Predicting Customer Retention at Ultimate Technologies Inc.

Problem Statement

Ultimate Technologies Inc. is a highly rated transportation network company. The focus of this project was to use exploratory data analysis to identify trends and daily cycles in user logins, and modeling to predict rider retention.

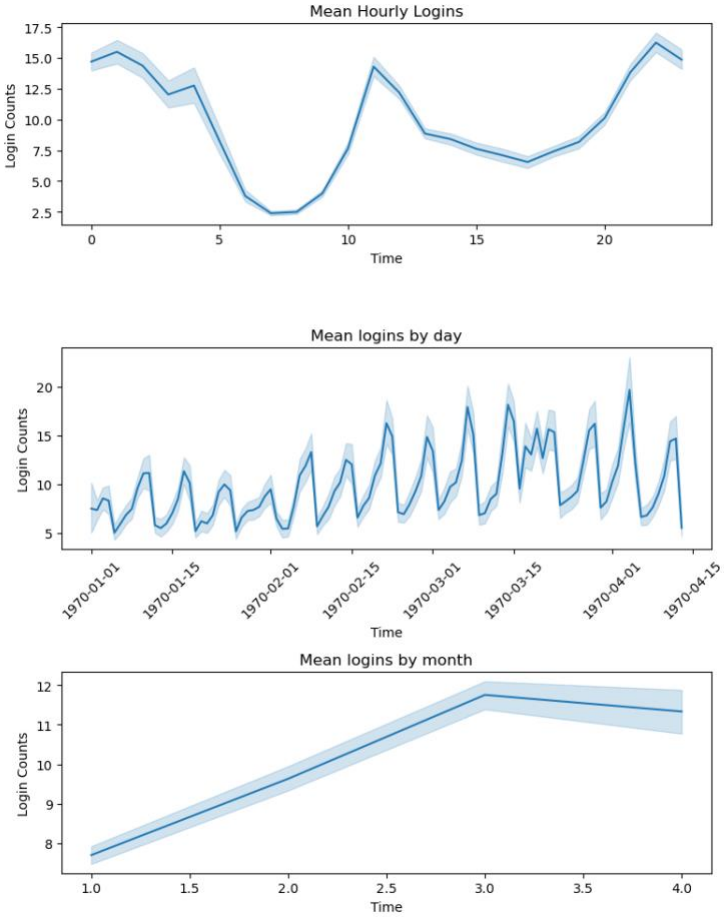
Part 1- EDA and Summary of Daily Cycles of Demand

There are several patterns that emerge when looking at the usage patterns for riders using Ultimate Technologies Inc. A summary of the key findings is below:

1. **Demand increases during non-business hours and on weekends.**
   The boxplots below illustrate that both the mean number of logins are higher on the weekends, and during non-business hours. This pattern is also evident when looking at the hourly patterns of daily usage.

2. **Demand increases during the lunch hour as well as during non-business hours.**
   There is a clear daily cycle to the number of user logins. There is a clear spike in demand during the lunch hour (around 11am-1pm) as well as during non-work hours.

3. **Demand increases across months.**
   There is an increase in demand from January through April, with an apparent peak in March. This pattern is most apparent when looking at mean values (rather than total logins). This is an interesting pattern, indicating that demand may increase during warmer months, although this depends on where Ultimate Tech transportation company is located.



Part 2- Experiment and metrics design

Problem Statement:
The neighboring cities of Gotham and Metropolis have complementary circadian rhythms: on weekdays, Ultimate Gotham is most active at night, and Ultimate Metropolis is most active during the day. On weekends, there is reasonable activity in both cities. However, a toll bridge,

with a two-way toll, between the two cities causes driver partners to tend to be exclusive to each city. The Ultimate managers of city operations for the two cities have proposed an experiment to encourage driver partners to be available in both cities, by reimbursing all toll costs.

1. **What would you choose as the key measure of success of this experiment in encouraging driver partners to serve both cities, and why would you choose this metric?**

I would choose the number of toll bridge reimbursement requests (as a proxy for number of bridge crossings) as a metric for determining success for this experiment. Increased toll reimbursement requests indicates that drivers are using the toll bridge rather than remaining exclusively in one city. Therefore, a greater number of reimbursement requests indicates greater movement of driver partners between cities.

Since both cities have a decent level of activity on weekends, driver partners might choose to remain in one city for the weekend. Therefore, choosing a period of time for the experiment that averages out the weekend effect would be beneficial -- namely, tracking reimbursement requests across at least several weeks or months.

2. **Describe a practical experiment you would design to compare the effectiveness of the proposed change in relation to the key measure of success.**

To compare the effectiveness of the proposed change -- namely, reimbursing all toll costs in order to encourage drivers to be available in both cities -- we would need to set up an experiment that allows us to compare driver availability in both cities both before and after the proposed change goes into effect.

To this end, we need a way to track movement between cities before the proposed change goes into effect. Presumably, we have access to driver license plates and could pull data on toll bridge usage for the months before the toll reimbursements initiative goes into effect. If we did not have access to this data, or to data showing which drivers picked up customers in each city, we could set up an experimental period of several months to track toll bridge usage before reimbursements are available. We would then track toll bridge usage for the same period of time after the reimbursements went into effect. Ideally, this time period would be at least several months, to give drivers time to get used to the new policy and change their patterns of behavior. For example, we could collect data for the 2 months prior to the proposed change, then allow 2-4 weeks to pass after the change before collecting another 2 months of data.

We would then conduct a two-tailed t-test on the total number of bridge crossings from before and after the change to determine if there was a significant increase ($p < 0.05$) in toll bridge crossings after the proposed change went into effect. This is a good test for determining whether there is a significant difference between two groups of data.

If the results showed a significant increase in bridge crossings after the proposed change went into effect, this would indicate that the proposed change was successful. Depending on the magnitude of the effect -- how much of a change between before and after -- we might need to suggest additional incentives (a monthly bonus?) to encourage drivers to move between cities. Even if there were an increase in bridge crossings, it would be important to also examine the overall number of rides given per driver, to see whether this new policy was actually having an impact on driver availability for customers. If we did not see any change in the number of rides drivers provided per day, then we would need to reexamine our assumption that encouraging movement between cities is beneficial to customers.

Part 3 - Predictive modeling

Problem Statement

Ultimate Technologies is interested in predicting rider retention. Using four months of data, the goal was to build a model to predict what features retain riders.

Data Wrangling and EDA

The dataset provided by Ultimate Technologies contains the following metrics:

- **city:** city this user signed up in
- **phone:** primary device for this user
- **signup_date:** date of account registration; in the form 'YYYYMMDD'
- **last_trip_date:** the last time this user completed a trip; in the form 'YYYYMMDD'
- **avg_dist:** the average distance in miles per trip taken in the first 30 days after signup
- **avg_rating_by_driver:** the rider's average rating over all of their trips
- **avg_rating_of_driver:** the rider's average rating of their drivers over all of their trips
- **surge_pct:** the percent of trips taken with surge multiplier > 1
- **avg_surge:** The average surge multiplier over all of this user's trips
- **trips_in_first_30_days:** the number of trips this user took in the first 30 days after signing up
- **ultimate_black_user:** TRUE if the user took an Ultimate Black in their first 30 days; FALSE otherwise
- **weekday_pct:** the percent of the user's trips occurring during a weekday

Data Quality and Missing Values

The data quality was good – there were no issues with out-of-range or missing values, and only slight class imbalances for categorical data.

There were some missing values in several features that I explored. Since the total percent of missing values in the data was only 1.3%, I decided drop these values. This decision was based on the fact that there were very few missing values (percentage-wise), and that the features that were missing values were ones that would be hard to impute/interpolate without potentially introducing bias into the data. The most concerning of these was the missing values in the 'avg_rating_of_driver' column, representing 16.2% of that column's data. However, there

is no good way to interpolate or infer these missing values since it represents a rating by the customer, and any assumptions on interpolating this value could skew the results. Therefore, I think the best decision is to drop these rows since we still have a good amount of data even without them. A caveat to this decision is that there may be some bias in people who don't leave a review -- perhaps if they had a very average experience they might not be inspired to leave any review. This is something to consider when analyzing the overall efficacy of the model.

Feature Engineering and Addressing Outliers and Kurtosis

There appears to be quite a lot of skew (some negative and some positive) in all of the numeric variables apart from weekday_pct. I used a log transform (with +1 offset) to address the skew. There are many choices for transformations – the main concern here is dealing with the skew and negative values created by a log transformation of 0, which is addressed by using log transform with an offset.

The only major collinearities between features in the data were between surge_pct and avg_surge, which makes sense. I kept both of these columns, but might consider dropping one later on to improve model fit. I performed minimal other feature engineering, although this could be an area to revisit to improve model performance.

Creating the Predictor Variable

A rider was considered “retained” if they logged in within the last 30 days prior to pulling the data. There was a slight imbalance in class data for the predictor variable (total percent of users retained was 41.10%). This can be addressed later if necessary using resampling -- oversampling (of the minority class) or under-sampling (of the majority class).

Predictive Modeling

The final dataset for modeling included 10 features and a binary predictor variable (user retained or not retained). I used ColumnTransformer to create a pipeline that performed one-hot-encoding on categorical variables and log transformation on numeric variables. This is a binary classification problem; I tested four classification models first to see how they performed.

1. Logistic regression
2. Random forest
3. Gradient boosting
4. SVM

I assessed model accuracy, as well as precision and recall to assess how well the model performed. Of the four models tested, the gradient boosting classifier performed best in all metrics, with the random forest classifier a close second.

After identifying the gradient boosting classifier model as the best model for the data, I performed hyperparameter tuning with 5-fold cross-validation on several model parameters

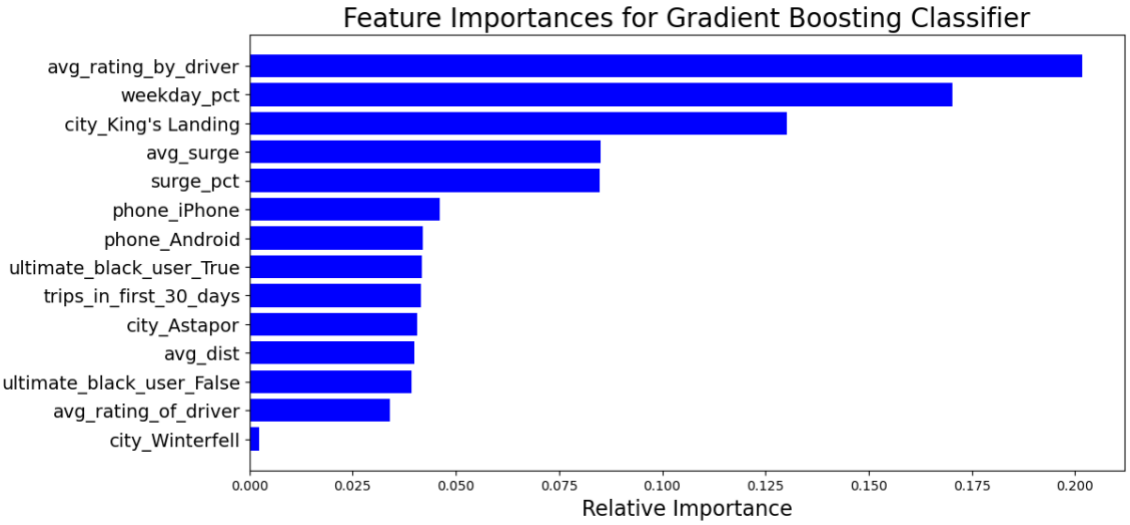
(n_estimators, max_depth, max_features, learning_rate) and then fit the final model using the best hyperparameters.

Summary of Key Findings

The final model performed fairly well (78% accuracy), although there is definitely room for improvement. Recall (true positive rate) was only 69% for the positive class, while precision (the accuracy of positive predictions made by the model) was 76%.

Key findings from the model:

1. **Average rating by driver was the most important predictor of rider retention.**
   This is interesting, since it indicates that how well the driver rated the rider was more important than the rider's perceived experience of the driver. Perhaps polite, happy customers lead to satisfied drivers and also satisfied customers! This indicates that the company should work on methods/promotions that will keep the drivers happy.
2. **The percent of the user's trips occurring during a weekday (weekday percent) was also a strong predictor of rider retention.**
   Users who used the service during weekdays were more likely to be retained. This could indicate that people who rely on using the taxi service for commuting purposes are more reliable customers for long-term use. The company should therefore target common commuting routes and demographics that might rely on taxis for commuting.
3. **Riders in King's Landing were more likely to be retained than riders in other cities**.
   Users who signed up in King's Landing (only ~21% of all users) were more likely to be retained than in the other two cities. This could lead to targeted campaigns to increase retention in other cities, or alternately increasing taxi availability in King's Landing where retention is already high. An analysis of why this difference between cities exists would also be important.



Discussion of Recall vs Precision

Deciding whether to prioritize precision or recall depends on Ultimate Technologies' business goals. For this model, since we are interested in overall user retention, precision is likely the most important metric – we want to be able to accurately identify retained users. High precision is important when false positives are costly – in this scenario, predicting a user will be retained when they were actually not retained. Too many false positives would lead to a model that might over-estimate user retention and lead to poor business decisions. With a high precision model, the taxi service can confidently target users for retention efforts without wasting resources on users who might not actually be retained.

Another possible direction the company could take is prioritizing high recall, and launching targeted promotions or incentives for users who are not retained, trying to entice them back to the company.

	precision	recall	f1-score
Not Retained	0.79	0.84	0.82
Retained	0.76	0.69	0.72
accuracy			0.78
macro avg	0.77	0.77	0.77
weighted avg	0.78	0.78	0.78