**NYC Taxi Trip Time Prediction – Regression**

**Kajal Mahajan**

**Data Science Trainee,**

**AlmaBetter, Bangalore.**

**Abstract:**

New York City taxi rides form the core of the traffic in the city of New York. The many rides taken every day by New Yorkers in the busy city can give us a great idea of traffic times, road blockages, and so on. Predicting the duration of a taxi trip is very important since a user would always like to know precisely how much time it would require of him to travel from one place to another. Given the rising popularity of app-based taxi usage through common vendors like Ola and Uber, competitive pricing has to be offered to ensure users choose them. Prediction of duration and price of trips can help users to plan their trips properly, thus keeping potential margins for traffic congestions. It can also help drivers to determine the correct route which in-turn will take lesser time as accordingly. Moreover, the transparency about pricing and trip duration will help to attract users at times when popular taxi app-based vendor services apply surge fares

**Keywords : matplotlib, seaborn, barchart, supervised machine learning**

**Problem Statement:**

### Your task is to build a model that predicts the total ride duration of taxi trips in New York City. Your primary dataset is one released by the NYC Taxi and Limousine Commission, which includes pickup time, geo-coordinates, number of passengers, and several other variables.

**Introduction:**

Earth is filled with an enormous population that tends to move from one place to another. Advancement in technologies had led to different ways of transportation. These include buses, autos and especially taxi services. New York City is one of the highly advanced cities of the world with extensive use of taxi services. Along with a vast population, the requirement of commonly available transportation serves the common purpose as it provides a very large transportation system. New York facilitates one of the largest subway systems in the world and comprises various green and yellow cabs which approximately count of around 13,000 taxis. Most of the population of New York depends upon public transport, and it has been estimated that 54 percent of the people do not own a car or a personal vehicle. As a matter of fact, it accounts for almost 200 million taxi trips per year.

**Variables description:**

New York City Taxi Duration dataset is taken from the Kaggle website which provides free access to complex challenges. This dataset helps us to predict the trip duration of a taxi ride taking into account the different factors that affect the ride duration. Along with the above-mentioned, one more dataset gets included which involves the climatic conditions of the city. Both of these datasets are combined using pre-processing techniques to create a single dataset that can be used further for accurate trip duration prediction. Some of the important attributes of the dataset are discussed below:

* id, which provides a unique identification to a trip.
* vendor id, a unique code which gets assigned to the different cab companies.
* pickup datetime, starting statistics of the pickup.
* dropoff datetime, ending statistics of the pickup.
* passenger count, passengers travelling in a particular trip.
* pickup longitude, longitudinal location of the pickup.
* pickup latitude, latitudinal location of the pickup.
* dropoff longitude, longitudinal location of the drop off.
* dropoff latitude, latitudinal location of the drop off.
* store and fwd flag, a code to identify whether the data is stored on the device and then gets forwarded to the database.
* trip duration, the total time of the trip in seconds.

The second dataset comprises the climatic data of the city which includes vital information such as the time of rainfall, sunlight, and various other factors which can be used for better prediction of the taxi trip.

**Steps Involved:**

**Data Collection:**

Data collection is the process of collecting, measuring and analysing different types of information using a set of standard validated techniques. The main objective of data collection is to gather information-rich and reliable data, and analyse them to make critical business decisions. Once the data is collected, it goes through a rigorous process of data cleaning and data processing to make this data truly useful for businesses. It refers to the process of finding and loading data into our system.

Pandas library is used to loading our data in our system in python. Using pandas we can manipulate data easily.

**Data Cleaning:**

Data cleaning refers to the process of removing unwanted variables and values from your dataset and getting rid of any irregularities in it. Such anomalies can disproportionately skew the data and hence adversely affect the results. Some steps that can be done to clean data are:

* Handling missing values: There are always some missing values in dataset. If we don’t remove or handle those missing values then that can cause a trouble in our analysis. Removing or replacing those missing values with something meaningful is very important so that our data will have no missing values.

## 

**Exploratory Data Analysis (EDA):**

Exploratory Data Analysis is a data analytics process to understand the data in depth and learn the different data characteristics, often with visual means. This allows you to get a better feel of your data and find useful patterns in it.

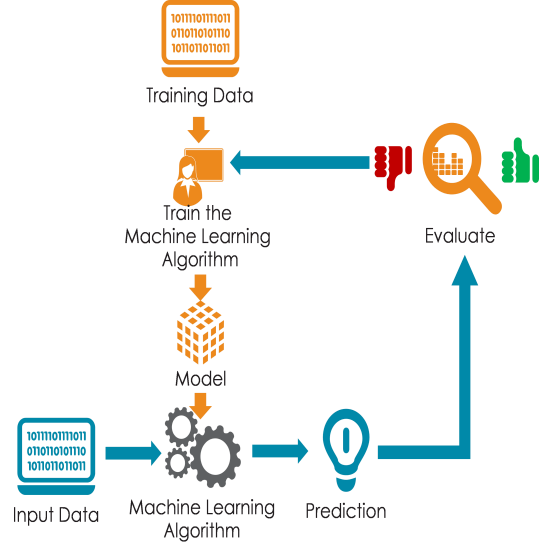
It is crucial to understand it in depth before you perform data analysis and run your data through an algorithm. You need to know the patterns in your data and determine which variables are important and which do not play a significant role in the output. Further, some variables may have correlations with other variables. You also need to recognize errors in your data.

All of this can be done with Exploratory Data Analysis. It helps you gather insights and make better sense of the data, and removes irregularities and unnecessary values from data.



**Model Training:**

Model training is the process of fitting a data into machine learning model from which model learns the patterns in data to predict the dependent variable. Model do it so by assigning a weight to each variable. After our model is trained, we test our model on test data to check how our model is performing.



We used six different types of model to train and test performances.

* Linear Regression
* Lasso Regression (Lasso)
* Ridge Regression (Ridge)
* Random Forest
* XGBoost

**Linear Regression :**

Regression models describe the relationship between variables by fitting a line to the observed data. Linear regression models use a straight line. Linear regression uses a linear approach to model the relationship between independent and dependent variables. In simple words its a best fit line drawn over the values of independent variables and dependent variable. In case of single variable, the formula is same as straight line equation having an intercept and slope.

y\_pred=β0+β1x

where

β0 and β1

are intercept and slope respectively.

In case of multiple features the formula translates into:

y\_pred=β0+β1x1+β2x2+β3x3+.....

where x\_1,x\_2,x\_3 are the features values and

β0,β1,β2.....

are weights assigned to each of the features. These become the parameters which the algorithm tries to learn using Gradient descent. Gradient descent is the process by which the algorithm tries to update the parameters using a loss function . Loss function is nothing but the difference between the actual values and predicted values(aka error or residuals). There are different types of loss function but this is the simplest one. Loss function summed over all observation gives the cost functions. The role of gradient descent is to update the parameters till the cost function is minimized i.e., a global minima is reached. It uses a hyperparameter 'alpha' that gives a weightage to the cost function and decides on how big the steps to take. Alpha is called as the learning rate. It is always necessary to keep an optimal value of alpha as high and low values of alpha might make the gradient descent overshoot or get stuck at a local minima. There are also some basic assumptions that must be fulfilled before implementing this algorithm

**Lasso Regression:**

This is a regularization technique used in feature selection using a Shrinkage method also referred to as the penalized regression method. Lasso is short for Least Absolute Shrinkage and Selection Operator, which is used both for regularization and model selection. If a model uses the L1 regularization technique, then it is called lasso regression.

In this shrinkage technique, the coefficients determined in the linear model from equation are shrunk towards the central point as the mean by introducing a penalization factor called the alpha α (or sometimes lamda) values.



Alpha (α) is the penalty term that denotes the amount of shrinkage (or constraint) that will be implemented in the equation. With alpha set to zero, you will find that this is the equivalent of the linear regression model and a larger value penalizes the optimization function. Therefore, lasso regression shrinks the coefficients and helps to reduce the model complexity and multi-collinearity.

Alpha (α) can be any real-valued number between zero and infinity; the larger the value, the more aggressive the penalization is.

**Ridge Regression :**

Ridge [regression](https://www.mygreatlearning.com/blog/what-is-regression/) is a model tuning method that is used to analyse any data that suffers from multicollinearity. This method performs L2 regularization. When the issue of multicollinearity occurs, least-squares are unbiased, and variances are large, this results in predicted values being far away from the actual values. Similar to the lasso regression, ridge regression puts a similar constraint on the coefficients by introducing a penalty factor. However, while lasso regression takes the magnitude of the coefficients, ridge regression takes the square.

**Random Forest :**

Random forest is a supervised learning algorithm which is used for both classification as well as regression. But however, it is mainly used for classification problems. As we know that a forest is made up of trees and more trees means more robust forest. Similarly, random forest algorithm creates decision trees on data samples and then gets the prediction from each of them and finally selects the best solution by means of voting. It is an ensemble method which is better than a single decision tree because it reduces the over-fitting by averaging the result. Random Forest is a supervised machine learning algorithm that is composed of individual decision trees. This type of model is called an ensemble model because an “ensemble” of independent models is used to compute a result.

**XGBoost**

XGBoost is short for “Extreme Gradient Boosting” which comes in association with various ensemble learning algorithms. It represents a flexible type of implementation where-in the concepts of decision trees (Gupta et al. [2020](https://link.springer.com/article/10.1007/s13198-021-01130-x#ref-CR9)) get wholly acknowledged. Moreover, it is found to be much faster when compared to more common algorithms like Adaboost.

Further, it has recently dominated the machine learning world and gotten much attention in Kaggle competitions. Execution speed and Performance (Qureshi et al. [2020](https://link.springer.com/article/10.1007/s13198-021-01130-x#ref-CR16)) are the two essential factors of using this algorithm in our work.

**Conclusion:**

During the time of our analysis, we did EDA on all the features of our datset. We first did a bit of data inspection, followed by various Univariate and Multivariate analysis. Those analysis have unveiled some interesting insight about the data, which we have listed in the corresponding places.

Next we have extracted some important featues from the raw dataset. The idea is to check the redundant featues and get rid of it (if required).Finally we implemented 5 machine learning algorithms: Linear Regression,Lasso, Rdge, Random Forest and XGBoost. We did hyperparameter tuning to improve our model performance. Now in order to avoid unnecesssary time complexity we have considered Feature Seleceted dataset for Random Forest, and XGBoost.

While compairing the results we have found that for XGBoost the results are very well, followed by Random Forest. However for the remianing three ML algorithms the results are very poor. The possible reason behind this is also partially explored by menas of correlation. In future one can elaborate the non linear dependency result using Mutual Information or by any other tool.

**Reference:**

<https://www.datacamp.com/tutorial/tutorial-lasso-ridge-regression>

<https://towardsdatascience.com/introduction-to-random-forest-algorithm-fed4b8c8e848>

<https://www.geeksforgeeks.org/ml-linear-regression>