NYC Taxi Trip Time Prediction

By Vivek Kumar Soni

Abstract:

A lot of streets and roads in New York city are quite busy due to traffic jams, construction, or roadblockage etc. Therefore, it is very important to predict the trip duration of taxi so that the user will know how much time it will take tocommute from one place to other. Also, due to the increasing popularity of app-based taxisuch as ola or uber and there competitive pricing levels. Decisions has to be taken by the user for opting which one to choose basedon trip pricing and duration. This prediction also helps drivers to choose route havinglesser trip time. We were provided with dataset which is released by NYC Taxi and Limousine Commission. This datasetcontains pickup time, drop-off time, geo- coordinates, number of passengers, trip duration and several other variables.

Our primary motives are to analyze the dataset, perform feature engineering to comes up with suitable independent features and building a good model that will help us in predicting the trip duration of NYC taxi.

Here, for prediction the taxi trip duration we have applied a linear regression, lasso, and ridge regression and then we have applied XGBoost and LightGBM. To find out which will give better acuracy and with lesser amount of prediction time. At last, a comparison of the two mentioned algorithms facilitates us to decide that XGBoost is more fitter and efficient than Multi-Layer Perceptron for taxi trip duration-based predictions

Keywords: machine learning, surge pricing, dynamic pricing, classified labels

1 Problem Statement

The dataset is based on the 2016 NYC Yellow Cab trip record data made available in Big Query on Google Cloud Platform. The data was originally published by the NYC Taxi and Limousine commission (TLC).

The main objective is to build a predictive model, which could help them in predicting the trip duration of taxi. This would in turn help them in matching the right cabs with the right customers quickly and efficiently.

- id - a unique identifier for each trip
- vendor_id - a code indicating the provider associated with the trip record
- pickup_datetime - date and time when the meter was engaged
- dropoff_datetime - date and time when the meter was disengaged
- passenger_count - the number of passengers in the vehicle (driver entered value)
- pickup_longitude - the longitude where the meter was engaged
- pickup_latitude - the latitude where the meter was engaged

- dropoff_longitude - the longitude where the meter was disengaged
- dropoff_latitude - the latitude where the meter was disengaged
- store_and_fwd_flag - This flag indicates whether the trip record was held in vehicle memory before sending to the vendor because the vehicle did not have a connection to the server - Y=store and forward; N=not a store and forward trip
- trip_duration - duration of the trip in seconds

2 Introduction

More than 7 billion people exist on earth. With necessities of food, water and shelter there also a key requirement of commutating from one place to other. Rapid advancement in technology in the last two decades leads to adaption of a more efficient way of transportation via internet and app-based transport system. New York city is one of such advanced city with extensive use of transportation via subways, buses and taxi services. New York has more then 10,000 plus taxi and nearly 50% of population doesn't have a personal vehicle. Due to this facts most people used taxi has a there primary mode of transport and it accounts for more than 100 millions taxi trips per year.

The dataset is based on the 2016 NYC Yellow Cab trip record data made available in Big Query on Google Cloud Platform. The data was originally published by the NYC Taxi and Limousine commission (TLC). This dataset contains around 1458644 records and 11 features.

Out of numerous machine learning algorithm we have selected Xgboost and LightGBM repressors for our used case. More accurately prediction will lead to make better taxi trip duration prediction not in New York but also applicable to other city as well in future and make user taking better decision for choosing right taxi for there commute.

2.1 Trip duration & Trip Duration Variation

Trip duration normally be calculated based on the distance between pickup and drop-off point and average speed of the vehicle covering this distance. However, there are many factors which affects the trip duration. Following are some of the factors:

- Peak hours: there are certain hours where route are might get busy due to moment of peoples commutating from office to home or vice versa.
- bad weather conditions (rain, snow, etc)
- big events or festivals
- traffic conditions

3 Feature Engineering:

3.1 Data Loading and generalcheckups

We have loaded the data from thegiven csv files using a function from pandas library. Then we checked the general information about data. We observed that the data contains 1458644 records and 11 features. Wesee that our data contains threedifferent data types i.e. floats, stringsand datetime objects.

3.2 Null values Treatment

We inspected the dataset and found out that our dataset has no null valuepresent in it. So, no need to do null value treatment.

3.3 Exploratory Data Analysis

We begin our EDA by first checking the distribution of our dependent variable i.e. trip duration. Weobserved that the data is highly positively skewed. We also plottedthe box plot and observed that there are many outliers present in the variable. To cross check this trip duration we have calculated the difference in pick and drop off timingand matched with trip duration we observed no difference. Thus, there isno miscalculation or falsified entries. To eliminate the outliers, we have segregated the data variable into different segment and observed that majority of trip duration is within an hour some observations are within twodays but a very few observation are having more than two days. We eliminate such values from out dataset.

We removed id variable as it doesn't give much interpretation. We then calculated the distance based on haversine formula from pickup and drop-off latitude and longitude. Thenwe plotted the box plot for the variable and observed there are many outlier so we segregate this variable and see that most of the trip are within 10km, some trip are within50km while a very few trip crosses 50km. so we eliminate trip with 0 andabove 50km distance.

We then checked for categorical variable store_and_fwd_flag and passenger_count. We observed the store and fwd. flag contain majority of one category. So we drop this feature. Passenger count variable has entries from 0 to 9. Since there is no trips with 0 passenger either this a miss entry or the driver forgot to enterpassenger count of that trip. Also in ataxi maximum six person are allowedto sit including minor. So we eliminate 0 and 7-9 records from our dataset.

We also created some more feature i.e. pickup month, pickup weekday and pickup hour. To get a good insight of trip duration and drop pickup date and drop-off timecolumn. Then we checked forcorrelation between variables and observed that geographic coordinates are very less correlated and VIF is also high between this variables so wedrop off this variable from our data set.

3.4 Encoding of categorical columns

Since some of our categorical variable are in string format. So we cannot passed this variable to our model directly so we have to use one hot encoding to convert it into numerical variable having binary integers 0 and 1.

3.5 Standardization of features

This is one of the important step for getting good accuracy as you can seethere are some columns having different ranges of values then other column. Therefore. It is important to do scaling the data so that our data set will have uniformity and we getgood accuracy. So, here we use MinMaxscaler function.

3.6 Fitting different models

For modelling we tried various classification algorithms like:

1. **Linear Regression**
2. **XGBoost classifier**
3. **LightGBM**

3.7 Tuning the hyperparametersfor better accuracy

Tuning the hyper parameters of respective algorithms is necessary for getting better accuracy.

4 Algorithms:

4.1 Linear Regression:

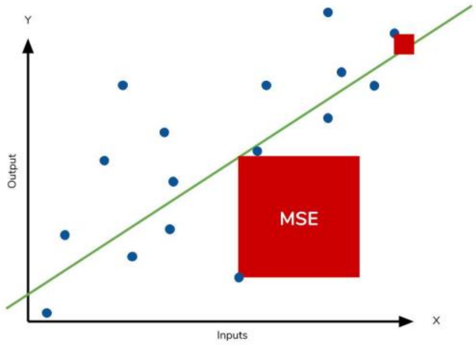
Linear Regression is a regression of dependent variable on independent variable. It is a linear model that assumes a linear relationship between dependent (y) and independent variables (x). The dependent variable (y) is calculated by linear combination of independent variable (x).

$$Y=B_0+B_1x_1+B_2x_2$$

The cost function for linear regression isgiven by:

Minimum sum of square error

$$MSSE= \sum_{1}^{n}(Y_i act - Y_i pred)^2$$



4.2 XGBoost:

Sometime in building a model. We cannot just rely on the result of a single model. Ensemble offer a systematic solution for this by combining the prediction of multiple model. The resultant model is superior then individual model called base learner and is obtained from aggregation of base learner prediction. Bagging and boosting are two types of ensemble method.

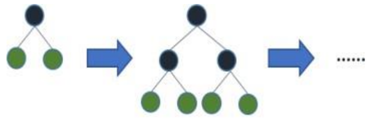
XGBoost comes under boosting and is known as extra gradient boosting. GBM first calculates the model usingX and Y then after the prediction is obtain. It will again calculates the model based on residual of previous model, here loss function will give more weightage to error of previous model. and this process continuous until MSE gets minimizes. XGBoost is just an extension of GBM with following advantages.

- Regularization
- Parallel Processing
- High Flexibility
- Handles Missing values
- Tree pruning
- Built in cross validation
- Continuous on existing model

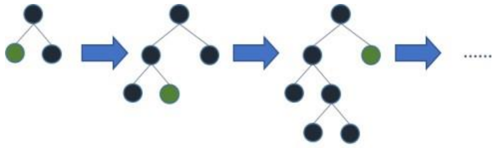
4.3 LightGBM:

Sometime in building a model Light GBM is a fast, distributed high performance gradient boosting framework. It is widely used for ranking, classification, regression, and many other machines learning task.

Light GBM is based on decision tree algorithm. But it splits the tree leaf wise rather then level wise like other boosting algorithm. So when growing on the same leaf in Light GBM, the leaf-wise algorithm can reduce more loss than the level-wise algorithm and hence results in much better accuracy which can rarely be achieved by any of the existing boosting algorithms.



Level-wise tree growth



Leaf-wise tree growth

5 Model performance:

The model performance can beevaluated by various regressionmetrics such as:

5.1 Mean Squared Error (MSE):

Mean squared error is the most widely used evaluation metric for regression task. It is the average of squared difference between actual and predicted value of dependent variable

$$\mathrm{MSE} = \frac{1}{n} \sum_{i=1}^{n} (y_i - \tilde{y}_i)^2$$

5.2 Mean Absolute Error (MAE):

Mean absolute error (**MAE**) is a measure of errors between paired observations expressing the same phenomenon. Examples of *Y* versus *X* include comparisons of predicted versus observed, subsequent time versus initial time, and one technique of measurement versus an alternative technique of measurement. MAE is calculated as the **sum of absolute errors** divided by the sample size.

$$\mathrm{MAE} = \frac{\sum_{i=1}^{n} |y_i - x_i|}{n} = \frac{\sum_{i=1}^{n} |e_i|}{n}.$$

5.3 R2 Score :

Coefficient of determination also called as R2 score is used to evaluate the performance of a linear regression model. It is the amount of the variation in the output dependent attribute which is predictable from the input independent variable(s). It is used to check how well-observed results are reproduced by the model, depending on the ratio of total deviation of results described by the model.

R2= 1- SSres / SStot

Where,

SSres is the sum of squares of the residual errors.

SStot is the total sum of the errors.

5.4 Adjusted R2 Score:

Adjusted R2 is a corrected goodness-of-fit (model accuracy) measure for linear models. It identifies the percentage of variance in the target field that is explained by the input or inputs.

R2 tends to optimistically estimate the fit of the linear regression. It always increases as the number of effects are included in the model. Adjusted R2 attempts to correct for this overestimation. Adjusted R2 might decrease if a specific effect does not improve the model.

Adjusted R squared is calculated by dividing the residual mean square error by the total mean square error (which is the sample variance of the target field). The result is then subtracted from 1.

$$Adjusted\ R^2 = 1 - \frac{(1 - R^2)(N - 1)}{N - p - 1}$$

Where

R^2 Sample R-Squared

N Total Sample Size

p Number of independent variable

Adjusted R2 is always less than or equal to R2. A value of 1 indicates a model that perfectly predicts values in the target field. A value that is less than or equal to 0 indicates a model that has no predictive value. In the real world, adjusted R2 lies between these values.

6 Hyper parameter tuning:

Hyperparameters are sets of information that are used to control the way of learning an algorithm. Their definitions impact parameters of the models, seen as a way of learning, change from the new hyperparameters. This set of values affects performance, stability and interpretation of a model. Each algorithm requires a specific hyperparameters grid that can be adjusted according to the business problem

Hyperparameters alter the way a model learns to trigger this training algorithm after parameters to generate outputs.

We used Grid Search CV, Randomized Search CV and Bayesian Optimization for hyperparameter tuning. This also results in cross validation and in our case we divided the dataset into different folds. The best performance improvement among the three was by Bayesian Optimization.

6.1 Grid Search CV-Grid Search combines a selection of hyperparameters established by the scientist and runs through all of them to evaluate the model's performance. Its advantage is that it is a simple technique that will go through all the programmed combinations. The biggest disadvantage is that it traverses a specific region of the parameter space and cannot understand which movement or which region of the space is important to optimize the model.

6.2 Randomized Search CV- In Random Search, the hyperparameters are chosen at random within a range of values that it can assume. The advantage of this method is that there is a greater chance of finding regions of the cost minimization space with more suitable hyperparameters, since the choice for each iteration is random. The disadvantage of this method is that the combination of hyperparameters is beyond the scientist's control.

7 Conclusion:

That's it! We reached the end of our exercise.

Starting with loading the data so far, we have done EDA, null values treatment, encoding of categorical columns, feature selection andthen model building.

In all these models our accuracy revolves in the range of 70 to 74%.

And there is no such improvement in accuracy score even after hyperparameter tuning.

So, the accuracy of our best model is 73% which can be said to be good for this large dataset. This performance could be due to various reasons like no proper pattern of data, too much data, not enough relevant features.

8 References:

1. MachineLearningMastery
2. GeeksforGeeks
3. Analytics Vidhya