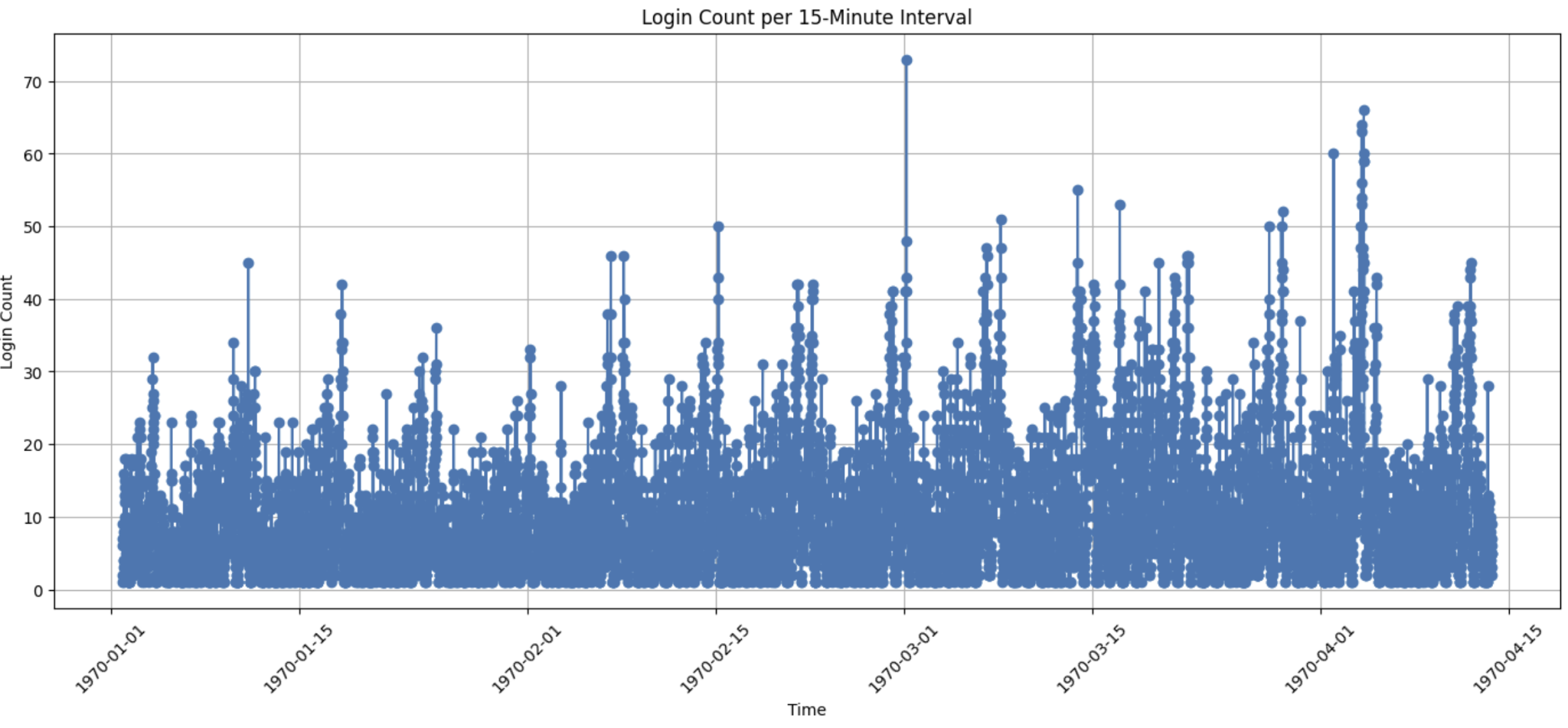
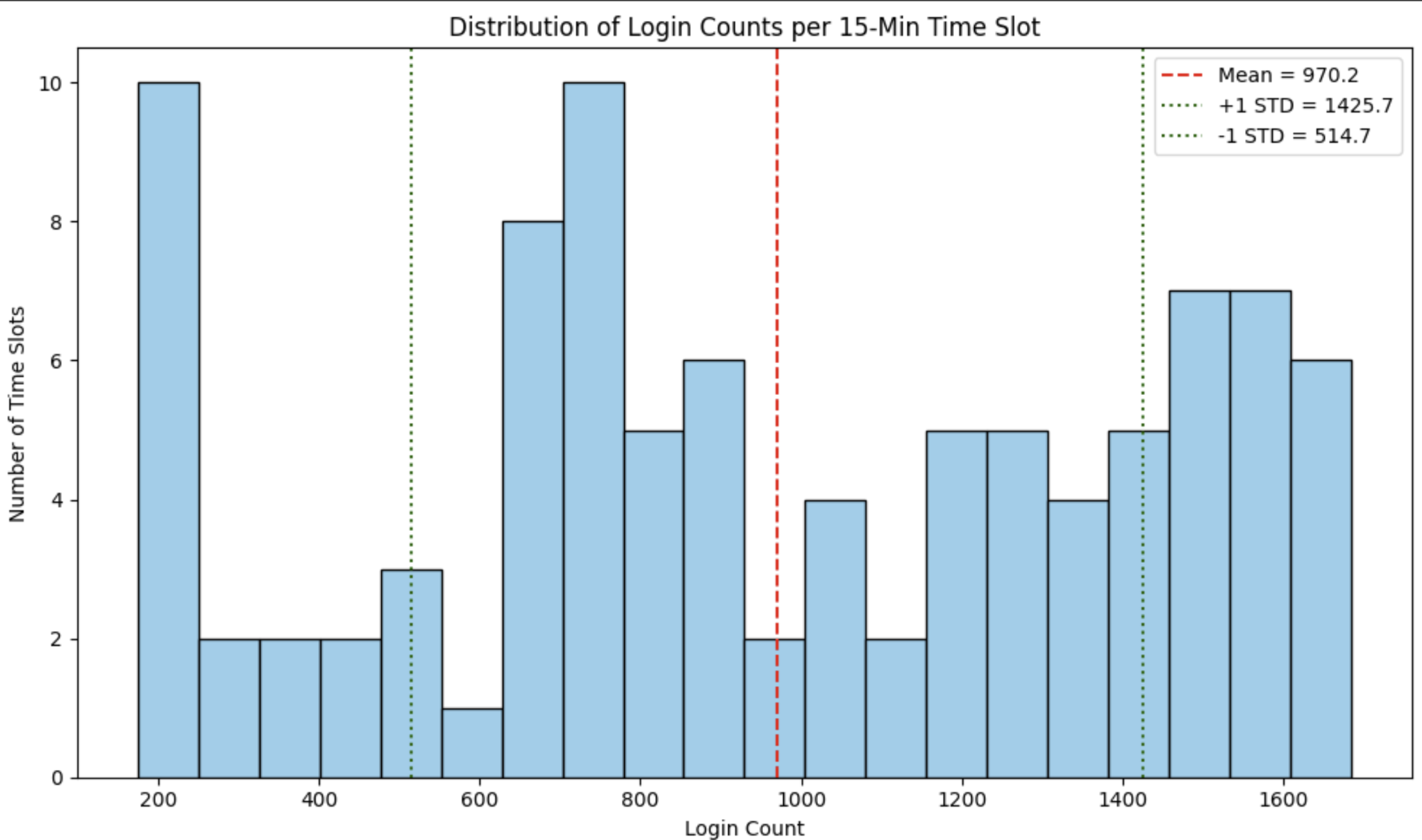
The Google Colab Notebook for Part 1 and Part 2 is [here](https://github.com/BTExpress1/ultimate-tech/blob/2a9a82bb3e32c53c85f8913bc25cd02e78aaabbe/notebooks/ultimate_tech.ipynb).

# Part 1

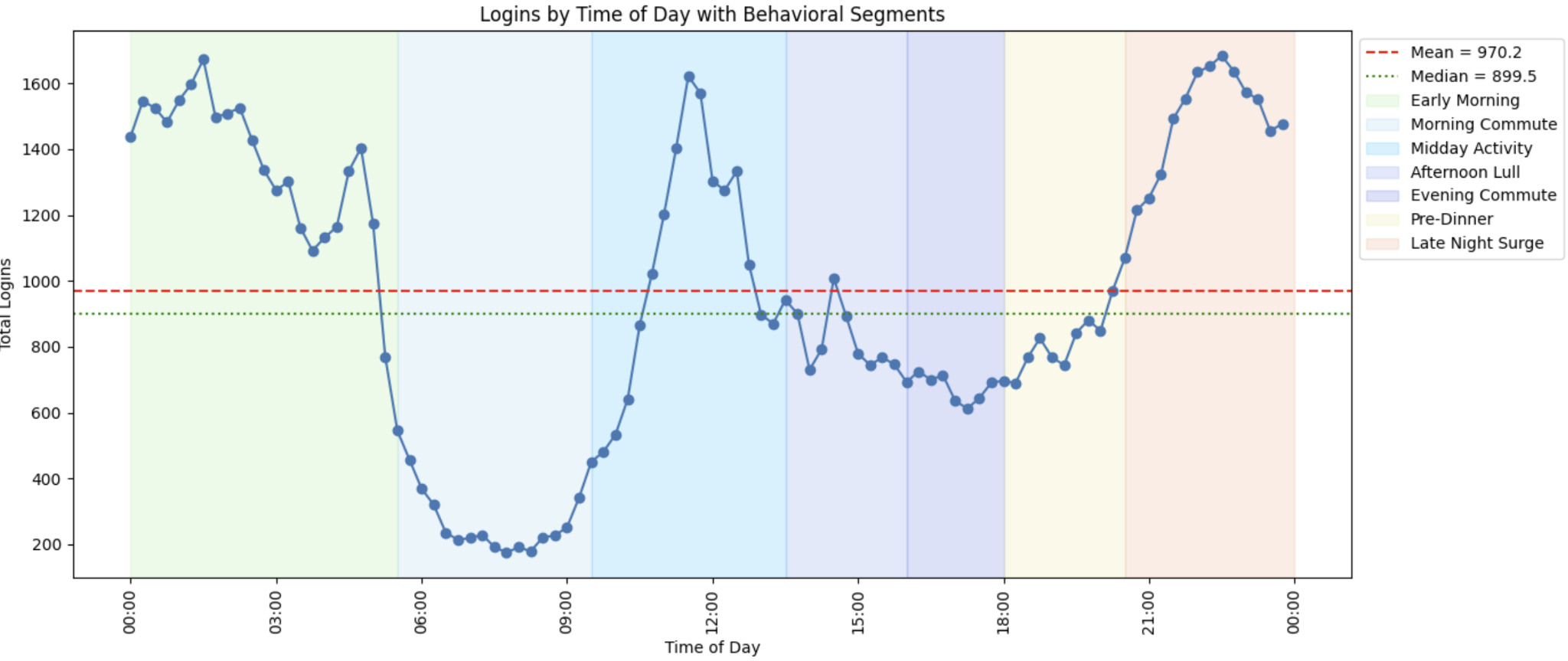
The logins.json file contains 93142 rows. There is no missing data. I made the assumption that these are related to taxi service. A user login equates to a user calling for a cab. After grouping the logins by 15 min increment, the following plot shows a fairly concentrated login count day to day. There are multiple outliers, and the initial assumption is that these could be related to the day of the week or other significant days in the month. Without additional context, it is hard to determine the cause of the spikes.



The distribution of login counts per 15-minute time slot (interval) shows a multimodal distribution. Once again, no specific trend to glim from it. The histogram below is a visual representation. However, the distribution suggests a high traffic period that could be tied to user behavior. The need to look at the logins by time of day is explored.



Breaking down the data into time of day, we are able to identify a trend. There are three times of the day when traffic is high. During early mourning hours (midnight to 5:30 am), midday (9:30 am to 1:30 pm), and late night (10:30 pm to midnight).The segments are color-coded in the chart below. During the early morning hours, people make their way home from late-night outings, e.g., at bars. This segment is followed by the late-night block, during which I assume people will be heading to activities by taxi, allowing them to indulge freely in alcohol. Therefore, the return home requires a similar taxi ride. The midday block, could be tied to people running errands by taxi during their lunch or extended break at work. Conversely, during to and from office commute hours, and family dinner time, the taxi traffic is low.



I identified the data containing duplicate rows. Based on the assumption that multiple people could call for a cab during the same time slot, I decided not to remove duplicates. This data might be useful in the future when additional analysis is performed.

# Part 2

**Pilot Program:** Toll Reimbursement Initiative for Cross-City Ride-share Expansion between Ultimate Gotham and Ultimate Metropolis

**Objective**

Ultimate Gotham and Ultimate Metropolis are two cities separated by a two way toll bridge. Ultimate operations find that drivers in each city prefer to operate on their side of the bridge. The goal is to implement a toll reimbursement program to encourage drivers to serve both Gotham and Metropolis, aiming to balance supply across cities and increase cross-city trip availability.

**Methodology**

* **Design:** Pre-post analysis using **Difference-in-Differences (DiD)** statistical modeling, OLS regression.
* **Comparison:** Treatment group (drivers eligible for toll reimbursement) vs. Control group (drivers not reimbursed).
* **Success Measures:**
  + Increase in active drivers.
  + Increase in cross-city trips.
  + Higher average distance traveled.
  + Volume of reimbursement requests.
  + Trip volume and driver retention.

**Pilot Implementation**

* Marketing campaign: Announce the program externally by focusing on higher earnings potential and toll reimbursement benefits.
* In-App notification: Push real-time reminders to drivers, especially when demand spikes in the other city.
* Signup incentives: Offer a bonus for new drivers who join during the program window and cross cities.
* Cross-city activity bonus: Offer a performance bonus, e.g., serve both cities 5+ times in the first 30 days to earn an extra cash bonus.
* Weekly highlight emails: Celebrate top earners or top cross-city drivers each week - keep momentum and social proof.
* Reimbursement process clearly communicated to drivers. In the future, this should be automated.

**Expected Key Results**

* Statistically significant increases observed in key metrics post-implementation during the pilot phase:  
  + 10% increase in active drivers.
  + 50% increase in cross-city trips.
  + 50% increase in average distance traveled.
  + 100% of reimbursement requests per eligible driver.
* The assumption is that control and treatment groups showed **parallel trends pre-implementation**, validating the causal inference.

**Post Pilot Recommendation**

Transition from **pilot** to **run-state program** if the results support measurable improvements in driver engagement and cross-city service levels.

**Caveats**

* **Driver Incentive Fatigue:** Impact may diminish over time if driver excitement wanes.
* **Toll Profitability Risk:** Rising tolls or lower fares could erode net driver profits even with reimbursements.
* **Parallel Trends Risk:** Natural market shifts could cause post-implementation divergence between groups.
* **Gaming the System:** Drivers may attempt to maximize reimbursements through artificially short cross-city trips.
* **Administrative Complexity:** Scaling manual reimbursement processes could strain operations unless automated.
* **Market Shifts:** External changes (e.g., new bridges, traffic reroutes) could distort program impact measurements.

**Next Steps**

* Automation of reimbursement processing.
* Expand marketing and in-app notification campaigns.
* Introduce tiered bonuses for sustained cross-city engagement.
* Monitor KPIs continuously post-rollout.

# Part 3

I am tasked with helping Ultimate in predicting rider retention. If a user is active within the last 30 days, they are considered retained. I made a few basic assumptions.

1. Every row is a unique user. The data set does not show historical trips and only the last trip.
2. The start of the program is 1/1/2014
3. The date the data is pulled is on 7/2/2014 (1 day after the max last trip)
4. I will use the date under no. 3 to determine activities in the preceding 30 days (June)

The plot below shows active users throughout June 2014. However, it does not identify if some of the same users logged in multiple times throughout the day. Eighteen thousand eight hundred four users, 37.6% of the total users, were retained in the 6th month. Two features contained null values. The average rating by driver is missing 0.4%. This is not significant; however, I imputed by the mean value. The average rating of driver is missing 16.24%. Imputing by the highest or lowest will likely skew the data. I imputed by the mean value. After this data manipulation, we have clean data to work with.

A graph of a number of people

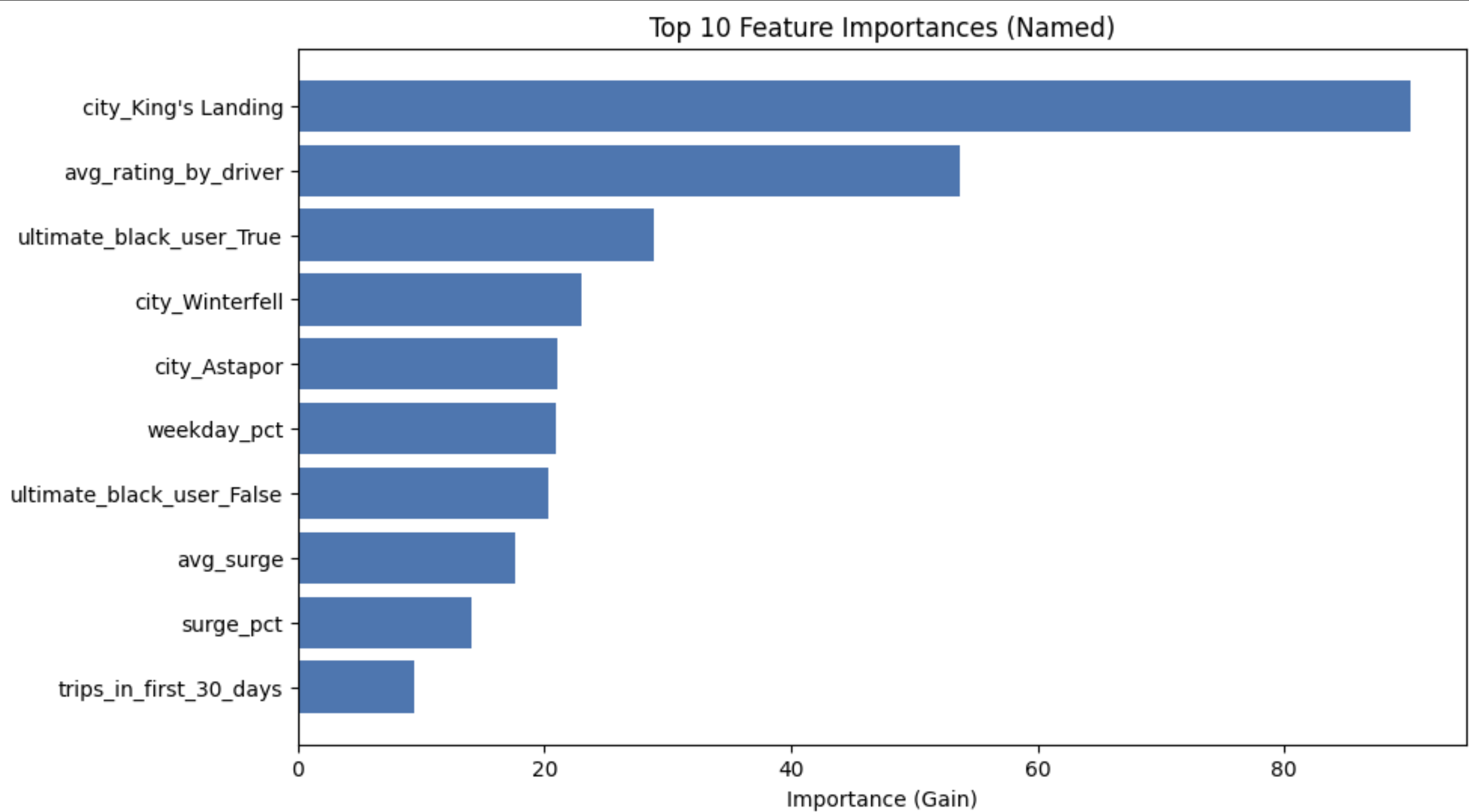
AI-generated content may be incorrect.

To build a predictive model, I started with the Logistic Regression algorithm. I assumed that there is a linear relationship between the features. In addition, with only a few columns of data, the problem was not too complex for the algorithm. The results were weak. The accuracy for this model was at 70% while AUC was at 0.724. The biggest weakness came from the class 1 recall at 43%. With a weak result, especially in the recall, I decided to try other algorithms. I moved to Support Vector Machine for its supervised learning power with an SVM classifier. The result was an improvement. The accuracy increased to 74.85%, and AUC is up to 0.8189. Class 1 recall was at 76%. Since Gradient Boosting is a superior algorithm for tabular data, I created an XGBoost model. The accuracy improved to 78%, and AUC went to 0.8458. However, class 1 recall dropped to 65%. Regardless, since it was the strongest overall, I decided to move forward with XGBoost hyperparameter tuning while prioritizing AUC and minimal errors.

I chose XGBoost hyperparameter tuning with RandomizedSearchCV primarily for the speed of execution, as this exercise is focused on the concepts versus the optimal result.

The final model result has an accuracy of 78.26%, an AUC of 0.8486, and a class 1 recall of 66%. With a small change in AUC, I can comfortably believe there is no overfitting and that the model is stable.

The next plot shows the most important features of the model. Users from King’s Landing are most likely to remain active in their 6th month. The average rating by driver and ultimate black car users are the other two features that most likely help us predict retention. My final step was to save the final tuned XGBoost Pipeline (model plus preprocessing) for future use.



By focusing on the five key features, Ultimate can design incentive programs to encourage users to remain active in their 6th month and beyond. It is important to include future features that might impact the model. Therefore, reevaluating assumptions and overall market conditions regularly is key to continued success.