**Summary on**

**Capstone Project - 2**

**Sachin Pandey**

**Data Science Trainee**

**Alma Better, Bangalore**

**Abstract:**

In today’s world Taxi rides are considered as the most integral part of the traffic system and the cities like New York which has a very busy traffic conditions, Taxi rides can give us insights 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 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. Thus in this research study, we used real-time data which customers would provide at the start of a ride, or while booking a ride to predict the duration and fare. This data includes pickup and drop-off point coordinates, the distance of the trip, start time, number of passengers, and a rate code belonging to the different classes of cabs available such that the rate applied is based on a regular or airport basis. Hereafter, we applied XGBoost and many more ML algorithms models to find out which one of them provides better accuracy and relationships between real-time variables.

**1.Problem Statement**

### We are provided with NYC\_Taxi\_data.csv dataset and our task is to 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.

**2. Introduction**

The objective of this project 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. Data Description 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 data was sampled and cleaned for the purposes of this project. Based on individual trip attributes, you should predict the duration of each trip in the test set. NYC Taxi Data.csv - the training set (contains 1458644 trip records).

**3. Steps Involved**

Discussion of NYC\_Taxi dataset will involve various steps such as:

* Loading the data into DataFrame.
* Data cleaning and performing EDA.
* Dividing the data in train and test split.
* Hyperparameter tuning.
* Model Training.
* Model Testing.
* Model adjustment on various Evaluation metrices.
* Questions that can be asked from the dataset.
* Conclusion.

**Data Preparation**

Data preparation is the process of cleaning and transforming raw data prior to processing and analysis. It is an important step prior to processing and often involves reformatting data, making corrections to data and the combining of data sets to enrich data.

**Gathering data:**

This step is about getting to know the data and understanding what has to be done before the data becomes useful in a particular context. This can be done by reading the CSV file and doing initial statistical analysis.

Though the dataset may seem to have the correct datatypes for each column, we need to check it. Inconsistent datatypes will create issues while dealing with problems.

**Cleanse and validate data:**

This step is crucial for removing faulty data and filling in gaps. Important tasks here includes :

* Removing extraneous data.
* Filling in missing values.
* Conforming data to a standardized pattern.

Dataset may contain duplicate values for particular application, so we have to delete the duplicate data in the dataset to make dataset more efficient to work on.

**Handling Null values:**

If our dataset contains null values which might tend to disturb our accuracy hence, we dropped them at the beginning of our project in order to get a better result.

**Detecting and removing the outlier:**

An outlier is an observation that lies an abnormal distance from other values in a random sample from a population. In a sense, this definition leaves it up to the analyst (or a consensus process) to decide what will be considered abnormal.

We can find outlier of the dataset using Boxplot and Scatterplot. Using Boxplot we have found the outlier of the dataset and then removed it from the dataframe.

# Exploratory Analysis and Visualization:

Exploratory data visualizations (EDVs) are the type of visualizations we assemble when we do not have a clue about what information lies within our dataset.

Like in this dataset we can find:

* The average time for a trip.
* We can find most busy pick up and drop off location.

And many more things can be visualised using the graphs and charts.

**Dividing the data in train and test split:**

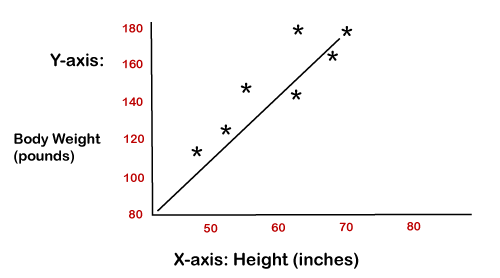
The dataset then will be divided into two parts i.e. Train and Test data. Training data will be used to train the model on the other hand Test data will be used in evaluation of a model.

**Algorithms used:**

1. **Linear Regression:**

Linear regression is one of the easiest and most popular Machine Learning algorithms. It is a statistical method that is used for predictive analysis. Linear regression makes predictions for continuous/real or numeric variables such as **sales, salary, age, product price,** etc.

Linear regression algorithm shows a linear relationship between a dependent (y) and one or more independent (y) variables, hence called as linear regression. Since linear regression shows the linear relationship, which means it finds how the value of the dependent variable is changing according to the value of the independent variable.The linear regression model provides a sloped straight line representing the relationship between the variables. The below diagram shows the linear regression for prediction of weight according to height:



1. **Decision Tree Regressor Classifier:**

Decision Tree is a supervised learning method used in data mining for classification and regression methods. It is a tree that helps us in decision-making purposes. The decision tree creates classification or regression models as a tree structure. It separates a data set into smaller subsets, and at the same time, the decision tree is steadily developed. The final tree is a tree with the decision nodes and leaf nodes. A decision node has at least two branches. The leaf nodes show a classification or decision. We can't accomplish more split on leaf nodes-The uppermost decision node in a tree that relates to the best predictor called the root node. Decision trees can deal with both categorical and numerical data.

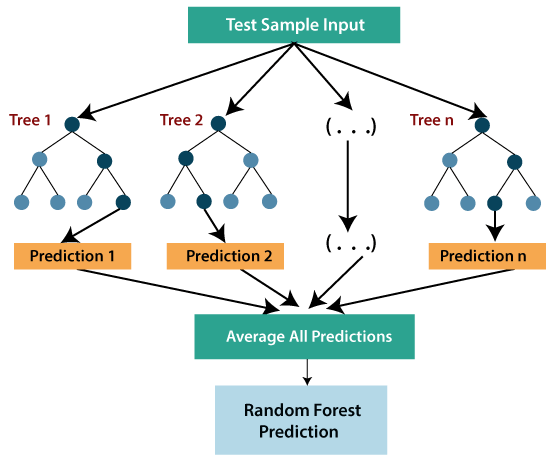
## **Key factors:**

### **Entropy:**

Entropy refers to a common way to measure impurity. In the decision tree, it measures the randomness or impurity in data sets.

### **Information Gain:**

Information Gain refers to the decline in entropy after the dataset is split. It is also called **Entropy Reduction**. Building a decision tree is all about discovering attributes that return the highest data gain.



1. **XGBoost:**

To understand XGBoost we have to know gradient boosting beforehand.

* **Gradient Boosting-**

Gradient boosted trees consider the special case where the simple model is a decision tree

****

In this case, there are going to be 2 kinds of parameters P: the weights at each leaf, w, and the number of leaves T in each tree (so that in the above example, T=3 and w= [2, 0.1, -1]).

When building a decision tree, a challenge is to decide how to split a current leaf. For instance, in the above image, how could I add another layer to the (age > 15) leaf? A ‘greedy’ way to do this is to consider every possible split on the remaining features (so, gender and occupation), and calculate the new loss for each split; you could then pick the tree which most reduces your loss.

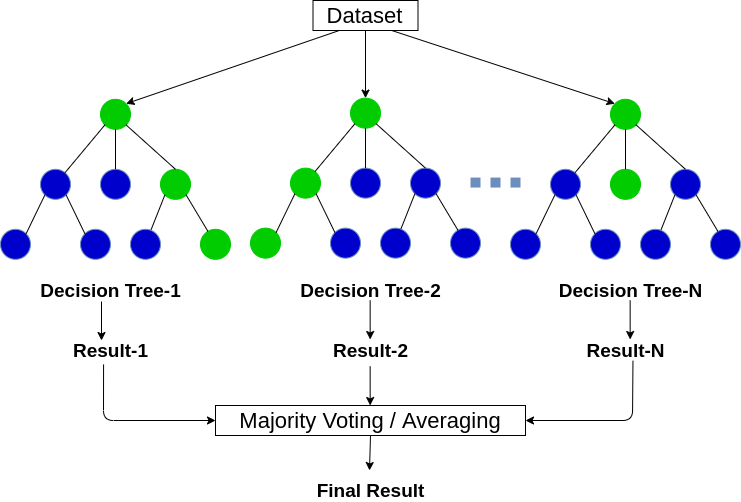
**XGBoost** is one of the fastest implementations of gradient boosting. trees. It does this by tackling one of the major inefficiencies of gradient boosted trees: considering the potential loss for all possible splits to create a new branch (especially if you consider the case where there are thousands of features, and therefore thousands of possible splits). XGBoost tackles this inefficiency by looking at the distribution of features across all data points in a leaf and using this information to reduce the search space of possible feature splits.

1. **Random Forest Regressor:**

Random forest is a [supervised learning algorithm](https://builtin.com/data-science/supervised-learning-python). The "forest" it builds, is an ensemble of decision trees, usually trained with the “bagging” method. The general idea of the [bagging method](https://builtin.com/data-science/tour-top-10-algorithms-machine-learning-newbies) is that a combination of learning models increases the overall result.

Random forest has nearly the same hyperparameters as a decision tree or a bagging classifier. Fortunately, there's no need to combine a decision tree with a bagging classifier because you can easily use the classifier-class of random forest. With random forest, you can also deal with regression tasks by using the algorithm's regressor.Random forest adds additional randomness to the model, while growing the trees. Instead of searching for the most important feature while splitting a node, it searches for the best feature among a random subset of features. This results in a wide diversity that generally results in a better model.

Therefore, in random forest, only a random subset of the features is taken into consideration by the algorithm for splitting a node. You can even make trees more random by additionally using random thresholds for each feature rather than searching for the best possible thresholds (like a normal decision tree does).

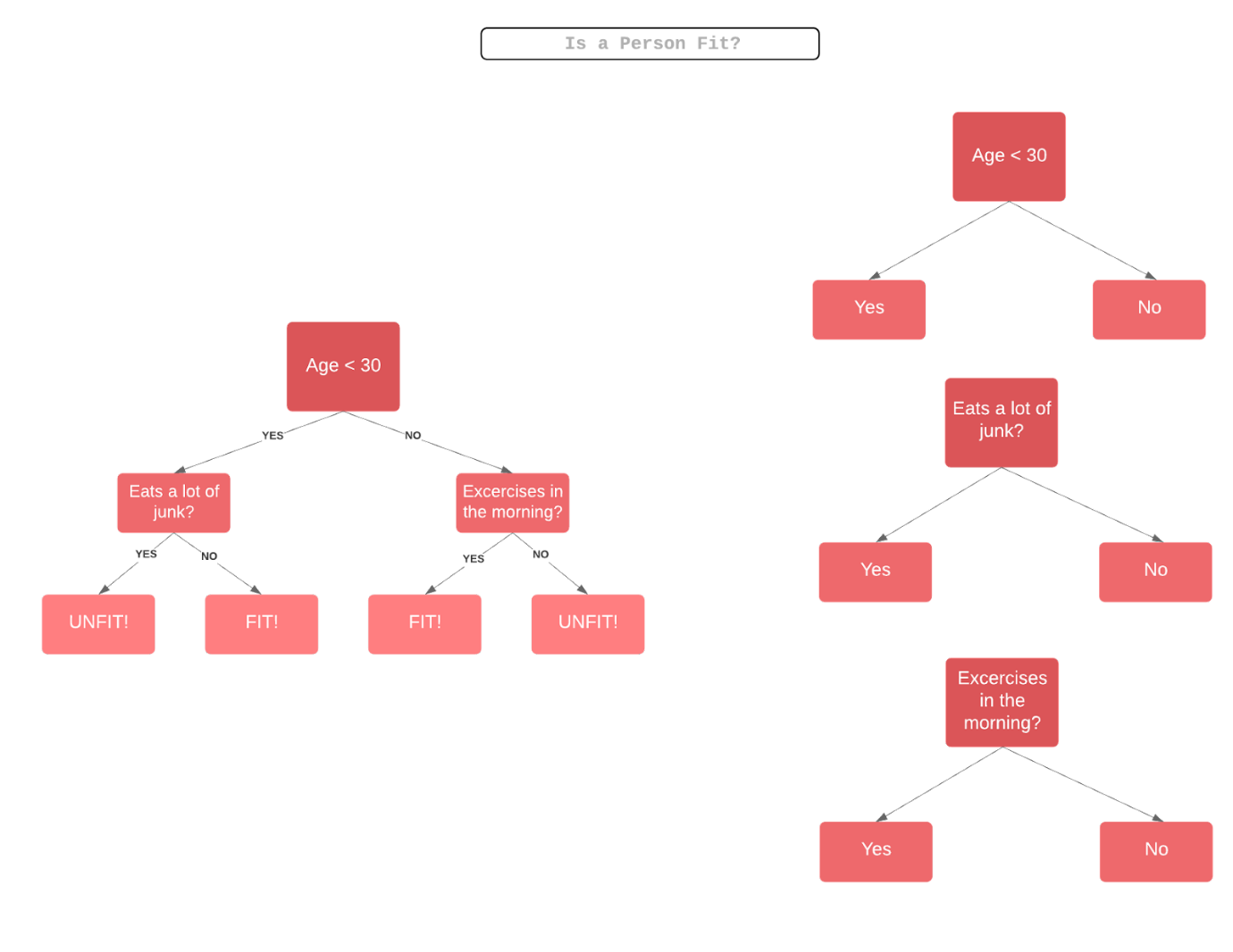


1. **ADA Boosting :**

***AdaBoost is short for Adaptive Boosting***. Basically, Ada Boosting was the first really successful boosting algorithm developed for binary classification. Also, it is the best starting point for understanding boosting. Moreover, modern boosting methods build on AdaBoost, most notably stochastic **gradient boosting machines**.

Generally, AdaBoost is used with short decision trees. Further, the first tree is created, the performance of the tree on each training instance is used. Also, we use it to weight how much attention the next tree. Thus, it is created should pay attention to each training instance. Hence, training data that is hard to predict is given more weight. Although, whereas easy to predict instances are given less weight.

We can see observation from the diagram below:



**Model Adjustment on various Evaluation matrices:**

**HyperParameter 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.

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.

1. **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

# Bayesian Optimization

# Bayesian Hyperparameter optimization is a very efficient and interesting way to find good hyperparameters. In this approach, in naive interpretation way is to use a support model to find the best hyperparameters. A hyperparameter optimization process based on a probabilistic model, often Gaussian Process, will be used to find datafrom data observed in the later distribution of the performance of the given models or set of tested hyperparameters.

**Model Adjustment on various Evaluation matrices:**

**Conclusion:**

On performing various operations on the Dataset the conclusion is as follows:

* The Trip Duration varies a lot ranging from a few seconds to more than 20 hours also some are going from 528 Hours to 972 Hours.
* It is observed that Vendor 2 taxi service provider is the most frequently used by the people in New York.
* Trip duration is generally longer for trips whose flag was not stored.
* There were few trips with Zero Passengers and few trips with 7,8 and 9 passengers and most number of trips were done by single or double passengers.
* Trip duration is the maximum around 3 pm and the lowest around 6 am.
* Trip duration is the longest on Thursdays closely followed by Fridays.
* From February, we can see trip duration rising every month and also significant drop in the Taxi trip count as month end approaches.
* For shorter trips (<5 hours), the pickup and dropoff latitude is more or less evenly distributed between 30 ° and 40 °.
* For longer trips(>5 hours ) the pickup and dropoff latitude is all concentrated between 40 ° and 42 ° degrees.
* The long duration trips(> 5 hours) are mostly concentrated with their pickup region near (40 °,75 °) to (42°,75°).