Data Analysis Final Project Report:

Analysis and Prediction of Uber and Lyft Price

Team Member: Yujie Cui (yuc113@pitt.edu)

I. Introduction:

Unlike public transportation, Uber and Lyft's ride prices are not constant and fluctuate greatly. And their demand is more dispersive and uncertain. In our project, we want to analyze the cab demand and predict the ride price based on given conditions. Our final goal is giving suggestions to drivers and passengers. The driver could find the location where may have higher demand; and the passenger may avoid high ride-price situation. Python is used as our data analysis tool, which is popular in the field of data mining and machine learning. To predict the ride price, we build a random forest model using given conditions. Besides, we also employ some packages in Python to show the relationship between cab demand and different features.

II. Problem Description:

There are three main problems we try to solve:

1. Can we accurately predict the cab price?
2. What features are more important on cab price?
3. When and where is the cab demand higher?

Our main goal is to build a model to predict ride price based on given conditions. As we all know, distance, time, car type and so on, may impact on cab price. If we know a certain condition, can we use it to predict a right ride price? Besides, we also wonder which features are more important on cab price. As costumers, we always want to reduce our cost. If we know what conditions may cause high cost, we can avoid them and save money. For drivers, they are more concerned for earning money. As a result, increasing cab order will benefit to drives. Then where and when the cab demand higher is important and it is our third problem.

III. Methodology

To solve those problems, we visualize the relationship between each feature and build a model to predict the cab price. To evaluate the results, we employ R^2 score in the prediction of model.

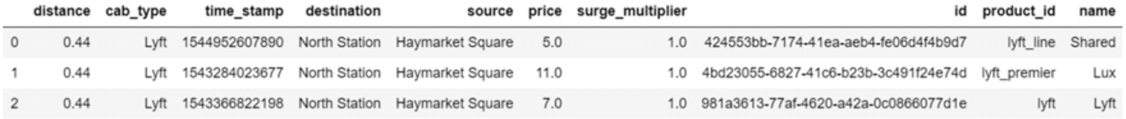
The visualization tools include pie chart, box plot and scatter plot. Pie chart is a great tool to visualize the proportion of different parts in a given feature, which could tell us which one is largest and which one is smallest. Box plot, in our project, is used to display the relationship between two features. One is category feature and the other one is numerical feature. We can see the 25%, 50% and 75% of the numerical feature in certain category. Finally, we also employ scatter plot to visualize the relationship between two numerical features.

The model in our project is random forest. Random forest is an ensemble tree-based model for classification and regression. The difference between random forest and general decision tree is that random forest contains multiple decision trees and its final model is based on the average of those trees, which reduces the variance and increase the accuracy of prediction. Besides, the package of random forest in python have a build-in feature called “feature importance” , which can help us directly visualize the importance of all of those feature we use.

IV. Dataset Description

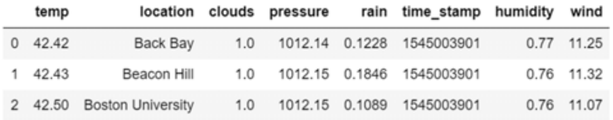
In our project, we choice cab and weather dataset from Kaggle predict cab prices against given features. This dataset is real-time data using Uber and Lyft api queries and corresponding weather conditions in Boston [1]. The cab ride data contains many types of cabs for Uber and Lyft and their price for some popular locations. Weather data contains weather features, such as temperature, rain, cloud, etc. for all the locations taken into consideration.

Table 1. Cab ride dataset



	distance	cab_type	time_stamp	destination	source	price	surge_multiplier	id	product_id	name
0	0.44	Lyft	1544952607890	North Station	Haymarket Square	5.0	1.0	424553bb-7174-41ea-aeb4-fe06d4f4b9d7	lyft_line	Shared
1	0.44	Lyft	1543284023677	North Station	Haymarket Square	11.0	1.0	4bd23055-6827-41c6-b23b-3c491f24e74d	lyft_premier	Lux
2	0.44	Lyft	1543366822198	North Station	Haymarket Square	7.0	1.0	981a3613-77af-4620-a42a-0c0866077d1e	lyft	Lyft

Table 2. Weather dataset

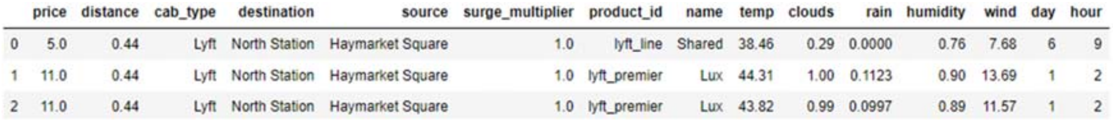


	temp	location	clouds	pressure	rain	time_stamp	humidity	wind
0	42.42	Back Bay	1.0	1012.14	0.1228	1545003901	0.77	11.25
1	42.43	Beacon Hill	1.0	1012.15	0.1846	1545003901	0.76	11.32
2	42.50	Boston University	1.0	1012.15	0.1089	1545003901	0.76	11.07

In cab ride dataset, there are 10 colums and 693071 rows; in the weather dataset, there are 8 columes and 6276 rows. The types of features contain continuous numeric, drecrete numeric and category features.

To predict cab price conprehensively, we need to combine the cab ride dataset and weather dataset. Firsly, we chance the “time_stamp” into “date_time” by function, “pd.to_datetime". Secondly, we create a new feature called “merge_date” to refelect same time for a location in both cab ride dataset and weather dataset. Thridly, we merge those two dataset based on the “merge_date” and create a new dataset which contains all the important features we need. There are 15 columes and 1164996 rows.

Table 3. Merged dataset



	price	distance	cab_type	destination	source	surge_multiplier	product_id	name	temp	clouds	rain	humidity	wind	day	hour
0	5.0	0.44	Lyft	North Station	Haymarket Square	1.0	lyft_line	Shared	38.46	0.29	0.0000	0.76	7.68	6	9
1	11.0	0.44	Lyft	North Station	Haymarket Square	1.0	lyft_premier	Lux	44.31	1.00	0.1123	0.90	13.69	1	2
2	11.0	0.44	Lyft	North Station	Haymarket Square	1.0	lyft_premier	Lux	43.82	0.99	0.0997	0.89	11.57	1	2

The data in new dataset is not clean and there are many missing values. For the rows without “price”, we drop those rows because our goal is to predict the cab price. For the 2964 rows without temperature, clouds, humidity and wind value, we also drop them because their proportion in the total 1164996 are really small. Finally, for the 1061692 rows without rain value, we fill the missing values into 0 to assume there are no raining.

V. Data Analysis

Based on our data visualization tool, we have preliminary summary statistics. Figure 1 is the pie chart of destination, which shows the proportion of demand in different location where passengers want to go. We can find the most popular destination is Financial District. Figure 2 is the pie chart of source, which can tell us the top 3 locations with highest demand are Financial, Northeastern University and Beacon Hill.

Figure 1. Pie chart of destination

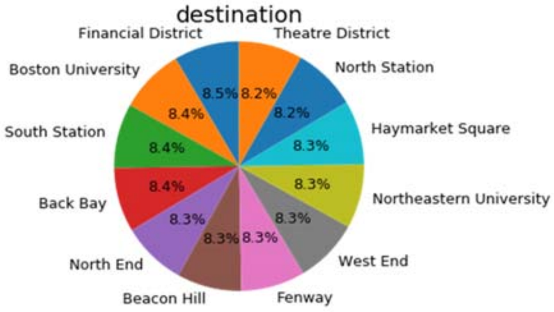
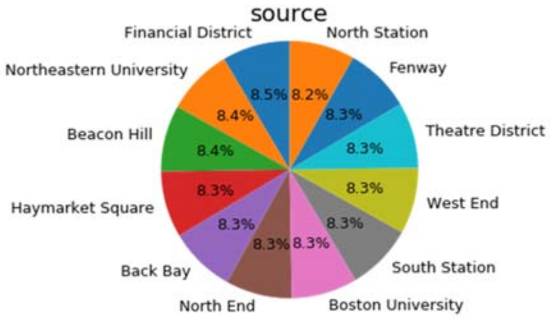


Figure 2. Pie chart of source



As figure 3 and 4 display, the proportion of hour in a day and day in a week indicates the time with higher cab demand. We can see that from 23:00 to 6:00 on Wednesday, Tuesday and Monday, there are more people need a cab ride.

Figure 3. Pie chart of hour in a day

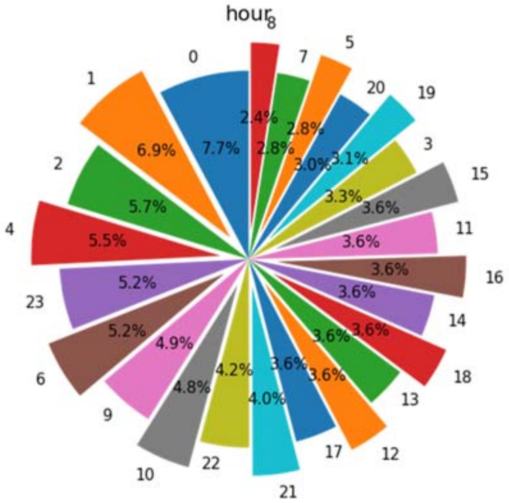
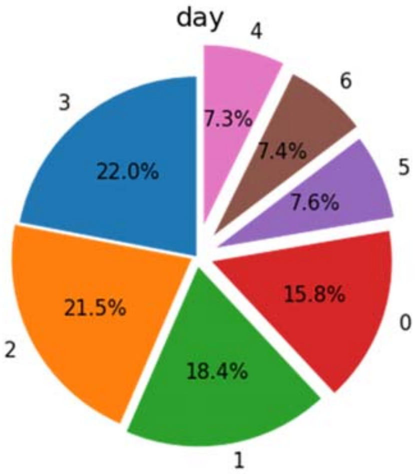


Figure 4. Pie chart of day in a week



As we all know, when we book a ride on Uber or Lyft, the different car type may cause the different cab price. As a result, we employ box plot to figure out the relationship between car type and cab price. Figure 5 and figure 6 shows the range of price in different car type. In Uber, Black SUV has the highest middle cab price; and Lux Black XL is the highest one in Lyft. Figure 7 and 8 are scatter plot, which shows the relationship between distance and cab price with different car types. Based on these two figures, we also find that both the interception and slope of distance vs. price are different in different car type.

Figure 5. Box plot of car type vs. price in Uber

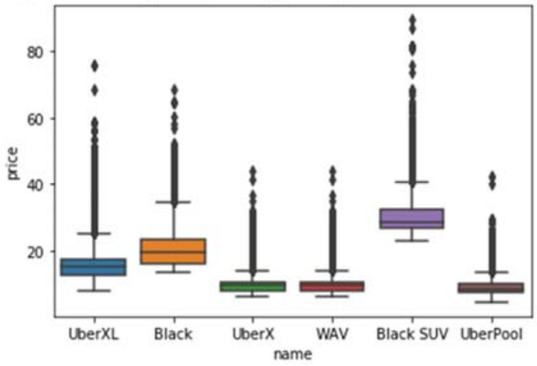


Figure 6. Box plot of car type vs. price in Lyft

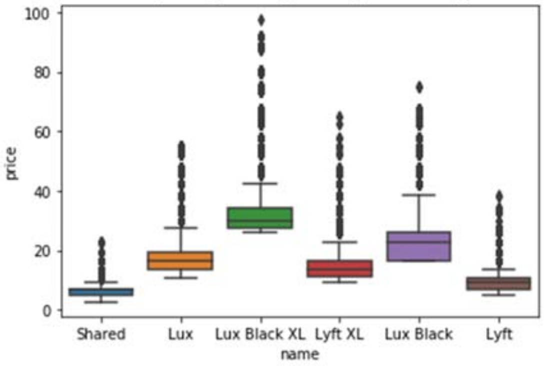


Figure 7. Scatter plot of distance vs. price in Uber

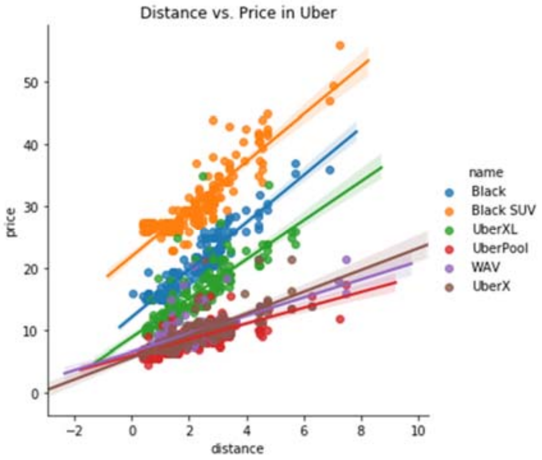
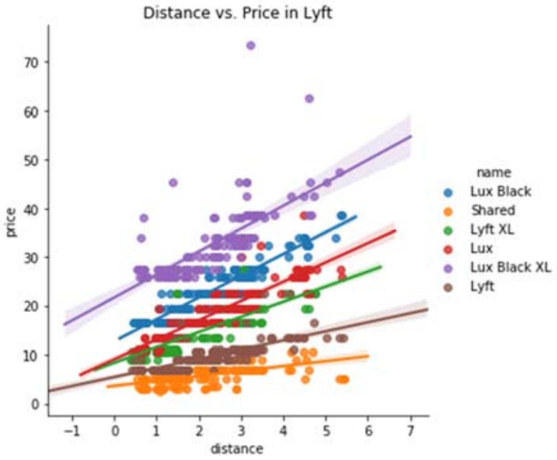
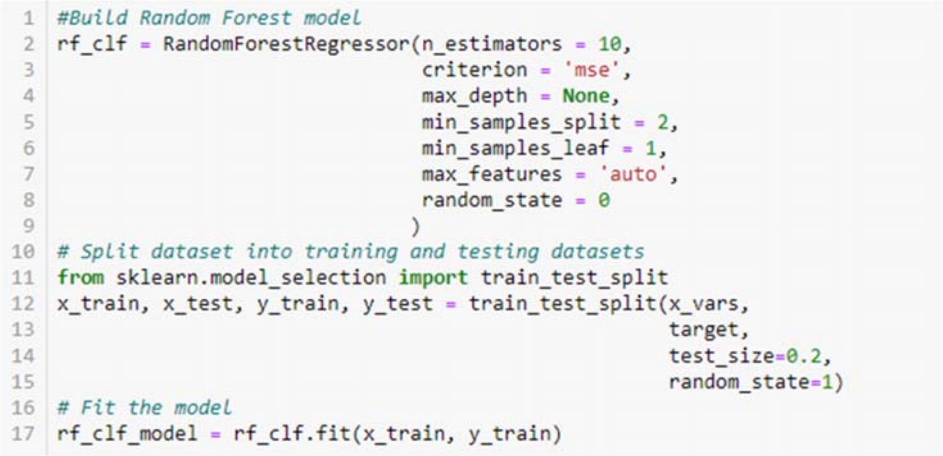


Figure 8. Scatter plot of distance vs. price in Lyft



The model we build to predict cab price is random forest. Because price is continuous numerical feature, this is a regression problem and we employ the package “RandomForestRegressor” from “sklearn.ensemble”. The build-in criterion we use in RandomForestRegressor is mean square error (MSE). Then we split the original dataset into two sets: training set and testing set; and fit the model by training dataset. Finally, if we want to forecast the cab price, we can use the build-in function “.predict” to get the predicted value. Besides, we also separate the original dataset into Uber dataset and Lyft dataset based on the feature “cab_type” because different company may have different business strategies. As a result, it may impact on the feature importance.

Figure 9. Random forest model codes



```python
#Build Random Forest model
rf_clf = RandomForestRegressor(n_estimators = 10,
                               criterion = 'mse',
                               max_depth = None,
                               min_samples_split = 2,
                               min_samples_leaf = 1,
                               max_features = 'auto',
                               random_state = 0
                              )
# Split dataset into training and testing datasets
from sklearn.model_selection import train_test_split
x_train, x_test, y_train, y_test = train_test_split(x_vars,
                                                    target,
                                                    test_size=0.2,
                                                    random_state=1)
# Fit the model
rf_clf_model = rf_clf.fit(x_train, y_train)
```

VI. Evaluation

In our project, we employ R^2 to evaluate the model performance. The range of R^2 is from 0 to 1. The higher the R^2, the better the model performs. Table 4 shows the R^2 of the original dataset, Uber Dataset and Lyft dataset. We can clearly find that the overall-dataset model, whose training R^2 is 0.989 and testing R^2 is 0.969, performs well and there is no overfitting or underfitting. The R^2 of Uber model is a little better than overall cab type and Lyft’s model does not perform well and a little overfits.

Table 4. R2 for different datasets

Dataset	All Cab Type	Uber	Lyft
Training R^2	0.989	0.997	0.979
Testing R^2	0.969	0.985	0.946

Cross-validation is used to reduce the overfitting of the prediction of model. In our project, since all of these three models perform well and do not overfit largely, we do not plan to employ cross-validation. Besides, splitting original data into training and testing data can help evaluate model fitting.

Apart from predicting cab price, random forest can also show us the feature importance in given dataset, which is displayed on figure 10 and 11. We can find that, for Lyft, the top 3 important feature are name (car type), distance and surge_multiplier; for Uber, the top 3 are name, distance and temperature.

Figure 10. Feature importance in Uber

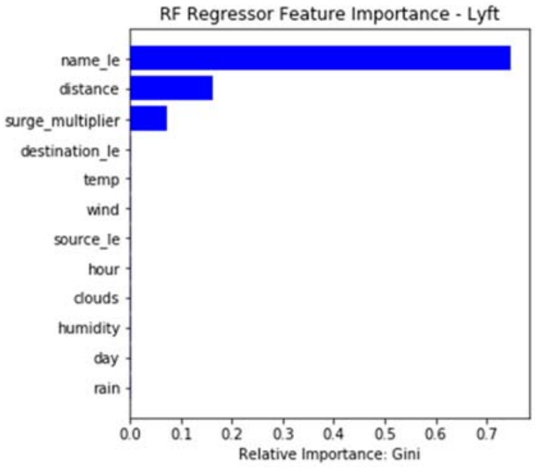
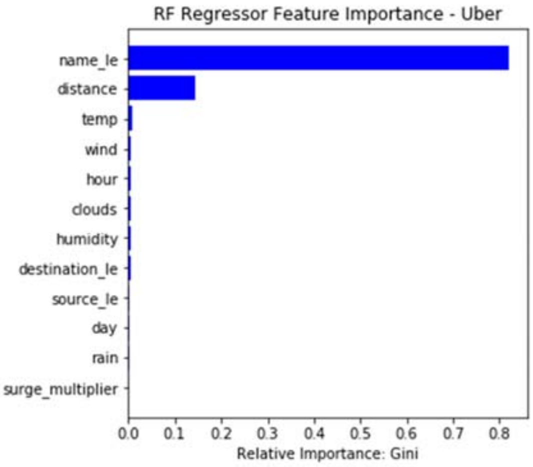


Figure 11. Feature importance in Lyf



VII. Discussion and Conclusions

In our project, based on the visualization tool and prediction model, we can draw the conclusion that:

1. Employing random forest regression model, we can accurately predict the cab price based on given conditions.

2. Based on the random forest model, the most important features for cab price are: car type, distance, surge_multiplier.

3. For the drivers in Boston, there is a higher chance of receiving an order on Financial District, Northeastern University and Beason Hill from 23:00 to 6:00 on Wednesday, Tuesday and Monday.

4. For Uber passengers, UberX, WAV and UberPool car type are cheaper; for Lyft passengers, shared, Lyft and Lyft XL are cheaper.

We can say with more confidence that our results are positive because we not only accurately predict the cab price by building random forest model on given data, but also find the reasonable important features that can give suggestions to both drivers and passengers. Although we encounter several problems during our project, we do our best to solve them. For example, when we build the model to predict cab price, we need a complete dataset but not two separate datasets. Then to solve this issue, we find the common part of these two datasets and merge them together.

However, we believe this is not the end our project and we can still improve our project. First thing is to accurate the demand on certain time and location and combine them together. For example, what time of the week is cab demand higher? The second one is the analysis on the influence of weather on cab price. In our understanding, taxis are more expensive when it rains; however, in the feature importance part, it does not show this trend. As a result, we want to improve our model and data processing and then find their internal relationship.

Reference:

[1] RaviMunde. Uber & Lyft Cab prices, Kaggle.

https://www.kaggle.com/ravi72munde/uber-lyft-cab-prices#cab_rides.csv