# KKBOX Churn Prediction

## 1) Problem Statement


KKBOX is Asia's leading music streaming service, holding the world's most comprehensive Asia-Pop music library with over 30 million tracks. They offer a generous, unlimited version of their service to millions of people, supported by advertising and paid subscriptions. This delicate model is dependent on accurately predicting the churn of their paid users. For a subscription business, accurately predicting churn is critical to long-term success. Even slight variations in churn can drastically affect profits. Churn prediction is a process where companies use ML models to forecast which customers are at the highest risk of ending their patronage. Churn prediction uses customer data based on user behavior and usage.

Different organizations define churn differently. For KKBOX, a user has churned if he has not renewed his subscription within 30 days of his current subscription expiry date. By adopting different methods, KKBOX anticipates they'll discover new insights as to why users leave so they can be proactive in keeping users dancing. For the organization to take proactive action, they need to find which user is at risk of going out of the subscription.

Improving customer retention and keeping your churn rate low is vital, especially since acquiring new customers is costly. Keeping the churn under 5% will greatly benefit the organization. Anything more than 5% will affect the overall revenue.

## 2) Data

The data is from the Kaggle competition. It contains members' data with churn indicator and their usage details which include transaction and user logs. The dataset is huge and needs a lot of analysis on what data will be useful for churn predictions.

Kaggle dataset

Following are the table descriptions in the dataset –

### a) Train

The train set, contains the user IDs and whether they have churned.

- **msno**: user id
- **is_churn**: This is the target variable. Churn is defined as whether the user did not continue the subscription within 30 days of expiration. is_churn = 1 means churn is_churn = 0 means renewal.

### b) Transactions

Transaction of users

- msno: user id
- payment_method_id: payment method
- payment_plan_days: length of membership plan in days
- plan_list_price: in New Taiwan Dollar (NTD)
- actual_amount_paid: in New Taiwan Dollar (NTD)
- is_auto_renew
- transaction_date: format `%Y%m%d`
- membership_expire_date: format `%Y%m%d`
- is_cancel: whether or not the user canceled the membership in this transaction.

### c) User logs

Daily user logs describing the listening behaviors of a user. Data collected until 2/28/2017.

- msno: user id
- date: format `%Y%m%d`
- num_25: # of songs played less than 25% of the song length
- num_50: # of songs played between 25% to 50% of the song length

- num_75: # of songs played between 50% to 75% of of the song length
- num_985: # of songs played between 75% to 98.5% of the song length
- num_100: # of songs played over 98.5% of the song length
- num_unq: # of unique songs played
- total_secs: total seconds played

### d) Members

user information.

- msno
- city
- bd: age. Note: this column has outlier values ranging from -7000 to 2015, please use your judgment
- gender
- registered_via: registration method
- registration_init_time: format %Y%m%d
- expiration_date: format %Y%m%d, taken as a snapshot at which the member.csv is extracted. Not representing the actual churn behavior.

## 3) Metrics

1) The evaluation metric for this prediction is Log Loss

$$Logloss_i = -[y_i \, ln \, p_i + (1 - y_i) \, ln(1 - p_i)]$$

where *i* is the given observation/record, *y* is the actual/true value, *p* is the prediction probability, and *ln* refers to the natural logarithm (logarithmic value using base of *e*) of a number.

2) Apart from Log Loss, Recall and Precision will also be good evaluation metrics since the success of the prediction is based on the churn customer.
3)

## 4) Data cleaning and Data wrangling

- **Data cleaning**

The data cleaning involved analyzing the data for missing values and irrelevant values which are by mistake or the outlier. For missing values, when there was too much missing data the feature was dropped, for example, 'Gender' was dropped since it had huge missing values. The outlier values were imputed with mean value as in the case of the 'Age' field. The 'Age' field was converted into the 'Age_group' field by binning the age.

- **Address the size of the data**

The data size was reduced by changing the datatype size, for instance, while importing the data using Pandas, the datatype was int64 which was converted to int32 for size reduction

- **Feature engineering**

Feature engineering involves creating new features based on the available data. Since the user usage data is transaction and log, creating new features is the key to successful churn prediction. More than 100 features were created based on the transaction and log data.

A few of the new features extracted from transaction data are average plan amount, sum of discounts, total number of transactions, number of times the list price was changed, days difference between last transaction date and expiry date, etc.

The log data, which contains everyday usage information is a very huge data and they were extracted in chunks. After reading the data in chunks, new features were created based on the user's latest active three months of log usage. The user is going to churn or not based on the latest user experience, hence active 3 months of days was used for feature creation. New features like mean of unique songs, sum of the songs played 50% etc were created. Finally, the newly extracted features were combined with the member data.

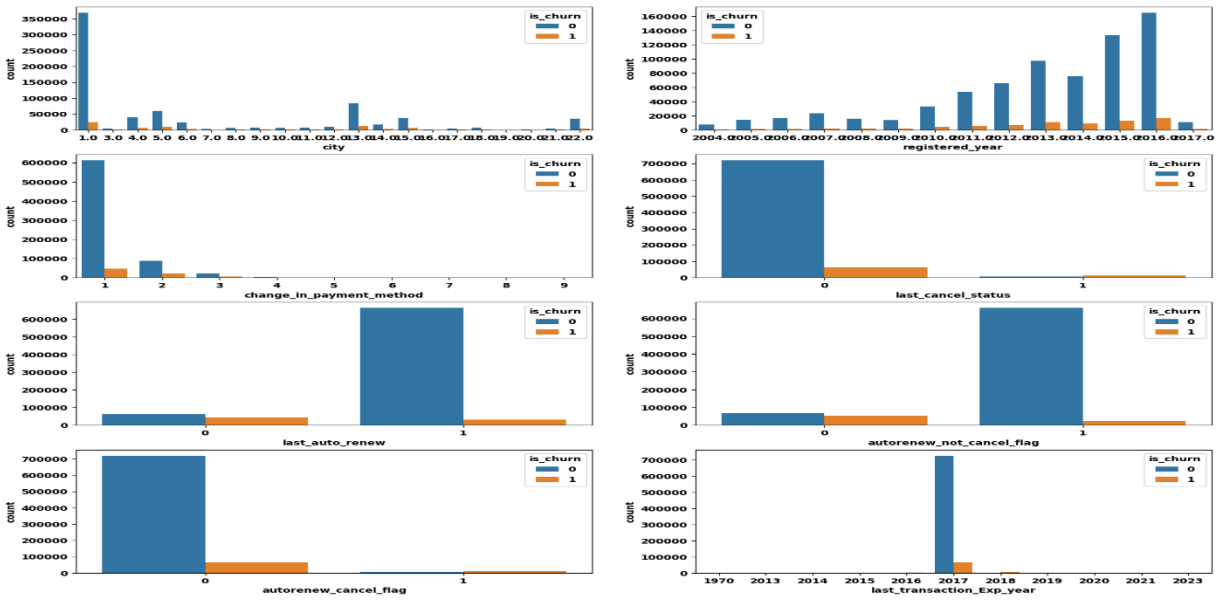
Notebook - Data wrangling - Member

Notebook - Data wrangling – Transaction

Notebook - Data wrangling – User log

## 5) Exploratory Data Analysis

During the EDA process, independent features were visualized to see if there was any trend associated with the churn feature as shown below, outliers were visualized and removed, and correlation between the independent features was visualized using a correlation matrix.



Though many newly created features were an advantage, it might create noise during model prediction if the right features are not selected. So, during the EDA process, various feature selection technique was implemented on the features.

Feature selection technique:

1) Recursive Feature Elimination (Sklearn RFECV)
2) SHAP values
3) ANOVA Inferential Value
4) Maximum Relevance — Minimum Redundancy (MRMR)

Based on the results from the correlation matrix and feature selection technique, each selected feature needs to be used for modeling to see which technique gives the best metrics.

Notebook - EDA Visualization

Notebook – Feature selection

## 6) Preprocessing and Modeling

During the preprocessing step,

- Categorical columns were converted to numerical variables using ordinal encoders.
- A standard scalar was used to standardize the numerical features
- The data was split into train and test split

After the preprocessing, various modeling algorithm was implemented with randomized CV to select the best hyperparameters. After selecting the best hyperparameter, the best one was implemented on the algorithm to find the final metrics.

Notebook for Preprocessing

Notebook for Modeling

**Performance metrics**

| Model | Accuracy | AUC | Recall | Precision | F1 | Log Loss |
|---|---|---|---|---|---|---|
| XGBoost - all features | 0.97 | 0.9839 | 0.88 | 0.93 | 0.9 | 0.078 |
| RandomForest - all Features | 0.94 | 0.96 | 0.71 | 0.94 | 0.78 | 0.12 |
| XGBoost - RFECV features | 0.96 | 0.98 | 0.88 | 0.93 | 0.9 | 0.08 |
| XGBoost - MRMR features | 0.97 | 0.99 | 0.89 | 0.93 | 0.91 | 0.078 |

## 7) Findings

XGBoost was the clear winner compared with the Random Forest. Random forest was giving a log loss of 0.12, whereas XGboost was under 0.08. The ideal log Loss is considered to be 0. The AUC, recall, Precision and the F1 score was also worse with the Random Forest model. On comparing the other three models within XGBoost, even though 'XGBoost with all features' was producing a good Log Loss of 0.078, having all features will create noise. The XGBoost with RFECV features was slightly overfitting and the Log Loss was 0.08. Overall XGBoost with MRMR optimal features was producing slightly better metrics with Log Loss - 0.078, Precision - 0.93 & Recall - 0.89. Hence the XGBoost with MRMR feature selection was selected as the final Model.

## 8) Idea for further research

Further feature engineering can be done to find hidden features that might produce better churn prediction. Improve recall, precision, and F1 score for the churn class by further fine-tuning the parameters. A customer satisfaction survey should be targeted at the churn customer and data need to be analyzed to see if there is any trend and find a way to improve customer retention. Segmenting the customer based on usage to find specific business needs for the segmented customer. For example, a customer might prefer hearing unique songs most of the time, in such cases right song suggestion might help in customer satisfaction.

## 9) Recommendation

To improve customer satisfaction and the retain the customer following can be implemented:

a. Get to the root of customer churn by getting customer feedback
b. Offer personalized and proactive customer communication
c. Reward customer loyalty
d. Follow up with customers
e. Provide customized user experience in the music based on the customer segmentation