Ultimate Coding Challenge – Unit 26.2

Submitted by Calanit Kamala

Ultimate Challenge Question 1

The logins.json file includes 93142 observations. The data includes timestamp data representing app logins times over a period of 102 days.

The first observable issue with the data is the year provided: all observations are labeled with the year 1970 which is an obvious mistake in the date label. I would investigate the query that pulled the data to understand where the error originated from. Since all the data is from the same year, using the mislabeled data will not affect the results of the analysis. The attribution error just needs to be corrected.

The EDA in the file ultimate_challenge_Q1 provides the logins data aggregated by 15 minutes, by one hour and by day.

A variable called 'event' was added and set equal to 1 in order to clearly represent aggregation.

The data exhibits a slight upward trend over the period provided.

The data was aggregated by day of week and by time of day. There is an obvious increase in demand throughout the week, with Monday having the lowest number of logins. Saturday and Sunday have significantly higher number of logins, with Saturday having the higher mean as well as the highest variability of all days.

Throughout the day, demand is generally higher from 9:00 pm to 4:00pm with two significant dips in demand: one from 6:00 am to 9:00 am and the other in the early evening hours, between 5:00 and 7:00 pm.

Ultimate Challenge Question 2

a. There are a few metrics that can indicate the outcome of an experiment to encourage drivers to meet demand in the two cities:

- **i.** Number of completed trips – if more drivers are available in Metropolis during the day and Gotham during the night, the amount of completed trips should increase.
- **ii.** Driver's app activity – if drivers increase their availability, the number of app logins and the time they spend on the app should increase.
- **iii.** % of completed trips by driver and by city origin – if the experiment worked we would like to see drivers trip origin change – for example, if a driver had 90% of their trips start in

Metropolis and after the experiment have 70% if their trips start in Metropolis, then it may serve as indication that the experiment had an impact.

b. One experiment that can explore whether a bridge toll refund can affect drivers' behavior is an A/B test. When designing the experiment, a time-frame has to be chosen, normally a 2-week span for multiple times, to make sure that the results of the experiment are robust to seasonal changes. Then a sample size has to be calculated for the desired power and significance. A common power for A/B tests is 0.8 and confidence interval is normally 0.05.

The null hypothesis in this experiment would be: *giving drivers a bridge toll refund has no impact on their trips origins*. The alternate hypothesis would be: *giving drivers a bridge toll refund changes their trip origin*, as they are more likely to drive across the bridge to pick up additional riders.

To implement the test, an equal number of drivers in each city will be exposed to the treatment randomly. For example: 500 drivers in Metropolis will be exposed and 500 drivers in Gotham will be exposed with drivers not given treatment during the experiment representing the control group for the respective city.

There are two main t-tests statistic to calculate in order to identify the impact of the experiment:

i. The first t-test would compare the means of treatment and control group in each city separately. If, for example, the mean of the % of rides originating in Metropolis for Metropolis drivers goes down after treatment, then it will indicate the experiment had an effect.

ii. Another interesting t-test would look at the differences in means of both treated groups – if there was a significant behavioral response in both cities, was there one city that had a stronger impact? It is possible that Metropolis drivers will be more affected than Gotham drivers, since night time rides are more expensive in general and night demand is higher.

c. If providing a toll refund is affecting drivers' behavior and changing the distribution of trip origin per driver, I would reject the null hypothesis and conclude that the toll refund provides sufficient incentive to affect drivers' supply. The next step would be to test revenue implications of providing such incentives – if indeed the incentives increase the number of completed rides, it also increases company's revenue and customer satisfaction. I would like to know the ROI to investment – was the increase in revenues sufficient to offset the incentive's cost? Is there a specific number of incentives that is required to offset the costs of the program?

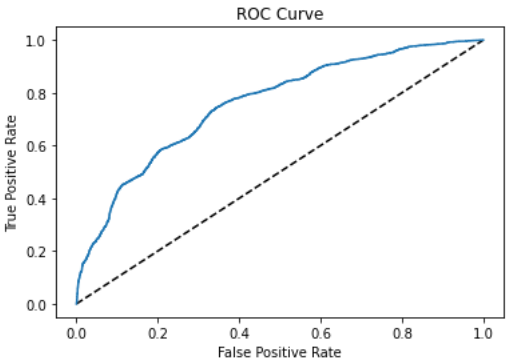
One caveat to this experiment is the implicit assumption that the only cost that drivers incur is the bridge toll. However, drivers will consider additional costs: time costs to drive across the bridge, possible time costs associated with bridge traffic, additional gas paid for driving a longer distance to pick up riders, etc. Providing a refund for the bridge toll addresses just one of these costs and may not be significant to alter drivers' behavior as desired.

The second issue with A/B testing in this context are network effects. If drivers exposed to treatment change their behavior as a result of treatment, that can affect the behavior of other drivers that are not directly exposed to the treatment and thereby make inference about the impact of the experiment biased. The complex network effects in the context of a shared resource pool (number of rides) means that A/B testing estimates' accuracy is questionabl. In the context of companies such as Uber and Lyft, switchback experimentation is often the framework used to address this issue.

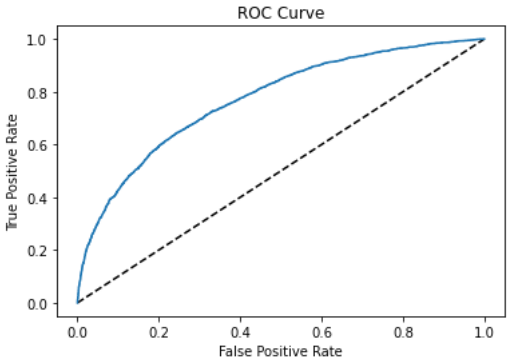
In swtichback experimentation design a driver will be sequentially and randomly assigned to a treatment for a fixed period of time. In this manner a driver can be assigned to either treatment or no treatment multiple times and that in turn can help estimate individual-level treatment effect and address inference in this context.

Ultimate Challenge Q3

1. Without eliminating null observations but after eliminating duplicate observations, 18,310 out of 49,992 were not active (number of rides in last 30 of data = 0), which is 36.62% of the sample. There are three features with null observations: average rating by driver (0.4% of observations missing), average rating for driver (16.24% of observations missing) and phone (0.79% of observations missing). Since average rating for driver has so many missing observations, I would be concerned about dropping these observations. This variable is hard to impute because it is rider-level data. I would probably run my models excluding nulls to find out whether this variable is significant. If it is not significant for estimation, I would go back to the full data set and drop the column, rather than the rows.
2. My initial model of choice would be a logistic regression since we have a binary dependent variable. However, a logistic regression requires a large amount of observations to provide accurate estimates. This sample may be too small to provide desired accuracy. After scaling the data, splitting it to train and test sets, I run logistic regression that resulted in accuracy of 0.69 and the following ROC curve



Given the ROC curve and the accuracy, I would like to see if I can do better. The first step is to run a recursive feature extraction. After running the model with RFE, the optimal number of features is 12 and accuracy increases to 0.72. The eliminated feature is 'average_rating_for_driver, which is also the feature with the highest number of null observations. Running the analysis again without this feature and with the original observations, improves accuracy to 0.73. The ROC curve is smoother.



I still would like to explore decision trees classifier. Decision tree classifier results in accuracy of 0.71 and a random forest classifier increases accuracy to 76%. For my last step I want to test cross-validation versus train test split. Five-fold cross validation results in accuracy estimates of 74-75%, so it seems like the random forest classifier with the train-test split has the best accuracy.

The top three features according to the random forest model, are average distance, average rating by driver and the percent of weekday rides. It looks like user's phone and city do not significantly affect the likelihood of a user to be an active rider. The company can use this information to target specific non-active users in order to reduce churn.