“Tonga” – Next Gen app for booking vehicles

Introduction

- A next gen app for booking vehicles like OLA, Uber
- Interface for a customer, driver, and manager
- Uses machine learning to predict accurate fare prices

Agenda

1.
Customer

Tonga cab service – Customers can book a cab, View Previous Bookings and Cancel Booking.

2.
Fare Prediction

Machine Learning Algorithm which will tell the fare to be charged for a passenger.

3.
Driver

Driver version of the Tango app has View Bookings option, using which Driver can see and select the ride to serve.

4.
Manager

Tonga Manager has options to Create a new driver and View available drivers. Also has option to View all bookings and cancel a booking.

Fare Prediction Strategy

Step 01

Understand the type of data. Identify the output variable and the factors which affect the output variable.

Step 02

Count and Remove the null value rows in the target variable..

Step 03

Perform train test split.

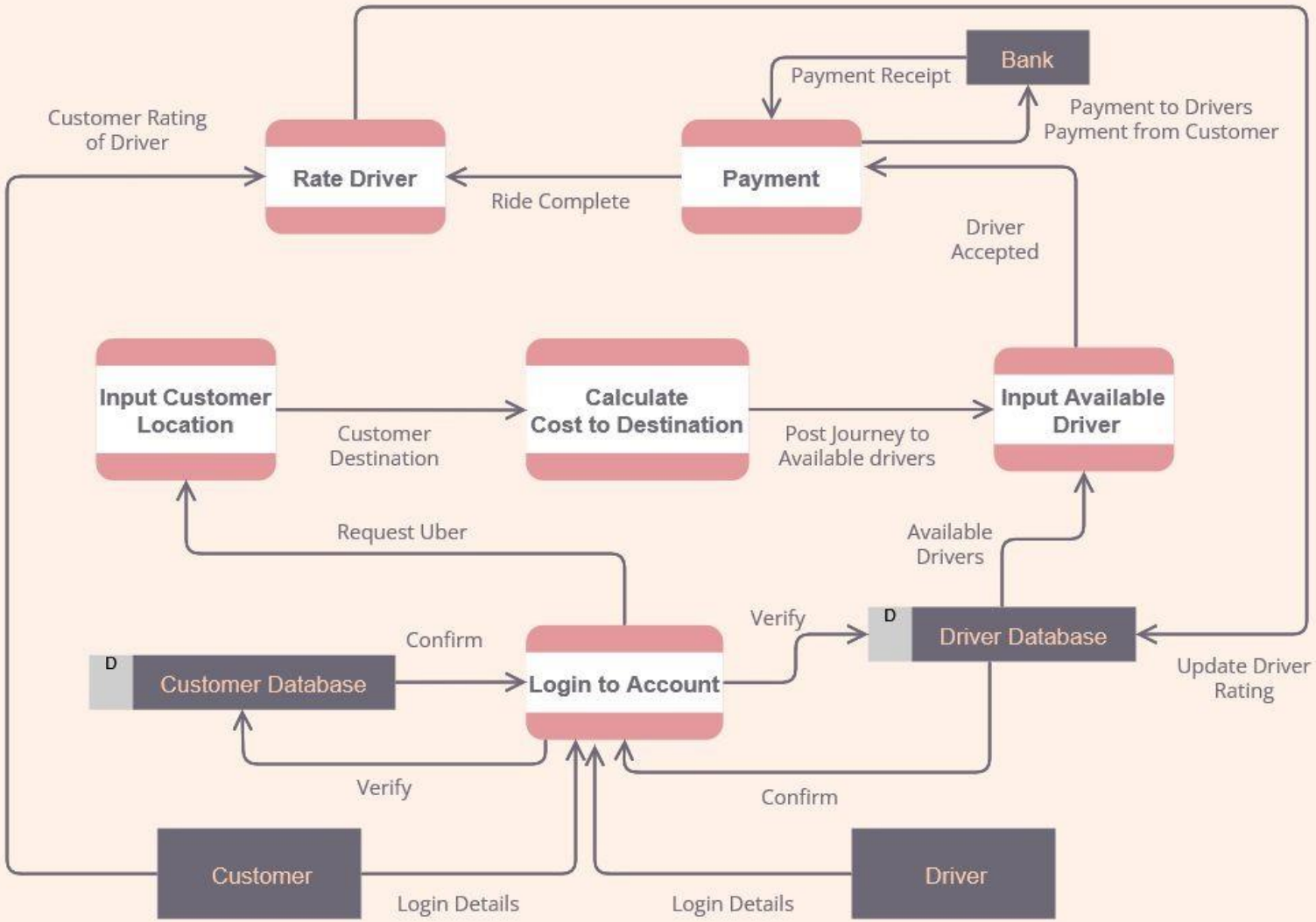
Step 04

Predict the accuracy using regression models.

Step 05

Check and compare the accuracy of the different models.

Tonga Work Flow Diagram



Random Forest

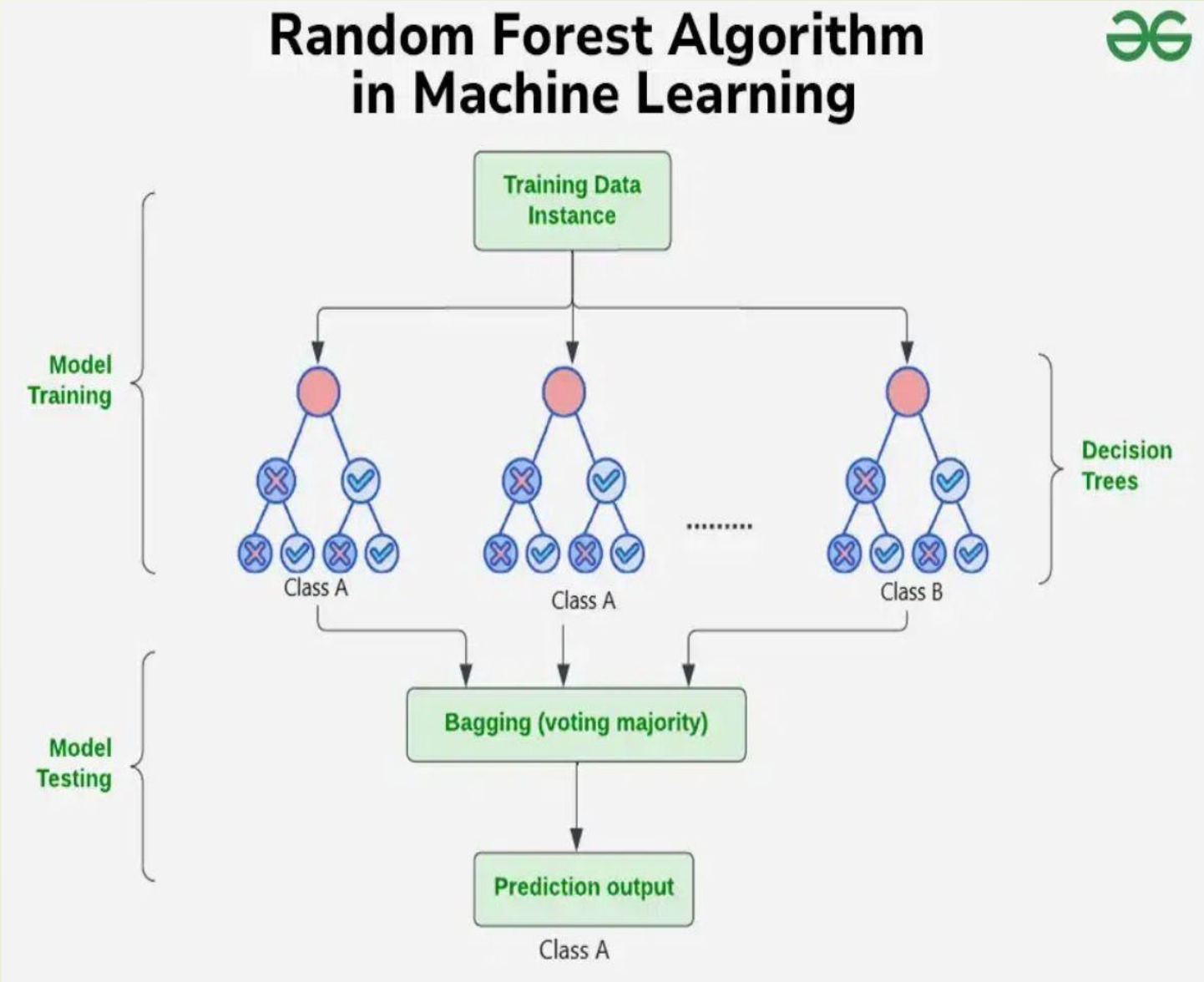
Ensemble Learning Method: Combines multiple decision trees.

Bagging Technique: Uses bootstrap aggregating (bagging) to train each tree on random subsets of data, reducing variance and preventing overfitting.

Random Feature Selection: Selects random subsets of features at each split.

Aggregated Predictions: For classification, predicts the majority class; for regression, predicts the average output from all trees.

Feature Importance: Estimates the impact of features on predictions by assessing the decrease in accuracy when feature values are permuted.



Performance v/s other models

- We used the Root Mean Squared Error to measure the performance of the model:

$$\text{RMSE} = \sqrt{\sum_{i=1}^{n} \frac{(\hat{y}_i - y_i)^2}{n}}$$

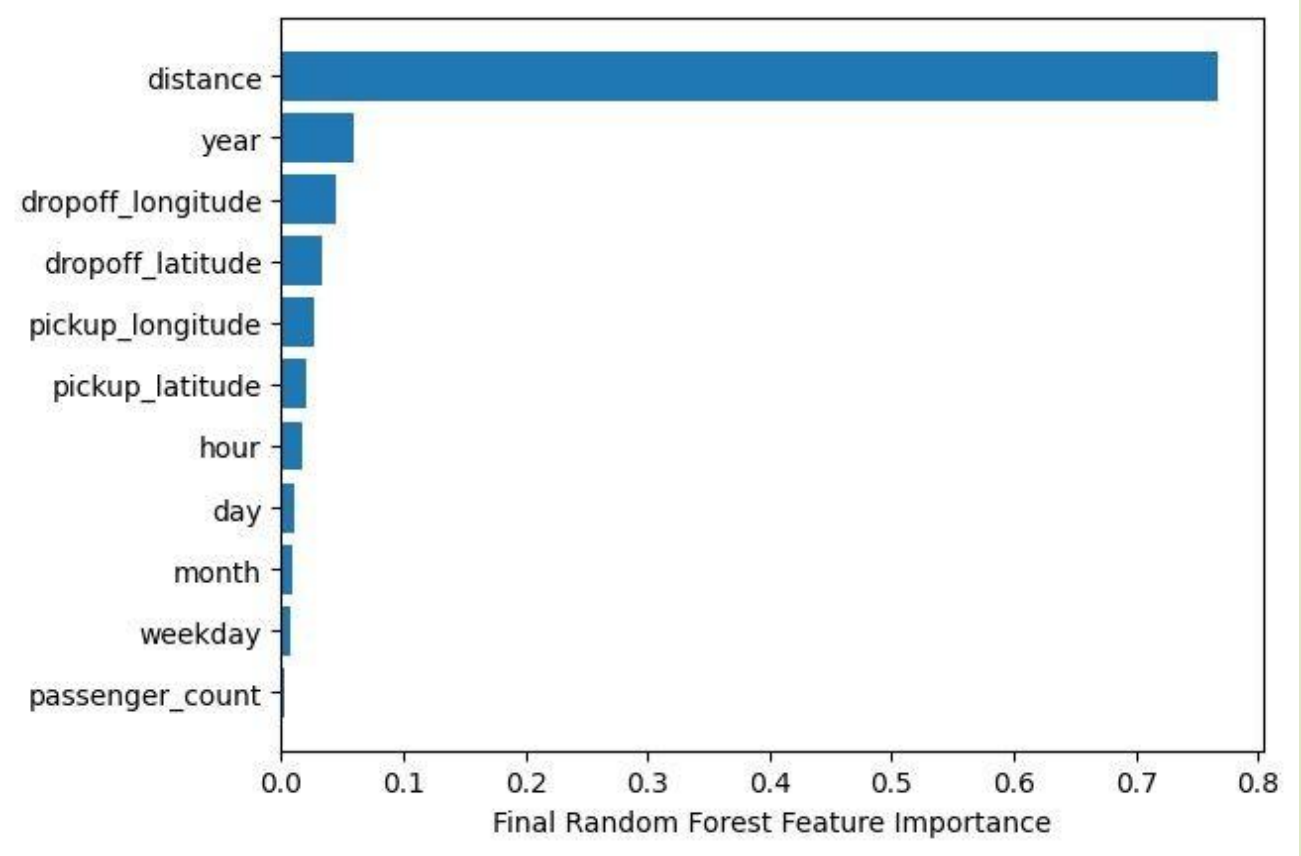
- RMSE value for Linear regression is: 6.091485402733827
- RMSE Value for Random Forest Regression is: 4.8408201738182335

- Comparing the two models, we find that Random Forest Regression performs better than Linear Regression and hence we will use it as our final model.

Results

	fare_amount
0	48.7165
1	459.2750
2	457.6750
3	48.8665
4	47.6270

Duration: 0:00:42.452150



Dataset

- A research group (FiveThirtyEight) obtained the data from the NYC Taxi & Limousine Commission (TLC) by submitting a Freedom of Information Law request on July 20, 2015.
- The directory contains data on Uber pickups in New York City from April - September 2014, and more from January - June 2015. All the files were received on August 3, Sept. 15 and Sept. 22, 2015.

- We used 195710 unique entries (after cleaning)
- 11 feature columns - fare_amount, pickup_longitude, pickup_latitude, dropoff_longitude, dropoff_latitude, passenger_count, day, hour, weekday, month, year

Dataset preprocessing

- Note that the is minimum value of fare is negative which is -52 which is not the valid value, so we need to remove the fare values which are negative, just for caution.
- Secondly, passenger_count minimum value is 0 and maximum value is 6.
- There are some variables with missing values:

```
key                  0
fare_amount          3
pickup_datetime      0
pickup_longitude     0
pickup_latitude      0
dropoff_longitude    1
dropoff_latitude     1
passenger_count      0
dtype: int64
```

- Finally, the minimum pickup and dropoff longitudes and latitudes have some weird large values. Probably, there is a problem here.

Tools and libraries

- Libraries used:

(a) seaborn: statistical data visualization, a Python data visualization library based on matplotlib

(b) geopy: Python client for several popular geocoding web services

(c) scikit-learn: machine learning in Python — Simple and efficient tools for predictive data analysis

(d) pyproj: Python interface to PROJ cartographic projections and coordinate transformations library

- Tools:

-Jupyter lab

-Tableplus

Conclusion

Models

Generate 2 models and make comparison.
Linear regression
Random Forest regression

Conclusion

Random Forest Regression performs better than Linear Regression in this case and therefore we will be using the Random Forest Regression Model to make predictions on the test dataset provided.

Next Steps

Season (spring, summer, fall, winter)
Holiday and Working day split
Traffic conditions
Combine with open data sources like weather data

Thank you