Holly Cohan,Yesh Onipede, Yu Xia

**Week 1 Report: Identify the Problem Statement and Dataset**

## Problem Statement

We aim to develop a predictive model that accurately forecasts rideshare prices in Boston during the months of November and December, utilizing a range of key features such as time of day, day of the week, source, destination, distance, temperature, and other weather-related features. Our focus is on predicting the cost of Uber and Lyft rides by analyzing how these situational and environmental factors influence pricing. The dataset, sourced from Kaggle, contains about 690,000 rows of data from November and December 2018. The project will eventually focus on approximately 100,000 observations from the original dataset. This subset will be identified either by focusing on a single company or ride type, or through random or stratified sampling.

Given that the data only spans two months, our model's predictions will be limited to this specific timeframe. While this may prevent generalization across other times of the year, the immediate goal is to build a model that performs well on this dataset and uncovers key patterns and insights related to rideshare pricing during this period.

## Articulation of value

The models we are building could hold significant value for both rideshare companies and their customers. By accurately predicting rideshare prices, companies can optimize their dynamic pricing strategies. Understanding how external factors such as weather conditions influence demand can help companies adjust their prices in real time, ensuring both maximal revenue and customer satisfaction.

The insights gained from this analysis could help businesses better anticipate demand surges during adverse weather conditions, allowing them to allocate resources efficiently. The model could offer more transparent pricing predictions for consumers, helping them make informed decisions about when to book rides. Overall, we hope to provide both business insights and operational improvements for the rideshare industry.

# Calculation of the potential economic value

## Assumptions

- Uber completed 2.57 billion rides through Q1 in 2024 [1]
- There is a 75% increase in requested rides completed during surge periods (positive hit) [2]
- There is a 140% decrease in rides requested due to the presence of a surge (negative hit) [3]
- Surge for positive hit example = 3x [4]
- Surge for negative hit example = 1.4x[5]
- 10% of rides are affected by Uber's dynamic ("surge pricing")

## Value calculation

*Impact of positive hit - Impact of negative hit*

**Impact of positive hit** = % of time a positive hit occurs * surge multiplier for positive hit example * impact of positive hit * total number of rides

**Impact of positive hit** = 0.1 * 3 * 0.75 * 2.57B = **578.25 million**

---

[1] According to Uber Statistics: How Many People Ride With Uber (https://backlinko.com/uber-users). There was no information available regarding the number of rides requested and therefore we will use the number of rides completed by Uber in 2024 Q1.

[2] This was calculated from a figure in a case study related to surge pricing titled "The Effect of Uber's Surge Pricing: A Case Study. This case study looked into a situation where demand for rides was high as it was New Year's Eve but surge pricing was not in effect due to a technical glitch. The metric requested rides completed was calculated from subtracting (% of requested rides completed during the surge outage - % of rides completed at the final time point prior to the surge outage).
(https://economicsforlife.ca/wp-content/uploads/2015/10/effects_of_ubers_surge_pricing.pdf)

[3] This was calculated from a figure from the above case study that looked into the supply and demand for Uber rides following a sold out concert in 2015 when a surge period was taking place post-concert. Figure 3 in the case study plotted the percentage increase over pre-surge baseline for ride requests, users opening the app, and driver supply before and after the surge period. The negative hit quantifies the % of users who were interested in booking a ride during the surge period (as indicated by opening the Uber app) - the % of users who were actually booked an Uber ride. The idea is that these rides were "lost" as a result of the surge pricing during this period.

[4] The case study specified for the surge outage New Years Eve example from which we are calculating our positive hit from, that the surge price multiplier fluctuated from 1-6x once the surge outage was over. We will use 3x as the surge multiplier for this example as it is the midpoint between the minimum and maximum surge multiplier values.

[5] The case study specified for the post-concert surge example from which are our negative hits are from, that the surge multiplier fluctuated from 1x to 1.8x the pre-surge prices. There we will use 1.4x as the surge multiplier as the average surge hit multiplier for our negative hit example.

**Impact of negative hit** = % of time a negative hit occurs * surge multiplier for negative hit example * impact of negative hit* total number of rides

**Impact of negative hit =** 0.1 * 1.4 * 1.4 * 2.57B = **504.92 million**

*Impact of positive hit - Impact of negative hit* = **578.25 million - 504.92 million = 73.33 million**

***% increase in rides* =** (Increase in rides / Total number of rides ) * 100 = (73.33 million / 2.57 B) * 100 = 2.85%

Dynamic pricing ("surge pricing") will result in the addition of **73.33 or 2.85%** more rides completed

## Project plan

Build a 13-week plan. Identify the steps, identify the weeks, and what you will do in each step (you can look at the syllabus to build this plan)

| | Steps | Deliverables |
|---|---|---|
| Week 1 | - Ingest data into JupyterHub<br>- Identify the target variable (Price)<br>- Identify the feature variable we are going to use through data visualization and correlation analysis<br>- Explore the variables in the dataset | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 2 | - Split dataset into training, validation and test dataset and save them as parquet in the `savedData` folder under the `data` folder<br>- Perform EDA on the datasets<br>- Identify problem, opportunities and pre-processing needed | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 3 | - Conduct data preprocessing based on findings from Week 2 and any new findings from Week 3 | - Written Report<br>- Jupyter Notebook<br>- Code Output |

| | | |
|---|---|---|
| | - Perform preprocessing across training, validation and test datasets and store them in the `savedData` folder under the `data` folder<br>- Build a simple linear regression model using all features using the training dataset | - Merge all artifacts into GitHub |
| Week 4 | - Create and engineer new features if needed<br>- Augment your dataset with new data if needed<br>- Reduce dimensions of your dataset if needed through PCA or other more suitable methods<br>- Create final training, validation and test datasets that will be used in modeling, evaluation and testing | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 5 | - Build a very simple set of models appropriate<br>  - Linear Regression<br>  - Decision Tree<br>- Tune hyper-parameters of the model<br>- Save the model in the `models` folder under the `data` folder<br>- Select model evaluation metrics, evaluate variations and pick the winning model<br>- Analyze how each feature contribute to Uber/Lyft's pricing model | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 6 | - Build a more complex set of models<br>  - Regularization<br>  - Random Forest<br>- Tune hyper-parameters of the model<br>- Save the model in the `models` folder under the `data` folder<br>- Select model evaluation metrics, evaluate variations and pick the winning model | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 7 | - Build another set of models appropriate for you problem and dataset<br>  - XG Boosting<br>- Tune hyper-parameters of the model<br>- Save the model in the `models` folder under the `data` folder<br>- Select model evaluation metrics, | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |

| | | |
|---|---|---|
| | evaluate variations and pick the winning model | |
| Week 8 | - Evaluate multiple models<br>- Select the best model<br>- Retrain the model on the training and evaluation set<br>- Calculate performance on the test dataset | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 9 | - Improve your model by improving the data | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 10 | - Explain the model by understanding feature importance and prediction outcomes<br>- Identify model risks<br>- Identify and quantify bias in your input dataset and model output<br>- Identify and measure bias | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 11 | - Save and deploy your model<br>- Build a monitoring plan | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 12 | - To put all weeks together and develop an end to end modeling project with all the artifacts | - Written Report<br>- Jupyter Notebook<br>- Code Output<br>- Merge all artifacts into GitHub |
| Week 13 | - Review a peer's end to end modeling work and related artifacts<br>- Provide written feedback to a peer's work and artifacts | - Go through the other person's artifacts – no deliverable<br>- Written peer feedback report<br>- |

## Discuss the dataset

The dataset consists of 693,071 observations and 57 variables, with the target variable being the price of a rideshare trip. It includes 56 features that cover various aspects of the ride, and we found the dataset through Kaggle.

### Key Features:

1. **Time Variables:**
    - Date and time of the ride request.
    - Day of the week (Weekday/Weekend).
    - Time of day (morning, afternoon, evening, night).
2. **Location Variables:**
    - Source location.
    - Destination location.
    - Distance between source and destination.
3. **Weather Variables:**
    - Temperature.
    - Precipitation.
    - Weather conditions (Clear, Rain, Snow, etc.).
4. **Cab Type:**
    - **Uber** or **Lyft**.
5. **Product Type**
    - Product_id: UberXL, etc

## Identify the type of modeling

The modeling approach for this dataset falls under supervised learning, specifically a regression problem, as the goal is to predict a continuous target variable—price. Several models can be applied to solve this problem. Linear regression serves as a straightforward method for predicting price based on the features. Regularization techniques, such as Ridge or Lasso, can improve the model by addressing overfitting. Tree-based methods, including Decision Trees, Random Forests, and XGBoost, are also well-suited for this task, as they can capture complex relationships between the features and the target variable, often leading to more accurate predictions.