Part 2 – Experiment and metrics design

The goal of this experiment is to measure the effectiveness of a feature change on the driving patterns of its users. In this example we intend to make the users more available in a city they didn't register. Specifically, there are two cities of interest and we intend to have users from one city spend more time picking up riders in the other city. In order to measure this, I propose using the average of an entropy measure from the population as the performance metric.

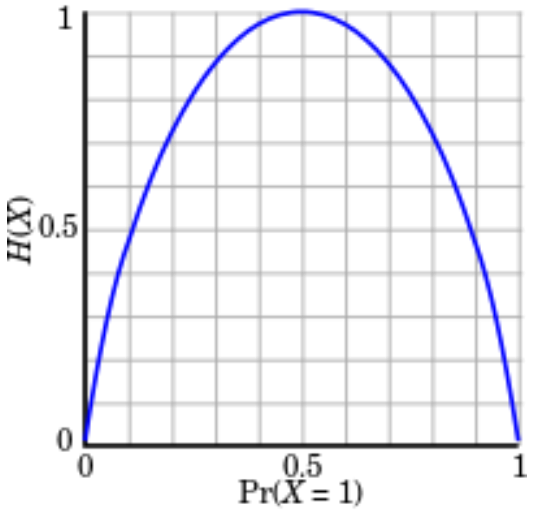
$$E[H_p(p)] = \frac{1}{n}\sum pln(p)$$

In this case p represents the probability of a driver accepting a ride in the city they're not registered in. Calculated as shown below:

$$p = \frac{\#\ of\ rides\ given\ in\ their\ registered\ city}{Total\ \#\ of\ rides}$$

The probability is specific to the user and is dependent upon being able to gather data on the locations the users picks up rides as well as what their registered city is.

Entropy is a metric derived originally in information theory and has a number of interpretations. It is often used in machine learning to represent how unsure we are when classifying something and used in some algorithms as a performance metric to tuning the model. The function goes to 0 as the probability goes to 0 or 1 and is maximized when at 50%. This makes it an ideal metric for measuring the success of our change because we are trying to get drivers to split time between cities in a more even way. Our desired results line up well with the function as a more even distribution between cities is shown as a higher value in the entropy function.



In order to implement the experiment, I suggest (if feasible) to randomly expose half of the drivers to the change. I would even do I stratified sample by city so plenty of drivers in each city are chosen. After a period of time calculate a sample mean of the entropy for users with the change and users without the change. Using the sample metrics run a T-test (or an equivalent bootstrap test) to determine the significance, getting a p-value representing how likely the results are due to chance alone. If the results are significant enough the team can be confident that the change was successful making it worth the overhead of the reimbursements. If splitting the users is too difficult, we can try measuring the mean entropy before the change and after the change. If by using the same process we see there is no significant change in the users we can roll back these changes.

One caveat to be aware of is the possibility of confounding variables ruining the integrity of the test. For example, if there is some other short-term draw to one of the cities then entropy may increase but only because of this other change not the change we are testing for. This is something that the operation teams need to be vigilant of. Another possible issue is that if the percent of the rides in the driver's home city decrease below 50% (however unlikely that is) then the entropy will actually begin decreasing again. This is a topic that should be discussed with the operations team. If this behavior is unwanted then we don't need to change the metric because it will naturally decrease our performance results. However, if this is something that shouldn't be punished, we can create a condition where if the true probability is below 50% then set to 50% before calculating the entropy, this way there is no reward or punishment for drivers having more rides in the other city and the performance will be maximized by 50% or less rides in drivers home cities. Finally, if there are other cities besides the two of interest that drivers are picking up riders from then we can adjust the probability equation to the following to eliminate the impact of the irrelevant cities.

$$p = \frac{\#\ of\ rides\ in\ registered\ city}{\#\ of\ rides\ in\ cities\ of\ intrest}$$