Report on User Adoption Rate

We’re given a table of 12000 users who registered in the last two years, along with important features for each user’s registration. We’re also given all the time-logins for all the users (accounting for a total of 207917 logins) and our first step is to find the users who have been **adopted** (13.3% of them are adopted users). Once this is taken care of, we remove some of the irrelevant features - names, email addresses, last time of login - and generate new ones. In the end, we found ourselves with a table containing our target ‘adoption_user’ and the other relevant feature variables: creation source, the week day they created the account, whether they opted in to the mailing list and whether they are on the regular marketing email drip.

Basic Statistical Analysis

adopted_user	0	1
was_invited		
0	0.876411	0.123589
1	0.857722	0.142278

At first, I looked for how each feature individually impacted the target variable. We notice that the ‘creation_source’ and whether the user was invited (as shown on this table) have a significant impact on the rate of adoption. Still though, those are univariate analyses, so we now look to build a more complex classifier model which will account for all variables at once. Note, ‘creation_source’ and ‘was_invited’ are collinear features, so we drop ‘was_invited’ from our model.

Machine Learning Classifier

features	ratio odds
SIGNUP_GOOGLE_AUTH	1.007098
opted_in_to_mailing_list	1.002682
enabled_for_marketing_drip	1.002551
creation_saturday	1.002515
creation_wednesday	1.001635
SIGNUP	1.001572
creation_tuesday	0.999540
creation_thursday	0.999206
creation_friday	0.998641
ORG_INVITE	0.998444
creation_monday	0.998335
PERSONAL_PROJECTS	0.979154

While I attempted multiple classifier fitted through a pipeline (LogisticRegression, KneighborsClassifier and RandomForestClassifier), I ended up choosing the logistic regressor from Scikit-learn to build a predictive model aimed at classify adopted vs non-adopted users, based on the features mentioned above. The model is built against the recall of the target variable, since our interest is to capture as many adopted users as possible. Once the model is built (with a precision score of 68%) we rank the importance of each feature by sorting the respective odds ratio against the default user: one invited by guest, that did not opt-to the mailing list, that did not enable for market drip and that created their account on a sunday. The features are ranked by importance and we note that those with odds’ ratio > 1 positively correlate with the adoption rate.

Other features that were not considered in this model but that could have been implemented are: whether the user was invited (as long as we remove the ‘creation_source’ variable), the number of referrals sent per user and finally the organization group the user belongs to. Implementing those to the model may improve its performance and give us a better idea on more relevant features.