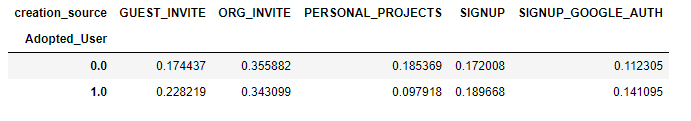
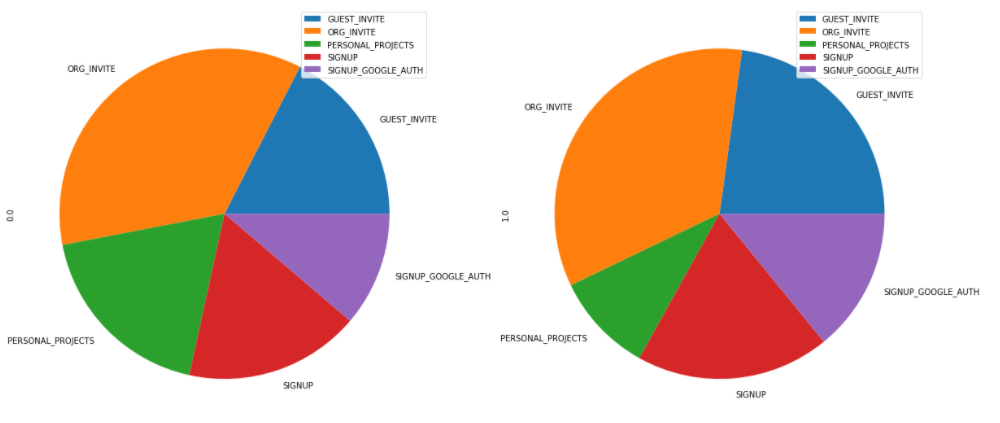
Adopted users in the Relax user data are a minority class in a highly imbalanced dataset. I found one feature that was associated with an Adopted User classification: Creation Source.

To identify the Adopted Users, defined as people with three logins within a seven-day period, I put the two datasets into data frames. I created a new column in the login data frame, exactly seven days after the login date, then merged the login data frame with a copy of itself on user\_id. With the new, self-joined data frame, I created a Boolean field that returned True if the second login was greater than the first login and less than the seven-day mark. Through a series of pivots, I identified the user\_ids where there at least 3 logins within one week and I identified 1,297 Adopted Users out of the 12,000 total users (or about 11%).

I examined each of the features and decided to use chi square tests to establish if a given categorical variable was associated with Adopted Users, and then charted the frequencies for the Adopted v non-Adopted users. Looking at creation source, the chi square p value was sufficiently low (less than 0.05) to reject the null hypothesis that there was no difference between creation sources. Adopted Users were more likely from guest invites and less likely from personal projects.





I ran similar tests for enabled for marketing drip and opted in to mailing list, but these did not yield low p values (0.82 and 0.31 respectively), and I did not reject the null hypothesis in either case.

I ran a logistic regression, which was not successful due to the imbalanced nature of the dataset. While is achieved 89% accuracy overall, it simply predicted that all users were non-Adopted Users and is useless at the predicting Adopted Users. I attempted to use the Imbalance Learn library but ran into a validation error. After looking up the error on Stack Overflow, I learned that I needed to update my version of SciKit Learn, and then found out it is not available in Anaconda. I would like to use the SMOTE methodology, randomly oversample the minority class and randomly under sample the majority class. This will have to be done after this library is available in Anaconda.