**Symbiosis Skills and Professional University**

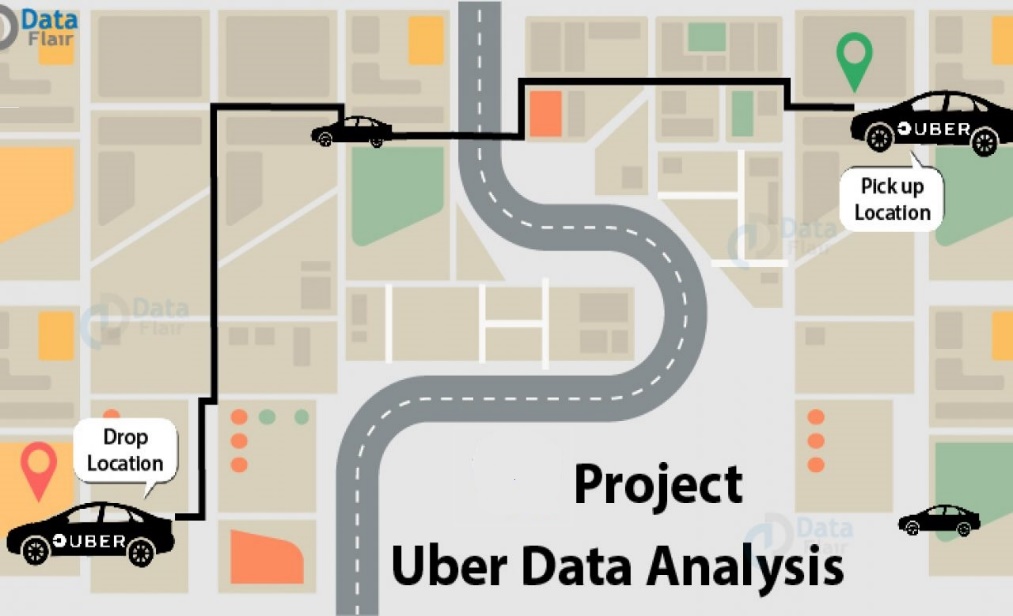
## Kiwale, Pune

**-------------------------------------------------------------------------**

# PROJECT REPORT

**On**

# “Uber Data Analysis”

****

**Submitted by**

1. **Pratik Kale**
2. **Tasarif Pathan**

**DA- Batch FST02:DA03**

# Under The Guidance of :-

**Trainer's Name:**

**1) Ritviz Singh**

# Shiv Patel

**STUDENT DECLARATION AND ATTESTATION BY TRAINER**

I’m Pratik Kale & Tasarif Pathan, A student at **Symbiosis Skills and Professional University,** hereby declare and attest that we understand and agree to comply with the rules and regulation pertaining to the Face Recognition and Attendance System implemented by the intuition.

**Name of Students**: - 1) Pratik Kale

* 1. Tasarif Pathan

**Course:**

**Data Associate**

# CERTIFICATE

This is to certify that the report entitled, **“Uber Data Analysis”**, submitted by “**1) Pratrik Kale, 2) Tasarif Pathan**” to Symbiosis Skills and Professional University, Pune, Maharashtra, India, is a record of bonafide Project work carried out by him under my supervision and guidance and is worthy of consideration for the completion of certificate course in ‘Data Associate”.

Signature of Trainer Name of Trainer

Date: / / 2024

|  |  |  |
| --- | --- | --- |
|  |  |  |
| Supervisor |  | Supervisor |

Date:

# ACKNOWLEDGEMENTS

We extend our sincere appreciation to all those who contributed significantly to the successful development of the “Uber Data Analysis” by Team Pratik Kale & Tasarif Pathan under the guidance of **Ritviz Singh** and **Shiv Patel** at **Symbiosis Centre of Distance Learning.**

Our team members have exhibited unwavering dedication, bringing a wealth of diverse skills to the project, resulting in the creation of an innovative system that seamlessly integrates technology with practical application.

Heartfelt thanks are extended to our esteemed project guides **Ritviz Singh** and **Shiv Patel** for their invaluable mentorship and technical expertise. Their guidance has not only shaped the project's technical aspects but has also ensured its alignment with the high academic standards of “**Symbiosis Centre of Distance Learning”.**

Team -

* + 1. Pratik Kale
    2. Tasarif Pathan

**Pratik Kale**

**Tasarif Pathan**

**Abstract**

*Uber was founded just eleven years ago, and it was already one of the fastest-growing companies in the world. In Boston, UberX claims to charge 30% less than taxis – a great way to get customers' attention. Nowadays, we see applications of Machine Learning and Artificial Intelligence in almost all the domains so we try to use the same for Uber cabs price prediction. In this project, we did experiment with a real-world dataset and explore how machine learning algorithms could be used to find the patterns in data. We mainly discuss about the price prediction of different Uber cabs that is generated by the machine learning algorithm. Our problem belongs to the regression supervised learning category. We use different machine learning algorithms, for example, Linear Regression, Decision Tree, Random Forest Regressor, and Gradient Boosting Regressor but finally, choose the one that proves best for the price prediction. We must choose the algorithm which improves the accuracy and reduces overfitting. We got many experiences while doing the data preparation of Uber Dataset of Boston of the year 2018. It was also very interesting to know how different factors affect the* pricing of Uber cabs.

**CONTENTS**

|  |
| --- |
| Declaration ii |
| Certificate iii |
| Acknowledge iv |
| Abstract v |
| List of figures vi |
| List of Tables vii |
|  |
| **CHAPTER 1 Introduction 1** |
| 1.1 Overview and Motivation 1 |
| 1.2 Objective 1 |
| 1.3 Issues and Challenges 2 |
| 1.4 Contribution 3 |
| 1.5 Organization of the Project 3 |
|  |
| **CHAPTER 2 Literature Review 4** |
|  |
| **CHAPTER 3 Machine Learning 6** |
| 3.1 What is Machine Learning? 6 |
| 3.2 Types of Machine Learning 6 |
| 3.2.1 Supervised Machine Learning 6 |
| 3.2.2 Unsupervised Machine Learning 7 |
| 3.2.3 Reinforcement Machine Learning 7 |
|  |
| **CHAPTER 4 Proposed work and Implementation 8** |
| 4.1 Data Preparation  8 |
| 4.2 Data Visualization 9 |
| 4.3 Feature Engineering 11 |
| 4.3.1 Label Encoding 12 |
| 4.3.2 Filling Nan Values 12 |
| 4.3.3 Recursive Feature Elimination  13 |
| 4.3.4 Drop Useless Columns 15 |
| 4.3.5 Binning 15 |
| 4.4 Modelling 16 |
| 4.4.1 Linear Regression 16 |
| 4.4.2 Decision Tree 16 |
| 4.4.3 Random Forest 17 |
| 4.4.4 Gradient Boosting 17 |
| 4.4.5 K-fold Cross Validation 18 |
| 4.5 Testing19 |
| 4.5.1 Mean Absolute Error 20 |
| 4.5.2 Mean Squared Error 20 |
| 4.5.3 Root Mean Squared Error 20 |
| 4.6 Price Prediction Function 22 |
|  |
| **CHAPTER 5 Conclusion 24** |
|  |
| **CHAPTER 6 References** **25** |

**List of Figures**

3.1 Types of ML Courtesy of Packt-cdn.com 6

4.1 Data Head 8

4.2 Strip-plot between Name and Price 9

4.3 Strip-plot between Icon and Price 9

4.4 Bar-Chart of Month 10

4.5 Bar-Chart of Icon 10

4.6 Bar-Chart of UV-Index 11

4.7 Feature Engineering Courtesy of Digitalag 12

4.8 Bar Chart of Price 13

4.9 Recursive Feature Elimination Courtesy of Researchgate 14

4.10 Final Dataset after Feature Engineering 15

4.11 Cross-Validation Courtesy of Wikimedia 19

4.12 Scatter Plot for Linear Regression 20

4.13 Dist Plot for Linear Regression 21

4.14 Scatter Plot for Random Forest 22

4.15 Dist Plot for Random Forest 22

**List of Tables**

4.1 RFE Accuracy Table 14

4.2 Model Accuracy Table 18

4.3 Error table for Linear Regression 21

4.4 Error table for Random Forest 22

**INTRODUCTION**

**1.1 Motivation and Overview**

Uber Technologies, Inc., commonly known as Uber, was a ride-sharing company and offers [vehicles for hire](https://en.wikipedia.org/wiki/Vehicles_for_hire), [food delivery](https://en.wikipedia.org/wiki/Food_delivery) ([Uber Eats](https://en.wikipedia.org/wiki/Uber_Eats)), [package delivery](https://en.wikipedia.org/wiki/Package_delivery), [couriers](https://en.wikipedia.org/wiki/Courier), [freight transportation](https://en.wikipedia.org/wiki/Freight_transport), and, through a partnership with [Lime](https://en.wikipedia.org/wiki/Lime_(transportation_company)), [electric bicycle](https://en.wikipedia.org/wiki/Electric_bicycle) and [motorized scooter](https://en.wikipedia.org/wiki/Motorized_scooter) rental. It was founded in 2009 by Travis Kalanick and Garrett Camp, a successful technology entrepreneur. After selling his first startup to eBay, Camp decided to create a new startup to [address San Francisco’s serious taxi problem](http://www.tc.umn.edu/~ssen/IDSC6050/Case15/Group15_index.html).

Together, the pair developed the Uber app to help connect riders and local drivers. The service was [initially launched in San Francisco](http://blog.uber.com/tag/history/) and eventually expanded to Chicago in April 2012, proving to be a highly convenient great alternative to taxis and poorly-funded public transportation systems. Over time, Uber has since expanded into smaller communities and has become popular throughout the world. In December 2013, [USA Today](https://en.wikipedia.org/wiki/USA_Today) named Uber its tech company of the year.

In Supervised learning, we have a training set and a test set. The training and test set consists of a set of examples consisting of input and output vectors, and the goal of the supervised learning algorithm is to infer a function that maps the input vector to the output vector with minimal error. We applied machine learning algorithms to make a prediction of Price in the Uber Dataset of Boston. Several features will be selected from 55 columns. Predictive analysis is a procedure that incorporates the use of computational methods to determine important and useful patterns in large data.

**1.2 Objective**

The objective is to first explore hidden or previously unknown information by applying exploratory data analytics on the dataset and to know the effect of each field on price with every other field of the dataset. Then we apply different machine learning models to complete the analysis. After this, the results of applied machine learning models were compared and analyzed on the basis of accuracy, and then the best performing model was suggested for further predictions of the label ‘Price’.

**1.3 Issues and Challenges**

1. **Overfitting in Regression Problem:-** Overfitting a model is a condition where a statistical model begins to describe the random error in the data rather than the relationships between variables. This problem occurs when the model is too complex. In [regression analysis](https://statisticsbyjim.com/glossary/regression-analysis/), overfitting can produce misleading [R-squared](https://statisticsbyjim.com/glossary/r-squared/) values. When this occurs, the [regression coefficients](https://statisticsbyjim.com/glossary/regression-coefficient/) represent the noise rather than genuine relationships. However, there is another problem. Each [sample](https://statisticsbyjim.com/glossary/sample/) has its unique quirks. Consequently, a regression model that becomes tailor-made to fit the random quirks of one sample is unlikely to fit the random quirks of another sample. Thus, overfitting a regression model reduces its generalizability outside the original dataset.
2. **Strip-plot and Scatter diagram:-** One problem with strip plots is how to display multiple points with the same value. If it uses the jitter option, a small amount of random noise is added to the vertical coordinate and if it goes with the stack option it increments the repeated values to the vertical coordinate which gives the strip plot a histogram-like appearance.

Scatter plot does not show the relationship for more than two variables. Also, it is unable to give the exact extent of correlation**.**

1. **Label Encoding:-**  It assigns a unique number(starting from 0) to each class of data which may lead to the generation of priority issues in the training of data sets. A label with high value may be considered to have high priority than a label having lower value but actually, there is no such priority relation between the attributes of the same classes.
2. **Computational Time:-** Algorithms like support vector machine(SVM) don’t scale well for larger datasets especially when the number of features are more than the number of samples. Also, it sometimes runs endlessly and never completes execution.

**1.4 Contribution:**

Each team member is responsible and has willing participation in the group. The work within the group is equally done by each team member. First, the project work is divided like one has to be done the exploratory data analysis part, two members work on feature engineering, and the rest work of modeling and testing was equally divided among all four members. And the second part i.e. written work is done in pairs like two members work on report and the other two works on presentation.

**1.5 Organization of the Project Report:**

The first section of this paper presents the concept of exploratory data analysis which told general information about the dataset. Then from the next section feature engineering part was started in which we plot many charts and deal with columns to extract the features helpful for our predictions in many ways. In the last part, we did modeling and testing in which we apply different models to check the accuracy and for further price prediction.

**LITERATURE REVIEW**

As we are researching on Uber and found what different researchers had done. So, they do research on the Uber dataset but on different factors. The rise of Uber as the global alternative has attracted a lot of interest recently. Our work on Uber's predicting pricing strategy is still relatively new. In this research, "Uber Data Analysis" we aim to shed light on Uber's Price. We are predicting the price of different types of Uber based on different factors. Some of the other factors that we found in other researches are:

Abel Brodeurand & Kerry Nield (2018) analyses the effect of rain on Uber rides in New York City after entering Uber rides in the market in May 2011, passengers and fare will decrease in all other rides such as taxi-ride. Also, dynamic pricing makes Uber drivers compete for rides when demand suddenly increases, i.e., during rainy hours. On increasing rain, the Uber rides are also increasing by 22% while the number of taxi rides per hour increases by only 5%. Taxis do not respond differently to increased demand in rainy hours than non-rainy hours since the entrance of Uber.

Surge Pricing is an algorithmic technique that Uber uses when there is a demand-supply imbalance. It occurs when there is a downward shift in both the rider's demand and driver's availability. During such a time of the rise in demand for rides, fares tend to usually high. Surge pricing is essential in a way that it helps in matching the driver's efforts with the demand from consumers. (Junfeng Jiao, 2018) did an investigation of Uber on surge multiplier in Austin, Texas founds that during times of high usage, Uber will enhance their prices to reflect this demand via a surge multiplier. According to communications released by (Uber, 2015), this pricing is meant to attract more drivers into service at certain times, while also reducing demand on the part of riders. (Chen & Sheldon, 2016) While some research is mixed, in general, surge pricing does appear to control both supply and demand while keeping wait time consistently under 5 minutes.

Anna Baj-Rogowska (2017) analyses the user's feedback from social networking sites such as Facebook in the period between July 2016 and July 2017. Uber is one of the most dynamically growing companies representing the so-called sharing economy. It is also a basis for the ongoing evaluation of brand perception by the community and can be helpful in developing such a marketing strategy and activities, which will effectively improve the current rating and reduce possible losses. So, it can be concluded that feedback should be an important instrument to improve the market performance of Uber today.

Anderson (2014) concluded from surveying San Francisco drivers that driver behavior and characteristics are likely determining the overall vehicle miles traveled (VMT). Full-time drivers are likely to increase overall VMT, while occasional drivers are more likely to reduce overall VMT. We also analyze the research on the driving behavior of the driver while driving on the road. The driver has been categorized based on ages and genders that focus on their driving reactions from how they braking, speeding, and steer handling. For gender differences, male driver practice higher-risk of driving while female drivers are lacks of pre-caution over obstacles and dangerous spot. More or less, adult drivers which regularly drive vehicles can manage the vehicle quite well as compared with young drivers with less experience. In conclusion, the driver's driving behavior is related to their age, gender, and driving experiences.

Some papers take a comparison between the iconic yellow taxi and its modern competitor, Uber. (Vsevolod Salnikov, Renaud Lambiotte, Anastasios Noulas, and Cecilia Mascolo, 2014) identify situations when UberX, the cheapest version of the Uber taxi service, tends to be more expensive than yellow taxis for the same journey. Our observations show that it might be financially advantageous on average for travelers to choose either Yellow Cabs or Uber depending on the duration of their journey. However, the specific journey they are willing to take matters.

**MACHINE LEARNING**

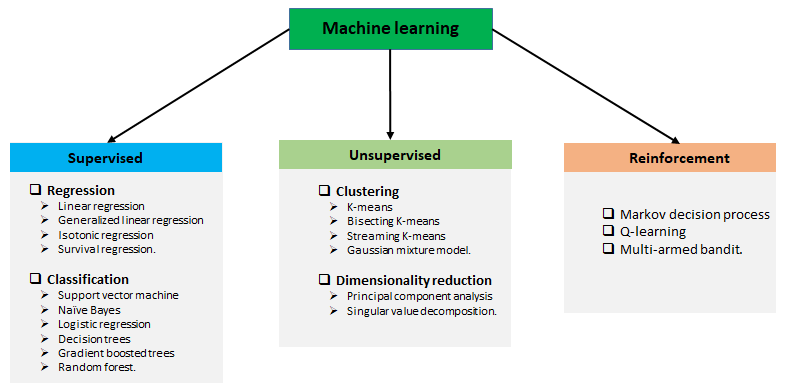
**3.1 What is Machine Learning?**

Machinelearning (ML) is the [scientific study](https://en.wikipedia.org/wiki/Branches_of_science) of [algorithms](https://en.wikipedia.org/wiki/Algorithm) and [statistical models](https://en.wikipedia.org/wiki/Statistical_model) that [computer systems](https://en.wikipedia.org/wiki/Computer_systems) use to perform a specific task without using explicit instructions, relying on patterns and [inference](https://en.wikipedia.org/wiki/Inference) instead. It is seen as a subset of [artificial intelligence](https://en.wikipedia.org/wiki/Artificial_intelligence).

Machine learning algorithms are used in a wide variety of applications, such as [email filtering](https://en.wikipedia.org/wiki/Email_filtering) and [computer vision](https://en.wikipedia.org/wiki/Computer_vision), where it is difficult or infeasible to develop a conventional algorithm for effectively performing the task.

**3.2 Types of Learning Algorithms**

The types of machine learning algorithms differ in their approach, the type of data they input, and the type of task or problem that they are intended to solve.

**Fig. 3.1 Types of ML Courtesy of Packt-cdn.com**

**3.2.1 Supervised learning**

Supervised learning is when the model is getting trained on a labelled dataset. The **labelled** dataset is one that has both input and output parameters. Supervised learning algorithms include [classification](https://en.wikipedia.org/wiki/Statistical_classification) and [regression](https://en.wikipedia.org/wiki/Regression_analysis). Classification algorithms are used when the outputs are restricted to a limited set of values, and regression algorithms are used when the outputs may have any numerical value within a range.

**3.2.2 Unsupervised learning**

Unsupervised learning algorithms take a set of data that contains only inputs, and find structure in the data, like grouping or clustering of data points. The algorithms, therefore, learn from test data that has not been labeled, classified, or categorized.

**3.2.3 Reinforcement learning**

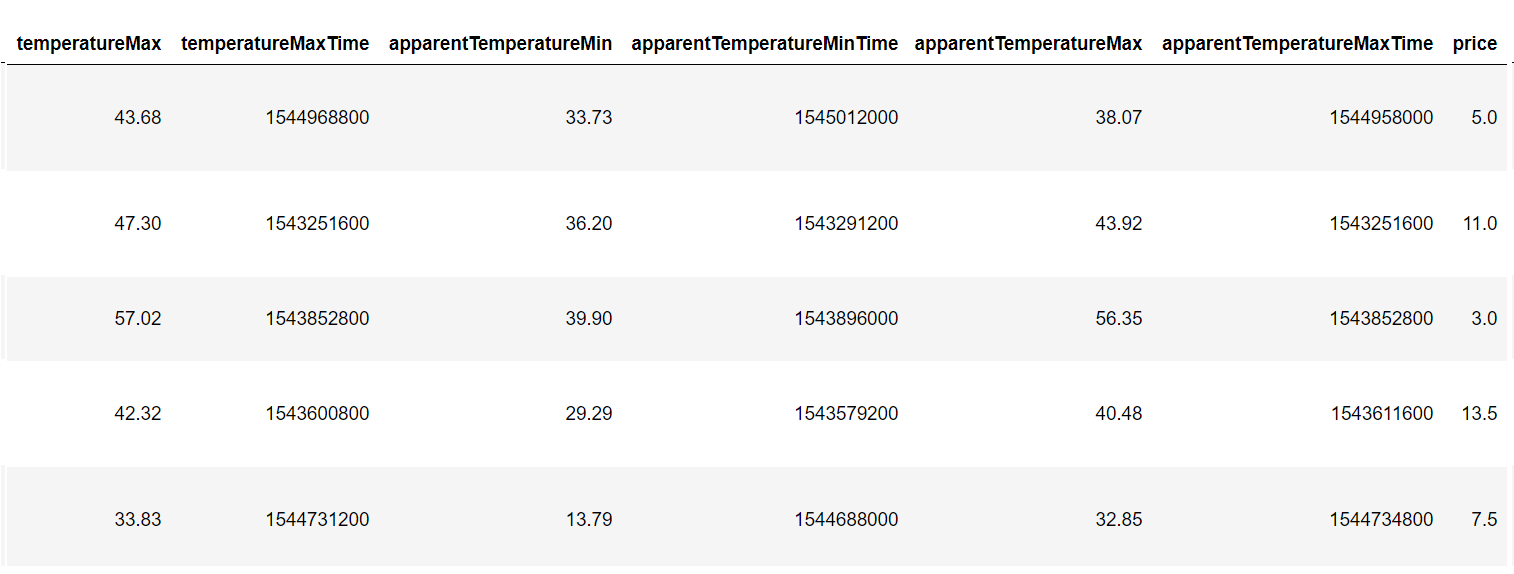
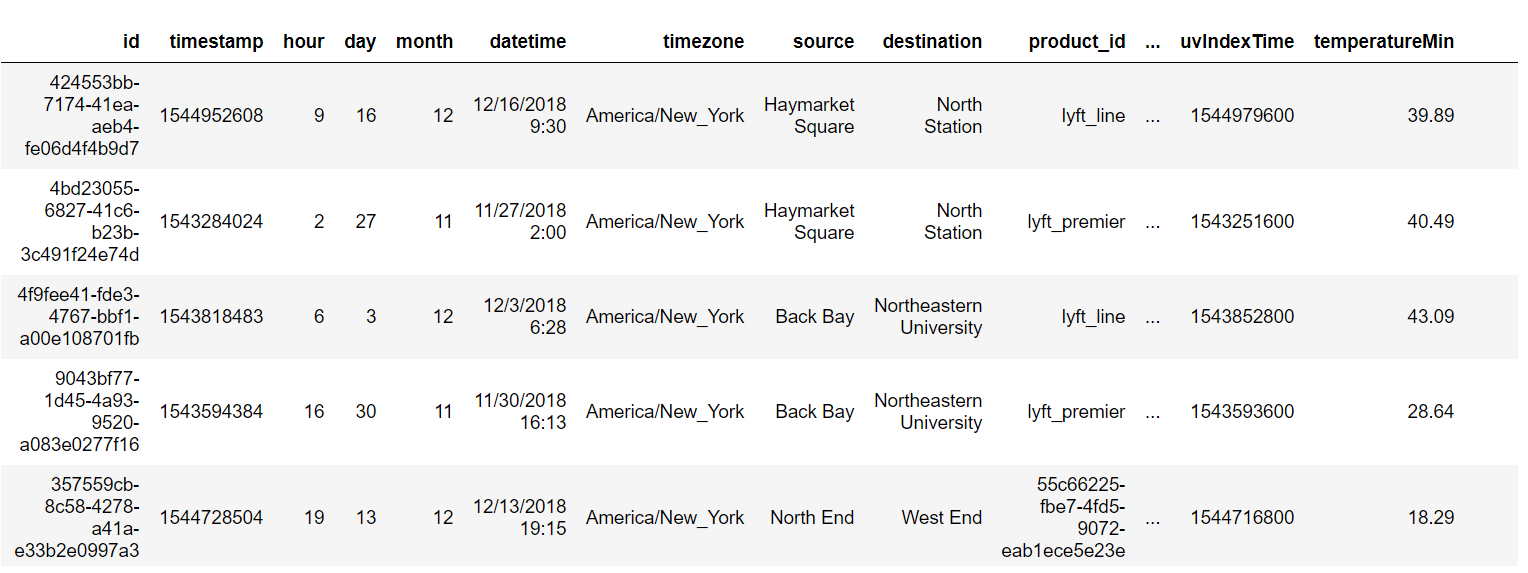
Reinforcement learning is an area of machine learning concerned with how [software agents](https://en.wikipedia.org/wiki/Software_agent) ought to take [actions](https://en.wikipedia.org/wiki/Action_selection) in an environment to maximize some notion of cumulative reward. In this learning, system is provided feedback in terms of rewards and punishments as it navigates its problem space.

**PROPOSED WORK & IMPLEMENTATION**

**4.1 Data Preparation**

The data we used for our project was provided on the www.kaggle.com website. The original dataset contains 693071 rows and 57 columns which contain the data of both Uber and Lyft. But for our analysis, we just need the Uber data so we filter out the data according to our purpose and got a new dataset that has 322844 rows and 56 columns. The dataset has many fields that describe us about the time, geographic location, and climatic conditions when the different Uber cabs opted.

Data has 3 types of data-types which were as follows:- integer, float, and object. The dataset is not complete which means we have also null values in a column named price of around 55095.

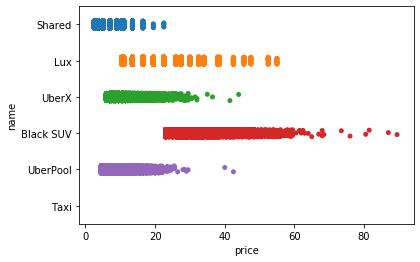


**Fig. 4.1 Data Head**

**4.2 Data Visualization**

Data visualization is a graphical representation of information and data. By using visual elements like charts, graphs, and maps, data visualization tools provide an accessible way to see and understand trends, outliers, and patterns in data.

For the same purpose, we have to import matplotlib and seaborn library and plot different types of charts like strip plot, scatter plot, and bar chart.



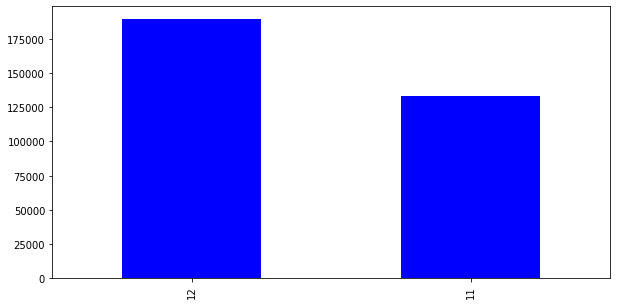
**Fig. 4.2 Strip-plot between Name and Price**

From the above chart, it was clear that Shared trip was cheapest among all and BlackSuv was most expensive. UberX and UberPool have almost same prices and Lux has moderate price. There is no graph for taxi which reveals that in the dataset there were no values of taxi was given.



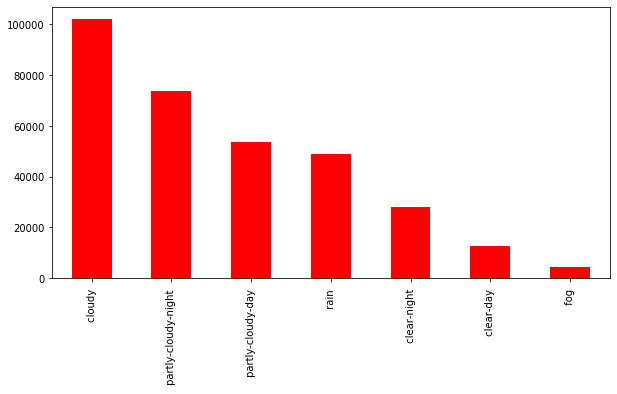
**Fig. 4.3** **Strip-plot between Icon and Price**

From the above chart, it was clear there were some outliers in cloudy type weather, some data has an anonymously high price above 80 while the other was below 60. In this plot, we analyze that in cloudy-day weather price was the highest while in foggy weather price was minimum.



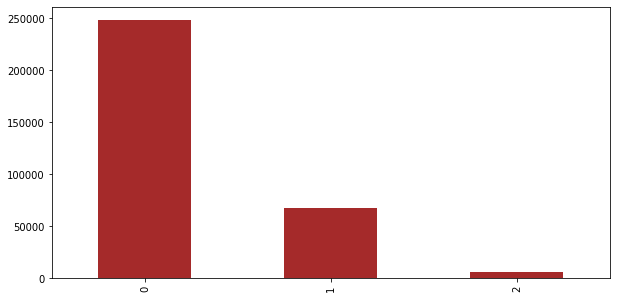
**Fig. 4.4** **Bar-Chart of Month**

From the above bar chart, it was clear that the data consists of all the information of only two months that is November and December.



**Fig. 4.5** **Bar-Chart of Icon**

The above bar chart represents the value count of the icon column and from the graph, it was clear that cloudy weather has the most data due to which we can say that may be in cloudy weather cab also opted most.



**Fig.4.6** **Bar-Chart of UV-Index**

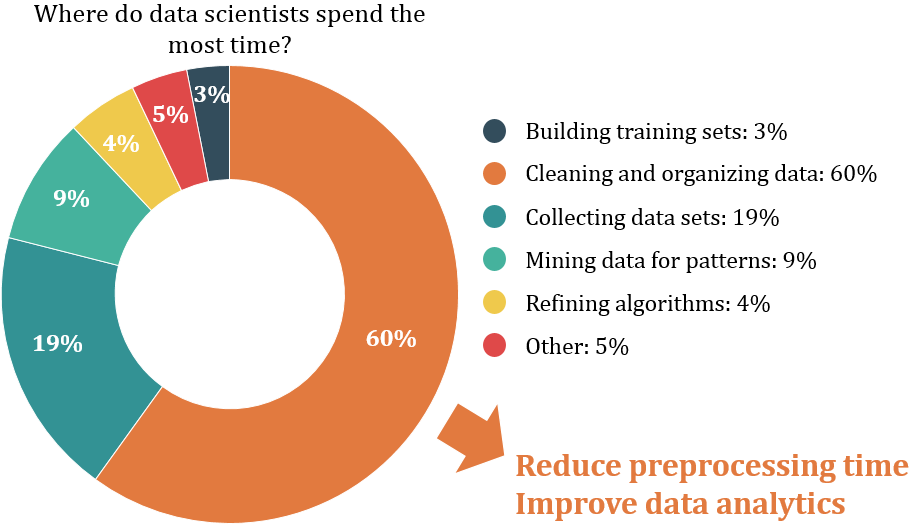
The above bar chart represents the value count of the UV-index column and from the graph, it was clear that when UV-index is 0, the dataset has the most data due to which we can say that when there is less UV-index cab was opted most.

**4.3 Feature Engineering**

Feature engineering is the most important part of the data analytics process. It deals with, selecting the features that are used in training and making predictions. All machine learning algorithms use some input data to create outputs. This input data comprise features, which are usually in the form of structured columns. Algorithms require features with some specific characteristics to work properly. A bad feature selection may lead to a less accurate or poor predictive model. To filters out all the unused or redundant features, the need for feature engineering arises. It has mainly two goals:

* Preparing the proper input dataset, compatible with the machine learning algorithm requirements.
* Improving the performance of machine learning models.

**“According to a survey in Forbes, data scientists spend 80% of their time on data preparation.”**



**Fig. 4.7 Feature Engineering Courtesy of Digitalag**

**4.3.1 Label Encoding**

Our data is a combination of both **Categorical variables**and**Continuous variables,** most of the machine learning algorithms will not understand, or not be able to deal with categorical variables. Meaning, machine learning algorithms will perform better when the**data is represented as a number** instead of categorical. Hence label encoding comes into existence. Label Encoding refers to converting the categorical values into the numeric form to make it machine-readable. So we did label encoding as well as class mapping to get to know which categorical value is encoded into which numeric value.

**4.3.2 Filling NAN Values**

To check missing values in Pandas DataFrame, we use a function isnull(). So we find that the price column in our dataset consists of 55095 Nan values. Now to fill these null values we use the fillna() function. We fill missing values with the median of the remaining dataset values and convert them to integer because price cannot be given in float. Now for the visualization purpose, we make a bar chart of the value count of price.



**Fig. 4.8 Bar Chart of Price**

**4.3.3 RFE (Recursive Feature Elimination)**

[Feature selection](https://heartbeat.fritz.ai/search?q=feature%20selection) is an important task for any machine learning application. This is especially crucial when the data has many features. The optimal number of features also leads to improved model accuracy. So we use RFE for feature selection in our data.

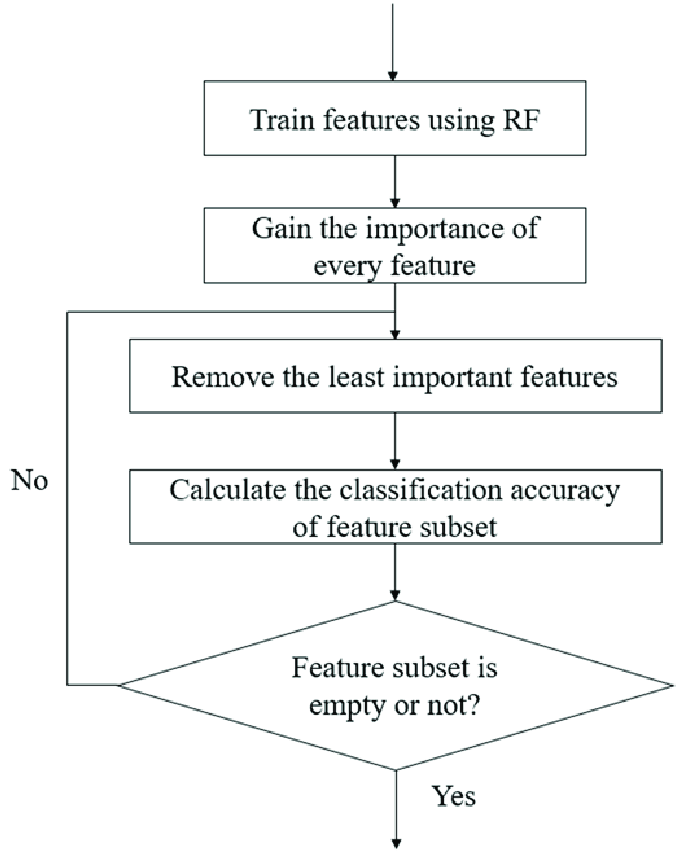
RFE is a wrapper-type feature selection algorithm. This means that a different machine learning algorithm is wrapped by RFE, and used to help select features. This is in contrast to filter-based feature selections that score each feature and select those features with the largest score.

There are two important configuration options when using RFE:

* The choice in the number of features to select (k value)
* The choice of the algorithm used to choose features.

RFE works by searching for a subset of features by starting with all features in the training dataset and successfully removing features until the desired number remains. This is achieved by fitting the given machine learning algorithm used in the core of the model, ranking features by importance, discarding the least important features, and re-fitting the model. This process is repeated until a specified number of features remain. Hence RFE technique is effective at selecting those features (columns) in a training dataset that are most relevant in predicting the target variable.

We are implementing recursive feature elimination through scikit-learn via sklearn.feature\_selection.RFE class.



**Fig. 4.9 Recursive Feature Elimination Courtesy of Researchgate**

On applying RFE in our dataset with Linear Regression model first we divide our dataset into dependent (features) and independent (target) variables then split it into train and test after that we found different accuracies in different number of features (k value) as follows:

**Table 4.1: RFE Accuracy Table**

|  |  |  |
| --- | --- | --- |
| **Serial No.​** | **No. of Feature (K)​** | **Accuracy​** |
| 1​ | 56​ | 0.8054834220​ |
| 2​ | 40​ | 0.8050662132​ |
| 3​ | 25​ | 0.8055355151​ |
| 4​ | 15​ | 0.8050457819​  ​ |

**Table 4.1 Continued**

From the above table, it was clear that 25 features have the highest accuracy as compared to all other k values which mean these 25 features are the best features given by RFE. So, we only consider these 25 features for further working and rest we eliminate. Now our dataset reduces from 56 features to 25 features.

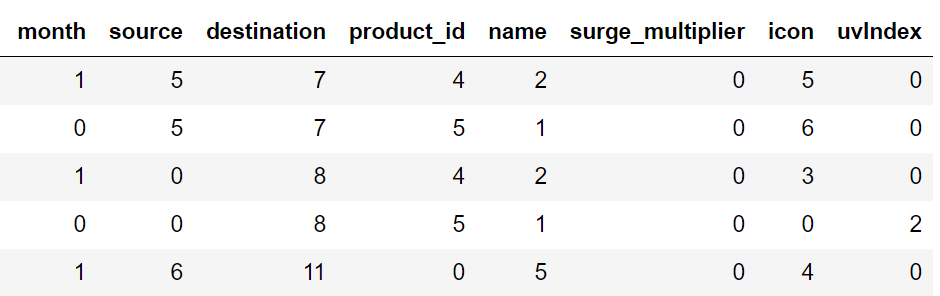
**4.3.4 Drop Useless Columns**

After applying RFE we get our 25 best features but still, there are many features which do not affect the price directly so we drop those features according to it. And eight features remained in our dataset. We use a method called drop() that removes rows or columns according to specific column names and corresponding axis.

**4.3.5 Binning**

Many times we use a method called data smoothing to make the data proper. During this process, we define a range also called bin and any data value within the range is made to fit into the bin. This is called the binning. Binning is used to smoothing the data or to handle noisy data.

So after dropping useless features, some features are not in range so to make all the features in the same range we apply binning and get our final dataset which is further used for modeling.



**Fig. 4.10 Final Dataset after Feature Engineering**

**4.4 Modeling**

The process of modeling means training a machine-learning algorithm to predict the labels from the features, tuning it for the business needs, and validating it on holdout data. When you train an algorithm with data it will become a model. One important aspect of all machine learning models is to determine their accuracy. Now to determine their accuracy, one can train the model using the given dataset and then predict the response values for the same dataset using that model and hence, find the accuracy of the model.

In this project, we use Scikit-Learn to rapidly implement a few models such as Linear Regression, Decision Tree, Random Forest, and Gradient Boosting.

**4.4.1. Linear Regression**

Linear Regression is a supervised machine learning algorithm where the predicted output is continuous in the range such as salary, age, price, etc. It is a statistical approach that models the relationship between input features and output. The input features are called the **independent variables**, and the output is called **a dependent variable**. Our goal here is to predict the value of the output based on the input features by multiplying it with its optimal coefficients. The name linear regression was come due to its graphical representation.

There are two types of Linear Regression:-

* **Simple Linear Regression**- In a simple linear regression algorithm the model shows the linear relationship between a dependent and a single independent variable. In this, the dependent variable must be a continuous value while the independent variable can be any continuous or categorical value.
* **Multiple Linear Regression**- In a multiple linear regression algorithm the model shows the linear relationship between a single dependent and more than one independent variable.

**4.4.2. Decision Tree**

Decision tree is a supervised learning algorithm which can be used for both classification and regression problem. This model is very good at handling tabular data with numerical or categorical features. It uses a tree-like structure flow chart to solve the problem. A decision tree is arriving at an estimate by asking a series of questions to the data, each question narrowing our possible values until the model gets confident enough to make a single prediction. The order of the question as well as their content is being determined by the model. In addition, the questions asked are all in a True/False form. Here in our project, we are focusing on decision tree regression only. It is used for the continuous output problem. Continuous output means the output of the result is not discrete. It observes features of an object and trains a model in the structure of a tree to predict data that produce meaningful continuous output.

**4.4.3.** **Random Forest**

Random forest is a supervised learning algorithm which can be used for both classification and regression problem. It is a collection of Decision Trees. In general, Random Forest can be fast to train, but quite slow to create predictions once they are trained. This is due because it has to run predictions on each tree and then average their predictions to create the final prediction. A more accurate prediction requires more trees, which results in a slower model. In most real-world applications the random forest algorithm is fast enough, but there can certainly be situations where run-time performance is important and other approaches would be preferred. A random forest is a meta-estimator (i.e. it combines the result of multiple predictions) which **aggregates many decision trees**, with some helpful modifications. Random forest first splits the dataset into n number of samples and then apply decision tree on each sample individually. After that, the final result is that predicted accuracy whose majority is higher among all.

Random Forest depends on the concept of ensemble learning. An ensemble method is a technique that **combines the predictions from multiple machine learning algorithms** together to make more accurate predictions than any individual model. A model comprised of many models is called an **Ensemble model**.

Random forest is a bagging technique and **not a boosting** technique. The trees in **random forests** are run in parallel. There is no interaction between those trees while building random forest model.

**4.4.4. Gradient Boosting**

Gradient boosting is a technique which can be used for both classification and regression problem. This model combines the predictions from multiple decision trees to generate the final predictions. Also, each node in every other decision tree takes a different subset of features for selecting the best split. But there is a slight difference in gradient boosting in comparison to random forest that is gradient boosting builds one tree at a time and combines the results along the way. Also, it gives better performance than random forest. The idea of gradient boosting originated in the observation by [Leo Breiman](https://en.wikipedia.org/wiki/Leo_Breiman) that boosting can be interpreted as an optimization algorithm on a suitable cost function. Gradient Boosting trains many models in a gradual, additive, and sequential manner.

The modeling is done in the following steps:-

* First, we split the dataset into a training set and a testing set.
* Then we train the model on the training set.
* And at last, we test the model on the testing set and evaluate how well our model performs.

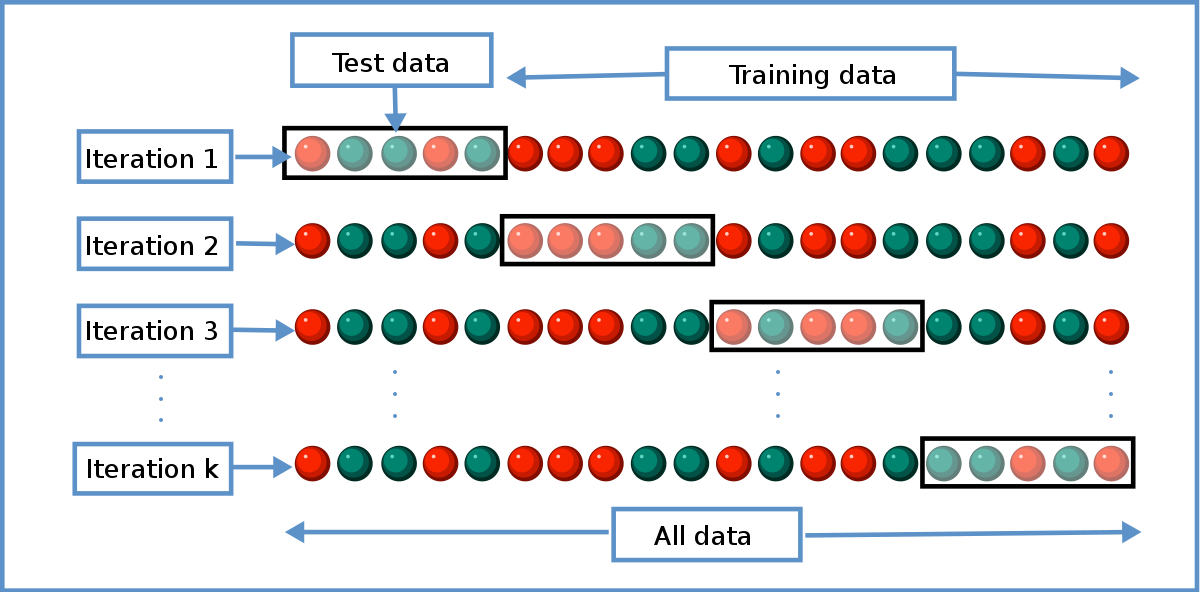
So after applying these models we get the following accuracy:

**Table 4.2: Model Accuracy Table**

|  |  |  |
| --- | --- | --- |
| **Serial No.​** | **Models** | **Accuracy​** |
| 1​ | Linear Regression​ | 0.747545073​ |
| 2​ | Decision Tree​ | 0.961791729​​ |
| 3​ | Random Forest​ | 0.962269474 |
| 4​ | Gradient Boosting Regressor | 0.963187213​  ​ |

**4.4.5 K-fold Cross Validation**

We also apply cross validation using linear regression algorithm. It is a technique where the datasets are split into multiple subsets and learning models are trained and evaluated on these subset data. It is a resampling procedure used to evaluate machine learning models on a limited data sample. It is one of the most widely used technique. In this, the dataset is divided into k-subsets (folds) and are used for training and validation purpose for k iteration times. Each subsample will be used at least once as a validation dataset and the remaining (k-1) as the training dataset. Once all the iterations are completed, one can calculate the average prediction rate for each model. The error estimation is averaged over all k trials to get the total effectiveness of our model.



**Fig. 4.11 Cross-Validation Courtesy of Wikimedia**

**4.5 Testing**

In Machine Learning the main task is to model the data and predict the output using various algorithms. But since there are so many algorithms, it was really difficult to choose the one for predicting the final data. So we need to compare our models and choose the one with the highest accuracy.

Machine learning applications are not 100% accurate, and approx never will be. There are some of the reasons why testers cannot ignore learning about machine learning. The fundamental reason is that these applications learning limited by data they have used to build algorithms. For example, if 99% of emails aren't spammed, then classifying all emails as not spam gets 99% accuracy through chance. Therefore, you need to check your model for algorithmic correctness. Hence testing is required. Testing is a subset or part of the training dataset that is built to test all the possible combinations and also estimates how well the model trains. Based on the test data set results, the model was fine-tuned.

**Mean Squared Error** (MSE), **Mean Absolute Error** (MAE), and Root Mean Squared Error (RMSE)are used to evaluate the regression problem's accuracy.

**4.5.1 Mean Absolute Error (MAE)**

It is the mean of all absolute error. MAE (ranges from 0 to infinity, lower is better) is much like RMSE, but instead of squaring the difference of the residuals and taking the square root of the result, it just averages the absolute difference of the residuals. This produces positive numbers only and is less reactive to large errors. MAE takes the **average** of the error from every sample in a dataset and gives the output.

Hence, **MAE = True values – Predicted values**

**4.5.2 Mean Squared Error (MSE)**

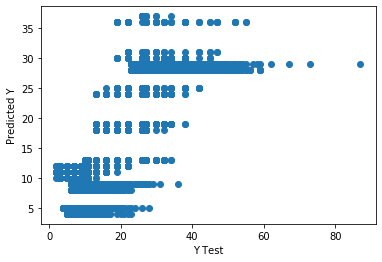
It is the mean of square of all errors. It is the sum, overall the data points, of the square of the difference between the predicted and actual target variables, divided by the number of data points. MSE is calculated by taking the average of the square of the difference between the original and predicted values of the data.

**4.5.3 Root Mean Squared Error (RMSE)**

RMSE is the standard deviation of the errors which occur when a prediction is made on a dataset. This is the same as MSE (Mean Squared Error) but the root of the value is considered while determining the accuracy of the model. RMSE (ranges from 0 to infinity, lower is better), also called Root Mean Square Deviation (RMSD), is a quadratic-based rule to measure the absolute average magnitude of the error.

In our project, we perform testing on two models: Linear Regression and Random Forest.

**Linear Regression Model Testing:**

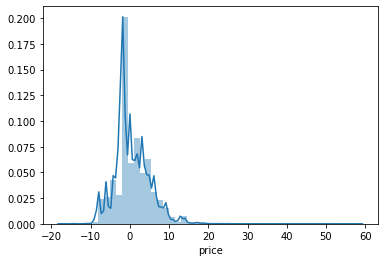


**Fig. 4.12 Scatter Plot for Linear Regression**

We draw a scatter plot between predicted and tested values and then find errors like MSE, MAE, and RMSE. After that, we also draw a distribution plot of the difference between actual and predicted values using the seaborn library. A **distplot**or distribution plot represents the overall distribution of continuous data variables.

**Table 4.3: Error table for Linear Regression**

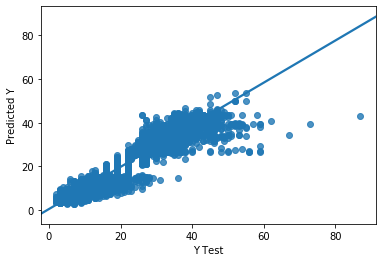
|  |  |  |
| --- | --- | --- |
| **Serial No.​** | **Models** | **Accuracy​** |
| 1​ | Mean Absolute Error | 3.40607721 |
| 2​ | Mean Squared Error | 20.0334370 |
| 3​ | Root Mean Absolute Error | 4.47587277 |



**Fig. 4.13 Dist Plot for Linear Regression**

**Random Forest Model Testing:**

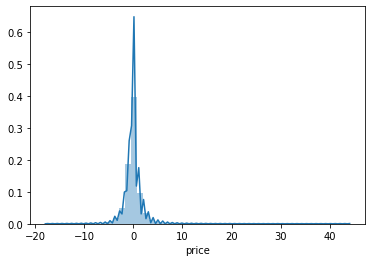
Similarly, we draw scatter plot, dist plot, and find all three errors for random forest also.



**Fig. 4.14 Scatter Plot for Random Forest**

**Table 4.4 Error table for Random Forest**

|  |  |  |
| --- | --- | --- |
| **Serial No.​** | **Models** | **Accuracy​** |
| 1​ | Mean Absolute Error | 0.99813700 |
| 2​ | Mean Squared Error | 2.94465361 |
| 3​ | Root Mean Absolute Error | 1.71599930 |



**Fig. 4.15 Dist Plot for Random Forest**

**4.6 Price Prediction Function**

After finding the errors for both linear regression and random forest algorithm, we build a function name “predict\_price” whose purpose is to predict the price by taking 4 parameters as input. These four parameters are cab name, source, surge multiplier, and icon (weather). As the dataset train on the continuous values and not on categorical values, these values are also passed in the same manner i.e. in integer type. We create a manual for users which gives instructions about the input like what do you need to type for a specific thing and in which sequence.

We use random forest model in our function to predict the price. First, we search for all the desired rows which have the input cab name and extract their row number. After then we create an array x which is of thelength of the new dataset and it’s initially all values are zero. After creating the blank array we assign the input values of source, surge multiplier, and icon to the respected indices. Following it we check the count of all desired rows if it was greater than zero or not. If the condition gets true, we assign the value 1 to the index of x array and return the price using the predict function with trained random forest algorithm.

It somehow works like a hypothesis space because it gives an output for any input from input the space.

**CONCLUSION**

Before working on features first we need to know about the data insights which we get to know by EDA. Apart from that, we visualize the data by drawing various plots, due to which we understand that we don’t have any data for taxi’s price, also the price variations of other cabs and different types of weather. Other value count plots show the type and amount of data the dataset has. After this, we convert all categorical values into continuous data type and fill price Nan by the median of other values. Then the most important part of feature selection came which was done with the help of recursive feature elimination. With the help of RFE, the top 25 features were selected. Among those 25 features still, there are some features which we think are not that important to predict the price so we drop them and left with 8 important columns.

We apply four different models on our remaining dataset among which Decision Tree, Random Forest, and Gradient Boosting Regressor prove best with 96%+ accuracy on training for our model. This means the predictive power of all these three algorithms in this dataset with the chosen features is very high but in the end, we go with random forest because it does not prone to overfitting and design a function with the help of the same model to predict the price.

**REFERENCES**

* Abel Brodeurand & Kerry Nield (2018) An empirical analysis of taxi, Lyft and Uber rides: Evidence from weather shocks in NYC
* Junfeng Jiao (2018) Investigating Uber price surges during a special event in Austin, TX
* Anna Baj-Rogowska (2017) Sentiment analysis of Facebook posts: The Uber Case
* Anastasios Noulas, Cecilia Mascolo, Renaud Lambiotte, and Vsevolod Salnikov (2014) OpenStreetCab: Exploiting Taxi Mobility Patterns in New York City to Reduce Commuter Costs
* <https://www.singlegrain.com/blog-posts/business/10-lessons-startups-can-learn-ubers-growth/>
* <https://www.kaggle.com/brllrb/uber-and-lyft-dataset-boston>
* https://matplotlib.org/1.3.1/users/legend\_guide.html
* https://www.sciencedirect.com/science/article/abs/pii/S0167268118301598
* <https://sci-hub.se/https://www.sciencedirect.com/science/article/abs/pii/S2210539517301165>
* <https://ieeexplore.ieee.org/abstract/document/8260068>
* <https://www.researchgate.net/publication/305524879_Dynamic_Pricing_in_a_Labor_Market_Surge_Pricing_and_Flexible_Work_on_the_Uber_Platform>
* [https://www.sciencedirect.com/science/article/abs/pii/S0167268118301598#:~:text=We%20look%20at%20the%20effect,rides%20in%20New%20York%20City.&text=The%20number%20of%20Uber%20(Lyft,higher%20when%20it%20is%20raining.&text=The%20number%20of%20taxi%20rides,higher%20when%20it%20is%20raining.&text=Taxi%20rides%2C%20passengers%20and%20fare,after%20Uber%20entered%20the%20market](https://www.sciencedirect.com/science/article/abs/pii/S0167268118301598#:~:text=We%20look%20at%20the%20effect,rides%20in%20New%20York%20City.&text=The%20number%20of%20Uber%20(Lyft,higher%20when%20it%20is%20raining.&text=The%20number%20of%20taxi%20rides,higher%20when%20it%20is%20raining.&text=Taxi%20rides%2C%20passengers%20and).
* https://arxiv.org/abs/1503.03021
* https://subscription.packtpub.com/book/big\_data\_and\_business\_intelligence/9781789808452/1/ch01lvl1sec19/label-encoding
* https://github.com/Ankush123456-code/house\_price\_prediction\_end\_to\_end\_project/blob/main/model/python.ipynb
* https://github.com/ankita1112/House-Prices-Advanced-Regression/blob/master/Housing\_Prediction\_full.ipynb
* https://scikit-learn.org/stable/modules/multiclass.html
* https://www.codegrepper.com/code-examples/python/confusion+matrix+python
* https://topepo.github.io/caret/recursive-feature-elimination.html
* https://arxiv.org/pdf/1503.03021.pdf
* https://www.sciencedirect.com/science/article/abs/pii/S0167268118301598
* https://www.kaggle.com/punit0811/machine-learning-project-basic-linear-regression
* https://gdcoder.com/decision-tree-regressor-explained-in-depth/
* https://medium.com/towards-artificial-intelligence/machine-learning-algorithms-for-beginners-with-python-code-examples-ml-19c6afd60daa
* https://www.tutorialspoint.com/machine\_learning\_with\_python/machine\_learning\_algorithms\_performance\_metrics.htm
* https://blog.paperspace.com/implementing-gradient-boosting-regression-python/
* <https://www.studytonight.com/post/what-is-mean-squared-error-mean-absolute-error-root-mean-squared-error-and-r-squared>
* <https://statisticsbyjim.com/regression/overfitting-regression-models/>
* <https://www.itl.nist.gov/div898/software/dataplot/refman1/auxillar/striplot.htm#:~:text=One%20problem%20with%20strip%20plots,increment%20to%20the%20vertical%20coordinate>.
* <https://www.toppr.com/ask/question/what-are-the-limitations-of-a-scatter-diagram/>
* <https://www.geeksforgeeks.org/ml-label-encoding-of-datasets-in-python/#:~:text=Limitation%20of%20label%20Encoding,in%20training%20of%20data%20sets>.
* <https://digitalag.osu.edu/sites/digitag/files/imce/images/ag_sensing/Figure3.png>
* <https://www.researchgate.net/publication/325800934/figure/fig1/AS:638132596768768@1529154075317/The-main-procedure-of-the-recursive-feature-elimination-RFE-method.png>
* <https://upload.wikimedia.org/wikipedia/commons/thumb/b/b5/K-fold_cross_validation_EN.svg/1200px-K-fold_cross_validation_EN.svg.png>
* https://static.packt-cdn.com/products/9781789345070/graphics/108f2a01-3e31-4907-a1a5-4baf441c3eed.png