KKBox Churn Prediction

Springboard Data Science Career Track
Capstone Project 3

Marco Riva
August 10, 2021

Client

- KKBox: Asia's leading music streaming service
- Business model: Subscription model
- https://www.kkbox.com/

Motivation & Business Impact

- Churn is of great concern for subscription model businesses
- Acquisition cost generally higher than retention cost
- Understanding customer behavior is key to drive value and growth
- Predictive tools for churn prediction are fundamental to maximizing customer retention

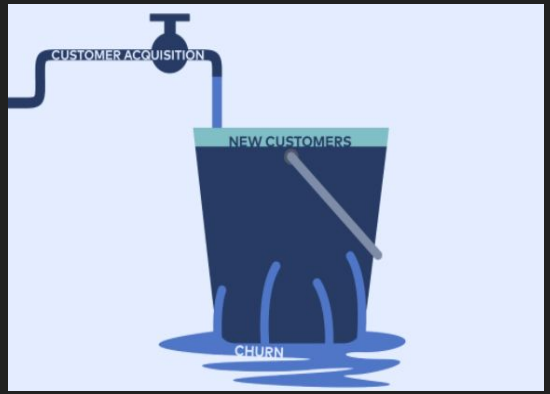


Image from: https://www.venrock.com/churn-is-the-single-metric-that-determines-the-success-of-your-subscription-service/

Problem Formulation

- Supervised Learning
- Binary Classification: Churn VS Retention
- Imbalanced dataset ~5% Churn - ~95% Retention

Objectives

- Utilize historical data provided by KKBox regarding user demographics, transaction and logs to predict customer churn in March 2017 (test set)
- Identify features correlated with churn and provide actionable insights

Evaluation Metric

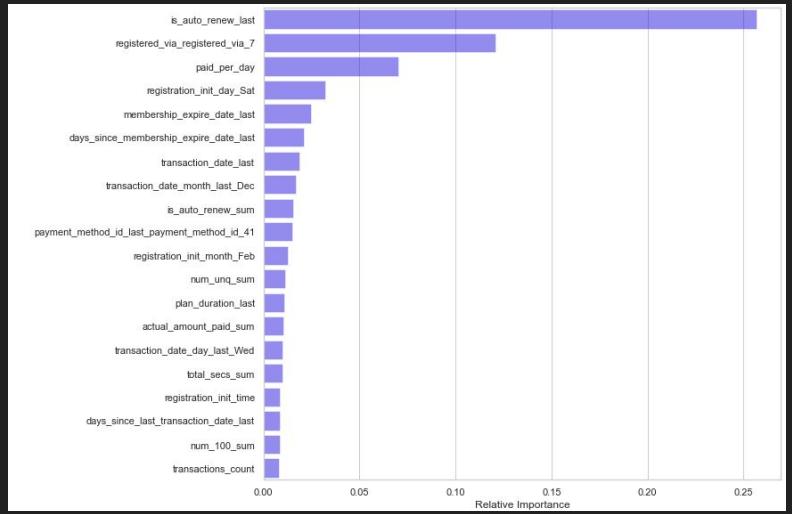
- Log Loss

Data Source

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

High Level summary and Key Findings

- User demographic, transactional and logs data were processed to generate predictors (features) for each user
- CatBoost, XGBoost and LightGBM models trained for churn prediction
  - Performance ~50% better than chance, Log Loss = 0.11 on test set (Kaggle)
- Key factors affecting churn:
  - Auto Renew Enabled
  - Price (averaged over day)
  - Registration Method
  - Temporal Features / Seasonality
  - Canceled Transactions
  - Usage: number of songs / time played



This presentation includes details of my work

Data Acquisition

Data

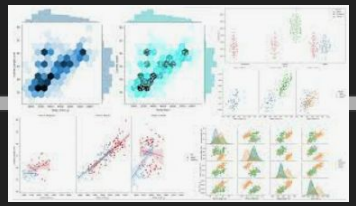
- Member info ~5M records
- Transactions ~22M records
- Logs ~400M records

Data Wrangling

Outliers

- Errors
- High Cardinality

Exploratory Data Analysis



Clean Data Exploration

Feature Engineering

Processing
Aggregation
Encodings

Modeling

Binary Classification

Insights

What we've learned

Available Data

- members.csv: User information
- transactions.csv: User transaction until end of February 2017
- transactions_v2.csv: User transaction in March 2017
- user_logs.csv: User logs until end of February 2017
- user_logs_v2.csv: User logs in March 2017
- train.csv: User ids and target variable for January 2017
- train_v2.csv: User IDs and target variable for February 2017
- submission.csv: User IDs and target variable for March 2017 (test set)

Data Wrangling

What Features?

- Member Info
- Transactions for each member
- Logs for each member

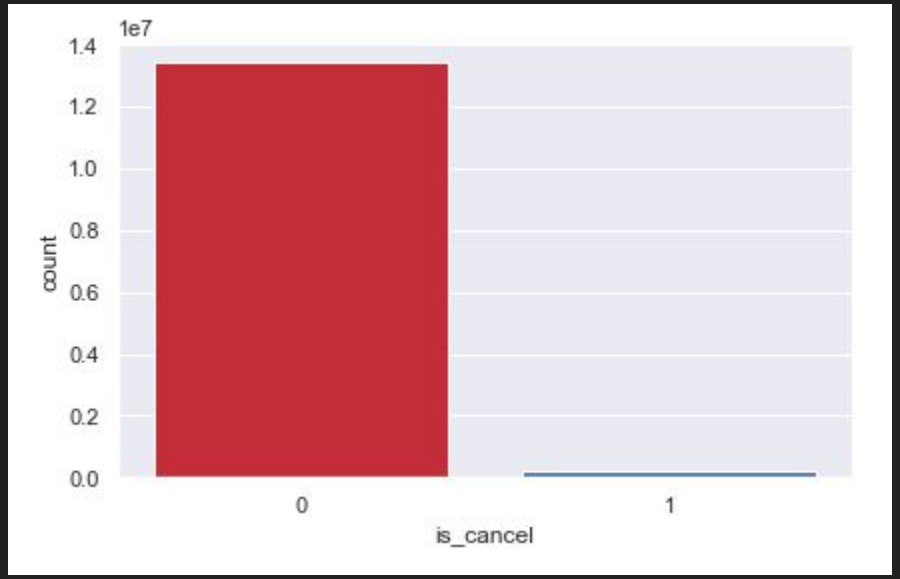
What Target?

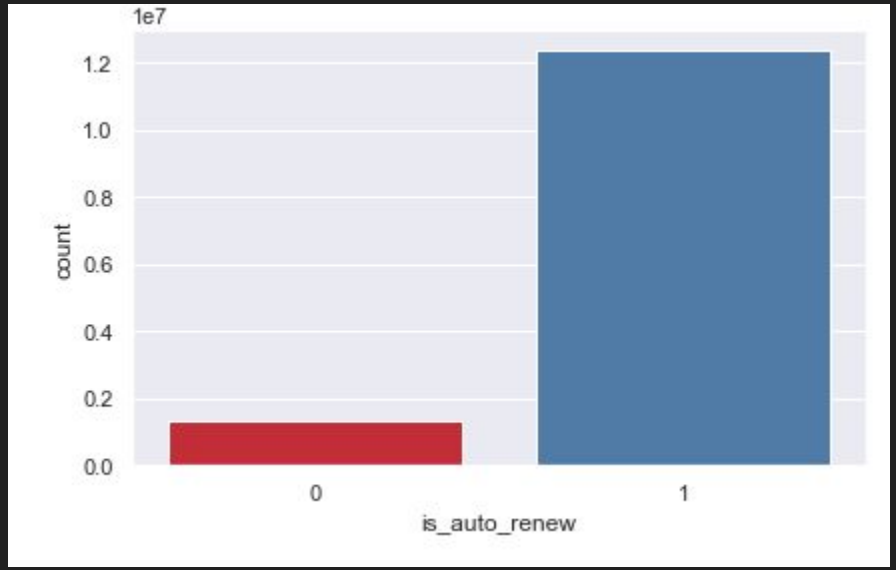
- Imbalanced binary variable
- Positive class implies churn

Data Cleaning

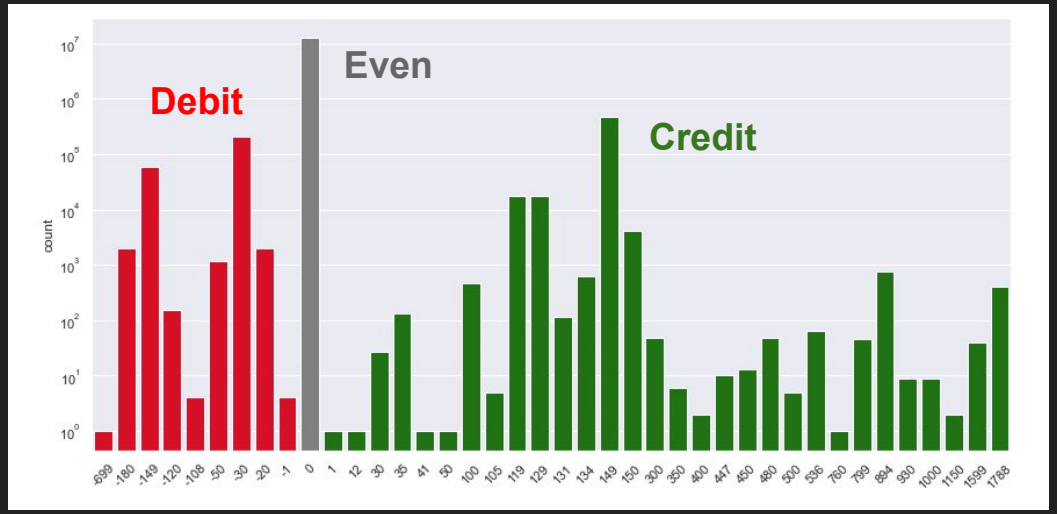
- Removed obvious outliers
- Dropped features with more than 40% missing values
- Reduced levels of high cardinality features
- No duplicates founds

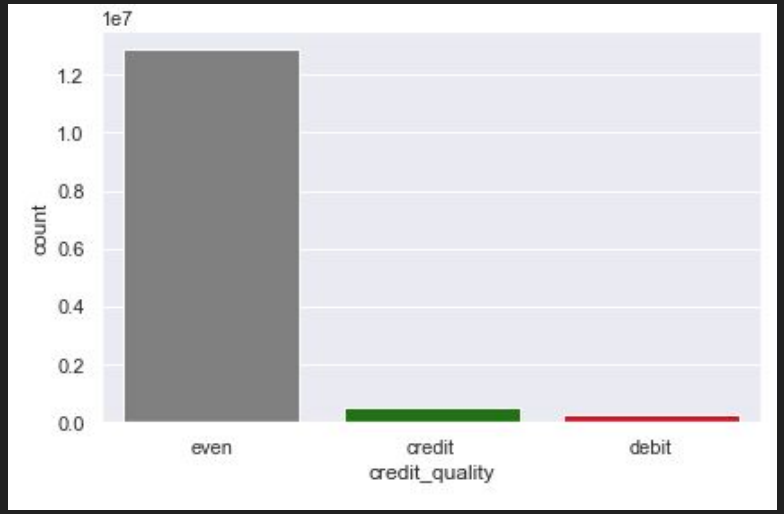
Exploratory Data Analysis



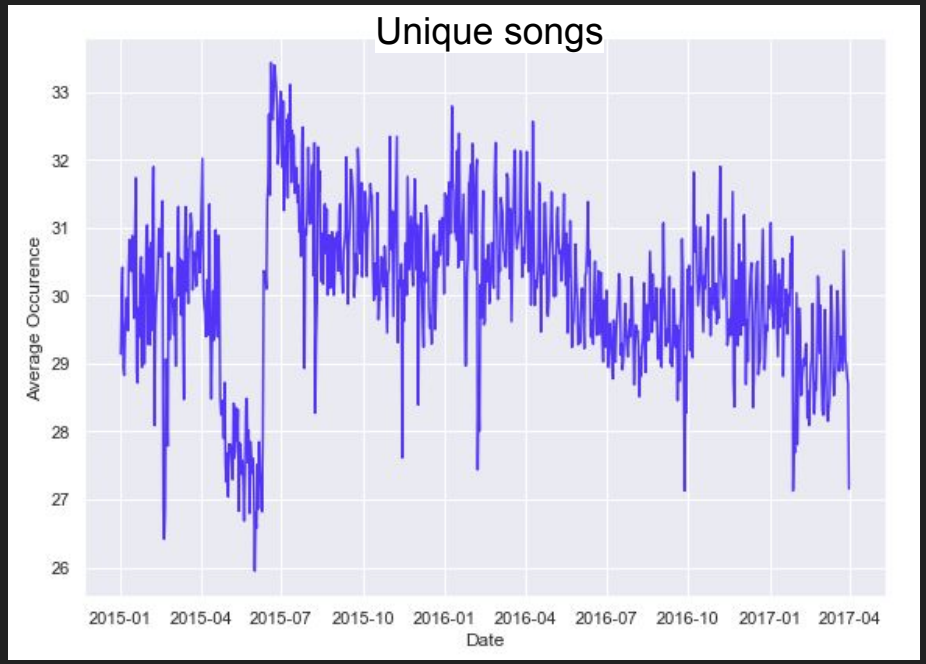


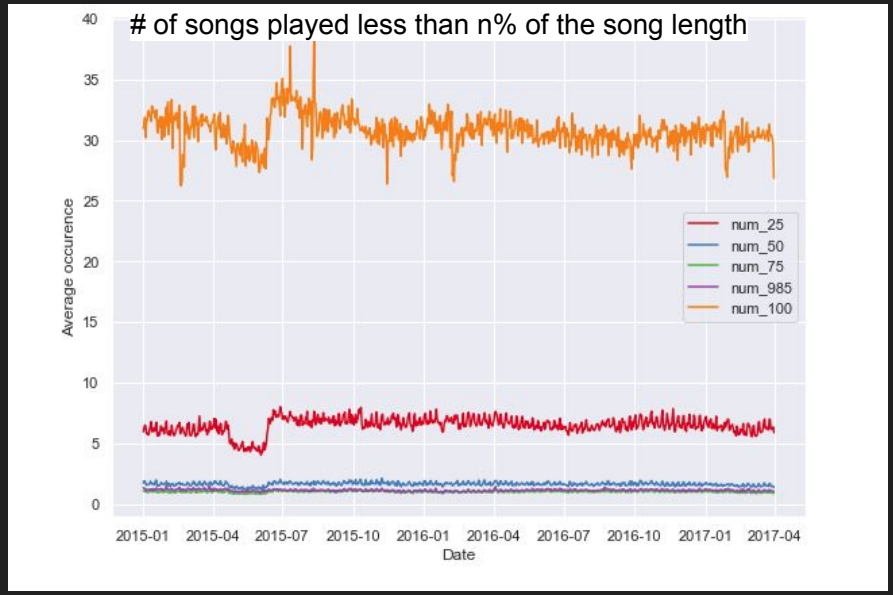
Exploratory Data Analysis



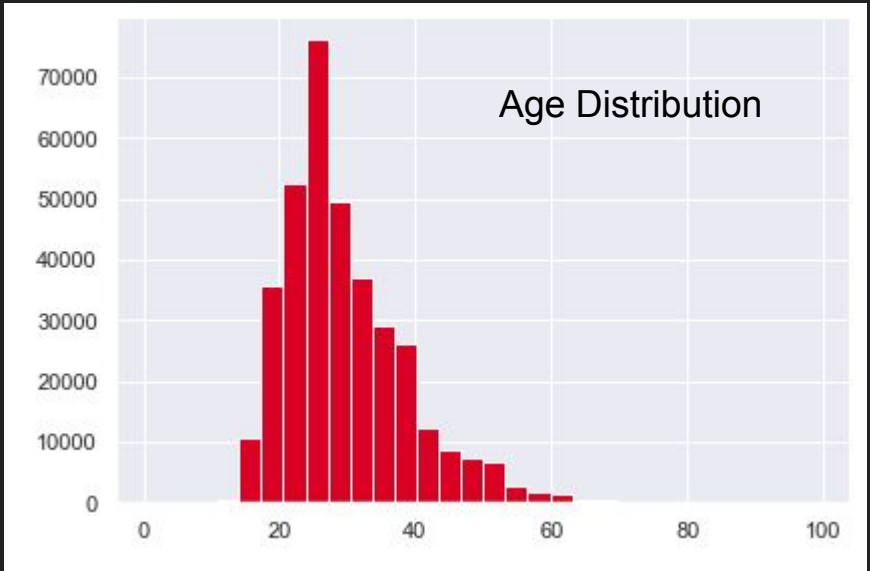


Exploratory Data Analysis

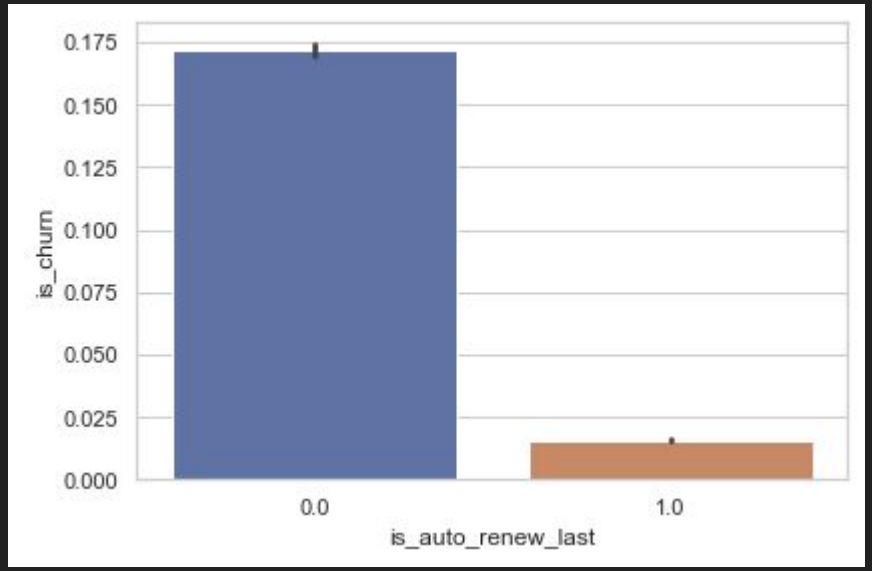


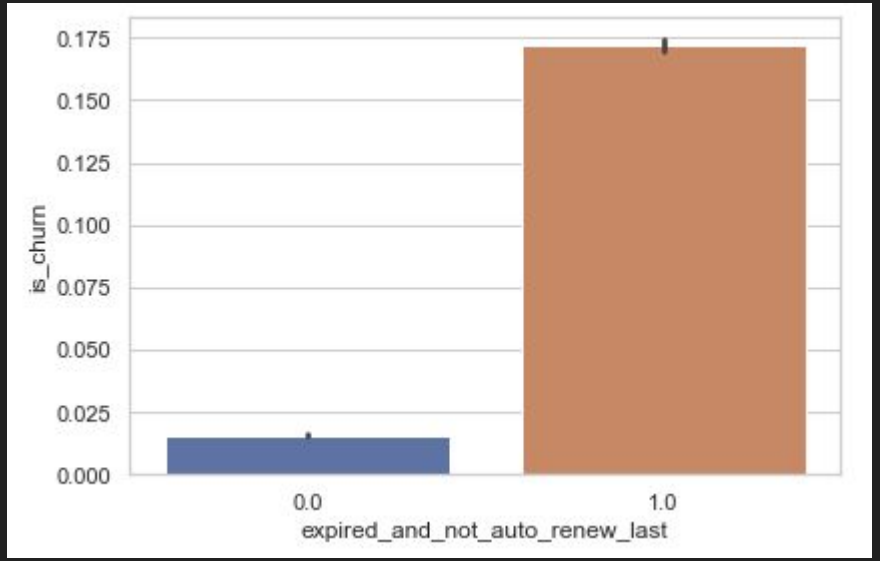


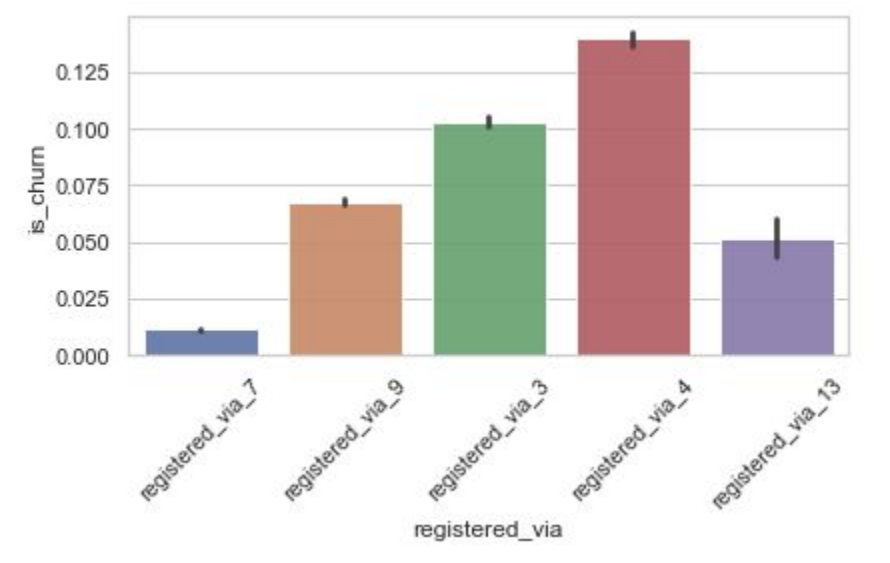
Exploratory Data Analysis

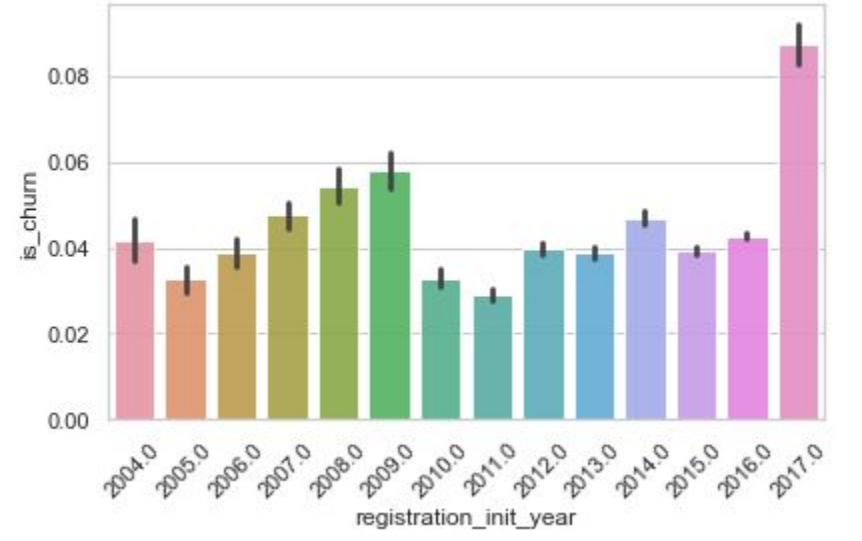


Candidate Features

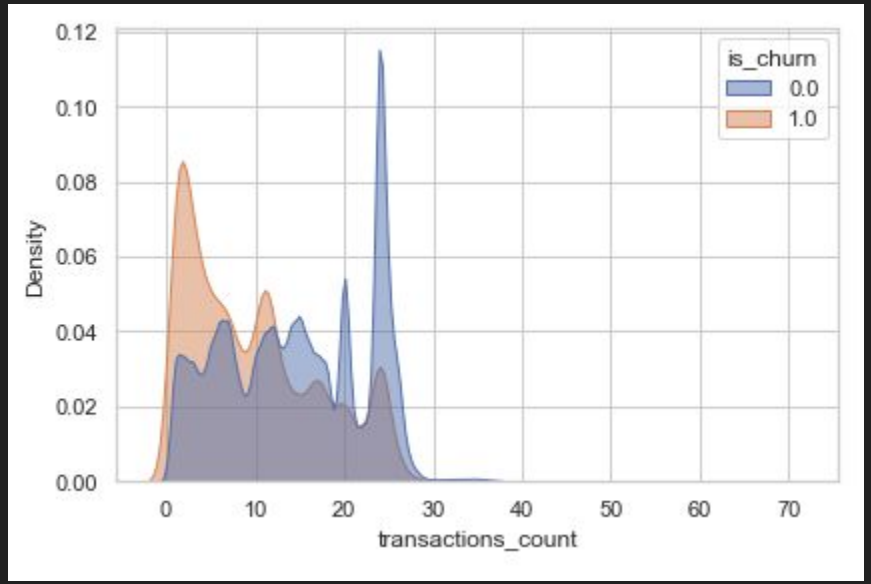


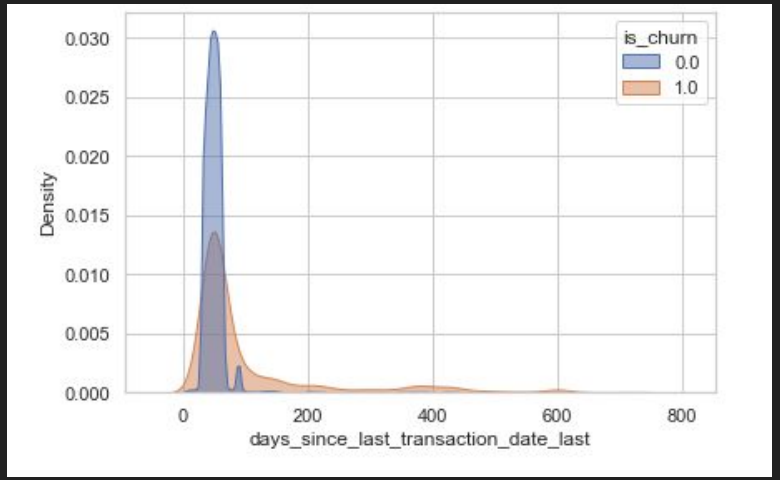


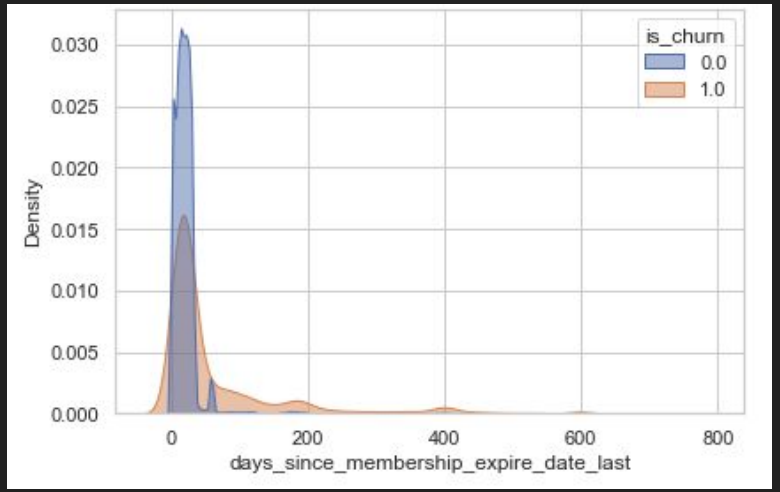

0.125
0.100
0.075
0.050
0.025
0.000
is_churn
registered_via_7
registered_via_9
registered_via_3
registered_via_4
registered_via_13
registered_via


0.08
0.06
0.04
0.02
0.00
is_churn
2004.0
2005.0
2006.0
2007.0
2008.0
2009.0
2010.0
2011.0
2012.0
2013.0
2014.0
2015.0
2016.0
2017.0
registration_init_year

Candidate Features





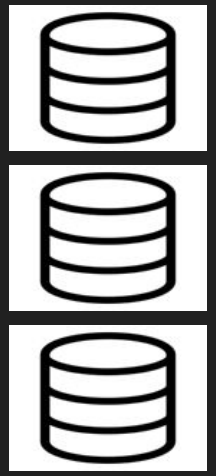


Feature Engineering

- Aggregated transaction and logs data to generate summary features (sum, average, count, etc.)

Train Test Split

Scikit-learn Environment



Train Set - Churn in February

Validation Set - Churn in March

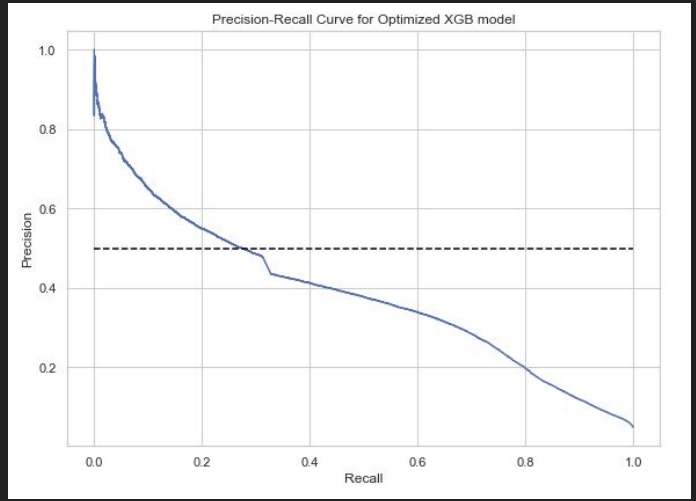
Test Set - Churn in April

Preprocessing Pipeline

- Missing values imputation
- One Hot Encoding
- 133 Features Total

Modeling

- Churn Rate 5%: dummy classifier Log Loss 0.19
- Boosting Models
  - CatBoost
  - XGBoost
  - LightGBM
- Tuned XGBoost provided best Log Loss of 0.11 on Test Set

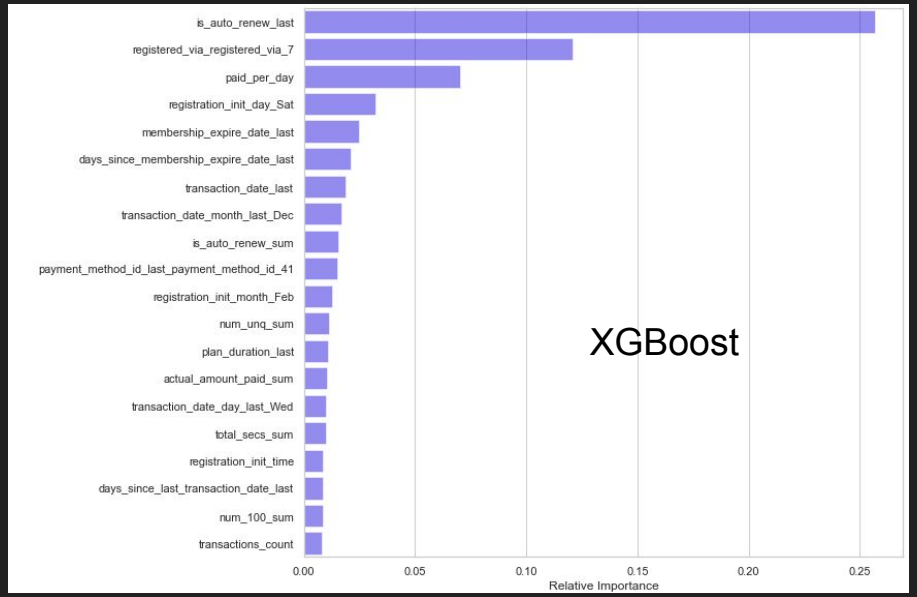


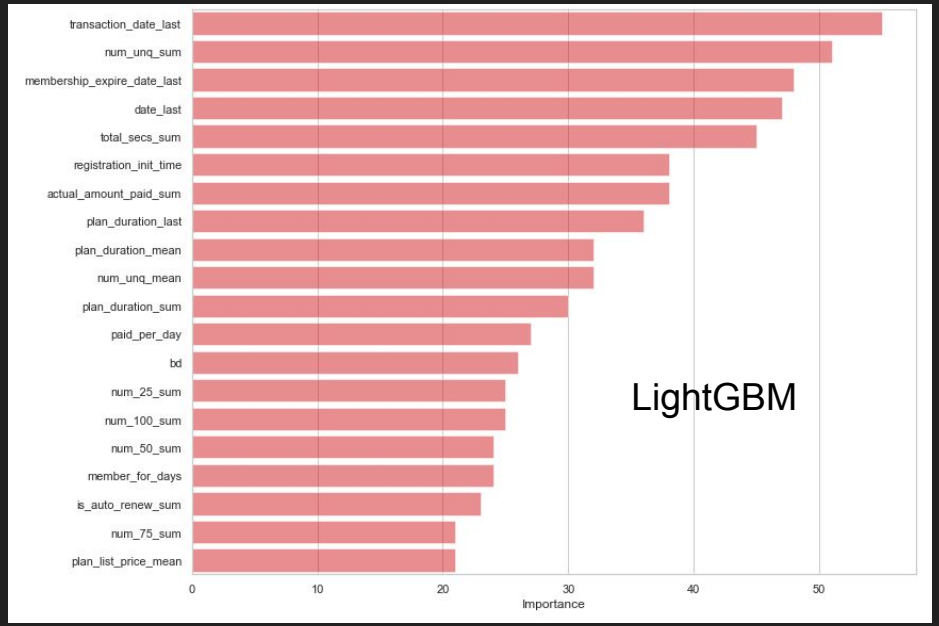
CatBoost

XGBoost

LightGBM

Feature Importance





Future Work

- Adjust sample weights to account for potential differences in feature / target distribution between train/valid/test sets
- Apply Oversampling techniques and/or SMOTE to improve predictions on minority class
- Adopt parallel computing solutions to investigate a wider range of hyperparameters

Thank you!