Capstone Project 1

KKBox Customer Churn Prediction

How will I retain my customers?

- Most companies utilize subscription business model, including KKBox
- KKBox shared their customer data on Kaggle to learn more about predicting churn rates.
- **Inquiry:** Can we predict if a user will make a new service subscription transaction within 30 days after the current membership expiration date based on their behaviors and interactions with the product?

KKBox is a Taiwan-based music streaming software

Data Provided

Train Dataset
msno
is_churn

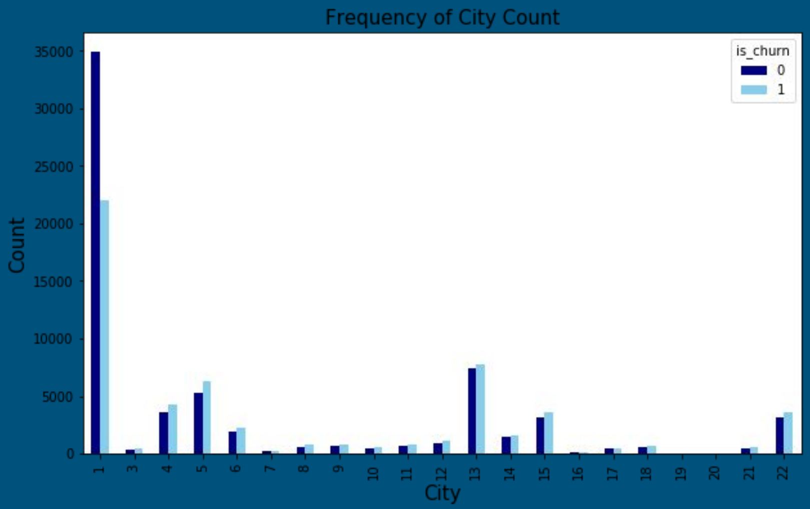
Member Data
msno
city
bd
gender
registered_via
registration_init_time

Transactions Data
msno
payment_method_id
payment_plan_days
plan_list_price
actual_amount_paid
is_auto_renew
transaction_date
membership_expire_date
is_cancel

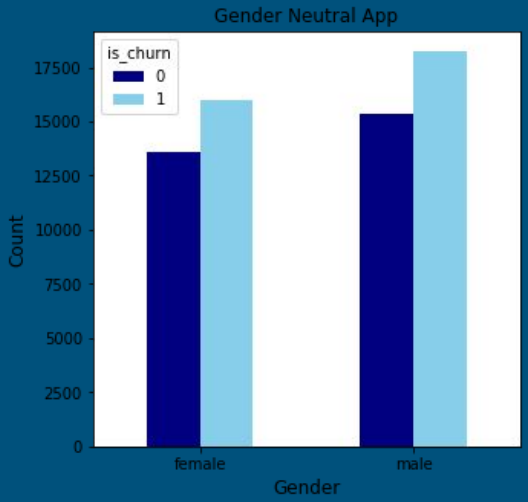
User Log Data
msno
date
num_25
num_50
num_75
num_985
num_100
num_unq
total_secs

Exploratory Data Analysis: Churn vs Not Churn

Both groups are similar in certain demographics.



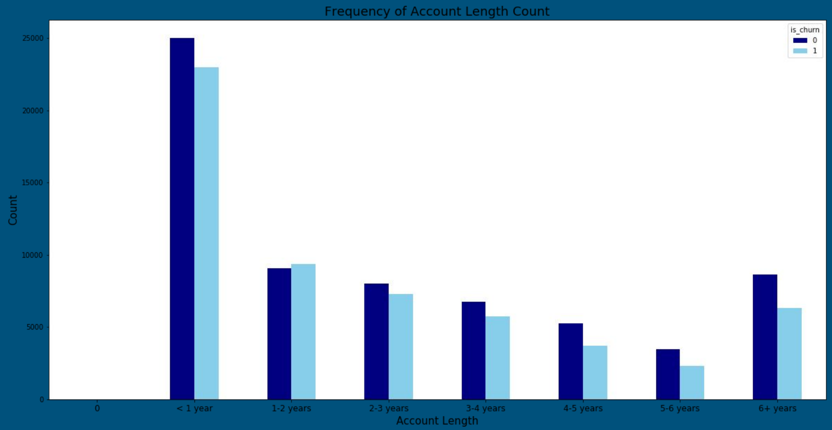
Most live in city 1



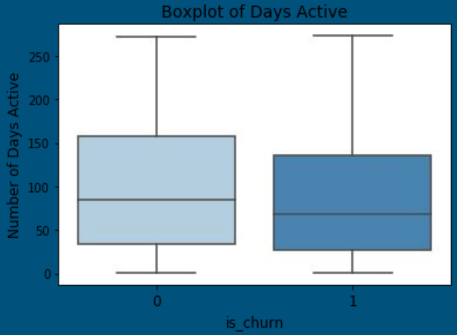
Even gender ratio

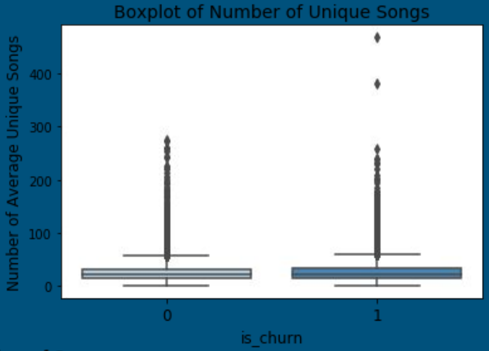
Exploratory Data Analysis: Churn vs Not Churn

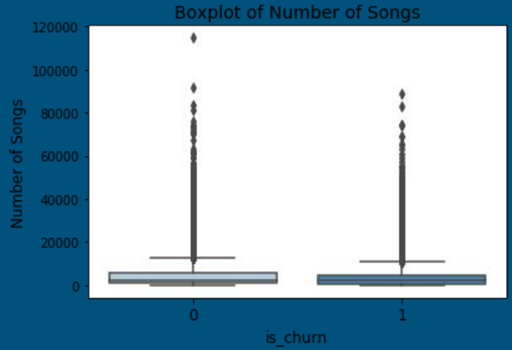
Their behaviors with the app was also similar.



Similar length of account distribution



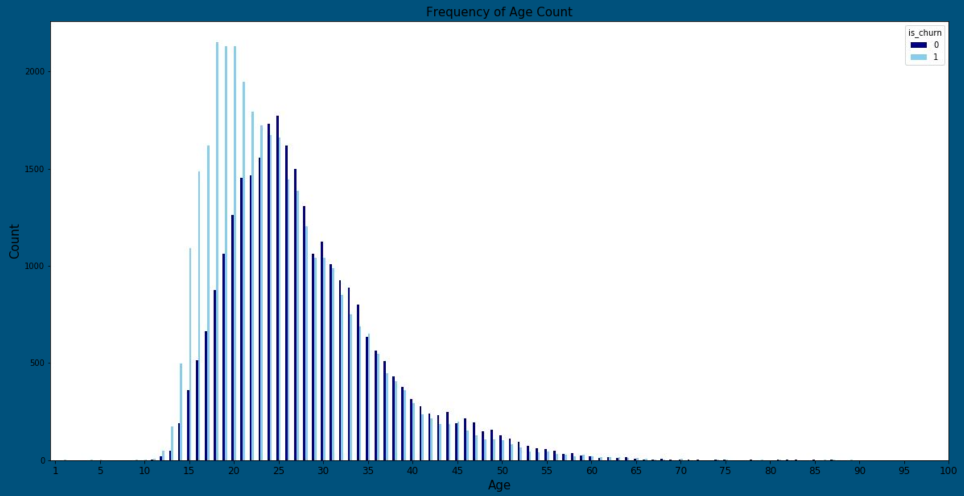




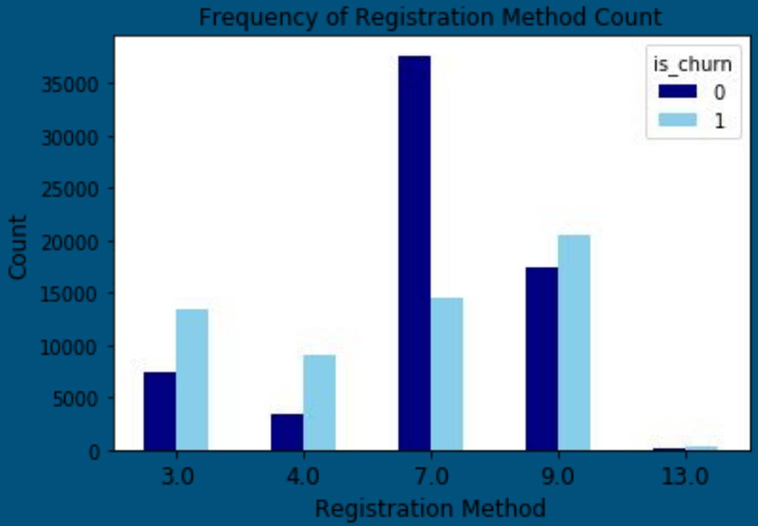
There's not much difference in the distribution of number of days and songs.

Exploratory Data Analysis: Churn vs Not Churn

A few demographics features did vary between the 2 groups.



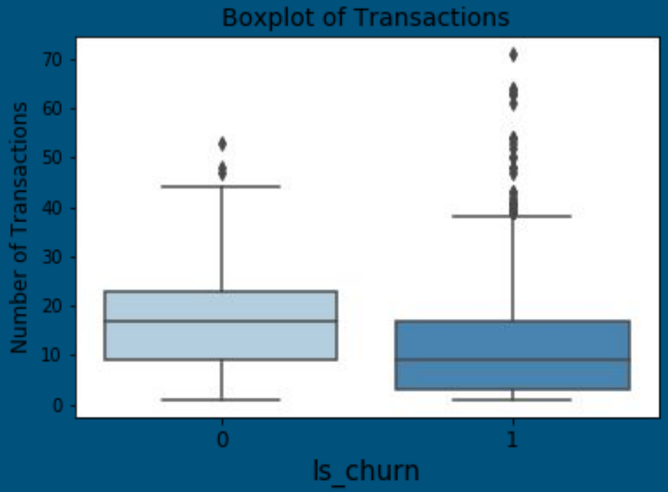
Most of churned users are in their early 20s while users who did not churn are mostly in their late 20s.



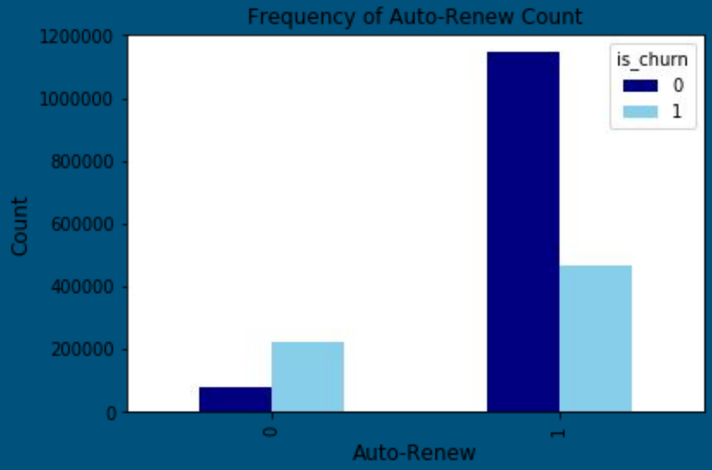
Users who did not churn mostly signed up via registration method 7 while most churned users registered via method 9.

Exploratory Data Analysis: Churn vs Not Churn

They also differed in the monetary transactions made.



Those who did not churn seemed to have made more transactions than those that churned.



The users who used the auto-renew feature tend to be less likely to churn.

Inferential Statistics

Although the distributions may seem to look like they differ visually, these hypotheses should be tested to see if the difference is statistically significant

Method used: Permutation (Bootstrap) Hypothesis Testing

Null Hypothesis	Mean Difference	95% Confidence Interval	P-value	Reject $H0$
There's no difference in the age between user who churned and did not churn.	2.574	[2.437, 2.711]	0	True
There's no difference in auto-renewal rate between user who churned and did not churn.	0.441	[0.437, 0.445]	0	True
There's no difference in the number of transactions between user who churned and did not churn.	5.520	[5.431, 5.609]	0	True
There's no difference in the number of unique songs listened per session between user who churned and did not churn.	-1.093	[-1.299, -0.888]	0	True

Building the Classification Model

To take advantage of the temporal aspect of the data, we will split the dataset into 2 separate time periods for training and testing data:

- Training Data: February (2017/02/01 - 2017/02/28)
- Testing Data: March (2017/03/01 - 2017/03/31)

Feature Engineering

Transactions Data
msno
payment_method_id
payment_plan_days
plan_list_price
actual_amount_paid
is_auto_renew
transaction_date
membership_expire_date
is_cancel

Transactions Data
msno
avg_actual_amount_paid
avg_is_auto_renew
avg_is_cancel
avg_payment_plan_7
avg_payment_plan_30
avg_discount_received
trans_count

Member Data
msno
city
bd
gender
registered_via
registration_init_time

Member Data
msno
age
gender_male
registered_via
days_since_reg
city_3
city_4
city_5
city_6
city_7
city_8
city_9
city_10
city_11
city_12
city_13
city_14
city_15
city_16
city_17
city_18
city_19
city_20
city_21
city_22
registered_via_4
registered_via_7
registered_via_9
registered_via_13

Feature Engineering

User Log Data
msno
date
num_25
num_50
num_75
num_985
num_100
num_unq
total_secs

→

User Logs Data
msno
avg_num_25
avg_num_50
avg_num_75
avg_num_985
avg_num_100
avg_num_unq
avg_rate_num_25
avg_rate_num_50
avg_rate_num_75
avg_rate_num_985
avg_rate_num_100
total_secs
log_count
days_since_login

In total, there are 48 features.

Data Preprocessing

Not all users in the training data and testing data that had churn information had records of transactions or user activity.

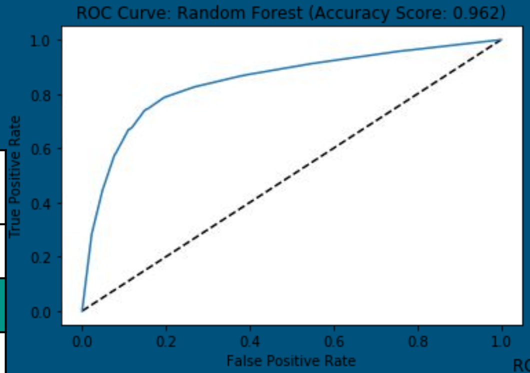
- If the user did not have any transactions data, the transactions count feature was filled with 0.
- If the user did not have any user log data, the days since log in was filled with 31.
- All the other features were filled with the sample mean for the column.

In the end, all the values were normalized.

Selecting Classification Model

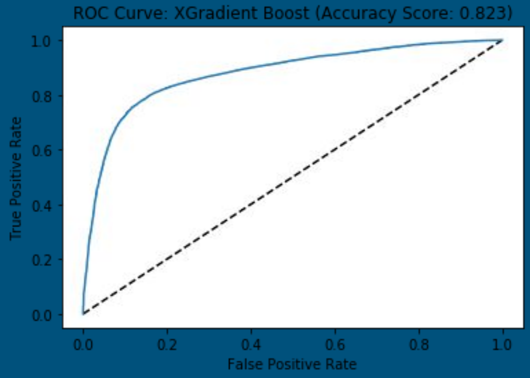
Fit 5 different classification models on the data using the default parameters and utilized log loss as the model metric.

Classification Model	Average Log Loss
Logistic Regression	0.463
Random Forest	1.464
Adaptive Boosting	0.681
Gradient Boosting	0.428
Extreme Gradient Boosting	0.427



Random Forest
Accuracy Score: 0.962

XGBoosting
Accuracy Score: 0.823



Fine Tuning Hyperparameters

Utilized GridSearch cross validation to select best parameters for extreme gradient boosting model:

- colsample_bytree = 0.8
- learning_rate = 0.05
- max_depth = 8

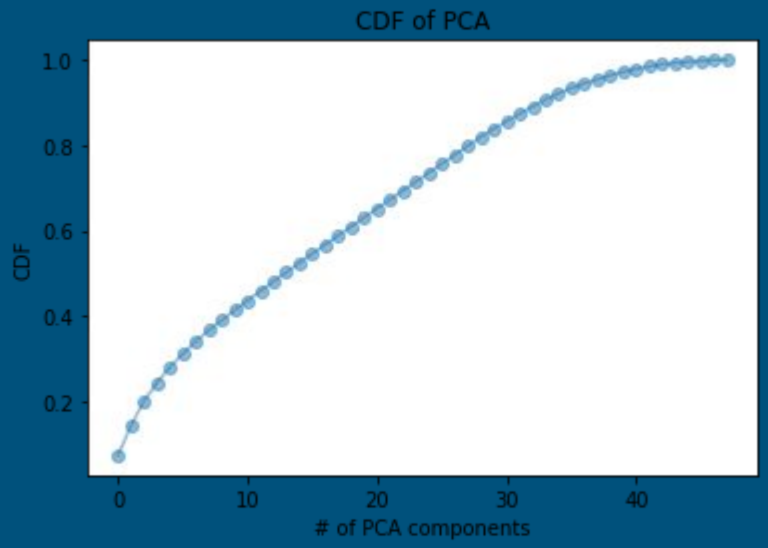
Feature Selection

Feature selection is the method of selecting the most important features to be used in the model. The scoring method used to rank each feature is Mutual Information.

Number of Top Features Kept	**Log Loss Score**
30	0.3919
32	0.3901
35	0.3896
40	0.3907
42	0.3899
All 48	0.3887

Principal Component Analysis

Unlike feature selection, PCA reduces dimensionality through a linear transformation into a lower dimension.



Number of PCA Components	Log Loss Score
29	0.4062
30	0.4067
31	0.405
32	0.4055
33	0.4038
34	0.4023
35	0.3998
36	0.3974
37	0.3958

Final Results

- Final Model
  - XGBoosting(colsample_bytree = 0.8, learning_rate = 0.05, max_depth = 8) using top 35 features
- Results
  - February Train Data
    - Accuracy: 83.7%
    - Log loss: 0.3934
  - March Test Data
    - Accuracy: 78.9%
    - Log loss: 0.4655

Limitations and Further Opportunities

- Model can be improved using server with higher CPU
  - Solving issue of unbalanced dataset by assigning higher weight to users that churned so that all the data can be used to train model
  - Fine tune more hyperparameters
- Model can be improved in a better understanding of the variables provided
  - Extreme negative values in columns that should strictly be positive (age and total seconds listened)

A model with a low log loss allows accurate prediction of the likelihood of churning. Clustering similar users together in terms of demographics and activity can pave the opportunity of a recommender system that provides the most effective promotion for the user.