Capstone Project 1 - Milestone Report

Objective: Predicting whether a user will churn after the subscription expires.

Evaluation: The evaluation metric for this project is Log Loss

$$\text{logloss}= \frac{-1}{N}\sum_{i=1}^{N}(yi\log(pi) + (1 - yi)\log(1 - pi))$$

where N is the number of observations, *log* is the natural logarithm, y_i is the binary target, and p_i is the predicted probability that y_i equals 1.

Client: KKBOX is Asia's leading music streaming service, holding the world's most comprehensive Asia-Pop music library with over 30 million tracks, supported by advertising and paid subscriptions. Currently, the company uses survival analysis techniques to determine the residual membership life time for each subscriber. By adopting different methods, KKBOX anticipates they'll discover new insights to why users leave so they can be proactive in retaining users.

Data: For this analysis, I will be using the KKBox's churn prediction dataset which was publicly available on Kaggle. Data is distributed across 4 different csv files as follows:

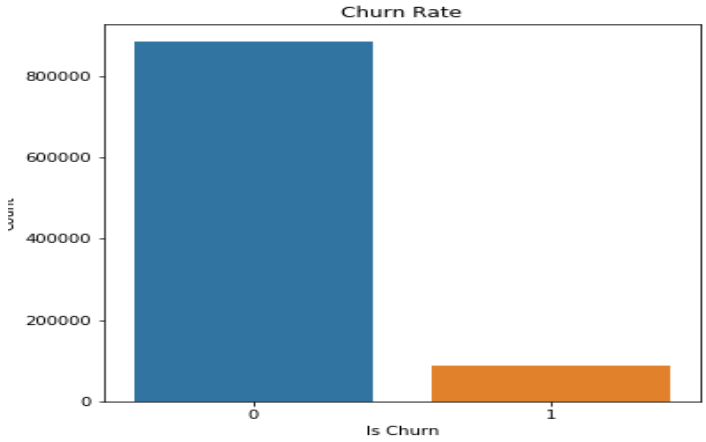
train_v2csv: The train set, containing the user ids and whether they have churned.
transactions.csv: The transaction set contains all the payment details till feb-2017.
transactions_v2csv: The transaction set contains all the payment details of march-2017.
members.csv: The members set contains the user information.
user_logs.csv: The user logs set contains the daily user logs describing listening behaviors of a user for the month march-2017.

Data Wrangling: All the .csv files were imported & converted as data frames. Then we concatenated the two transactions data frames ('transactions_1' & 'transactions_2') making sure we have all the transaction details in one data frame 'transactions'.

Memory reduction: Because all the data frames are very large in size and are using a lot of memory, we performed memory reduction to reduce the memory usage by changing the data types of some columns and splitting the date column into three different columns - year, month & day columns respectively. Finally, we made sure none of the data is stripped from the original data frames and then we dropped the date columns. This helped us to reduce the memory usage to half of its usage than before.

Data Exploration in Train

Churn Percentage: In the train data frame, we have the 'msno' – which are unique id no's for given to each customer & their churn detail are present.

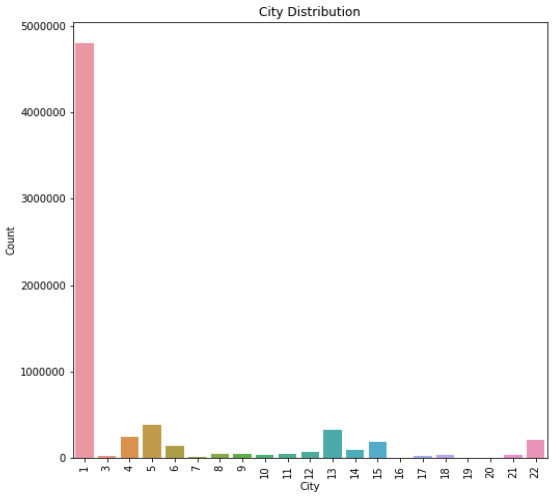


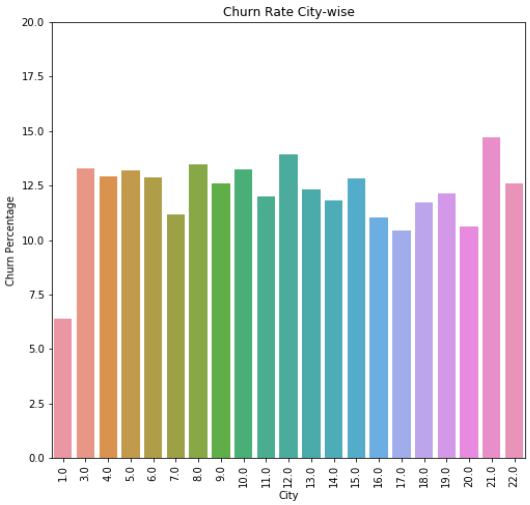
Only 9% people have churned which looks so successful, making it a highly imbalanced classification problem.

Data Exploration in Members & Train

Here we're merging train & members data sets using left join. After the merge, because not every member is present in the members data set some null values are observed.

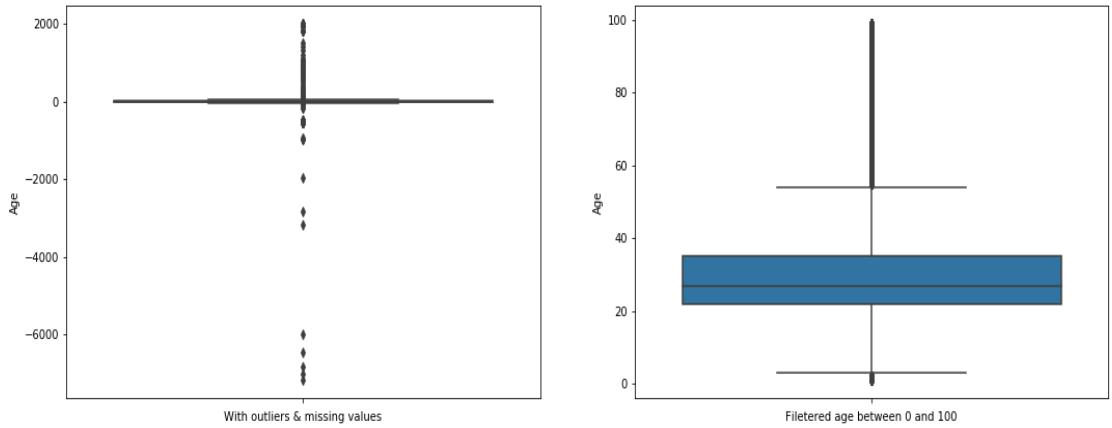
City:



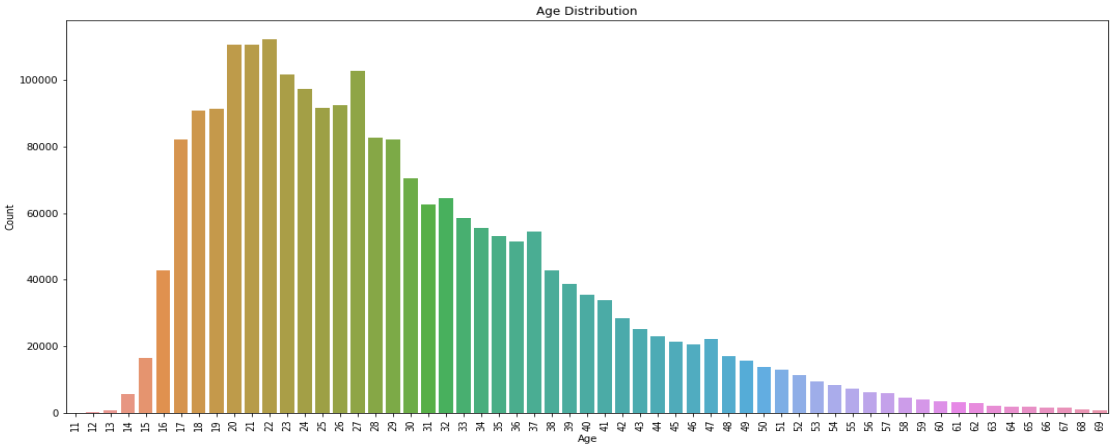


- There are total of 21 cities, there is no city '2'. We observe majority from city 1. Everything else looks similarly unpopular.
- The cities are quite similar at churn rates with the crucial exception of city 1. In this most popular city, the churn rate is significantly lower when compared to other cities. This has a big impact on the overall churn rate.

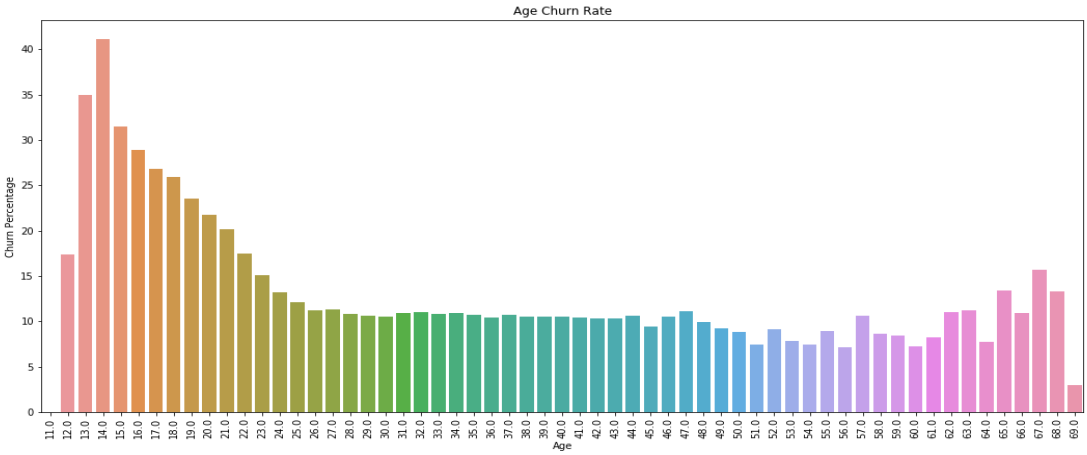
bd (age column):



- In the bd (Age) column we observed it has lot of values set to 0 and there are some outliers ranging from '-7168' to '2016'.

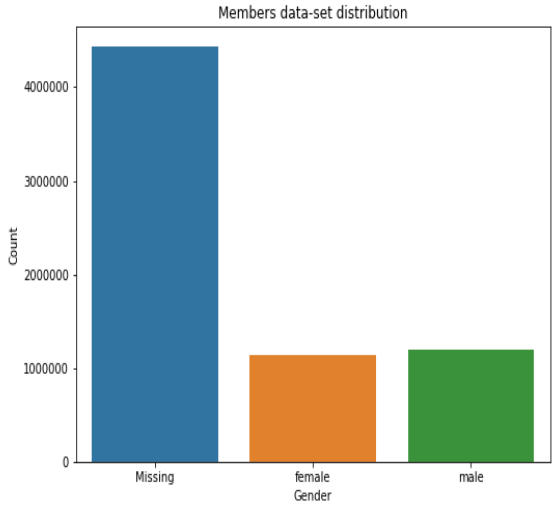


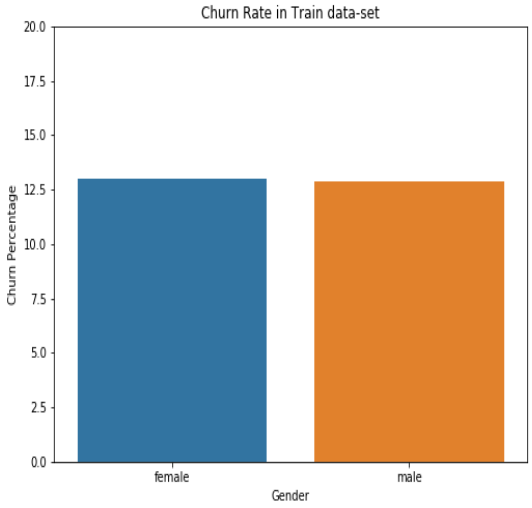
- Later we filtered the bd (Age) column between 10 and 70 and observed that most of the customers are aged between mid-teens to mid 40's.



- we find that younger users on average appear to be more likely to churn.

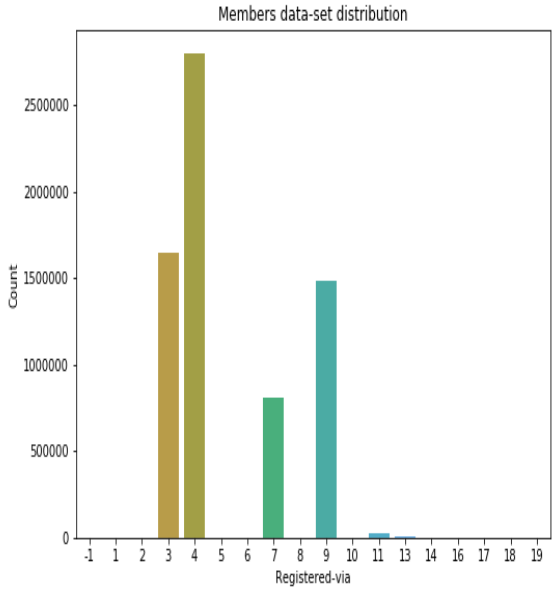
Gender:

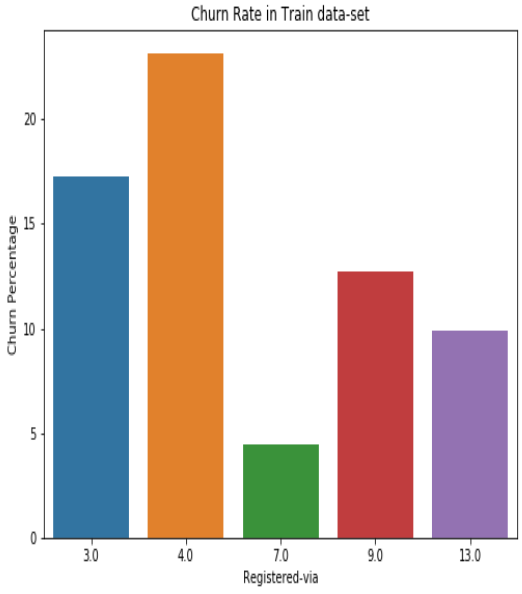




- Around 60% of the data is missing after the merge. With the data we have it seems both male and female are churning quite similar. We have to see how to deal with the missing values in future analysis.

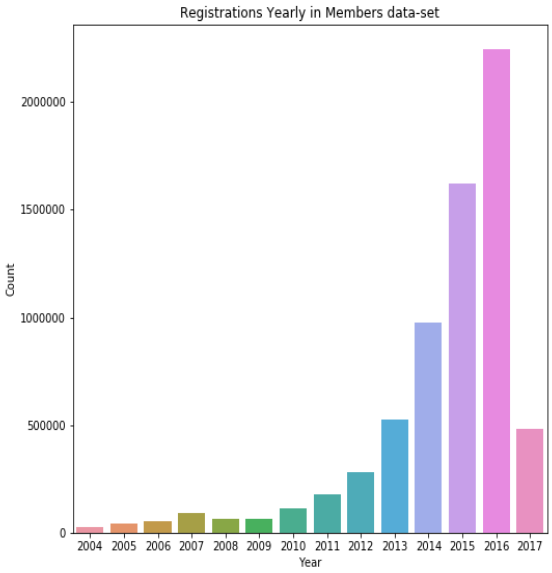
Registered-via:

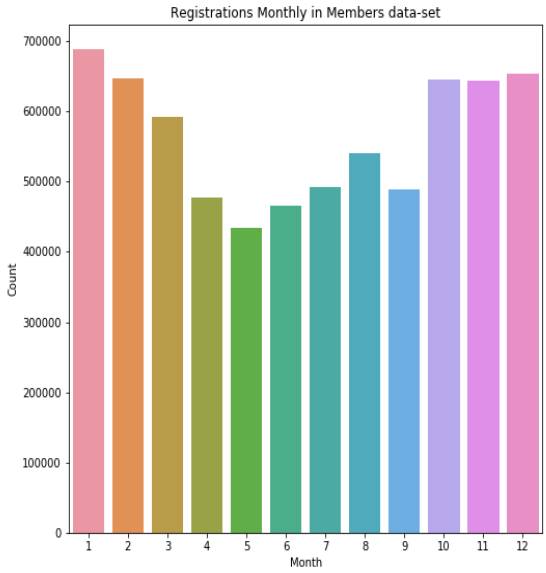




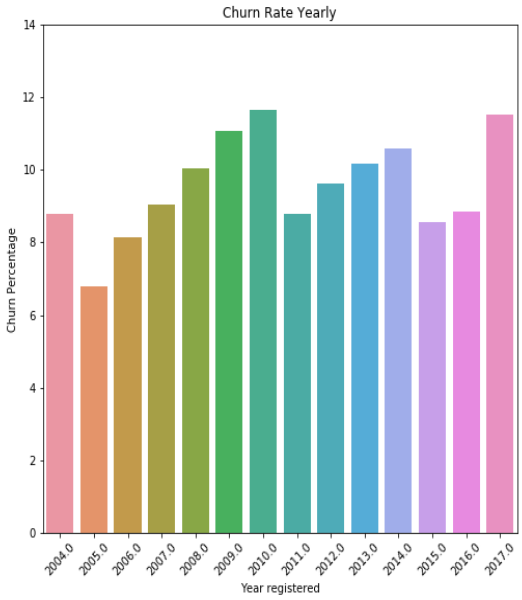
- There are 5 classes ('3', '4', '7', '9', '13') listed as registration method in x1. There are also some additional classes in the members dataset. As we merged the train and members they are missing. Also, there are noticeable differences in terms of registration method. Method ‘7’ appears to correlated with the most loyal users, while method ‘4’ has slightly higher churn rate of all.

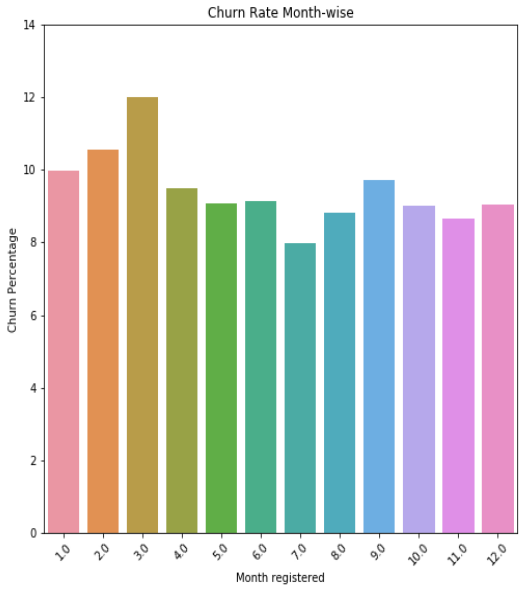
Registration & churning trends yearly & monthly:





- we observed that popularity started rising slowly after 2009 and it started to increase strongly from 2012. Registrations are high during the year end (oct, nov, dec) and year starting months (jan, feb).

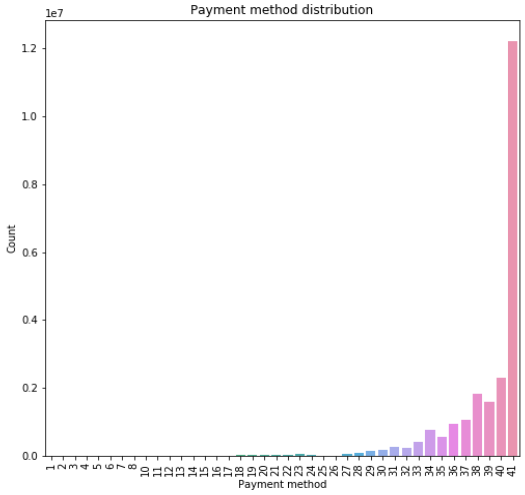


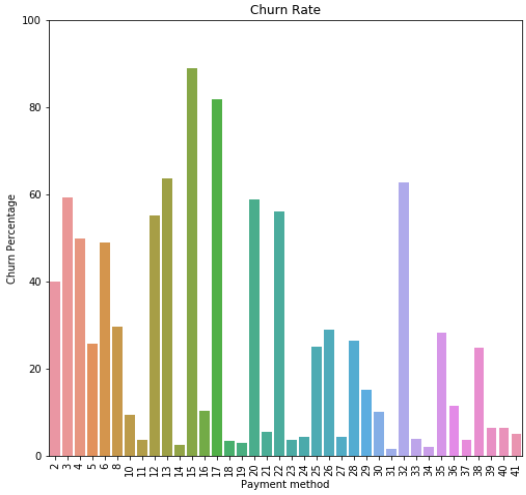


- The churn rate doesn't seem to follow a trend. It's been consistent and fluctuating between 8% to 12% regardless of increase in number of registrations every year.

Data Exploration in Transactions & Train

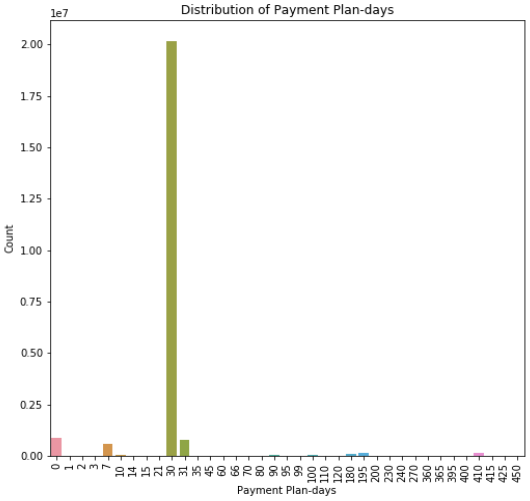
Here we're merging train & transactions data frames.
Payment method-id:

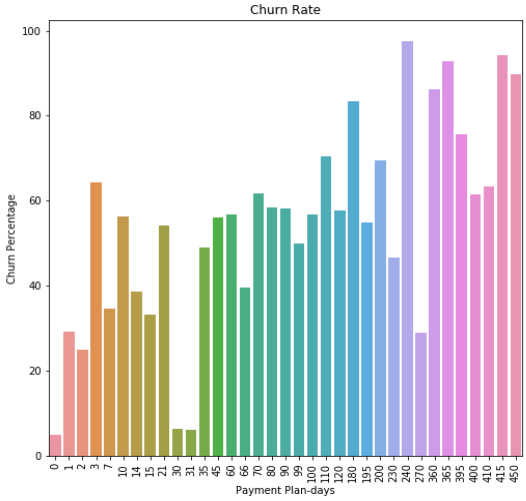




- There are 40 payment methods (method '9' is missing) and the payment - method '41' is by far the most popular one.
- Some payment methods are clearly associated to more loyal users than others. Note that several categories suffer from low-number statistics and the corresponding large churn rate bars. However, the vastly popular payment method "41" is easily in the top 10 of lowest churn.

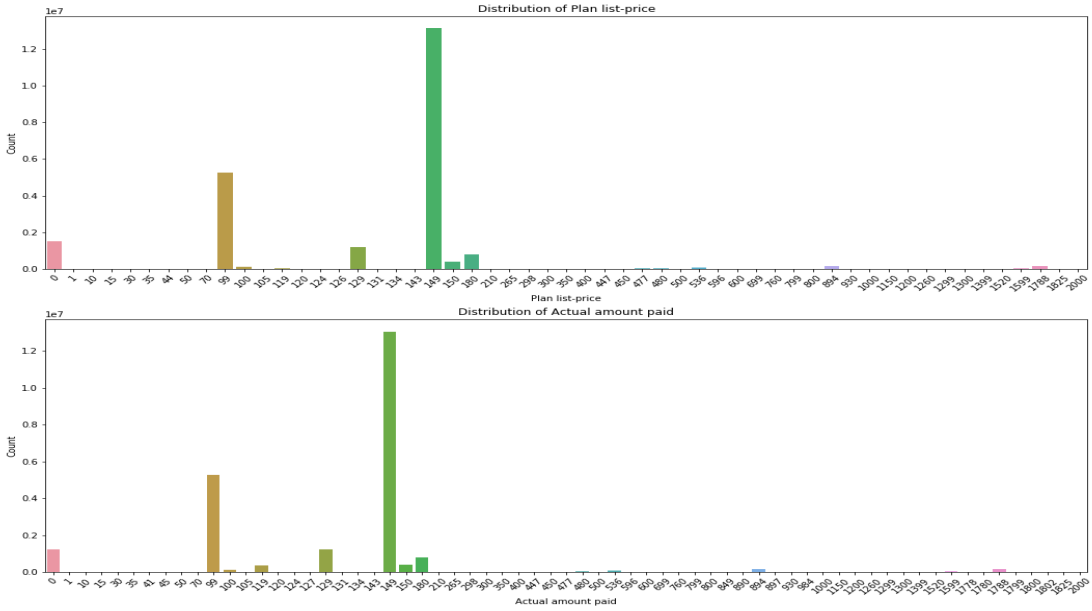
Payment plan days:



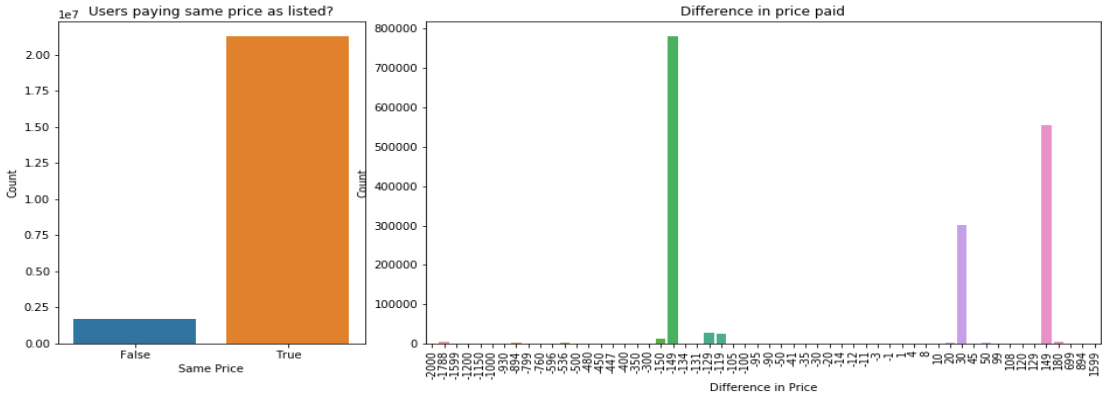


- The payment plan duration categories show strong differences in churn percentage. The lowest churn numbers (around 5%) are associated with the 30-days, 31-days, and the 0-day memberships (surprisingly). The churn percentage for next widely used plans 7-days is 35% & 410-days is 63%. For this feature as well, there are categories with low number statistics and large churn rate bars.

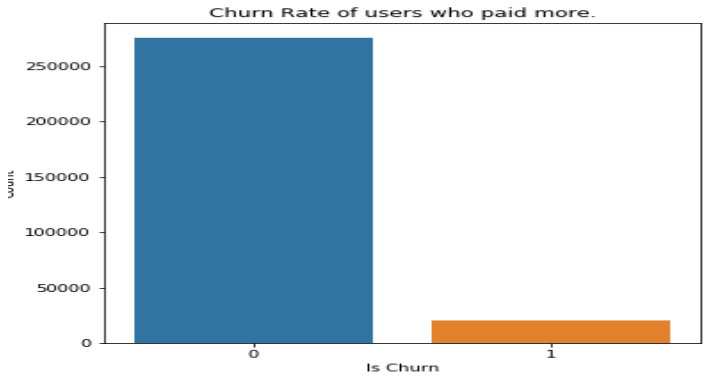
Plan list price & Actual amount paid:



- The overall distributions of planned vs actual payment are very similar, even though differences are slightly visible e.g. for 119 NTD. Since both features have the same discrete payment values we can directly compare their frequency.

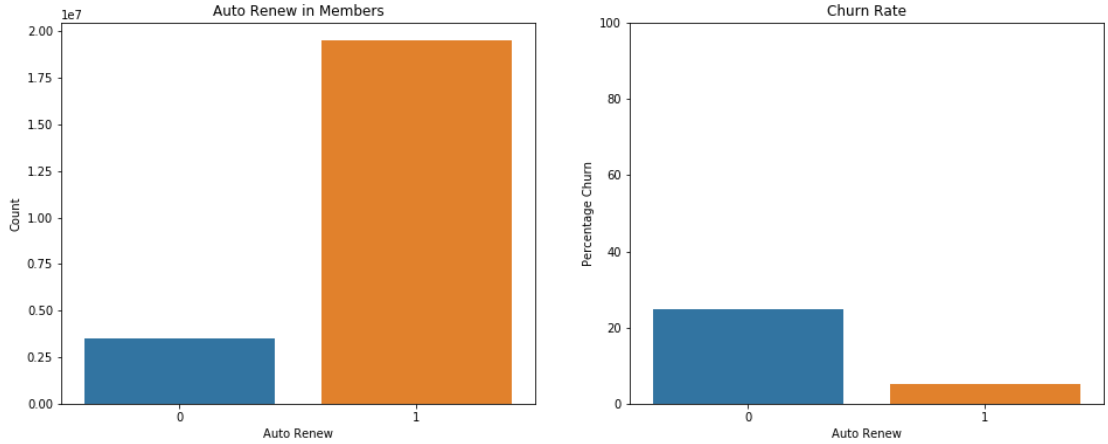


- Interestingly here, in most of the cases the users ended up paying more.



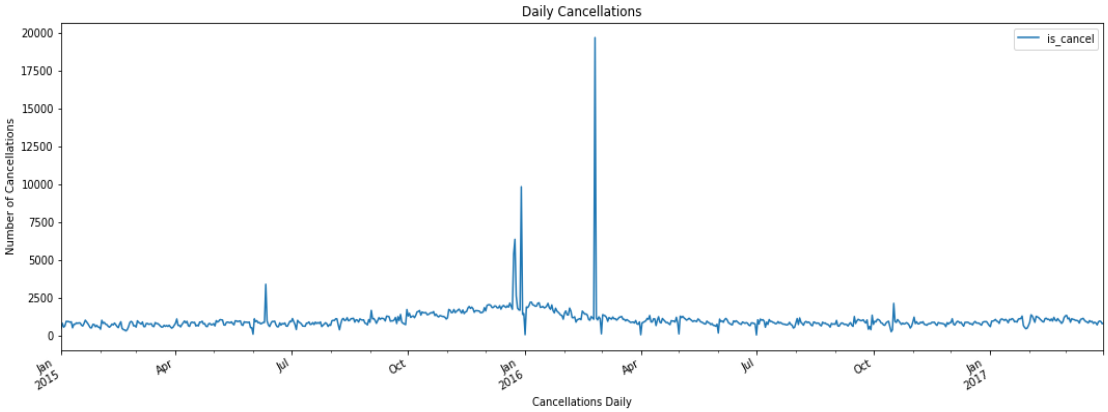
- There isn't any surprising trend in churn rate of users who paid more.

Auto-renew:

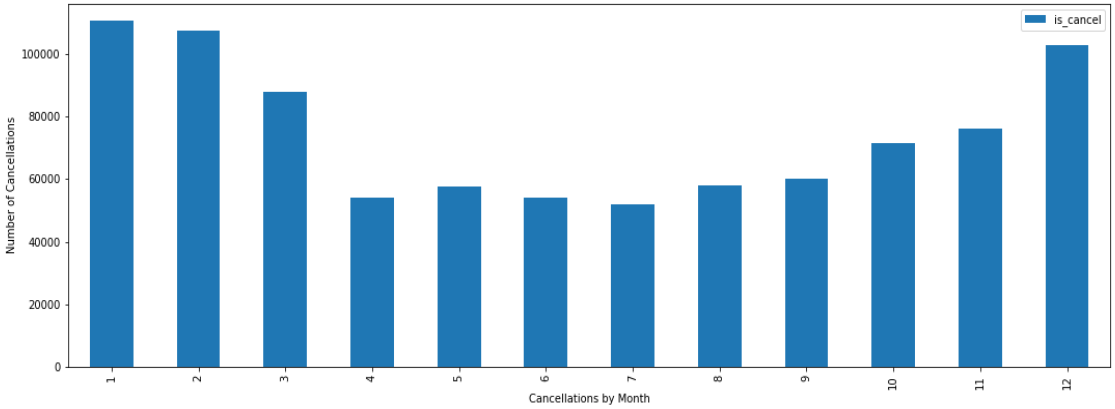


- The vast majority of users have automatic renewal of their subscriptions enabled and users who did not choose to auto renew were clearly more likely to churn.

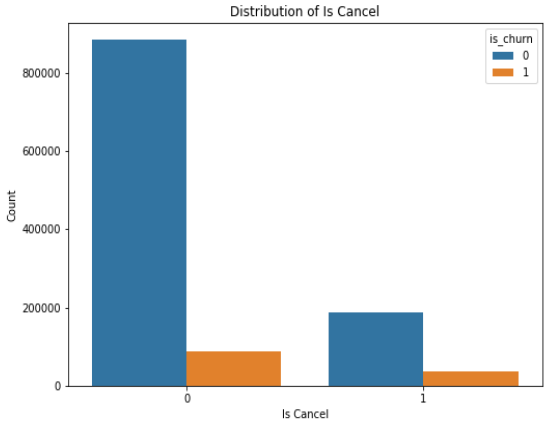
Is-cancel:

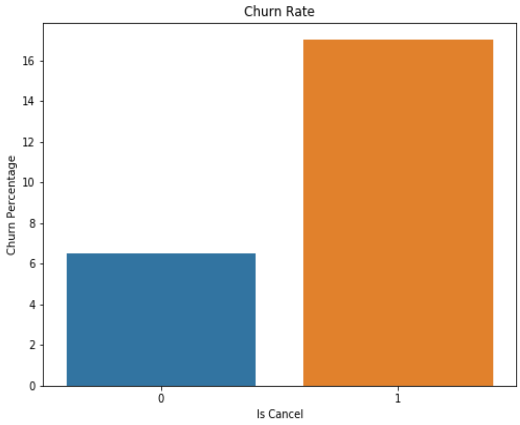


- The cancellation trend seems quite consistent overall with a couple of spikes in jan & mar 2016.



- Cancellations are high in the months of dec, jan, feb & mar. Rest all the months seem very similar.



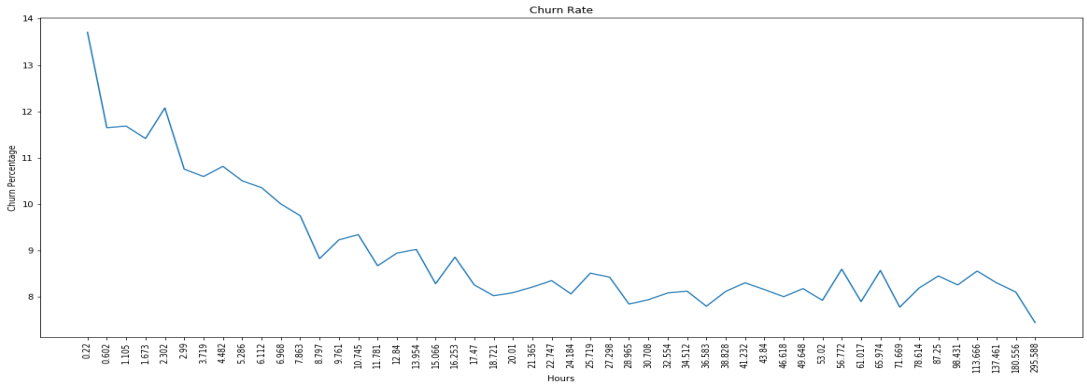


- Not all users who cancelled their subscription are churning. Majority of the cancelled users are re-subscribing within a month.

Data Exploration in User-logs & Train

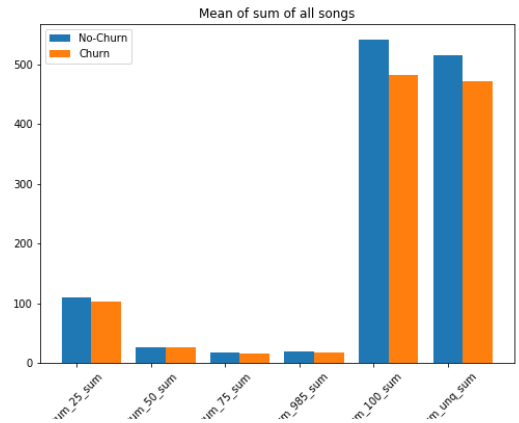
Here we merged the train data frame with user-logs which contains listening behaviors of a user for march.

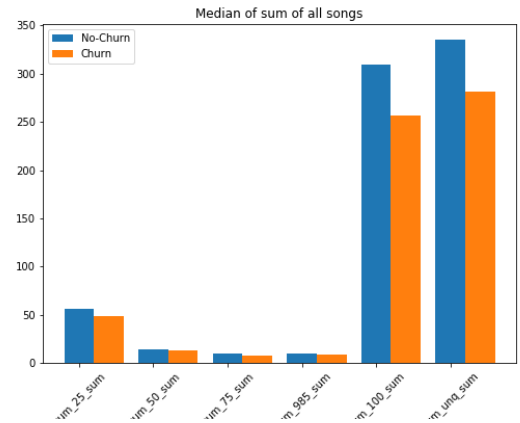
Life span of Churn Users vs No-Churn Users:



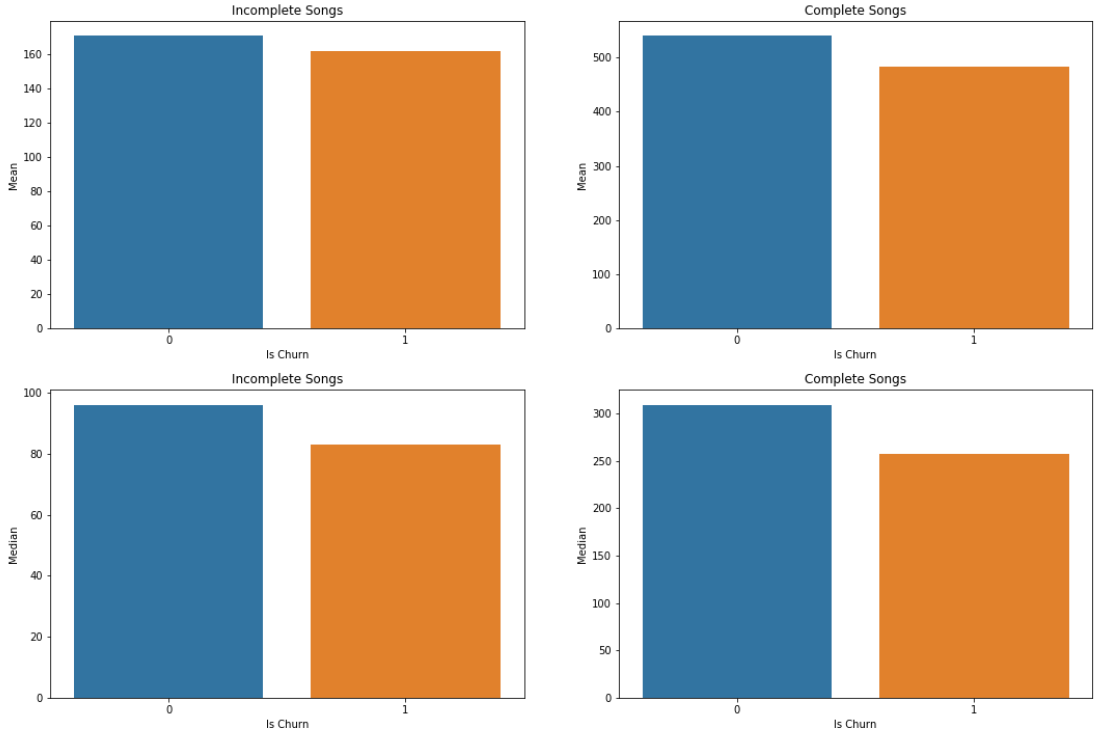
- Here we observed that users with less life span are slightly churning more.

Mean & Median of sum of all songs by Churn Users Vs No-Churn Users:



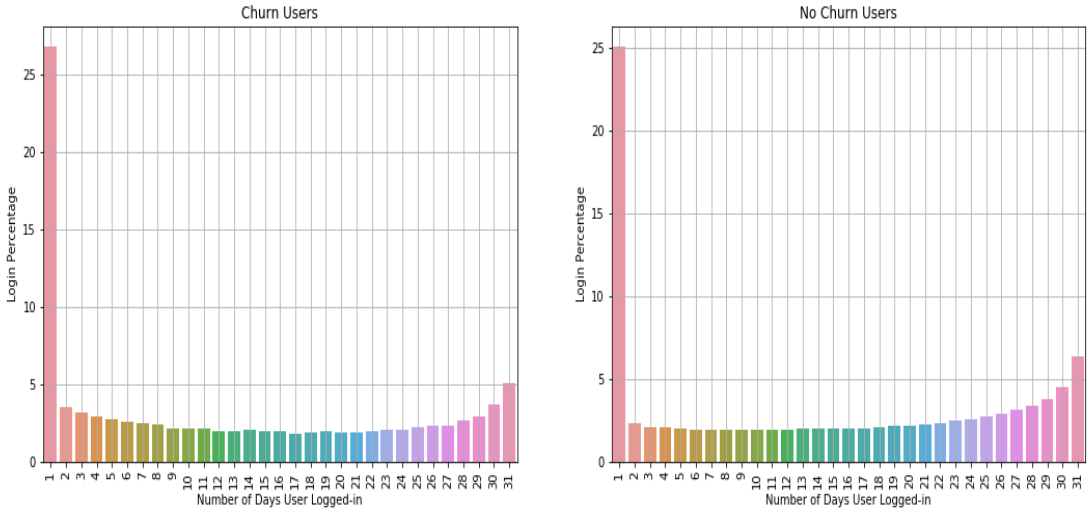


Number of complete songs against incomplete songs:



- Hence, it is clear from the above two plots that Churn users are listening to less number of songs when compared to No-Churn users.

Number of days user has logged in:



- Here, we observe that both the churn users & not churn users log-ins are very similar.