Project Report

On

Predicting Fare amount of Uber Rides in New York.

Submitted in partial fulfillment for the award of

Post Graduate Diploma in Big Data Analytics (PG-DBDA)

From Know-IT(Pune)

Guided by: Mrs. Trupti Joshi

Submitted By: Anuja Salunke (220943025007)
Shraddha Bagal (220943025039)
Snehal Patil (220943025046)
Vidhi Borele (220943025055)

CERTIFICATE

TO WHOMSOEVER IT MAY CONCERN

This is to certify that

Anuja Salunke (220943025007)

Shraddha Bagal (220943025039)

Snehal Patil (220943025046)

Vidhi Borele (220943025055)

Have successfully completed their project

On

Predicting Fare Amount of uber rides in New York.

Under the guidance of Mrs. Trupti Joshi.

ACKNOWLEDGEMENT

This project “Predicting the Fare amount of uber rides in New York” was a great learning experience for us and we are submitting this work to CDAC Know-IT (Pune).

We all are very glad to mention the name of Mrs. Trupti Joshi for her valuable guidance to work on this project. Her guidance and support helped us to overcome various obstacles and intricacies during the course of project work.

We are highly grateful to Mr. Vaibhav Inamdar, Center coordinator, Know-IT, C-DAC, for his guidance and support whenever necessary while doing this course Post Graduate Diploma in Big Data Analytics (PG-DBDA) through C-DAC ACTS, Pune.

Our most heartfelt thanks go to Mr. Shrinivas Jadhao, Vice-President (Training), Know-IT, CDAC who gave all the required support and kind coordination to provide all the necessities like required hardware, internet facility and extra Lab hours to complete the project and throughout the course up to the last day here in C-DAC Know-IT, Pune.

Submitted By:

Anuja Salunke (220943025007)

Shraddha Bagal (220943025039)

Snehal Patil (220943025046)

Vidhi Borele (220943025055)

TABLE OF CONTENTS

Abstract

Machine Learning (ML) is probably the most popular branch of AI to date. Most systems that use ML methods use them to perform predictive analysis. This project aims to conduct a predictive analysis of fare prices based on location distance.

To do this, we carried out the dataset from Kaggle and cleaned it properly for our correct prediction and analysis. We then considered all the features and did feature engineering and extraction for the analysis.

We analyzed a large dataset of uber.csv taken from Kaggle using data mining.

Processing included data cleaning, EDA etc. We did predictive analysis on fare prices, analysis on the accuracy of trip frequency and observed other trip traits based on changes in few attributes. The results were explored using graphical patterns using tableau for better Visualization.

We have also used geopy library of python to develop something similar to GUI application for user interaction to assist data-driven decision-making. It takes the address in New York, gives the measure of latitude and longitude and then predicts the fare depending on the pickup and the drop-off location.

The purpose of this study is to provide researchers, companies or anyone wishing to perform predictive analysis with clues that will enable them to choose the best ML method(s).

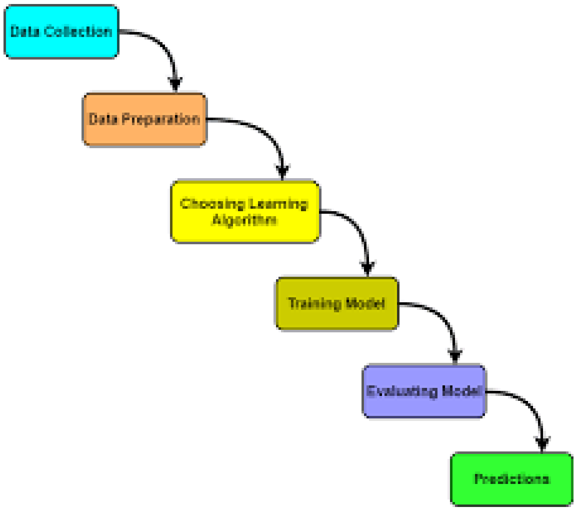
Introduction

- Uber is defined as a P2P platform. The platform links you to drivers who can take you to your destination safely.
- The dataset includes primary data of 9 columns and 200000 rows.
- Uber pickups and drop-offs with details including the date, time of the ride as well as longitude-latitude information, number of passengers etc.

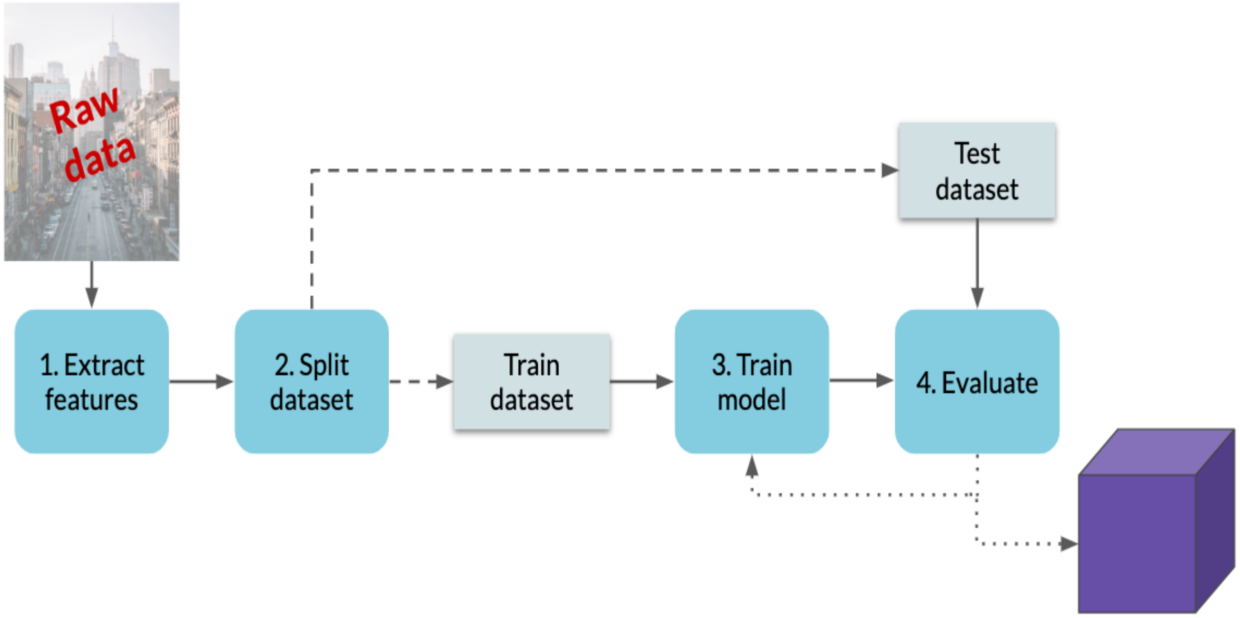
Effective taxi dispatching will facilitate each driver and passenger to know the fare amount depending on the distance of the trip.

- Every ride booked on Uber gives their team a large amount of information, including the riders' booking preferences, pickup, and drop-off trends, availability of drivers in the area, traffic patterns, duration, speed, weather factors, and more.
- Some popular uses include calculating a competitive fare to maximize profits (using predictive modelling algorithms).
- Estimating surge prices, tuning the requirements of drivers in a particular region, catching fake rides, and fake drivers, and estimating ride info like ETA.

Methodology



Machine Learning Workflow



Machine Learning Models

1)Linear Regression:

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.

Assumptions of Linear Regression

- Linear relationship between the features and target:
- Small or no multicollinearity between the features:
- Homoscedasticity Assumption
- Normal distribution of error terms
- No autocorrections

2) Lasso Regression:

In statistics and machine learning lasso (least absolute shrinkage and selection operator; also Lasso or LASSO) is a regression analysis method that performs both variable selection and regularization in order to enhance the prediction accuracy and interpretability of the resulting statistical model.

A hyperparameter is used called "*lambda*" that controls the weighting of the penalty to the loss function.

3) Elastic Net Regression:

The benefit is that elastic net allows a balance of both penalties, which can result in better performance than a model with either one or the other penalty on some problems.

Another hyperparameter is provided called "*lambda*" that controls the weighting of the sum of both penalties to the loss function.

A default value of 1.0 is used to use the fully weighted penalty; a value of 0 excludes the penalty.

Very small values of lambada, such as 1e-3 or smaller, are common.

- elastic_net_loss = loss + (lambda * elastic_net_penalty)
- Elastic net is a penalized linear regression model that includes both the L1 and L2 penalties during training.

4)Decision Tree Regression:

It is a tree-structured classifier with three types of nodes.

The Root Node is the initial node which represents the entire sample and may get split further into further nodes.

The Interior Nodes represent the features of a data set and the branches represent the decision rules.

Finally, the Leaf Nodes represent the outcome. This algorithm is very useful for solving decision-related problems.

With a particular data point, it is run completely through the entirely tree by answering *True/False* questions till it reaches the leaf node.

The final prediction is the average of the value of the dependent variable in that particular leaf node.

Through multiple iterations, the Tree is able to predict a proper value for the data point.

5)Random Forest Regressor:

Ensemble learning is the process of using multiple models, trained over the same data, averaging the results of each model ultimately finding a more powerful predictive/classification result.
Our hope, and the requirement, for ensemble learning is that the errors of each model (in this case decision tree) are independent and different from tree to tree.

Bootstrapping is the process of randomly sampling subsets of a dataset over a given number of iterations and a given number of variables.

These results are then averaged together to obtain a more powerful result. Bootstrapping is an example of an applied ensemble model.

The bootstrapping **Random Forest** algorithm combines ensemble learning methods with the decision tree framework to create multiple randomly drawn decision trees from the data, averaging the results to output a new result that often leads to strong predictions/classifications.

6) Gradient Boost:

Gradient boosting is a method standing out for its prediction speed and accuracy, particularly with large and complex datasets.

From Kaggle competitions to machine learning solutions for business, this algorithm has produced the best results. We already know that errors play a major role in any machine learning algorithm.

There are mainly two types of error, bias error and variance error. Gradient boost algorithm *helps us minimize bias error* of the model.

The main idea behind this algorithm is to build models sequentially and these subsequent models try to reduce the errors of the previous model.

7) Extreme Gradient Boosting:

Gradient boosting refers to a class of ensemble machine learning algorithms that can be used for classification or regression predictive modeling problems.

Ensembles are constructed from decision tree models. Trees are added one at a time to the ensemble and fit to correct the prediction errors made by prior models.

This is a type of ensemble machine learning model referred to as boosting.

Models are fit using any arbitrary differentiable loss function and gradient descent optimization algorithm.

This gives the technique its name, “*gradient boosting*,” as the loss gradient is minimized as the model is fit, much like a neural network.

Models And Their Accuracy

Machine Learning Model	Accuracy
Linear Regression	0.75
Lasso Regression	0.671
Elastic net	0.712
Decision Tree Regressor	0.6083
Random Forest Regressor	0.807
Gradient Boost Regressor	0.7969
XGBoost	0.8228

Geopy Library (Python)

Getting Address Information

Location objects have instances of address, altitude, latitude, longitude, point**.** "Address" returns a concatenated string of all information that belongs to the address. This may or may not include information like an amenity, road, neighborhood, city block, suburb, county, city, state, postcode, country, country code.

Getting Latitude and Longitude

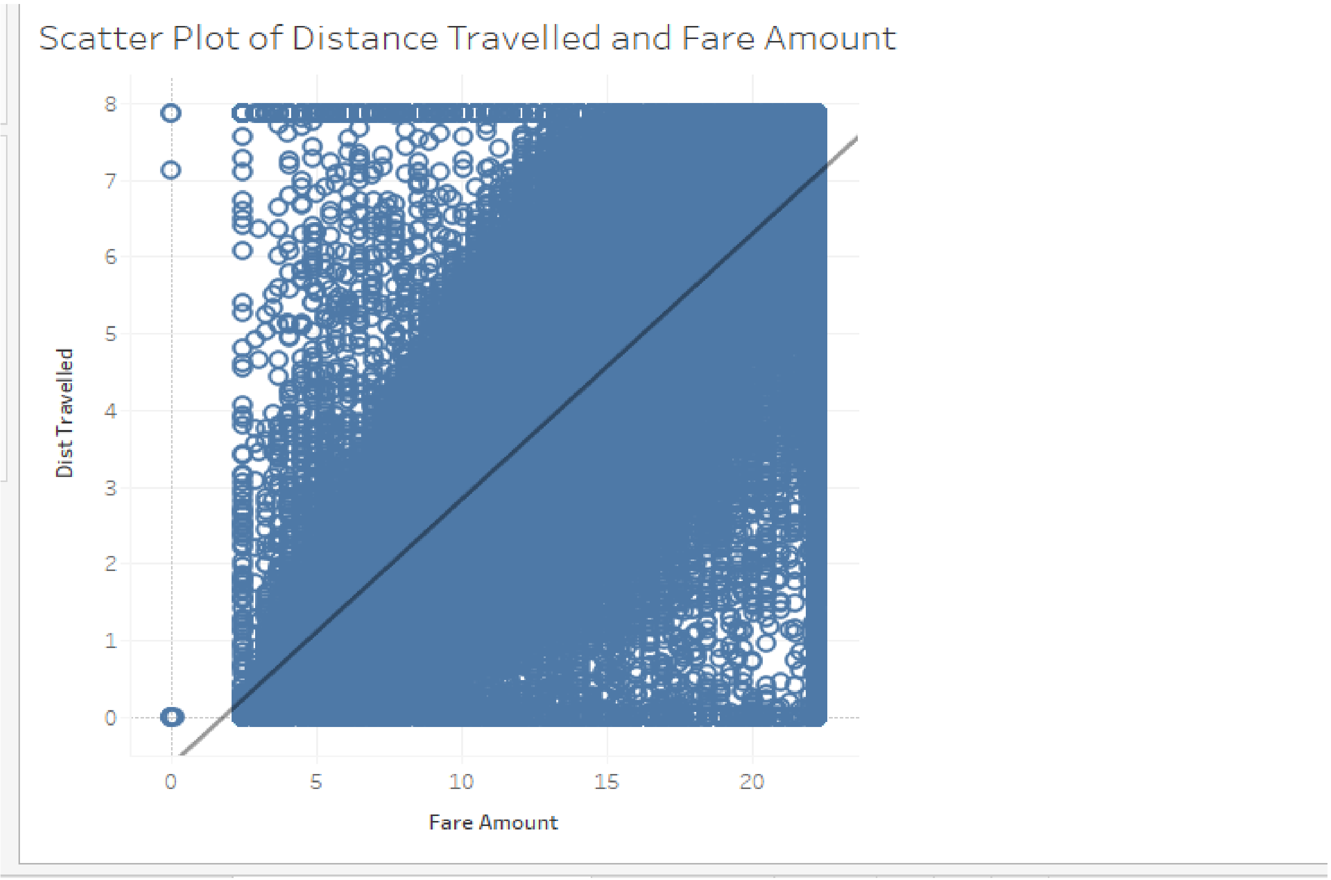
We noticed above that we are using the "reverse" function of our geocoder object. That's because the non-reversed, general use of this geocoder is to get the actual coordinates from the address.

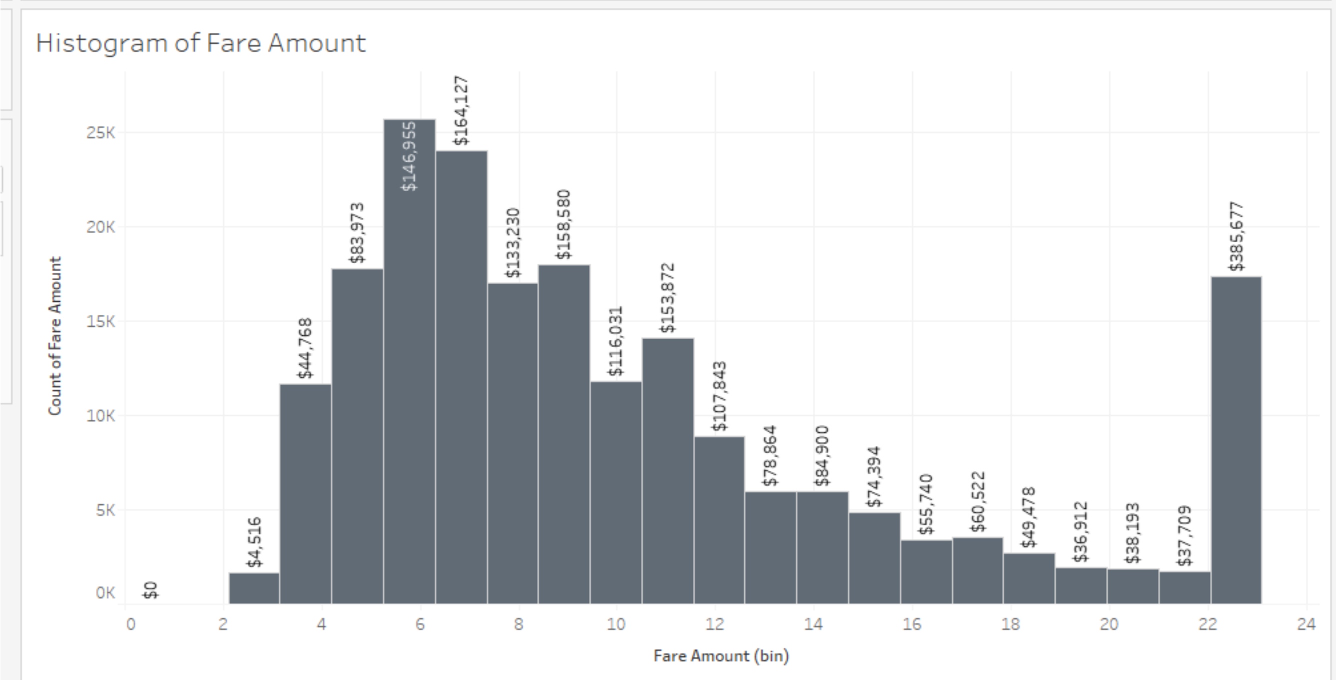
This can be handy if we want to assess the geographical effect or to quantify the categorical locations.

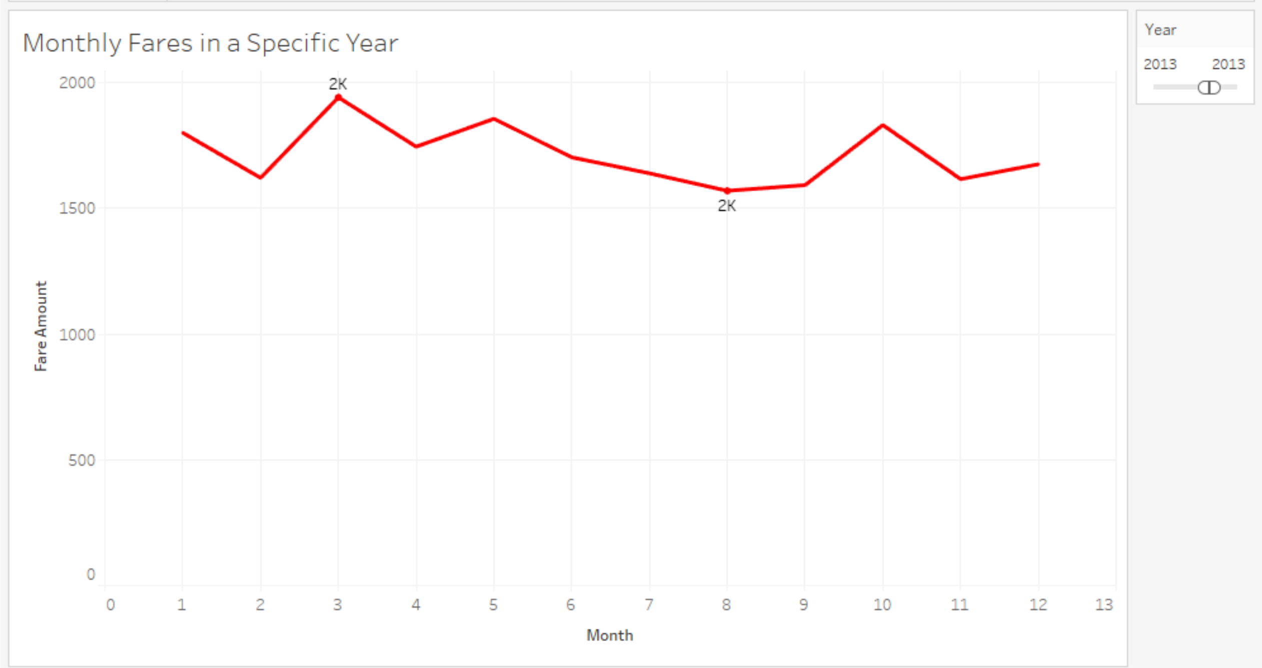
Getting Distance

Another important variable we can extract from the coordinates is the distance between different locations. This is a simple task using Geopy. Geopy offers both the great-circle distance and the geodesic distance.

Data visualization and Representation




Histogram of Fare Amount
Count of Fare Amount
0K
5K
10K
15K
20K
25K
$0
$4,516
$44,768
$83,973
$146,955
$164,127
$133,230
$158,580
$116,031
$153,872
$107,843
$78,864
$84,900
$74,394
$55,740
$60,522
$49,478
$36,912
$38,193
$37,709
$385,677
0
2
4
6
8
10
12
14
16
18
20
22
24
Fare Amount (bin)


Monthly Fares in a Specific Year
Fare Amount
2000
1500
1000
500
0
2K
2K
0
1
2
3
4
5
6
7
8
9
10
11
12
13
Month
Year
2013
2013

Conclusion and Future Scope

- People prefer to have a shared ride for their journeys.
- People avoid riding when it rains.
- When traveling long distances, the price also increases accordingly.
- Uber could be the first choice for long distances.

This guide briefly outlines some of the tips and tricks to simplify analysis and undoubtedly highlighted the critical importance of a well-defined business problem, which directs all coding efforts to a particular purpose and reveals key details.

References

1) https://towardsdatascience.com/things-to-do-with-latitude-longitude-data-using-geopy-python-1d356ed1ae30.

2) https://geopy.readthedocs.io/en/stable/

3) https://towardsdatascience.com/workflow-of-a-machine-learning-project-ec1dba419b94

4) https://www.kaggle.com/datasets/elemento/nyc-yellow-taxi-trip-data

5) https://www.datacamp.com/tutorial/pickle-python-tutorial

6) Bierman, L., “Random forests,” Machine learning, Vol. 45.

7) Hands-On Machine Learning with Scikit-Learn, Kera’s, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems.