**WSDM – KKBox’s Churn Prediction Challenge**

**Kaggle Research Prediction Competition**

Springboard Capstone Project

Beverly Rice Jan 2018

**I. Introduction**

KKBox is Asia’s leading music streaming service, holding the world’s most comprehensive Asia-Pop music library with over 40 million tracks. The company supports a variety of devices to include Windows, Mac OS X, IOS, Android, Symbian, Bada, and Java phones. They offer personal recommendations, high-quality sound, exclusive live concerts, and a “Listen With” feature that allows members to listen to music with favorite celebrities, artists or friends while chatting with them at the same time. Subscribers have praised the service’s relevance of search results and advanced search function over similar providers.

KKBox boasts having over 10 million subscribers based in Taiwan, Hong Kong, Japan, Macau, Malaysia, and Singapore. They work on a “freemium” business model that allows consumers to receive base services for free but are required to pay for premium options like listening in offline mode, supporting multiple platforms at once, and no advertisements. As a premium subscription business, predicting who will churn is critical to long-term success. Currently, the company uses survival analysis techniques to determine the residual membership life time for each subscriber. KKBox is interested in new methods to predict churn so they can be proactive in keeping users. They have donated a dataset for use in the 11th ACM International Conference on Web Search and Data Mining (WSDM 2018) to challenge kaggle participants to build an algorithm that predicts whether a user will purchase a new service subscription within 30 days after their current membership expiration date.

**II. The Dataset**

KKBox provided 5 files for use in building the model:

1. train\_v2.csv
2. members\_v3.csv
3. transactions\_v2.csv
4. user\_logs\_v2.csv
5. sample\_submission\_v2.csv

1. The train file containing 970,960 unique user IDs and whether they have churned

2. The member file with select user information for 6,769,472 members, to include user ID, city, age, gender, registration method, and registration initialization time

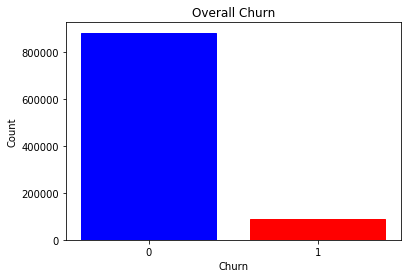
3. The transactions file with 1,431,009 transactions of users up to Mar 31, 2017 - detailing user ID, payment method, payment plan, plan price, amount paid, whether the member opted to auto-renew their subscription, transaction date, membership expiration date, and whether or not the user canceled the membership in a given transaction

4. The logs file with 18,396,362 daily user logs describing the listening behaviors of select users; includes user ID, date of log, the number of songs played less than 25% of the song length, between 25-50% of the song length, between 50-75% of the song length, between 75-98.5% of the song length, and over 98.5% of the song length; \ It also includes the number of unique songs and the total seconds played for the given log

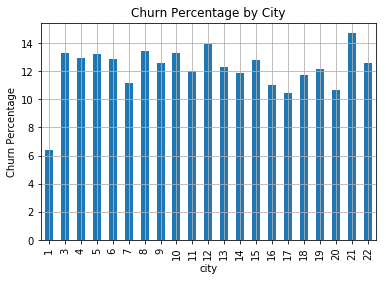
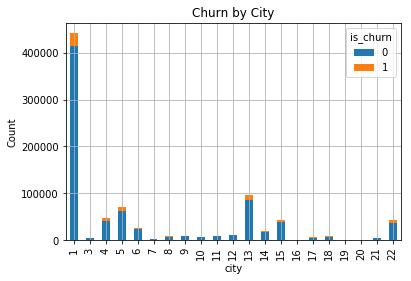
5. The sample submission file with a test set, containing the user IDs in the format WSDM expects it to be submitted in - it includes the user ID and churn prediction

**III. Exploratory Data Analysis**

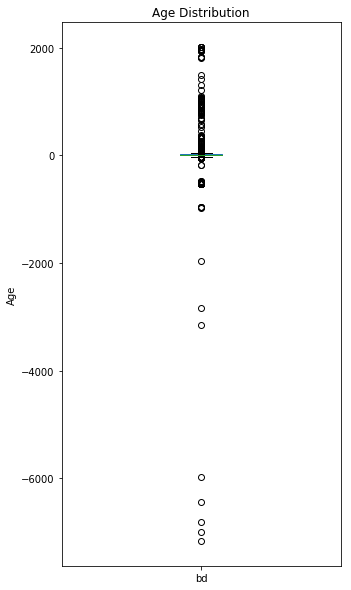
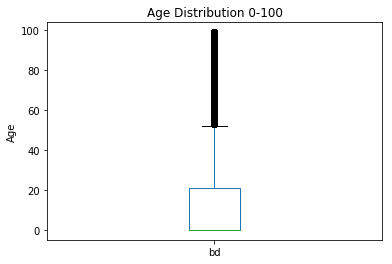
The first file explored was the train file which contained only 2 variables, unique user ID and whether the member churned.



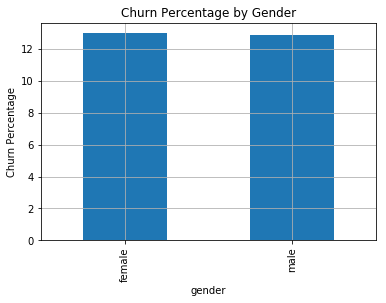
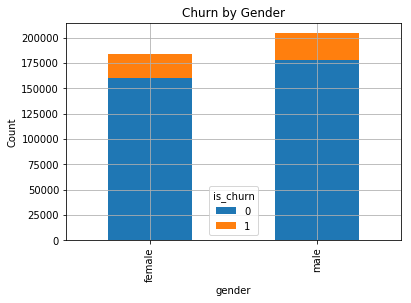
The file showed the vast majority of KKBox members continuing their subscriptions beyond March of 2017. 87,330 members terminated their membership making up 8.99% of the users at the time. The train file was then combined with the members file to analyze churn rates with respect to each additional variable provided in the new file: city, age, gender, registration method, and initial registration time.



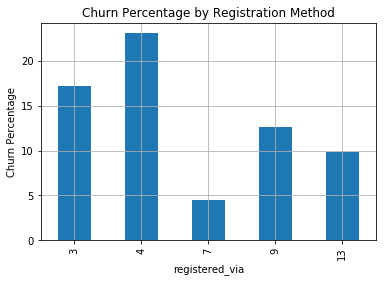
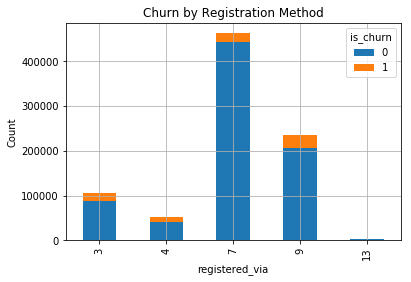
A majority of KKBox members come from City 1 of 21 independent cities listed. City 1 produced the lowest churn rate, just over 6%, while all other cities fell between 10-15%.



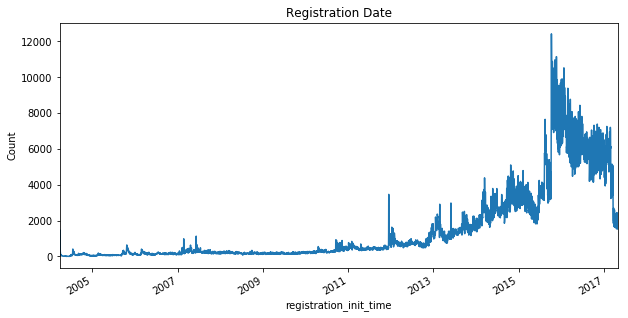
Reported ages ranged from -7168 to +2016 and over 67% of members reported an age of '0' suggesting many users falsified the information. After removing the users under the age of 1 and over the age of 100, the data reflected the remaining users largely fell between the ages of 15 and 55. The pool of users within this range made up less than 30% of the overall data provided and was too small to be considered an accurate representation of KKBox's member base as a whole. With all this in mind, the variable was discarded.

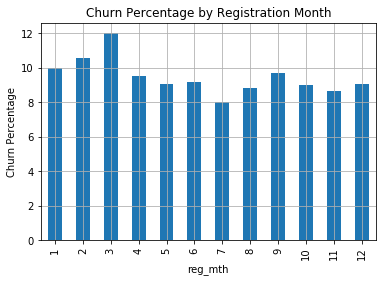
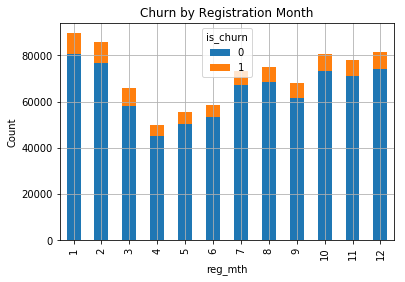


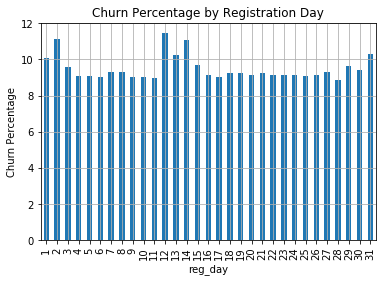
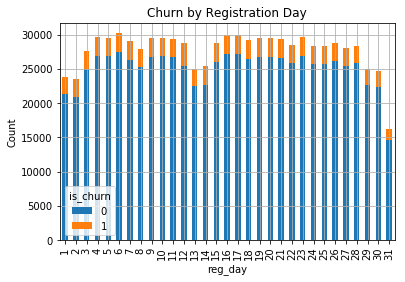
Gender was the only variable throughout the entire data set with missing values. Roughly 55% did not specify gender and between the two, there was only a 0.08% difference in churn rate, males at 12.90% and females at 12.98%. Considering only 45% reported gender and males and females churned at the same rate, the variable was discarded.



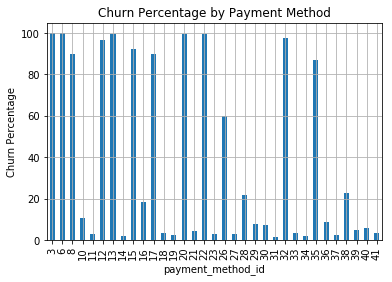
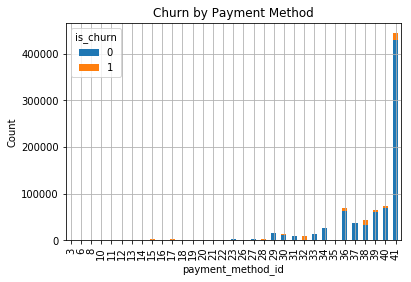
Most members registered via method 7 of 5 listed methods. Those who registered via this method were the least likely to churn. Method 4 produced a significantly higher churn rate than most other options.



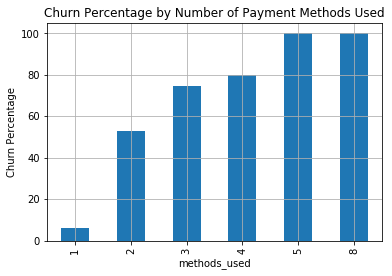
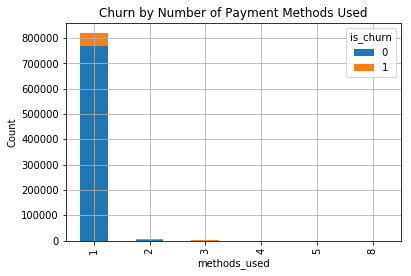




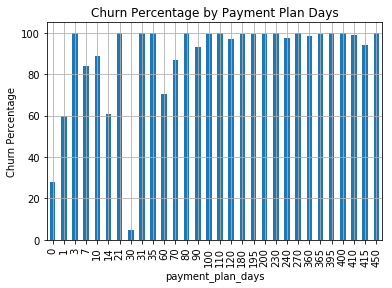
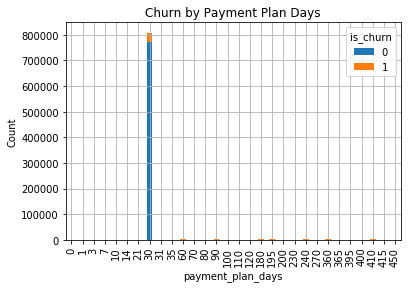
The number of registered members increased at a rapid rate starting in 2009. During this time period, KKBox went international and expanded their user base beyond Taiwan and into Hong Kong and Macau, and later to Japan, Malaysia, Singapore, and Thailand. Further, in 2011, KKBox received investments from KDDI Corporation and HTC Corporation likely contributing to the increase in users. The data above shows more people tend toregister in the beginning and end of the year. The data also depicts more users churning in March than any other month. Additionally, users tend to churn slightly more in the beginning (1-2), middle (12-14) and end (30-31) of the month.



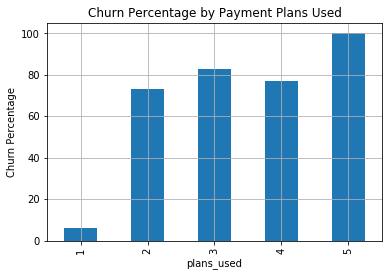
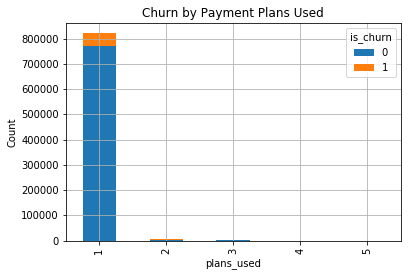
Most members used payment method 28 or higher – the most used being method 41. Several payment methods reflect high churn rates among members most notably in those methods less than 28. Some members used multiple payment methods so a new variable, methods\_used, was created:



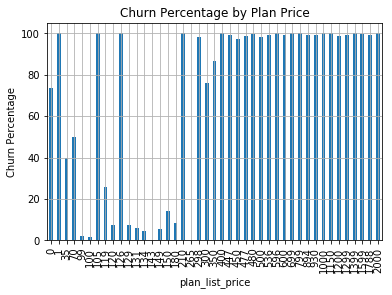
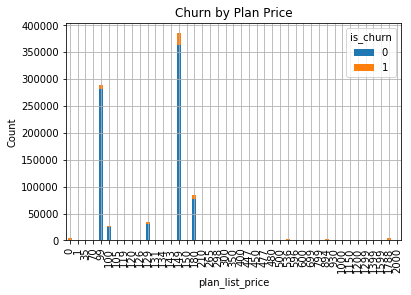
Most members used only 1 payment method but the data shows the more payment methods used, the more likely the user is to churn.



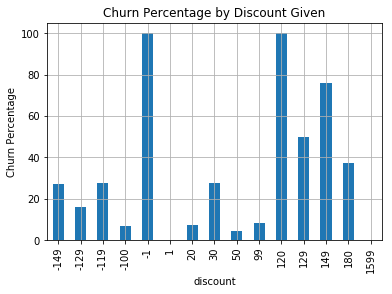
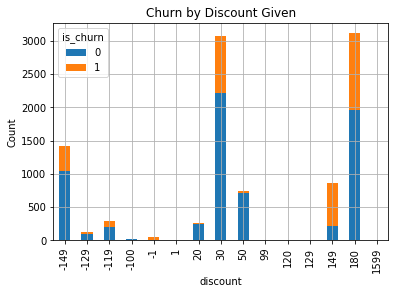
Most members used a 30-day payment plan and every other option reflected significantly higher churn rates. Some members used multiple payment plans and so a new variable was created, plans\_used:



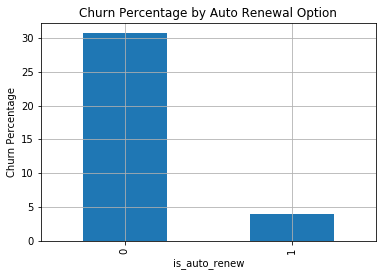
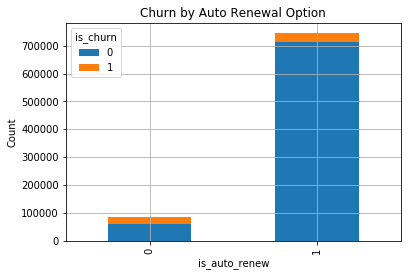
Just as with the variable ‘methods used,’ members who used more than one payment plan were likely to churn



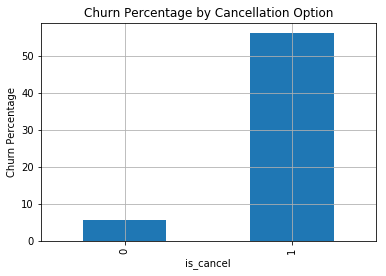
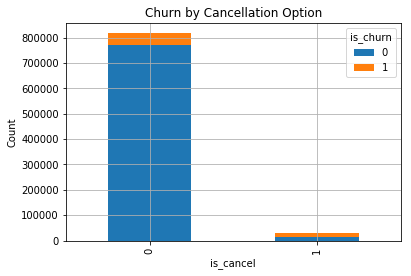
Most members purchased subscriptions at $99 and $149; most other options reflected significantly higher churn rates. Less than 0.5% of members didn’t pay the original plan price so we introduced a new variable, discount:



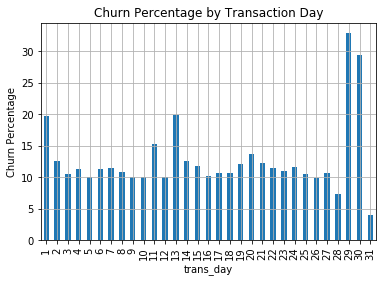
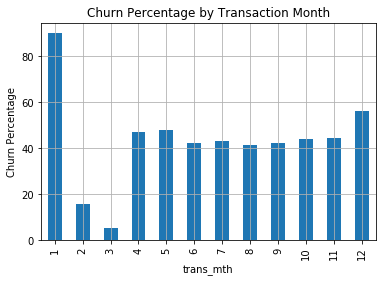
Generally, members who received discounts churned at a high rate. We also see that some members paid more than the plan price. A new variable, percent\_off, was also created for the model.



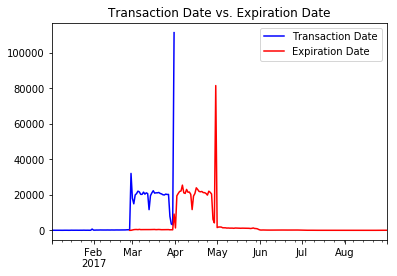
Members who have selected to auto-renew their subscriptions are much less likely to churn.



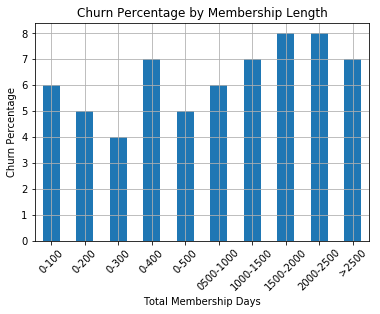
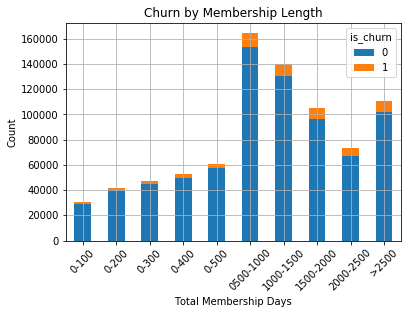
Members who actively cancel their subscription are unlikely to purchase a new membership within 30 days of cancellation.



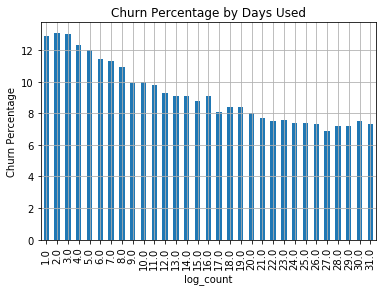
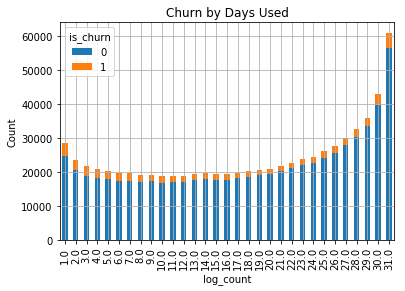
High churn rates are seen in members who purchase their latest subscription in the month of January. The data also shows that those who purchase their subscription towards the end of the month (29-30), are much more likely to churn than if purchased earlier in the month.



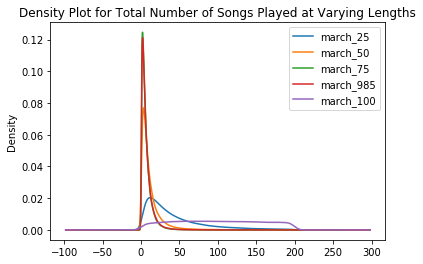
The transaction date and expiration date variables follow a very similar time-series pattern. Based on this information, a new variable was created to depict a members total subscription time (in days), mbr\_time, by subtracting the membership expiration date from the registration initialization date:



Users with a membership time between 1500-2500 days had the highest churn rate at 8%.

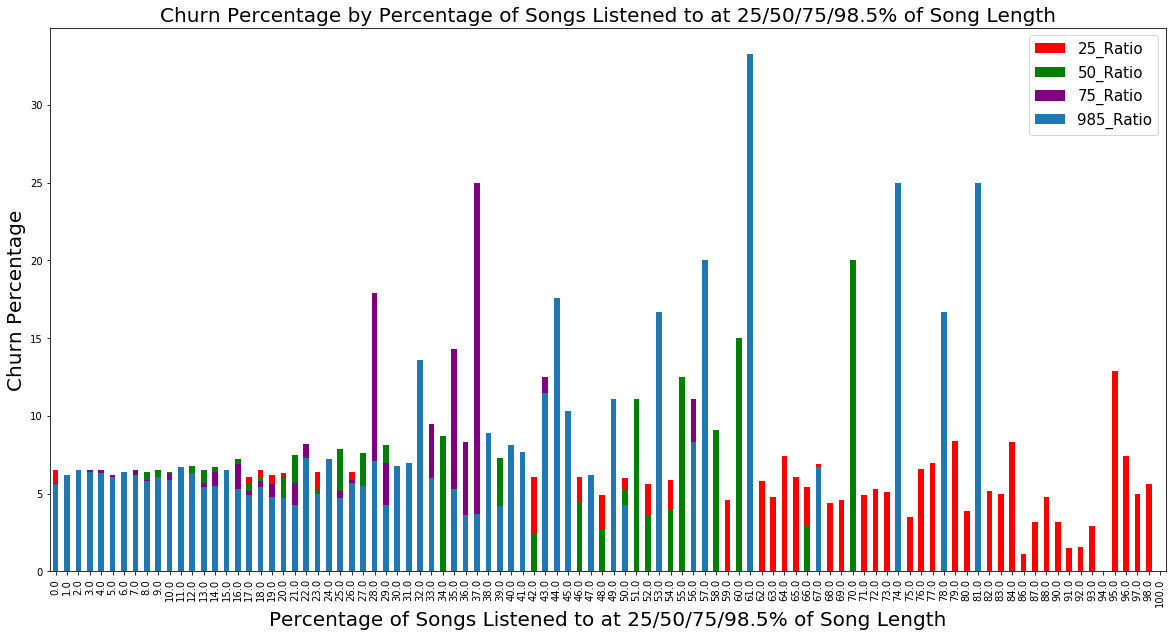


The logs file provided daily usage logs for each member. A new variable, log\_count, was created to depict the total number of days a member was actively using KKBox. After graphing the data, we see that the less number of logs or days used by a member, the more likely they are to churn.



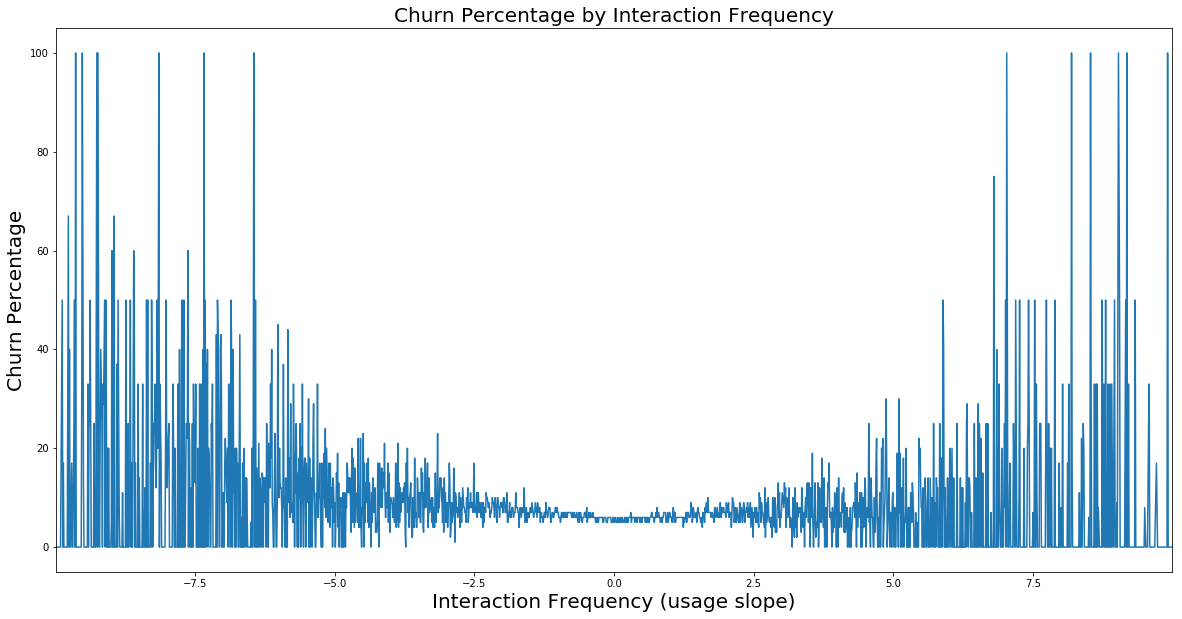
The logs file also provided the number of songs played at 25, 50, 75, 98.5, and 100% of the song length for the given day. The total songs each member listened to at 25, 50, 75, 98.5 & 100% of the song length for the entire month of March were calculated by grouping the data via member ID. Total number of songs played at their full length, march\_100, have a broader distribution compared to all other options. Tracks played at 25% the song length are the only other option that contributed past 25.

Based on the data above, I took the total number of songs played by a member at 25% of the song length and divided that by the total number of songs played at varying lengths throughout the entire month of March. The initial assumption was that if most songs were played at 25% or 50% of the song length, the service was not appealing to the user’s interest who often skipped to a new song a short way through. This calculation was made for the total songs played at 50%, 75% and 98.5% of the song length:



Generally, my initial assumption proved true. Members who listened to 0 to 25% of the songs at less than their full length churned at a steady rate of 6%. As the percentage of songs listened to at less than their full length increased, we see more variation and uptick in churn rate.

KKBox interaction frequency was also calculated for each member. This variable was measured by plotting the number of songs listened to over time in the month of March and calculating the slope of the distribution. An assumption made was if a member listened to fewer and fewer songs as the month progressed (negative slope), the more likely they will churn.



The data shows that people who interacted with KKBox at regular, steady intervals (slope = 0) were the least likely to churn. It also shows that those who interacted with KKBox less and less through the March (negative slope) were likely to churn also surprisingly along with those individuals who interacted with the service more frequently as the month progressed.

A number of other new variables were created based off of the information in the log file:

1. repeated\_songs: songs played more than once throughout the month of march; calculated by subtracting the total number of unique songs from the total number of songs played in the month of March
2. avg\_unq\_songs: the average number of unique songs played each day a user was active; calculated by dividing the number of total unique songs played by the number of logs/days active
3. total\_secs\_march: total seconds a user was actively playing a song/using KKBox throughout the month of March; calculated by summing the total\_secs variable for each unique user

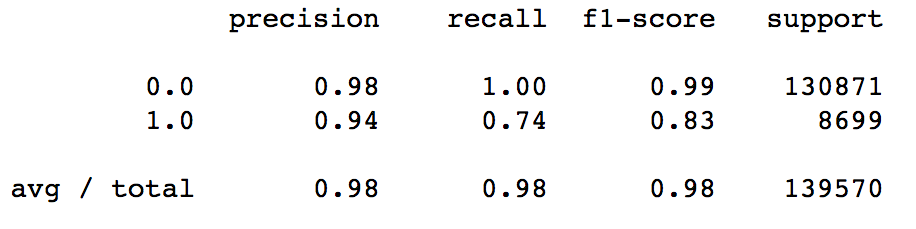
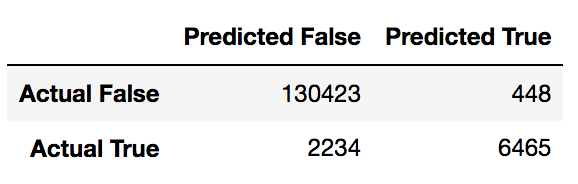
**IV. Data Wrangling**

The train file and members file were first combined to measure churn with respect to independent variables such as ‘city,’ ‘gender,’ ‘bd,’ and ‘registration method.’ As mentioned previously in the report, the ‘gender’ and ‘bd’ (age) variables were removed. Of the 970,960 users in the train file, 860,967 users had corresponding member information. After the train file and members file were combined, the transactions file was prepared for inclusion in the master dataset to be used in the model. The transactions file had 1,431,009 individual transactions with multiple transactions made by the same user. After the new/additional variables were created for each member, the transactions file was grouped by user id, then sorted by the user’s membership expiration date. The case with the latest expiration date was retained, and the duplicate transactions were removed and combined with the master dataset. Of the 860,967 users, all had at least one transaction file. After the transactions file was merged with the master dataset, the logs file was prepared for inclusion. The logs file had 18,396,361 log entries, multiple logs from the same user. After the new/additional variables were created for each member, the logs file was grouped by user id, then sorted by the user’s latest log. The case with the latest log date was retained, and the duplicate logs were removed and combined with the master dataset. Of the 860,967 users, 697,847 had at least one log file.

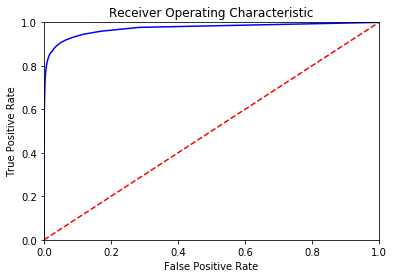
The final dataset to be used in our model included 36 independent variables for 697,847 unique users.

**V. Model**

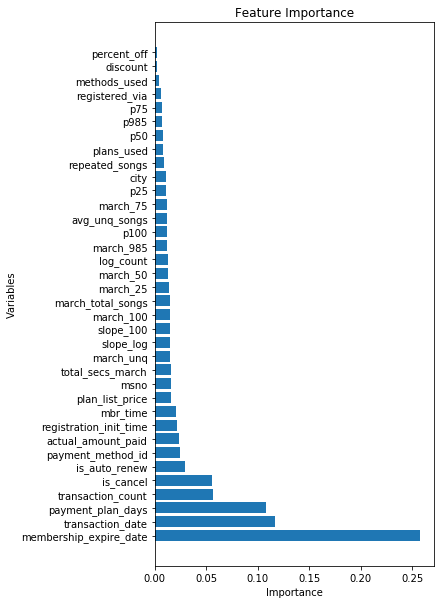
Of the KNN, logistic regression, and random forest classifier models used, the random forest classifier performed the best. An 80/20 train test split was selected for this model starting with 10 n\_estimators. The model produced an overall churn/no-churn prediction accuracy of 98%. Models with 50 and 100 n\_estimators were also tried with no significant changes to the overall accuracy.



6465 out of 8699 positive churn cases were accurately predicted, producing a 74% recall score. Essentially, this model accurately predicted 74% of the churners in the test set.



The Area Under Curve was measured at 87%



2.98%

5.56%

5.65%

10.81%

11.68%

25.79%

1.61%

1.61%

2.06%

2.17%

2.36%

2.41%

1.22%

1.24%

1.24%

1.26%

1.28%

1.41%

1.44%

1.45%

1.45%

1.48%

1.49%

1.57%

0.19%

0.26%

0.43%

0.59%

0.68%

0.71%

0.80%

0.84%

0.89%

1.10%

1.13%

1.16%

Of the 36 independent variables, membership expiration date, transaction date, payment plan days, whether the member actively cancelled their subscription, and total transaction count were the top 5 important features in predicting churn in the model.

Suggestions for model improvement:

1. Provide churn history beyond those who left in March 2017

2. Provide log files for the last year, not just for the month of March

3. Provide assessment on search and recommendation engines ideally by ability to cater to each individual user

**VI. Important Findings**

The most significant variables already given in the initial dataset when predicting churn include membership expiration date, latest transaction date, payment plan days, whether a member actively cancels their current subscription, and whether a member opts for the auto-renew feature. New variables to consider are transaction count, total membership time, and interaction frequency.

3 models were tested: KNN, logistic regression, and random forest classifier and of all 3, the model that performed the best with the given set of parameters was the random forest classifier. The random forest classifier produced a 75% recall score while the other two produced a less than 5% recall score for members who churned

Members who listened to a majority of their songs at less than their full length were more likely to churn than those who saw most of their songs through. Although this was observed in the data, it proved to be of little impact in the random forest classifier model.

**VII. Further Research for KKBox**

Although the percentage of songs listened to at less than their full length had little impact in the random forest classifier churn model, I would recommend KKBox look further into this variable by assessing their recommendation and song-to-user matching performance. For those who listen to a majority of their songs at 0-50%, it may be possible that the service is inaccurately predicting songs the user is interested in.

Another potential target group is those collective members who show less frequent interactions with the service as the month progresses. A mid-month assessment might identify these users early on.

One final recommendation is to reduce the payment plan length to less than 100 days and/or eliminate plans that cost $300 or more. Payment plan lasting over 100 days and costing more than $300 showed close to a 100% churn rate.

Competition Link: **https://www.kaggle.com/c/kkbox-churn-prediction-challenge/data**