Growing Smarter: Understanding User Acquisition

Mac Strelioff

06/27/2019

# Background

As part of my fellowship at [Insight Data Science](https://blog.insightdatascience.com/), I worked on a 2 week consulting project with an external company. Their product was a SaaS application enabling team collaboration on shared files, and has a 30-day free trial model. Multiple users can be associated with an account, and users can have different roles that enable different privileges within the service. The overall goal was to identify free trial accounts that would convert to subscription-based paying customers.

At the request of the client, some of the information in this post has been masked so as to not reveal any confidential information.

# Business Need

The survival of any company hinges on it’s ability to acquire new users. However, the majority of free trial users for my client failed to convert to customers. This leaves much opportunity to increase their userbase.

## Project Goals

As a consultant, I helped with two major goals focused on issuing nudges to users in order to increase user acquisition:

1. Identify, as early as possible, patterns in user behavior that indicate whether a free trial user is likely to become a customer after the trial.
2. Leverage those patterns to identify outreach strategies for users that might otherwise be unlikely to continue using the service..

# Data sources and processing

The original dataset consisted of account activity and product performance data for all accounts over a one year period. The dataset contained a large number of accounts, including some that were not germane to the project goals. Since the primary goal focused on behavior during the free trial, I excluded any accounts that were never on a free trial during the data collection period. In exploratory analysis, I found a large number of accounts that showed little to no activity. This prompted me to believe there are two types of trial accounts that do not convert to paying customers – 1) accounts that were created then never engaged with the product, and 2) accounts that engaged with the product, then failed to find the product valuable and decided to stop engagement. Since the strategies to address these accounts might differ, I decided to exclude any account without a minimal degree of engagement with the product and focus on the type of nonconversion. The criteria for a minimal level of engagement was chosen based on dependencies between product features, and decided during discussion with the client. Finally, some product features were not available during the full duration of data collection, as the product evolved over time. Since there was a large sample size, the easiest way to make the analyses pertinent for all features, and to keep accounts comparable to one another, was to exclude data collected before all product features were available.

## Feature Engineering and Exploration

The original account activity data was in terms of counts of each possible action (e.g. workspaces created) on each date during data collection. To make the accounts easily comparable regardless of observation dates, I created a variable for account age (days since the account was created). Also, rather than working with daily counts I computed cumulative sums, which represented an account’s total usage aggregated over all its users’ activity of a product feature up to a particular age of their account. Finally, to mitigate confounding of the counts of activity by the number of users, I divided the cumulative sums by the number of users. These were features that I focused on – total usage of each product activity per user up to a particular day since the creation of the account.

The time of conversion could be many months after a trial had ended, as it may take time for a user associated with an organization to gain approval to purchase a subscription, or it may take time for a large organization to negotiate a price with my client. To focus on classifying free users as *potential* customers based on activity, I created a variable that indicated whether an account *ever* ended up as a customer. I then conceptualized the problem as a classification problem where, based on account activity over time since the account’s creation, I estimated the probability that the account would ever convert to a customer.

I initially thought that the accounts that converted would differ in terms of the distribution of these features (cumulative activity per user), relative to those that did not convert. Hence, in exploration, I focused on probing this intuition by plotting the median and to quantiles of cross sections of these features across account age, split by users who ended up converting versus those who did not. An example, focusing on the number of workspaces created per user, is shown in the figure below. Classification would be easiest and most interpretable if there were features that the converters clearly used more than those who never converted.

Median (solid line) and middle 60% (shading) of the distribution of cumulative workspaces created per user, split by those who ever paid (blue) and those that were on a free trial forever (red)

Median (solid line) and middle 60% (shading) of the distribution of cumulative workspaces created per user, split by those who ever paid (blue) and those that were on a free trial forever (red)

# Modeling

## Considerations

My intuition about differing distributions is natrually expressed in [linear discriminant analysis](https://en.wikipedia.org/wiki/Linear_discriminant_analysis) or [quadratic discriminant analysis](https://en.wikipedia.org/wiki/Quadratic_classifier). However, these algorithms hinge on an assumption that the features are Gaussian distributed, which was clearly not the case – values were strictly non-negative, and distributions were skewed such that there were lots of values around 0 and some values far from the mean. Because of these violations, I also considered a [support vector classifier](https://en.wikipedia.org/wiki/Support-vector_machine) with radial basis functions, and tree based algorithms – [random forests](https://en.wikipedia.org/wiki/Random_forest), and [gradient boosting decision trees](https://en.wikipedia.org/wiki/Gradient_boosting) (GBT). An advantage of the tree-based approaches is their robustness to any distribution of the features, and any functional relationship between the features and the probability of an account continuing product use after the free trial.

Another issue was class imbalance, since the majority of accounts did not continue. I used [SMOTE](https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis), an algorithm that generates synthetic data from the underrepresented class (continuing customers), to address the class imbalance when fitting the model in training sets. I also considered metrics beyond accuracy, such as [AUC](https://en.wikipedia.org/wiki/Receiver_operating_characteristic) and the [F1 score](https://en.wikipedia.org/wiki/F1_score), when selecting a final model.

## Initial Performance

To quickly hone in on a model, I assessed the algorithms mentioned above with default hyperparameter values using [scikit learn](https://scikit-learn.org/stable/index.html) in Python. I focused on classification accuracy in a test set, and the tree-based algorithms outperformed the others by approximately 10%. This outperformance is likely due to the non-standard distributions of the features.

## Hyperparameter Tuning

To assess performance across hyperparameters for the tree-based algorithms, I created [validation curves](https://scikit-learn.org/stable/auto_examples/model_selection/plot_validation_curve.html) with 5-fold cross validation and a set of hyperparameter values. The performance of both tree based algorithms was mostly stable in terms of [F1 scores](https://en.wikipedia.org/wiki/F1_score), except for poor performance when there were very few () estimators. To mitigate potential overfitting, I increased the minimum number of observations in a leaf to 10, but left all other hyperparameters at their default values.

## Fitting Procedure

I fit the random forests and GBT algorithms to cross-sections of the data at 7, 14, and 30 days since account creation. For each cross section, I split the data into 5 folds, and used [SMOTE](https://en.wikipedia.org/wiki/Oversampling_and_undersampling_in_data_analysis) when training the algorithms within each fold to account for the class imbalance. In each fold, I logged the feature importances and performance metrics from both algorithms.

# Actionable Results

I used the mean feature importances across folds from the GBT algorithm to identify the features that differentiated between successful accounts and accounts with users that may have needed more onboarding. Since these features seperate continuing accounts from those that did not continue after the free trial, these are the features to prioritize when considering interventions to add value to the user’s experience.

Mean (bar length) and standard deviation (black error bars) of feature importance evaluated across the 5 folds for the gradient boosting trees algorithm. Feature names have been obscured at the client’s request

Mean (bar length) and standard deviation (black error bars) of feature importance evaluated across the 5 folds for the gradient boosting trees algorithm. Feature names have been obscured at the client’s request

To understand the form of the relationship between these features and continued engagement after the trial, I binned accounts based on product feature usage and plotted the proportion of accounts that continued using the product after the free trial across these bins. Figures like this, paired with data on an individual account activity, could help in personalizing user outreach to focus on aligning users’ feature usage with that of users in more successful accounts. For example, **based on the figure below, if an account has fewer than 1 workspace per user, their experience might be improved by resources that make workspace creation easier to understand or engage with**.

Proportion of accounts that convert as a function of workspaces per user. There seems to be a bump in conversion rates from around 20% to around 40% above around 1 workspace per user.

Proportion of accounts that convert as a function of workspaces per user. There seems to be a bump in conversion rates from around 20% to around 40% above around 1 workspace per user.

To identify struggling accounts for outreach, I applied the GBT algorithm to all data and focused on the confusion matrix, shown below. Different actions could be taken with respect to accounts in each quadrant. Accounts in the *bottom right quadrant* are currently customers, and were identified as such by the algorithm. These are the accounts with users who were successful in identifying value in the product and establishing an ongoing relationship with my client. Accounts in the *top right quadrant* are those who have not yet become customers, but are engaging with the product in ways that are similar to those who have become customers. Users in these accounts have likely identified valuable aspects of the product, and could be contacted by a customer success team to help them find a subscription plan that suits their needs.

Accounts on the *top left* and *bottom left quadrants* were identified as accounts that would not continue with the product after the free trial – **these are the accounts that may benefit from outreach** that demonstrates the value this product can add to their workflows. Accounts in the *top left quadrant* were correctly identified as accounts that would not continue with the product. Users in these accounts may have failed to identify valuable aspects of the product, and may have had a better experience if they had been contacted early by customer support or had access to educational resources that could have helped them use the product. Accounts in the *bottom left quadrant* are those that became paying customers, but were incorrectly identified as accounts that would not continue with the product based on their activity in the first 7 days. Misclassifying these accounts has essentially no harm, as it would only encourage efforts to improve their experience early in their trial.

Confusion matrix based on all data, for identifying who to contact.

# Summary

I began with an intuition that accounts that converted would have a different distribution of product feature usage per user, relative to those that did not convert, when feature usage was assessed at cross sections based on account age. To explore this, I looked at the quantiles of distributions across feature usage. Focusing on this with formal models, I found that tree-based algorithms could perform well across accuracy, precision, recall, and F1 scores, in identifying the accounts that converted, based on daily snapshots of aggregate feature usage.

To derive insights from this data and analysis, I focused on confusion matrices which identified free trial accounts that acted as if they would continue product usage, and accounts that were currently unlikely to continue with the product (those in need of educational resources, and/or contact from support teams). I focused on the features found to be important by the tree based algorithms in order to identify specific features to target when reaching out to users in struggling accounts. To discover how feature usage relates to a propensity to continue using the product, I looked at the proportion of accounts that continued after trial across levels of engagement with the important product features.

Overall, my work provided valuable tools for identifying accounts to connect with for long term relationships or for onboarding and educational assistance, as well as feature usage patterns indicative of success with the product.