Taxi Fare**

New York City comprises 5 Boroughs cited where the Hudson River intercepts the Atlantic Ocean. It’s the most populous city in the United States, with a 2020 population of over 8.8 million people and it is the largest metropolitan area in the world.

Being described as the cultural financial and media capital of the world, New York city significantly influences commerce, entertainment, research, technology, education, politics, tourism, dining, art, fashion, and sports (en.wikipedia.org/New\_York\_City).

The five boroughs that make up NYC, are Brooklyn (kings county), Queens (Queens County), Manhattan ( New York County), the Bronx (Bronx County), and Staten Island (Richmond county). According to Wikipedia (en.wikipedia.org/wiki/Taxis\_of\_New\_York\_City), there are two categories of varieties of taxi cabs in New York City. The yellow medallion taxis, which can pick up passengers anywhere in the five Boroughs, and the apple green taxicabs are also known as boro taxis are allowed to pick up passengers in upper Manhattan, the Bronx, Brooklyn, Queens (excluding LaGuardia Airport and JFK International Airport) and Staten Island. Both types structured the fare charges the same way.

There were 51,396 individuals licensed to drive medallion taxicabs in 2014, which dropped slightly to 13,587 medallions. Competition from ridesharing companies caused a decline in taxi patronage since 2011. This must have been as a result of the fare system adopted by the ridesharing companies unlike the Txicab system using their old system of fare charge.

**1.3 Problem statement**

**1.3.1**  **Business definition**

* Which of the data attributed influence the New York City Fare?
* When most and where most do the fare increase or decrease?
* New York City fare Prediction is ML-based.
* Distance, trip time, the hour of the day, any day of the week could have impacted positively or negatively.

**1.3.2**  **Business Objective**

* Determine which of the ML models accurately predict New York City Taxi fare.

**2.0 Data Source and collection**

The data source is from the Kaggle competition: <https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data>

It contains 3 data files namely (i) Train.csv, which has 55 million rows and 8 columns. The second file is Test.csv which contains 9914 rows and 7 columns. The third file is the sample submission.csv with 9914 rows and 2 columns. This last file is for those who wish to enter the competition and that file is in submission format.

The dataframe of the train.csv was assigned and stored in df\_train. This contains the names of the following column: keys, fare\_amount, pickup\_datetimme, pickup\_longitude, pickup\_latitude, dropoff\_longitude, dropoff\_latitude, and passenger\_count.

Also, we assigned test.csv to df\_test, which has the same column names without the fare\_amount column. The sample\_submission.csv is stored in the df\_sample\_submission and it has two columns namely key and fare\_amount.

**2.1 Introduction to Data Wrangling**

This section involves performing some processes to bring the data to a standard that would be able to give the required results needed for analysis. It includes converting and mapping data from one raw form into another, perhaps format to achieve convenient consumption and presentation of the dataset.

Most of the errors occurred due to methods applied during data acquisition or sources or merging data from different sources. It may have occurred as a result of the instruments used in data collection. Moreso, human errors may introduce such effects in the data, and if ignored would lead to spurious analytic interpretation and erroneous conclusions or inferences.

Most often, acquired data are unstructured and unorganized which must require organizing and extraction of useful components for easier computation and analysis.

**2.1.1 Data Wrangling**

Out of the 55 million rows of the train.csv. One percent was used for this investigation. Assuming the local machine for this investigation has GPU and Cuda, Cudf, the entire 55million rows would have been analyzed.

Essentially, the columns were examined to ascertain whether the datasets were categorical or ordinal, or numerical or continuous. Identifying these data types would enable the transformation of the data for exploratory data analysis.

Most datasets are pruned to have missing values or wrong data entry. These were applied and replaced with mean, median, and mode of respective columns or attributes. Only the dropoff\_longitude and dropoff\_latitude have missing values from the raw data. In these cases, using Sklearn impute, the NumPy NaN values were imputed with strategy attribute “most frequent”

**2.1.2 Feature Engineering**

To have some insights into the data, Machine Learning fits mathematical notations to the sets. Features are inputs to any model, and these features are numerical representations of an aspect of real-world datasets.

* The fare\_amout column contains $ symbols, which made all the values to be strong and can’t be used for computation. The column had to be converted to numeric values using the Pandas pd.to\_numeric function.
* Pickup\_datetime decomposition. To understand the exact time, the pickup\_datetime was transformed into the year, month, day, hour, and minute. This was achieved using the Pandas dot underscore DateTime. After this transformation or decomposition, the pickup\_datetime was dropped.
* Outliers: Figure 1.1 shows outliers of boxplots of fare\_amount and passenger\_count.

**Table 1.1: Summary Statistics of df\_train showing ranges**

Table

Description automatically generated

* Also, summary statistics of the train.csv dataframe

(Table 1.1 above) explicitly depicted outliers such as far\_amount range from -$44.90 to $500 (-44.90 won’t be acceptable), pickup\_longitude with range -3377.68 to 2140.6; dropoff\_longitude with range -3383.297 to 40.85. Based on these insights, the outliers on passenger\_count, fare\_amount, pickup & dropoff latitudes, pickup longitudes, and dropoff longitudes would be trimmed within the following ranges:

* + - * + fare\_amount: 1 to 500
        + longitudes: -75 to -72
        + latitudes: 40 to 42
        + passenger\_counts: 1 to 6

A picture containing diagram

Description automatically generated

**Figure 1.1: Boxplots of fare\_amount and passenger\_count outliers**

Figure 1.2(a) shows the pair plot of fare amount and passenger count after removing the outliers. Figure 1.2(b) is the corresponding histogram for the two attributes.

A picture containing calendar

Description automatically generated

Chart

Description automatically generated

**Figure 1.2 (a & b): Pair-plot and Histogram of Fare amount and Passenger count**

* Converting longitudes and latitudes to distance. To access pickup coordinates, the distance of the trip must be known as well as the duration of such a trip. Haversine formula was used to achieve this, after converting decimal degrees to radians. Then the pickup longitudes and latitudes with dropoff\_longitudes and latitudes were dropped from the dataframe. Hence, a distance column was created. Tot minute as the trip duration was created by converting hour and minute for each of the rows. The next step involved converting all the new attributes (Hour, Minute, date, day, mouth, Year) t integers.

**3.0 Introduction to Exploratory Data Analysis (EDA)**

We used Python visualization techniques to examine the effects of some attributes on the taxi fare amount. These attributes were created from the original dataset attributes or columns. Distributions and correlations would be presented in this section. However, the emphasis would be on fare amount and passenger count.

**3.1 Co-ordinate System**

Pickup longitudes and pickup latitudes were the trip start point at pickup DateTime, and the trip ended at the dropoff longitudes and dropoff latitudes. From these, trip distances were derived as well as the equivalent total trip durations (known as the tot\_minute). The co-ordinate system could only be related to the taxi fare through the trip distance per row and trip Hour, trip day, trip minute per row. In the next section 3.2, graphical representations of these would be presented.

**3.2 Pickup Datetime and Datetime system**

As shown in sections 2.1.1 and 2.1.2, the feature engineering transformation, we can visualize the fare amount variation with hour, see figure 2.1 where the fares were high between 3 am and 6 am probably for passengers getting to the airports for flights.

**Chart, bar chart

Description automatically generated**

**Figure 2.1: Average Fare Amount over Hour**

And also increases between 1 pm and 3 pm. The average fare amount over day shows the same trending for all days (figure 2.2).

Chart, bar chart

Description automatically generated

**Figure 2.2: Average fare amount over the days**

Examining the average fare amount over Years demonstrates exponential fare increase over the years. Starting from the year 2009 to the year 2015 ( see figure 2.3), with the year 2014 and2015 almost at the same level. These were accumulated or volume of trips per year.

Chart, bar chart

Description automatically generated

**Figure 2.3: Average Fare Amount over the Years**

Figure 2.4 shows the average fare amount over total trip time per row. There were sharp spikes in the fare amount, which appears like a sinusoidal wave.

Chart, bar chart, histogram

Description automatically generated

**Figure 2.4: Average Fare Amount over the Total Minute per row**

Also, shown is the relationship between the average fare amount and the number of passengers or passenger count (figure 2.5). The result indicated that a passenger count of 6 had the highest average fare followed by a passenger count of 2.

Chart, bar chart

Description automatically generated

**4.0 Machine Learning Algorithms**

Machine Learning Algorithms are classified into 4 types namely: Supervised Learning, Unsupervised Learning, Semi-supervised Learning, and Reinforcement Learning. Machine Learning is to model relationships and dependencies between a target prediction output and the independent features to predict the output values for new data based on the relationships learned from the previous datasets.

In Reinforcement Learning, the agent applies observations gathered from interaction with the environment to simulate actions that would maximize the reward or decrease the risk. This type of learning continuously iteratively learns from the environment.

Unlike Supervised learning, Unsupervised learning involves no assistance or supervision, and the system learns about patterns in a dataset. The algorithms try to use techniques based on the input data to mine for rules, detect patterns, and summarize and group the data points. The input data has no labels.

Semi-supervised Learning falls between Supervised and Unsupervised making use of both labeled training data and unlabeled training data, which would result in considerable improvement in learning accuracy.

In this study, Supervised Learning algorithms would be used. The first step was to subject the dataframe, df\_train for training and validation by using the sklearn dot model underscore selection import train underscore test underscore split into X and Y variables. The X independent variable contains all the attributes of the df\_train with exception of the fare\_amount which is the target variable. The Y then stores the target variable, fare\_amount which would be predicted as a result of the effects of the independent variables in X.

**4.1 GridSearch – XGBOOST Regressor**

Gradient Boosting describes a class of ensemble machine learning algorithms for classification or regression predictive modeling problems. These ensembles are constructed from decision tree models, in which trees are added one at a time and fit to correct the prediction errors made by previous models. Extreme Gradient Boosting (XGBoost) is an effective open-source implementation of gradient boosting algorithms.

To evaluate the XGBoost, after instantiating the following conditions were applied for this regressor: objective – reg-linear, colsample\_bytree = 0.3, learning\_rate = 0.1, max\_depth = 5, alpha = 10, n\_estimators = 10. The model was fitted and later subjected to predict. The RMSE was 9.145. We deployed KFOLD validation applying hyperparameters of num\_boost\_round = 100, and n-fold = 12, the result was an improved RMSE of 4.95.

Table 3.1 shows the results of predicted fare\_amount based on this model. Both actual fare\_amount and predicted fare\_amount are shown for sampled rows picked at random.

**Table 4.1: Actual fare amount vs Predicted fare amount for XGBoost**

|  |  |  |
| --- | --- | --- |
| S/N | Actual Fare\_Amount | Predicted Fare\_Amount |
| 0 | 7.50 | 7.27 |
| 1 | 11.30 | 6.94 |
| 2 | 16.67 | 7.16 |
| 3 | 8.50 | 7.73 |
| 4 | 10.50 | 7.60 |
| 5 | 5.00 | 6..90 |
| 6 | 5.00 | 6.92 |
| 7 | 6.50 | 6.71 |

Figure 4.1 presents the visual representation of the actual fare amount and the predicted fare amount.

A picture containing calendar

Description automatically generated

**Figure 4.1: XGBoost Actual fare amount vs the predicted fare amount.**

A Scatter plot of the actual fare amount against the predicted fare amount looks like the regressive display(see figure 4.2).

FChart, scatter chart

Description automatically generated

**Figure 4.2: Scatterplot of Actual fare amount vs Predicted fare amount**

**4.1.1 Shapley Interpretation of the XGBoost Model**

In the XGBoost model, each of the independent variables must have contributed a quota to the target variable. Shapley’s feature importance was used to examine this. Relative importance against these features shows individual contributions to the fare amount prediction. Distance stands out as the most effective with 0.95 relative importance and year contributing relative importance of 0.089, see figure 4.3.

Figure4FghShape

Description automatically generated

**Figure 4.3: Relative importance of features in XGB model**

Further application of Shapley confirms the above results as shown in the Shapley summary plot(figure 4.4). Towards the right, with red indicating high feature value distance scores high and, year less one in the score as shown in the figure below. Every other feature has dots of red but is insignificant to the contribution or impacting the fare amount prediction.

Graphical user interface, application

Description automatically generated

**Figure 4.4: Shapley summary plot of XGB model showing the impact on model output**

As shown in figure 4.5(a), the shap dependence plot for distance is the shap value for distance with tot\_minute trailing. Figure 4.5(b) is the dependence plot of passenger\_count as impacted over the year. Passenger\_count 2 and 6 showed impaction. However, figure 4.5(c) demonstrates how the tot\_minute affected the fare\_amount due to the effect of the distance which is in contrast to figure 4.5(a). Hour dependence over passenger\_count did not impact anything. The hour and passenger\_count are not related.

|  |  |
| --- | --- |
| a | b |
| c | d |

**Figure 4.5(a, b, c, d): Shapley Dependence\_plots: distance, passenger\_count, tot\_minute, Hour respectively**

Figure 4.6(a) demonstrates the effects of the features on the fare\_amount, the year 2009, day 5, passenger\_count 1, and tot\_minute of 483 contribute towards the zero for the 10th observation, but date = 14, month = 11, and distance of 3.676 contribute towards the positive point 1. The red is an indication of the positive contribution of the prediction of fare\_amount which in this case is 11.11, which is lower than the base value of 11.34 due to the negative effect of mostly year = 2009, day = 5, passenger\_count = 1, and tot\_minute = 483. On the other hand, figure 4.6(b) shows a stack of these features (Lundberg Scott M. et al 2018). The values are sorted by default in order of similarity. Other selections include output/target values, original order. Or by any feature.

On the left dropdown, the shap impact for a single feature can be filtered. From 0 to 2200, blue region, the prediction of fare\_amount had been pulled down while the prediction goes up as shown from the region of red stacks.

|  |
| --- |
| a |
| b |

**Figure 4.6(a, b): Shapley force plot 10th observation and force plot for entire**

**4.2 GridSearch LightGBM**

GridSearch LightGBM is the second algorithm to be tried in the taxi fare\_amount prediction. The advantage of the LightGBM algorithm over other machine learning algorithms is its ability to converge much faster. This is because it makes use of leaf-wise tree growth algorithms compared to the depth-wise tree growth algorithms associated with other algorithms. A possible drawback to this is that the leaf-wise growth may introduce overfitting in cases where appropriate parameters are not used.

To evaluate this algorithm, the original split df\_train dataframe was used, and the following hyper\_params applied: task: train, boosting\_type:gbdt, objective:regression, metric: 11, 12, learning rate: 0.005, feature fraction: 0.9, bagging fraction: 0.7, max\_depth = 8, num\_leaves = 128, Max\_bins: 512, and num\_iterations:10000. The model was fitted and subjected to prediction and outcome of the prediction is shown in Table 4.2.

**Table 4.2: Actual fare amount vs Predicted fare amount for LightGBM**

|  |  |  |
| --- | --- | --- |
| S/N | Actual fare amount | Predicted fare amount |
| 0 | 5.70 | 7.46 |
| 1 | 4.50 | 4.94 |
| 2 | 8.50 | 9.09 |
| 3 | 29.30 | 23.75 |
| 4 | 7.30 | 6.88 |

Icon

Description automatically generated

**Figure 4.7: Distributions of the actual and predicted fare\_amount**

A picture containing calendar

Description automatically generated

**Figure 4.8: Actual Fare\_amount and Predicted Fare\_amount for LightGBM algorithm**

**4.2.1 Shapley Interpretation of LightGBM**

In this LightGBM model, every independent variable contributed a quota to the target variable. Shapley’s feature importance was used to examine this. Relative importance against these features shows individual contributions to the fare amount prediction. The trip total time is known as tot\_minute, contributes mostly to the fare amount prediction using this algorithm. Distance also contributed as much as shown in figure 4.9. It could be seen that virtually all the independent variables impacted the fare amount prediction with only the passenger count having the least in relative performance.

Chart, bar chart

Description automatically generated

**Figure 4.9: Relative importance of features on fare amount prediction based on LightGBM model**

The Shap value impact on the model output shows that the distance contributes immensely to the model output as shown in figure 4.10. The red color indicates a high impact contribution to the prediction of the fare amount. Similarly, the year attribute has contributed to the prediction output.

Graphical user interface

Description automatically generated with medium confidence

**Figure 4.10: Shapley summary plot of XGB model showing the impact on model output**

|  |  |
| --- | --- |
| a | b |
| c | d |

**Figure 4.11 (a, b, c, d): Shapley Dependence\_plots: distance, passenger\_count,**

**tot\_minute, Hour respectively**

In figure 4.11(a) Shap dependence plot shows that distance impact the fare amount where there is a positive gradient and the associated year attribute did not impact the prediction of the fare amount as much. Figure 4.11(b) is the passenger count shap dependence plot, only passenger count 2 fails to contribute to the prediction. As had been shown in the feature importance, figures 4.11(c & d) demonstrate the effect of the tot\_minute and hour on the fare amount prediction in this LightGBM algorithm.

The shap force plot for the 10th observation presented in figure 4.12(a) shows that year, 2009, day 5, tot\_minute were shown in blue had negatively impacted the prediction or the final output which is 10.30 in contrast to the base value of 11.34. However, the month, 11, and distance shown in red arrows had contributed to the prediction towards positive value.

In figure 4.12(b), the values were sorted by default in order of similarity. It could be possible to select output/target value, original order, or by any feature. On the left-hand side of the x-axis, the red color was an indication of the impact of distance on the fare amount output. The effect of distance started dwindling from 400 or 500 as shown in the figure below.

|  |
| --- |
| a |
| b |

**Figure 4.12(a, b): Shapley force plot 10th observation and force plot for LightGBM**

**4.3 Ridge Regression**

Most often Ridge Regression algorithm is applied to analyze any data that suffer from multicollinearity. Its performance is based on L2 regularization. Ite issue of multicollinearity occurring leads to least-squares being unbiased and variances are large and this would cause the output results in predicted values to be far away from the actual values.

Importing Ridge from sklearn dot linear underscore model, and using the already trained dataset of df\_train, and applying alpha = 0.1 parameters with normalizing = True, the training data was instantiated followed by fitting and prediction. Table 4.3 shows the results of the prediction against the actual fare amount.

**Table 4.3: Actual fare amount vs Predicted fare amount for Ridge Regression**

|  |  |  |
| --- | --- | --- |
| S/N | Actual Fare Amount | Predicted Fare Amount |
| 0 | 6.50 | 6.69 |
| 1 | 6.90 | 6.82 |
| 2 | 2.90 | 5.25 |
| 3 | 6.50 | 9.64 |
| 4 | 4.90 | 4.61 |
| 5 | 8.00 | 11.17 |
| 6 | 54.54 | 31.92 |
| 7 | 6.50 | 8.07 |
| 8 | 4.50 | 6.17 |
| 9 | 6.50 | 6.96 |

Figure 4.13 shows the Ridge regression prediction, which is a sample of the entire result. The actual fare amount on the left top pane is not conspicuous as the predicted values on the bottom right pane. The result of correlating the actual and predicted values was shown on the top right and left bottom of figure 4.13.

A picture containing text, electronics, screenshot, display

Description automatically generated

**Figure 4.13: Ridge Regression of Actual and Predicted Values.**

The Ridge regression performance shows RMSE of 16.71 and MAPE of 26.5.

A picture containing chart

Description automatically generated

**Figure 4.14: Actual and Predicted values from the Ridge Regression**

**4.4 Random Forest Regressor**

Random Forest is also an ensemble technique with the capability of performing both regression and classification tasks with the application of multiple decision trees and a technique of Bootstrap Aggregation known as bagging. Random forest is described as a meta estimator that can fit several classifying decision trees on various sub-samples of the dataset and uses averaging to improve the predictive accuracy and control over-fitting. The Random forest supports both continuous and categorical features.

Next, we instantiate the Random Forest Regressor using the trained dataset of X independent variables and target variables. The following parameters were used: max\_depth of 10, random state of zero, n\_jobs of negative one, and n estimators of 10.

Based on this model it was subjected to fit and predict on the trained dataset. Table 4.4 shows the actual and predicted fare amount values with performance evaluations showing MSE of 23.98, RMSE of 11.99, and MAPE of 21.26. Figure 4.15 shows the actual and predicted fare amount, and figure 4.16 is the heatmap of the values.

**Table 4.4: Actual fare amount vs Predicted fare amount for Random Forest Regressor**

|  |  |  |
| --- | --- | --- |
| S/N | Actual Fare Amount | Predicted Fare Amount |
| 0 | 6.50 | 6.06 |
| 1 | 6.90 | 6.15 |
| 2 | 2.90 | 4.69 |
| 3 | 6.50 | 9.13 |
| 4 | 4.90 | 4.72 |

Scatter chart

Description automatically generated

**Figure 4.15: Actual and Predicted Fare amount for Random Forest Regressor**

A picture containing icon

Description automatically generated

**Figure 4.16: The HeatMap of the Predicted and Actual Fare Amount for Random Forest Regressor Model**

**4.4.1 Shapley Interpretation of Random Forest Regressor**

The Shapley interpretation of RandomForest regressor will examine the feature importance as well as the shap dependence plot of some of the features. Figure 4.17 is the shap summary plot of the feature importance. It demonstrates that the most important feature that contributes to the prediction was the distance, followed by the slight performance of year and tot\_minute.

**Shape

Description automatically generated with low confidence**

**Figure 4.17:Relative importance of features on fare amount prediction based on RandomForest**

**Graphical user interface, text, application

Description automatically generated**

**Figure 4.18:Shapley summary plot of RandomForest model output**

|  |  |
| --- | --- |
| a | b |
| c | d |

**Figure 4.19(a, b, c, d): Shapley Dependence\_plots: distance, passenger\_count,**

**tot\_minute, Hour respectively**

Shap summary plot for the RandomForest Regressor shows that distance still contributed immensely to the final prediction in comparison to the year, see figure 4.18 above. Figure 4.19(a) demonstrates the effect of distance on fare amount prediction while year does not impact according to the shap dependence plot. Similarly, passenger count for RandomForest regressor made no impact as there was no red color indicator as a positive contribution (see figure 4.19(b)). Figure 4.19(c) shows some impacts by the tot\_minute. The same applied to Hour feature as shown in figure 4.19(d).

For the shap force plot for the 10th observation, the base value was 11.82 and the predicted value was 7.05 an indication of poor prediction since the distance contributed negatively as shown by the extend of the blue arrow while the tot\_minute and year of 2014 were short arrows of red high. The effect was the prediction of 7.05 below the base value, see figure 4.20(a). Between the distances 0 and 400, there was a shap force effect on the prediction of fare amount, see figure 4.20(b).

|  |
| --- |
| a |
| b |

**Figure 4.20 (a, b):** **Shapley force plot 10th observation and force plot for RandomForest**

**4.5 Ordinary Least Square Regression**

Ordinary least Square Regression can predict an output result with an acceptable error margin if a set of input parameters are applied. It is one of the Machine Learning algorithms, that falls under the Supervised learning categories. It has the capability of estimating the unknown parameters by recreating a model that minimizes the sum of the squared errors between the observed data and the predicted values. OLS can work for both univariate datasets and multivariate datasets.

To execute this algorithm, the previously trained dataset of X-independent variables and independent target variables were also used in this algorithm. After instantiating the LinearRegression which OLS is a member class, fit and predict the changes were performed(see Table 4.5).

**Table 4.5:Observed Fare Amount and Predicted Fare amount from OLS**

|  |  |  |
| --- | --- | --- |
| S/N | Actual Fare Amount | Predicted Fare Amount |
| 0 | 6.50 | 6.23 |
| 1 | 6.90 | 6.37 |
| 2 | 2.90 | 4.66 |
| 3 | 6.50 | 9.52 |
| 4 | 4.90 | 3.90 |
| 5 | 8.00 | 11.16 |
| 6 | 54.54 | 33.97 |
| 7 | 6.50 | 7.75 |
| 8 | 4.50 | 5.65 |
| 9 | 6.50 | 6.51 |
| 10 | 16.90 | 17.46 |
| 11 | 4.50 | 6.97 |

**Figure 4.21: Sample plot of Actual and Predicted Fare Amount from OLS model**

As seen in figures 4.21 and 4.22, actual fare amount vs predicted values, least-square minimization of errors was achieved. Only at data point six does the actual value exceed the predicted value by a certain large percent. A scatter plot showed that the actual vs predicted has a positive slope. The R2 is between 0.670 and 0.89. Figure 4.23 demonstrates the histograms of the two values with count values predicted less than the actual fare amount.

A picture containing calendar

Description automatically generated

**Figure 4.22: Pair-plot of Actual and Predicted Fare Amount from OLS model**

A picture containing text, shoji, crossword puzzle, clipart

Description automatically generated

**Figure 4.23: Histograms for Actual Fare Amount and Predicted Fare Amount from OLS Model**

**5.0 Discussion and Model Assessment**

We shall evaluate the performance of these algorithms based on performance evaluation metrics, the run-speed, and Shapley interpretation. The five algorithms had shown good predictions, however, their error margins signaled low score grading to some of them. LightGBM showed a very low RMSE and MAPE, (see Table 5.1 below) which has to give it the best algorithm for this problem with the fast ability to run and execute fast.

**Table 5.1:Evaluations of the different algorithms’ performances**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Model** | **RMSE** | **MSE** | **MAPE** | **R2** |
| XGB | 9.15(4.95) | 83.64(24.50) | 47.62 | 0.89 |
| LightGBM | 0.2469 | 20.22 | 20.40 | 0.79 |
| RandomForest | 11.99 | 23.98 | 21.30 | 0.76 |
| Ridge Regression | 16.71 | 33.43 | 26.50 | 0.65 |
| OLS | 5.70 | 32.43 | 24.90 | 0.67 |

A look at the attributes of the df\_train dataframe, figure 5.1 shows the histograms with fare\_amount, passenger\_count, distance, day, and year well discretized. Fare amount shows a correlation with distance, which has a positive slope (figure 5.2). Figure 5.3 is the heatmap of the df\_train dataframe attributes, only distance correlated well up to 0.8 with fare\_amount while Hour correlated with tot\_minute to 1.0

**A picture containing text, crossword puzzle, clipart

Description automatically generated**

**Figure 5.1: Individual Histograms of the attributes of the dataframe df\_train**

In terms of Shapely interpretation, the relative importance of features impacting fare\_amount prediction shows distance and year for XGBoost (figure 4.3), but in LightGBM the entire attributes contributed (see figure 4.9), and in Random Forest Regressor only distance and year contributed to fare\_amount, figure 4.17. Moreover, the Shapley force plot for 10th observations for XGBoost showed the date, month, distance having short red arrows resulting in the predicted value of 11.11, however below the base value of 11.44 better than in LightGBM that has 10.30 below the 11.34 base value; Random Forest Regressor force plot was very poor because of a 7.05 prediction compare to the base value of 11.82 (see figures 4.6a, 4.12a, and 4.20a)

**Chart, scatter chart

Description automatically generated**

**Figure 5.2: Correlation of Fare\_amount with Distance**

**A picture containing qr code

Description automatically generated**

**Figure 5.3: Heatmap of the feature engineering df\_train dataframe attributes**

**5.1 Conclusion**

### Five techniques were deployed to determine the Taxi fare\_amount predictions. The purpose of trying the five models is to ascertain which of them would be able to give the best predictions as evaluated by the error standard and running time. These models are XGBoost, LightGBM, Ridge, RandomForestRegressor, and Ordinary Least Square(OLS)

### xgboost has an RMSE of 9.15 and after changing some of the params it attained 4.95 which however was not better than LightGBM whose RMSE of 0.2465. This shows that among these models LightGBM is the best model that should be utilized in New York City Taxi Fare amount prediction. However, R^2 for the RFR, Ridge, and OLS is 0.74, 0.6596, and 0.66 respectively.

### The observation is that the key parameters that would have been used for determining fare\_amount had not been employed in this scenario. . These key parameters are distance and time. 1% or 550000 was used in this study out of the 55 million rows of the dataset.

* 1. **Recommendations**

1. LightGBM, best-selected model based on its metric, robustness, fastness, and very low RMSE.
2. It would guide taxi fleet operators to know the fare range to charge their commuters as distance, locations, and time indicators show.
3. This model is easy to deploy operationally by taxi fleet operators and commuters as well. And would be a great compass for operators against overcharging their customers.
4. The model can generate predictions within milliseconds, and it’s known to have a low computational requirement.
5. A real-time web application can be developed based on this algorithm for this market segment.

**References**

Balika J.Chelliah, et al,(2021) Taxi fare prediction system using key feature extraction in

Artificial Intelligence. Turkish Journal of Computer and Mathematics

Education, Vol. 12.No.6, pg3803 -3808.

Biagioni James, et al, (2011) Rasytracker: automatic transit tracking, mapping, and arrival

Time prediction using smartphones, Proceedings of the 9th ACM Conference

on Embedded Networked sensor systems. ACM.

Christophoros Antoniades, et al, (2016) Fare and Duration Prediction: A study of New York

City Taxi Rides.

Greta Nagai and Alex Combessie, (2019) Taxicab fare prediction Machine Learning in

Real-time

<https://www.kaggle.com/c/new-york-city-taxi-fare-prediction/data>

<https://en.wikipedia.org/wiki/semi-supervised_learning>.

Lundberg Scott M, et al (2018) Explainable machine learning predictions for prevention of

hypoxaemia during surgery, Nature Biomedical Engineering Vol 2,

pg 749 - 760

Monalisha Ojha, et al, (2019) Taxi Fare Prediction. https://www.overleaf.com/articles/Taxi-

fare-prediction.

Vanajakshi, L., S.C. Subramanian, and R. Sivanandan,(2009) Travel time prediction under

heterogeneous traffic conditions using global positioning system data from

buses IET intelligent transport systems, 1-9

[www.wikipedia.org/New\_York\_City](http://www.wikipedia.org/New_York_City)