Take-Home Challenge: Relax Inc.

Goal: To explain the variation in 'adoption rate' between uses of the service.

Preface

Issues with the data – Traditionally, when a user id is generated, it is done sequentially. There are some "user ids" with a lower id number have a later creation_date than others. This could be either test accounts that are in production or could be correct due to how the id generation process is done. Further investigation is required.

Feature Engineering

Adopted user flag – To solve the problem, a feature was created to identify if an 'adopted user'. To create the flag, I grouped usage summary table by user_id and week (summing visited by each group). I then created a binary feature for users with a visited frequency > 2 on a given calendar week. Finally, I dropped the week and created a unique list of users that had a single week with 3+ visits. The final output was joined to the user table for modeling.

User Group Size – To determine if the size of a group correlates to a user becoming an 'adopted user', I grouped the users table by org_id and counted the number of users within a specific group. One hypothesis is that users in a larger group might have more interaction with other users and show a propensity to become an adopted user.

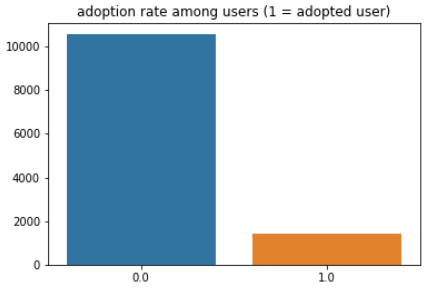
User Invites – In a similar vein to the "User Group Size" feature, the "user invites" feature was added to count the number of total invites a user contributed to. The idea being that a user with a higher invite count might foster more interaction in a group. Those associated with this user might have a higher "adoption rate".

User Create Date Rank – To capture any change in time, the feature creation_dates was converted to a rank.

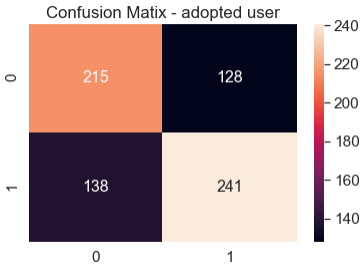
To Note: for any NaN values in "User Invites", a -1 was filled in.

Machine Learning

With the binary classifier "Adopted User Flag", a logistic regression was fit to the data. Due to the imbalance in the data, a high accuracy was achieved. To combat this, classes were balanced by sampling the *non-adopted user* to the same size as *adopted users*. After the balancing, the logistic regression achieved 63% accuracy (50% being random).



Looking at the confusion matrix, we see that the model is similarly inaccurate for both 'adopted' and 'non-adopted' users.



Next Steps

63% is not an accurate score. Further improvements are required. One idea is to change the "adopted user" flag to 3 visits in a given calendar week versus 7 consecutive days.

Furthermore, taking more recent dates in the sample might be more representative in terms of recent adoption rate since the difference in the platform has likely changed overtime.

Removing the data described in the preface might reduce some of the noise in the data.

Finally, removing out some of the features that do not contribute to the overall model could improve accuracy slightly.