Capstone Project Antan Tyagi

Machine Learning Engineer Nanodegree October 27th, 2017

Definition

Project Overview

Customer churn modelling for a music company

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

Customer Churn can be crucial to evaluate customer satisfaction over periods of time, especially when measuring negative impacts which recent changes on the website, product supplier and others may have caused customers to leave. Churn is especially relevant in contractual circumstances, which are often referred to as a "subscription setting," as cancellations are explicitly observed.
Since the KK Box model is subscription based model, customer churn is really critical metric for them.

Problem Statement

Since this is a problem which will have two outputs wether the customer churns or not, this is a binary classification problem.

In this task, we will be predicting whether a user will churn after their subscription expires.Specifically, we want to see if a user make a new service subscription transaction within 30 days after their current membership expiration date.

For a subscription based service, churn is really critical and is a direct measurement of customer engagement. Hence this is a critical problem to address for the organisation.

Problem statement Link : https://www.kaggle.com/c/kkbox-churn-prediction-challenge

Metrics

The evaluation metric for this problem is

Log Loss

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

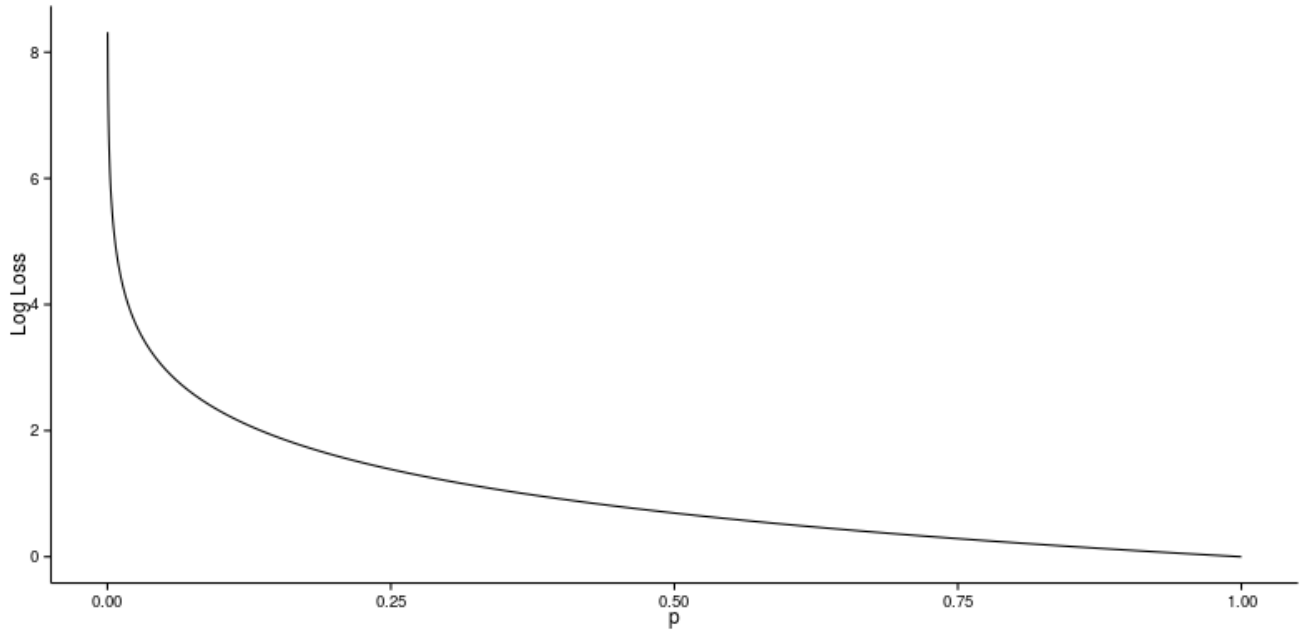
where N is the number of observations, log is the natural logarithm, yi is the binary target, and pi is the predicted probability that yi equals 1.

Note: the actual submitted predicted probabilities are replaced with $\max(\min(p, 1-10^{-15}), 10^{-15})$

Link : https://www.kaggle.com/c/kkbox-churn-predictionchallenge#evaluation

The reason we are going for log loss is that log loss heavily penalizes classifiers that are confident about an incorrect classification i.e. if the prediction is incorrect with a great score. For a churn model, we do not want to target those customers which are not churning as this will cause a significant time and money waste but also wrong customer targeting might push the customer to leave our service.

The graph shows the Log Loss contribution from a single positive instance where the predicted probability ranges from 0 (the completely wrong prediction) to 1 (the correct prediction)



Analysis

Data Exploration

We have the following data files and their descriptions

Tables

train.csv

The train set, containing 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.

msno	is_churn	
0	waLDQMmcOu2jLDaV1ddDkgCrB/jl6sD66Xzs0Vqax1Y=	1
1	QA7uiXy8vIbUSPOkCf9RwQ3FsT8jVq2OxDr8zqa7bRQ=	1

sample_submission_zero.csv

The test set, containing the user ids, in the format that to submit
• msno: user id
• is_churn: This is what we will predict. 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.

The sample submission set is provided as the test set to be used for the model as explicitly mentioned in the problem description.

msno	is_churn	
0	ugx0CjOMzazClkFzU2xasmDZaoIqOUAZPsH1q0teWCg=	0
1	f/NmvEzHfhINFEYZTR05prUdr+E+3+oewvweYz9cCQE=	0

transactions.csv

Transactions of users up until 2/28/2017.

•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.

msno	**payment_method_id**	**payment_plan_days**	**plan_list_price**	**actual_amount_paid**	**is_auto_renew**	**transaction_date**	**membership_expire_date**	**is_cancel**
YyO+tlZtAXYXoZhNr3Vg3+dfVQvrBVGO8j1mfqe4ZHc=	41	30	129	129	1	20150930	20151101	0
AZtu6Wl0gPojrEQYB8Q3vBSmE2wnZ3hi1FbK1rQQ0A4=	41	30	149	149	1	20150930	20151031	0

user_logs.csv

daily user logs describing 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

Msno	Date	Num_25	Num_50	num_75	Num_985	num_100	num_unq	total_secs
rxIP2f2aN0rYNp+toI0Obt/N/FYQX8hcO1fTmmy2h34=	20150513	0	0	0	0	1	1	280.335
rxIP2f2aN0rYNp+toI0Obt/N/FYQX8hcO1fTmmy2h34=	20150709	9	1	0	0	7	11	1658.948

members.csv

User information. Note that not every user in the dataset is available.

•msno
•city
•bd: age. Note: this column has outlier values ranging from -7000 to 2015, please use your

judgement.
•gender
•registered_via: registration method
•registration_init_time: format %Y%m%d
•expiration_date: format %Y%m%d

msno	city	bd	gender	registered_via	registration_init_time	expiration_date
URiXrfYPzHAlk+7+n7BOMl9G+T7g8JmrSnT/BU8GmEo=	1	0	NaN	9	20150525	20150526
U1q0qCqK/lDMTD2kN8G9OXMtfuvLCey20OAIPOvXXGQ=	1	0	Nan	4	20161221	20161224

Data Exploration and Exploratory Visualization

Preliminary Exploration :

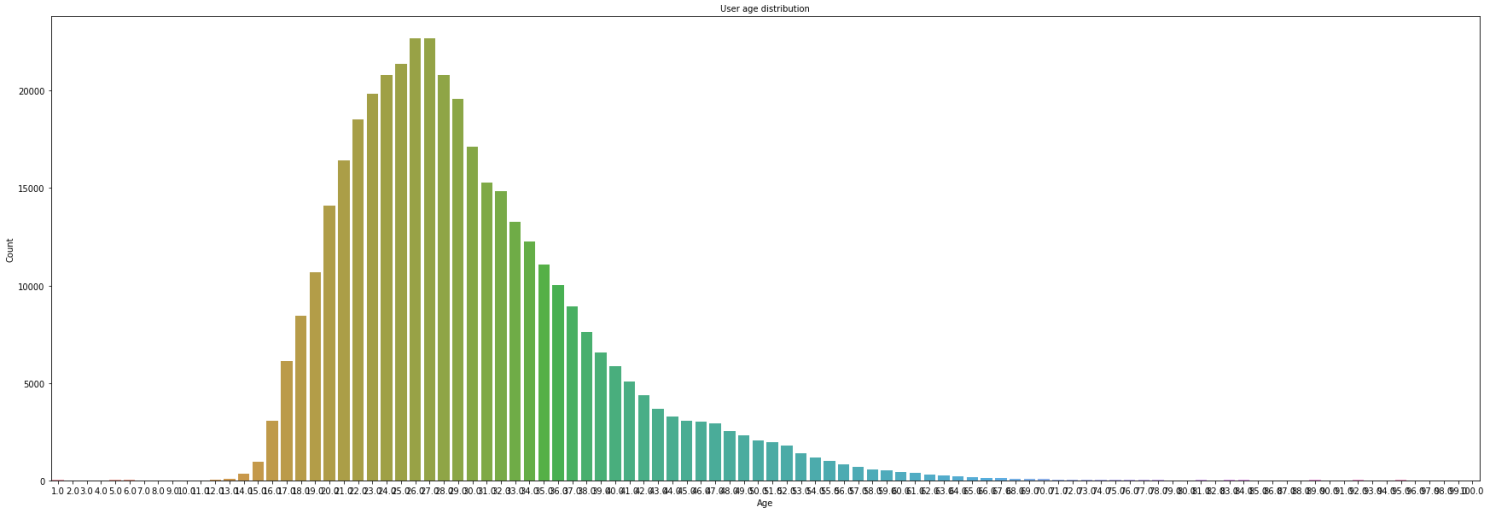
The size and dimensions of the tables are as follows:

Table	Number of Records	Columns
train	992931	2

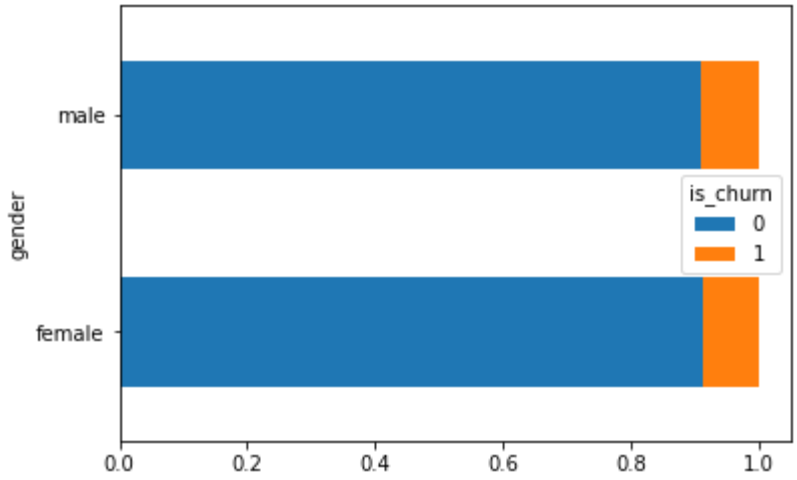
Sample submission	970960	2
members	5116194	7
transactions	21547746	9

Some features of the data :

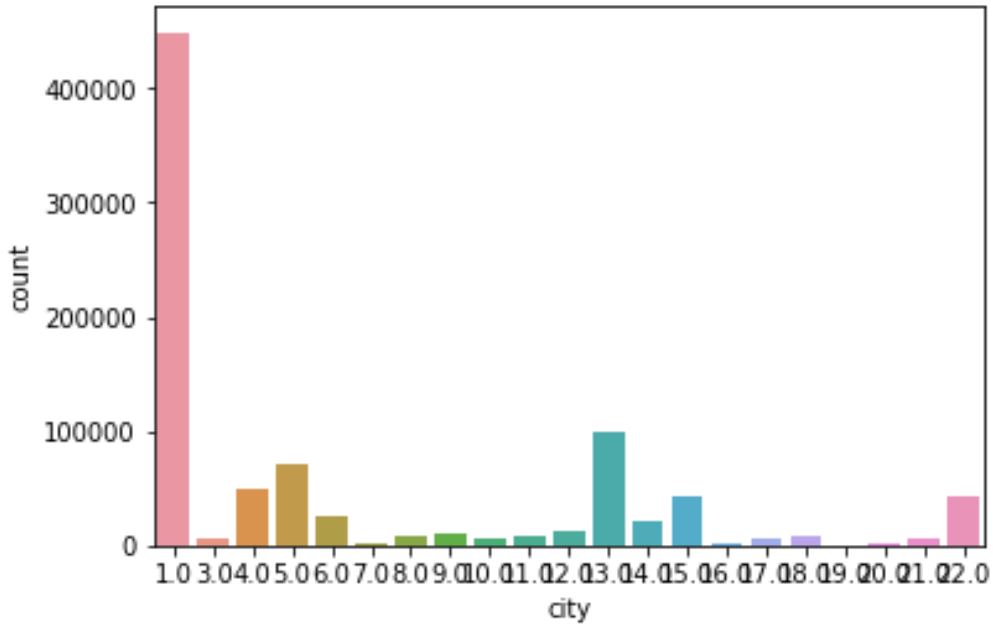
- The churn rate as obtained from the train data is 6.4% indicating that 6.4% of user has left the company
- The age variable in members information has values ranging from 2015 to-3152.0 indicating there are incorrect records for the age variable.
- The mean and median age of subscribers(after removing outlier) is 29.7 and 28 giving some information about the demographics of the customer base. The following plot shows the demographic dividend



- In the members information, the gender column has 60% null values showing incomplete information about these members.
- Out of the present value for genders, we have 171705 females out of which 16152 have churned and 189875 males out of which 18325 have churned.The churn ratio is shown as below

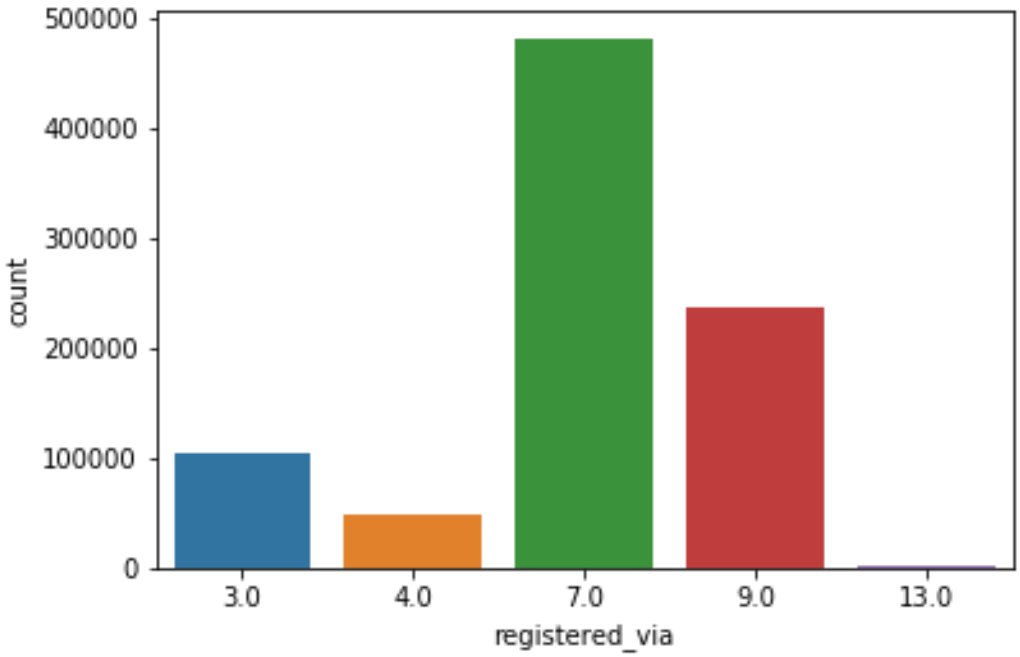


- Regarding the city distribution, we have the following plot



From the above plot, it is evident that most of our registration comes from city 1 . Also we don't have a city 2 in our data

- Looking for the registered_via we have 11% null values and majority of registration comes from city 7



Algorithms and Techniques

For this problem, we are going to use xgboost algorithm. XGBoost is an state of the art variant of gradient boosted trees methods.XGBOOST has become a de-facto algorithm for winning competitions.XGBoost is also known as ‘**regularized boosting**‘ technique because it has a regularizer which prevenets overfitting unlike other boosting algorithms.

XGBoost implements a gradient boosting decision tree algorithm. Boosting refers to an ensemble technique where new models are added over existing model(generally a weak classifier) to correct for their error.

It is called gradient boosting because it uses a gradient descent algorithm to minimize the loss when adding new models.

This approach supports both regression and classification predictive modeling problems.

The following parameters can be used to tune xgboost :

- eta : Learning rate.Makes the model more robust by shrinking the weights on each step.
- Objective : The loss function which is to be minimized
- eval_metric : The metric to be used for data validation.
- Seed : The random number seed
- max_depth : The depth upto where the splits are made. After that pruning is done backwards and removes splits where there is no positive gain
- silent : To stop print running messages.True stops message printing

Reference Link : https://www.analyticsvidhya.com/blog/2016/03/complete-guide-parameter-tuning-xgboost-with-codes-python/

Benchmark Model : The benchmark model as defined in the capstone submission.
For the benchmark model, we will be simply using the mean of the training dataset and apply that to the test dataset. Since in the training set we have the indicator that customer has churned, the mean value is then added to the test dataset and log loss function is used to calculate the score.

Using this approach, the baseline score we get is 0.30786

Methodology

Data Preprocessing

The following preprocessing steps are taken :

- The members information are combined with the train and test data
- The outliers in the age variables are removed.
- The genders which are currently defined as male and female are mapped as 1 and 2 respectively.
- The user log data is loaded and grouped for each user and combined with the train and test dataset.

Sources consulted :
https://www.kaggle.com/dguliani/methods-to-use-user-logs-and-transactions

http://journals.plos.org/plosone/article?id=10.1371/journal.pone.0180735

- The transaction data is now combined with the train and test dataset. We are not doing any processing for transaction data and we will let xgboost figure this out itself.
- The NA values are replaced by 0 in both test and train dataset.

Implementation :

The implementation process can be divided into following steps :

- Training model : The model was trained on the preprocessed data.A validation set was also created on a 30 percent split.A watchlist is also created which displays the train and validation sets accuracy.We also enabled a early stopping criteria if ‘valid-log loss’ accuracy does not increase in subsequent 10 rounds.
- The model is then used to predict the test data set.The sample submission is stipulated as the test data set as the problem requirements.

- The test file is then submitted to kaggle portal to get the log loss score for the model

Libraries Used :

- Package : Sklearn
    - Class : model_selection
    - Function : train_test_split
    - This function is used to split our training data into a train set for the model and a validation set which is used to validate our model.The split is done as 70% training set and 30% validation set.
- Package : xgboost
    - Dmatrix is an internal xgboost data structure which is optimised for efficieny and training speed. It is created from numpy array and we pass our matrices to it.
    - We train our model and using xgboost and validate it on the validation set which is created. We minimize the log loss error to optimize our model.
    - Then we predict on the test dataset and submit our model to kaggle for log loss evaluation.
    - We then plot the feature importance to ascertain our model behaviour and determine the most important features for churn rates.

Problems with Implementation :

- Initial submission was highly inaccurate. The log loss error was 1.689 which is way higher than baseline model. This was obtained by running a Grid Search CV from which we got a great training accuracy(in line of 0.012) but the model fell apart on the test set. So that approach was abandoned.Also took a lot of time.
- Kaggle only allows 5 submission per day from a single account. So I had to work for a couple of days to get the optimum model.

Refinement :

- Grid Search parameter used

  eta : { 1, 0.5,0.05,0.01,0.001,0.002,0.0005}
  max_depth : {2,3,5,6,7}
  number of rounds : {200,500,800,1000,1200,1500,2000}

  The best grid search parameters obtained were:
  eta : 0.001
  max_depth : 4
  number_of_rounds : 2000

  However the test accuracy we got from the grid seearch was 1.689.So I had to abandon this model and go for manual parameters selection.

- eta rate was progressively decreased as we went ahead with training.The final value obtained was 0.002
- max_depth was initialized to 2 and then finally the final value settled was 7 as going beyond that was overfitting our model.
- Number of training rounds were progressively increased to 1500.Going beyond that was overfitting and did not lead to improvemnet in accuracy.

Results

Model evaluation and validation :

The values of the parameters settled for final solution are as follows :

- eta : learning rate. Value is 0.002
- max_depth : 7

- number of rounds : 1500.
- We allowed the xgboost to determine itself the best tree limit for our model by specif

To validate our model, we compared the accuracy of our traing and validation sets and kept that in mind that our train and validation accuracy does not defer out too much. We also enabled early stopping rounds as a multiple of 10 in case our accuracy did not improve. This was done to prevent overfitting.

The final training and validation accuracy achieved are as follows

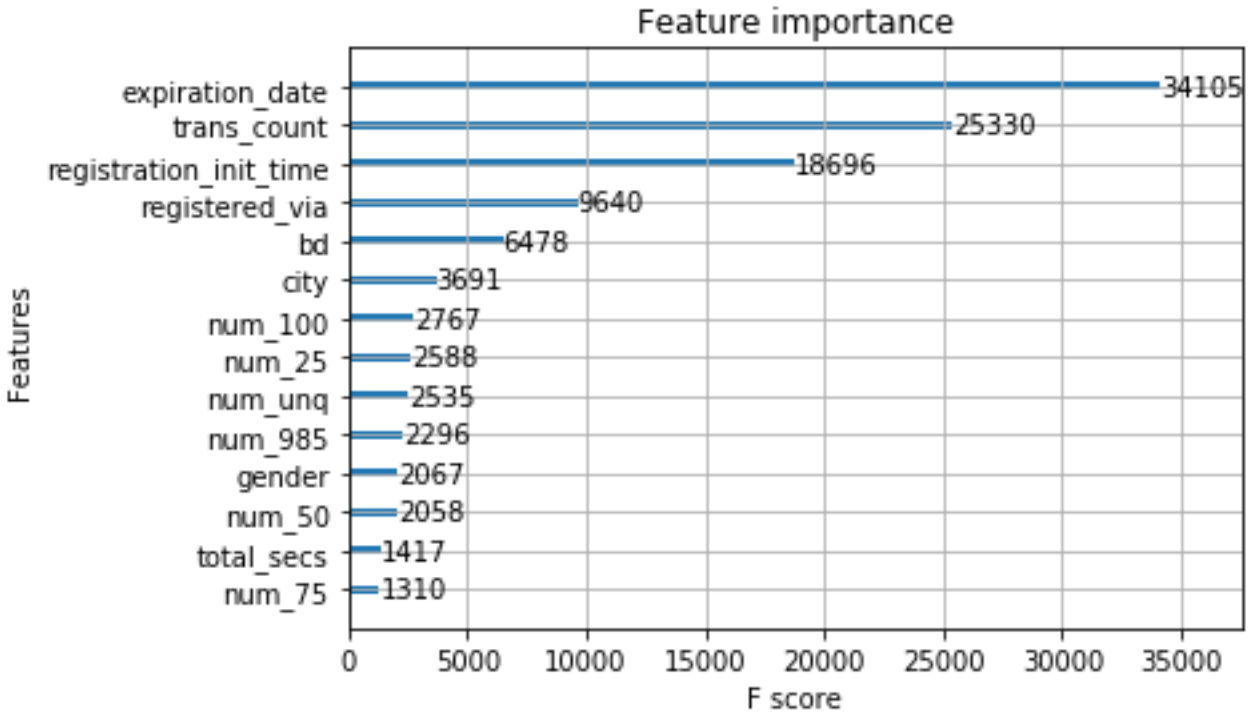
```
train-logloss:0.132714  valid-logloss:0.133781
```

Justification :

The trained model is then fitted to the sample submission data(as the test data specified by problem rules). The data is then submitted to Kaggle. After evaluation, the accuracy obtained was 0.2256.This is an improvement to our benchmark model which had the accuracy of 0.30786. This shows that this modelcan be used for churn prediction and has a pretty good accuracy. We knew that we had an imbalanced model(only 6% churned), thus we had a higher benchmark model accuracy. To improve on that by a really significant amount shows we have a good model.

Free Form Visualization :

The following graph shows the feature importance determined by the F-score for the features



- We see that expiration data is the most important feature for this model. Intutively this seems reasonable but has to be looked in combination with other features as this might be misleading.
- Transaction count also looks an important variable but it might be not be correct as a new user might have less transactions and might not have churned.
- Total number of songs listened is not a significant factor.

Reflection :

The process used for this project can be summarized as :

- The initial relevant problem and dataset were found.
- A benchmark was created for the project
- The data is then preprocessed to prepare for the model.
- The classifier was used to train the data(multiple times until optimum parameters were obtained)
- The model was tested on the test data and the model was further fine tuned to obtain the best accuracy.

I found the last two steps to be really difficult. Since xgboost has a large number of parameters, combined with our large data, Grid Search CV was not a good choice. Also RandomizedCV does not give satisfactory results with our dataset. The result was that I had to manually tune the parameters and had to limit the number of parameter which can be tuned. Secondly the treatment of log data because of its size was cumbersome and had to refer some other sources to carry out that.

Improvement :

The model has scope for improvement.Some of these can be

- We are taking only transaction count from the transactions, we could get payment plan days and payment amount and other information from the transactions table can be used to create a better model.

- Parameter can be tuned further. I did not find a good way to tune my parameter in a reasonable time. This could help us better tune our model.