Springboard Data Science Career Track - Relax Inc. Take-Home Challenge

After loading in the "takehome_user_engagement.csv", I stripped out the time on the time_stamps leaving only the date, which I used to group together all user logins. This left me with a list of users for each day. From there, I ordered them by date and collected all the user lists within seven day periods. I counted each user within each seven span and captured the user_id's that had counts greater than or equal to three. Then I created a column in the imported "takehome_users.csv" DataFrame and mapped the "adopted user" labelling to the user_id's

Right from the start, I could see that predicting anything with this data was going to be challenging. There are only 10 columns in the "takehome_users.csv" which is not much to go on. Also, many of them or not going to be particularly useful for predictions, like names, email address and user_id. Also, there are considerably more users labeled as "False" under "adopted_user" than are labeled True (10555 : 1445, respectively), which could lead to bias. To account for the latter, I would pull 1445 random samples from the "False" labels so the two are even. Also, I found that there was missing data in the "last_session_creation_time" and "invited_by_user_id" columns. "Last_session_creation_time" would not be a helpful feature so I can just remove that all together, but there are thousands of missing values in "invited_by_user_id" and that could be useful, so I filled in the NaN with zero's since these are all user ID's and not quantitative values. After that, I also removed the "creation_time" and "object_id" columns and encoded all non-numeric data into integers. These steps left me with only 5 usable features, so predictions were going to be rough.

I opted for a Random Forest classifier and it performed about as well as I expected. Five features is not a lot to go on and the model yielded an accuracy score of 0.53, which is not great. From those features, the model found "org_id" to be the most significant towards classification followed by "invited_by_user_id" and "creation_source". Investigation into the org_id's activity and invites received from other users could help understand why they would be relevant to classification and perhaps uncover some additional features for future modeling. What org's have users that received / sent out the most invites? How many of those invited were converted into new users? What is the function of these orgs groups? How big are they? These are all directions for additional analysis that would help us to understand how the service is being used and who is / will continue to use it.

```
              precision    recall  f1-score   support

           0       0.51      0.54      0.52       347
           1       0.55      0.52      0.54       376

    accuracy                           0.53       723
   macro avg       0.53      0.53      0.53       723
weighted avg       0.53      0.53      0.53       723

Accuracy Score: 0.53

Feature Importance:
org_id                        0.617329
invited_by_user_id            0.298529
creation_source               0.048725
opted_in_to_mailing_list      0.018250
enabled_for_marketing_drip    0.017167
dtype: float64
```