Passenger Saver based on Uber Surge Multiplier

Peng Zheng, Jinli Yan, Xipeng Chai, Haomin Lin, Yanjun Ding, Yuntian Zhang

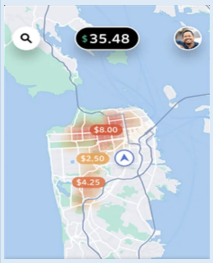
Georgia Institute of Technology

Introduction

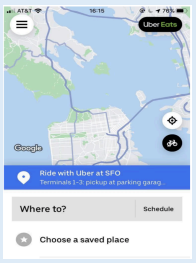
Uber is a company that is famous for managing a ride-sharing platform. One of the most important characteristic of Uber is flexibility of supply. During the times of high demand, uber will increase the price to attract more drivers to satisfy passenger. This dynamic pricing is called "surge multiplier". It can be affected by time, distance and region.

The problem is : The surge multiplier is only visible for drivers, so passengers can be charged a higher price unconscious.

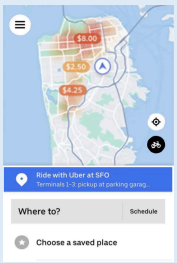
We will use data from past give passenger a guide to move to the nearest region with the lowest surge Multiplier. There is no Apps provide these information to passenger, that why we are unique. Our project will help customers save money and reduce traffic pressure.



Only Driver Knows



Customers know **nothing**



Customers **also** knows

Data

Data Set:

Due to the close of Uber's public API, we choose to use the data set "Uber and Lyft Dataset Boston,MA" from Kaggle. This dataset included more than 700,000 records.

Data set → ***Basic information***: timestamp, source, destination, cab type , price, distance etc

Data set → ***Weather*** : temperature, visibility etc

Approach

First : Clean data sets:

- Defined and Calculated Surge multiplier (Databricks with Scala)
- Eliminating Null values ,convent timestamp to real time (Python Pandas)

Different products have different launch price and there are fluctuations of price for the same type of products. In order to remove the affect of these factors, our multiplier is defined as : $s_i = \frac{p_i}{\sum_{j=1}^{n} p_j/n}$ (Pi is the price of ride i and the denominator is the average price categorized by source-destination and product type).

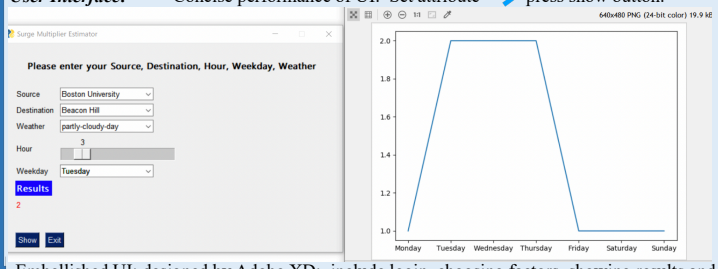
Second : build model

In order to predicting surge multiplier under different circumstances , we decided to use a machine learning with random forests as model. （Software: Scikit-learn)

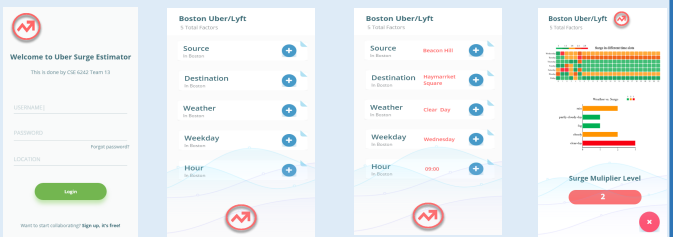
For training and Prediction: we choose 6 attributes from 24 columns (arranged by significant impact) . The attributes are {source , destination ,distance, weather summary , weekday ,hour}.

With a proper model ,we predicted the multiplier based on self-constructed attributes set.

User Interface: Concise performance of UI: Set attribute → press show button.



Embellished UI: designed by Adobe XD: include login ,choosing factors, showing results and returning results.



Experiment and Results

Testbed :

Applying different algorithm and parameters , the compare their final accuracy we decide how the program can give the best possible outcome finally.

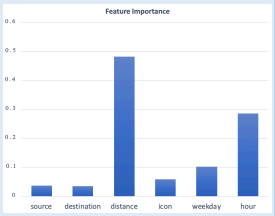
In order to verify our result, we divide the dataset into train dataset(80%) and test dataset(20%). This division is for testing the accuracy in different dimension.

Choose Experiment Model:

Random Forest
Bayesian Regression
Support Vector Machine
→ Compare the accuracy → Random Forest

Build model:

All Attributes → Build model → Check the importance score → Choose 6 most important attributes → Build model → Grid search improved → Cross validation → Find best model

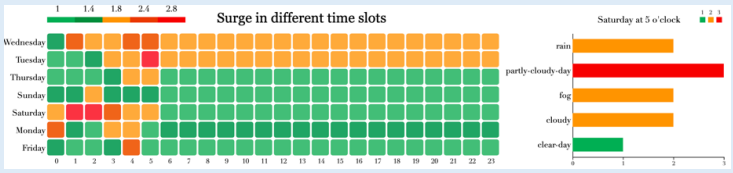


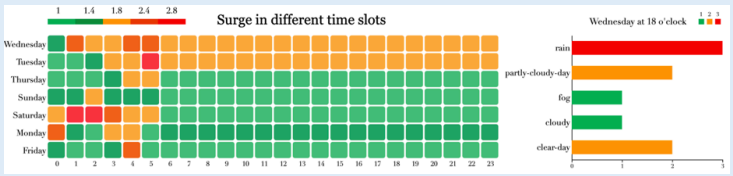
Experiment Visualization

After we finished tuning, we create a csv file that consists of all possible combinations of "destination, source, distance, weekly, hour and weather exists in the dataset. Then use the ***scikit-learn*** with our derived random forest algorithm to predict the surge multiplier of each combination.

Calculate the average result for each case and convert the final results to three distinct values"1","2","3". Using this result, we create a new table that contains all the result.

Using d3.js we implement a heatmap with interactive bar charts.





For example, at 18 o'clock on Wednesday. The rainy day makes surge go up because more people need a ride. However, fog days and cloudy days lead to lower surge multipliers because these weather make people less want to go out.

With several accountable cases, we can accept that out model is reasonable and produce right results.

Conclusion

Our goal in this project is to make Surge Multipliers accessible to passengers by estimating based on old data from Uber and Lyft. Due to complexities and impurities of dataset, we contributed a lot to data cleaning. Then we trained our dataset to achieve a high-accuracy prediction model. Finally, with chosen parameters ,our model achieves an accuracy of approximately 75% (80% for training ,20% for testing). What 's more, we also commit to design interactive visualization (heatmap) based on D3.js, and a User Interface. Conclusively, passengers can receive an estimated level of surges, interactive user interface as well as visualization based on our project.

Reference

Niels Agatz, Alan Erera, Martin Savelsbergh, and Xing Wang. 2012. Optimization for dynamic ride-sharing: A review. European Journal of Operational Research 223, 2 (2012), 295 – 303. https://doi.org/10.1016/j.ejor. 2012.05.028

Sotiris Brakatsoulas, Dieter Pfoser, Randall Salas, and Carola Wenk. 2005. On Map-matching Vehicle Tracking Data. In Proceedings of the 31st International Conference on Very Large Data Bases (VLDB '05). VLDB Endowment, 853–864. http://dl.acm.org/citation.cfm?id=1083592.1083691

Le Chen, Alan Mislove, and Christo Wilson. 2015. Peeking Beneath the Hood of Uber. In Proceedings of the 2015 Internet Measurement Conference (IMC '15). ACM, New York, NY, USA, 495–508. https://doi.org/10.1145/2815675.2815681