Sparkify Capstone Project

This project is part of the Capstone project for Udacity's Data Science Nanodegree.

The project analyses log data for two months for "Sparkify" music streaming service (which is similar to Spotify or Pandora), **to observe the behavior for users who stayed vs. users who churned**. Churn analysis is a very important business problem to help companies increase customer retention and loyalty.

To gain insights on churn and non-churn users, the project computes several user attributes from the log data. The data show that **users that cancel, tend to be new/recently registered, and less engaged with the platform**.

To predict churn, the project further builds two churn models: 1) a Logistic Regression, and 2) a Multi-Layer Perceptron (MLP). F1 score was used as the optimization metric for both models. **The MLP model provided slightly better results, with an F1 score of 0.78 for test data (vs. 0.76 for the Logistic Regression).**

The rest of the document describes the followed project steps.

1. Load and Clean Dataset

The project uses Spark Python API (PySpark) running on IBM Cloud on a medium-sized dataset (230MB) of Sparkify log data. PySpark was the chosen analysis language in order to perform data science at scale (i.e. Bid Data).

The dataset contains in total **543,705 records** (i.e. log entries) spread across **448 users**, out of which **99 users have visited the "Cancellation Confirmation" page (i.e. churned)**. The dataset spans 2 months of records (1-Oct-18 to 1-Dec-18) for the users.

Figure 1 Columns of Sparkify log data

```
root
 |-- artist: string (nullable = true)
 |-- auth: string (nullable = true)
 |-- firstName: string (nullable = true)
 |-- gender: string (nullable = true)
 |-- itemInSession: long (nullable = true)
 |-- lastName: string (nullable = true)
 |-- length: double (nullable = true)
 |-- level: string (nullable = true)
 |-- location: string (nullable = true)
 |-- method: string (nullable = true)
 |-- page: string (nullable = true)
 |-- registration: long (nullable = true)
 |-- sessionId: long (nullable = true)
 |-- song: string (nullable = true)
 |-- status: long (nullable = true)
 |-- ts: long (nullable = true)
 |-- userAgent: string (nullable = true)
 |-- userId: string (nullable = true)
```

2. Exploratory Data Analysis

Churn frequency

Churn is defined using the Cancellation Confirmation page (presumably once a user visits this page he has deleted his account). This happens for both paid and free users. Since 99 of 448 users have churned, the **churn frequency is 22%**.

Data Exploration

To observe the behavior for users who stayed vs. users who churned, various user attributes are computed and compared. These (per-user) attributes are:

- Count of each visited page (attribute name: *About, Add Friend, Add to Playlist, Cancel, Cancellation Confirmation, Downgrade, Error, Help, Home, Logout, NextSong, Roll Advert, Save Settings, Settings, Submit Downgrade, Submit Upgrade, Thumbs Down, Thumbs Up, Upgrade,*)
- Gender (attribute name: *is_M, is_F*)
- Session statistics
    - Avg. length of listened songs (attribute name: *avg_length*)
    - Avg. number of items (e.g. songs) browsed or listened to in one session (attribute name: *avg_itemInSession*)
    - Count of sessions (attribute name: *sessionId*)
- Subscription level (attribute name: *is_free, is_paid*)
- Number of days since registration (attribute name: *days_registration*)

Figure 2 First 5 rows of computed user attributes

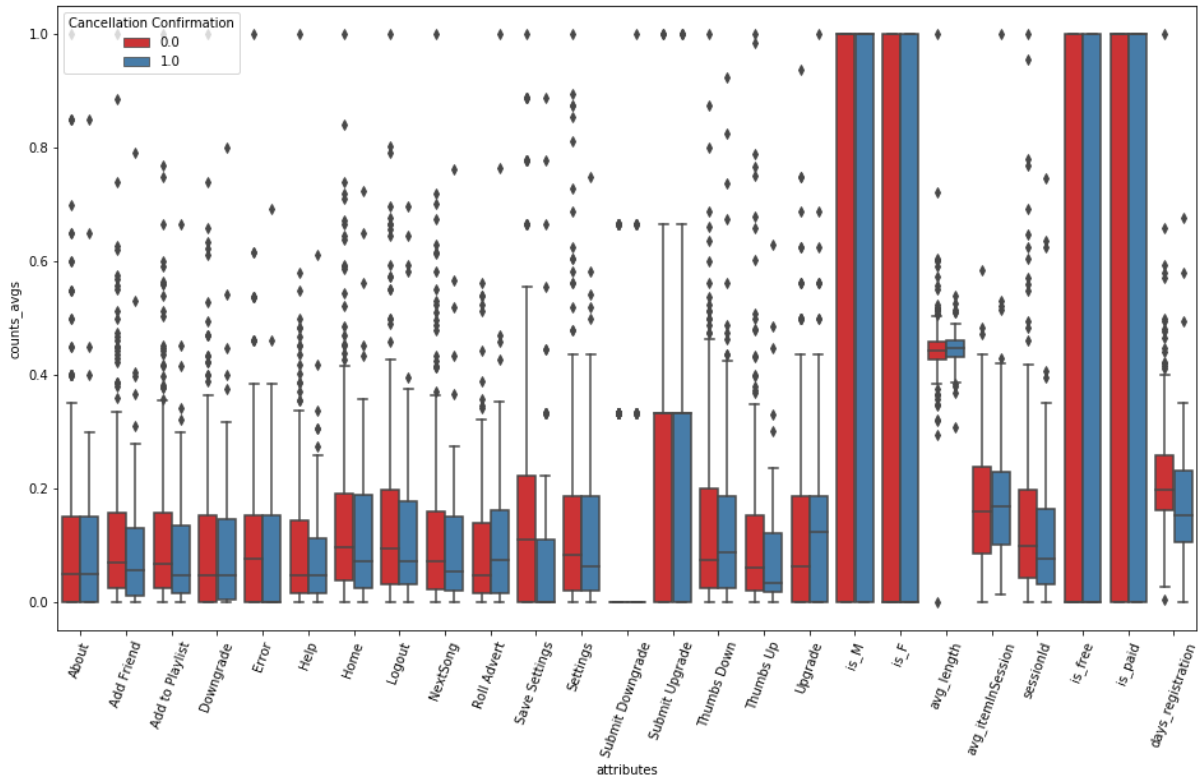
userId	About	Add Friend	Add to Playlist	Cancel	Cancellation Confirmation	Downgrade	Error	Help	Home	Logout	NextSong	Roll Advert	Save Settings	Settings	Submit Downgrade	Submit Upgrade	Thumbs Down	Thumbs Up	Upgrade	is_M	is_F	avg_length	avg_itemInSession	sessionId	is_free	is_paid	days_registration
100010	0	3	1	1	1	0	0	0	3	1	96	22	0	0	0	0	3	4	2	0	1	269.2635095535388	68.5	2	1	0	14
200002	1	2	6	1	1	3	0	1	23	11	310	11	0	2	0	1	5	15	2	1	0	247.33117322513322	83.2	5	0	1	54
296	0	2	3	1	1	1	0	2	7	3	112	9	0	1	0	1	0	8	1	0	1	265.44961898747046	20.6	5	0	1	26
125	0	2	2	0	0	0	0	2	2	0	62	6	0	2	0	0	1	2	0	1	0	281.4469189018566	27.666666666666668	2	1	0	106
124	3	26	45	1	1	21	0	10	70	17	1826	1	3	15	0	0	15	102	0	0	1	248.7753352490972	128.11764705882354	17	0	1	112

only showing top 5 rows

Data understanding

Next, the attributes were compared between churned and non-churned users. Since the attributes have different scales (e.g. count of pages vs. average length of session), each attribute was normalized and displayed on a 0-1 scale.

Figure 3 Scaled distribution of attributes for churn and non-churn users



Comparison of attributes reveals that **users that cancel tend to be new, and less engaged with the system**. In particular:

- Users that cancel visit fewer "engagement" pages and relatively more "negative" pages vs. users that do not cancel.
    - "Engagement" pages (visited less by churn users): Add Friend, Add to Playlist, Save Settings, Thumbs Up.
    - "Negative" pages (visited more by churn users): Roll Advert, Thumbs Down.
- Users that cancel have fewer sessions vs. users that do not cancel (as evidenced by count of sessions (*sessionId*))
- Users that cancel have registered recently (as evidenced by number of days since registration (*days_registration*))

3. Feature Engineering and Selection

The above attributes were calculated over different timeframes. Only the attributes with the highest discriminatory power between churn and non-churn users were selected for a future model. Followed steps:

- Attributes that varied with time were calculated over different timeframes. Chosen attributes were calculated over the last 1, 2 and 4 weeks. This step resulted in 93 attributes for each user.
- Data was split in training and test sets (80% training - 350 users / 20% test - 98 users).
- A two-sample Kolmogorov–Smirnov (K-S) test was used on training data to understand the discriminatory power of each attribute. K-S is a two-sided general nonparametric test, for the null hypothesis that 2 independent samples are drawn from the same continuous distribution. If the K-S p-value is low, then we reject the hypothesis that the distributions of the two samples are the same. The chosen p-value was 5%.
- The attributes with high discriminatory power (K-S p-value < 5%) were selected for a future model. To reduce collinearity, only one attribute per considered time-frames was selected (e.g. count of Roll Advert in the last 14 days correlates highly with count of Roll Advert in the last 28 days, and with the one in the last 7 days; only Roll Advert in the last 28 days was retained, as it has the lowest K-S p-value). To further reduce collinearity, attributes with a correlation coefficient higher than 0.9 with already selected attributes were also excluded.

Figure 4 K-S p-value for top 30 attributes' comparison between churn & non-churn users

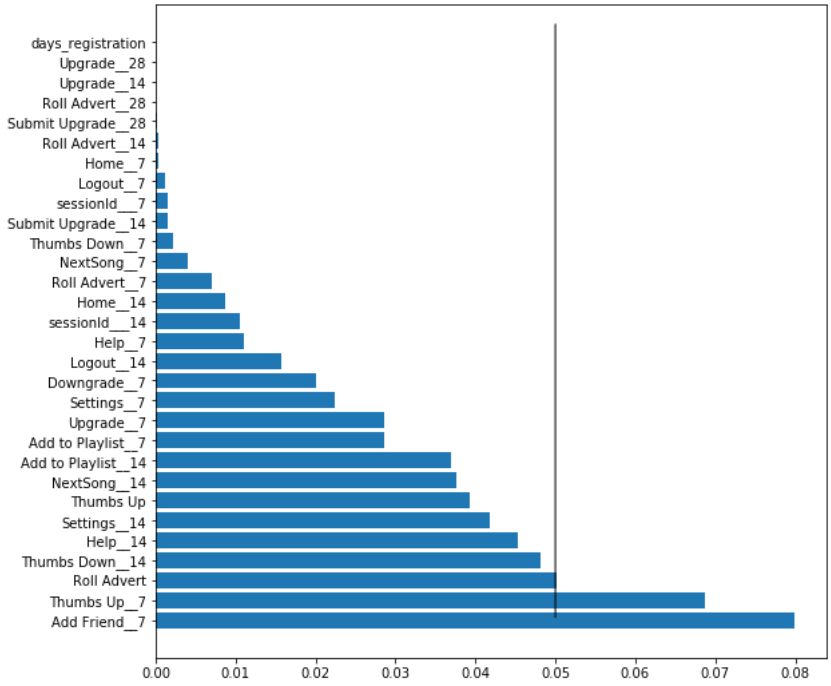
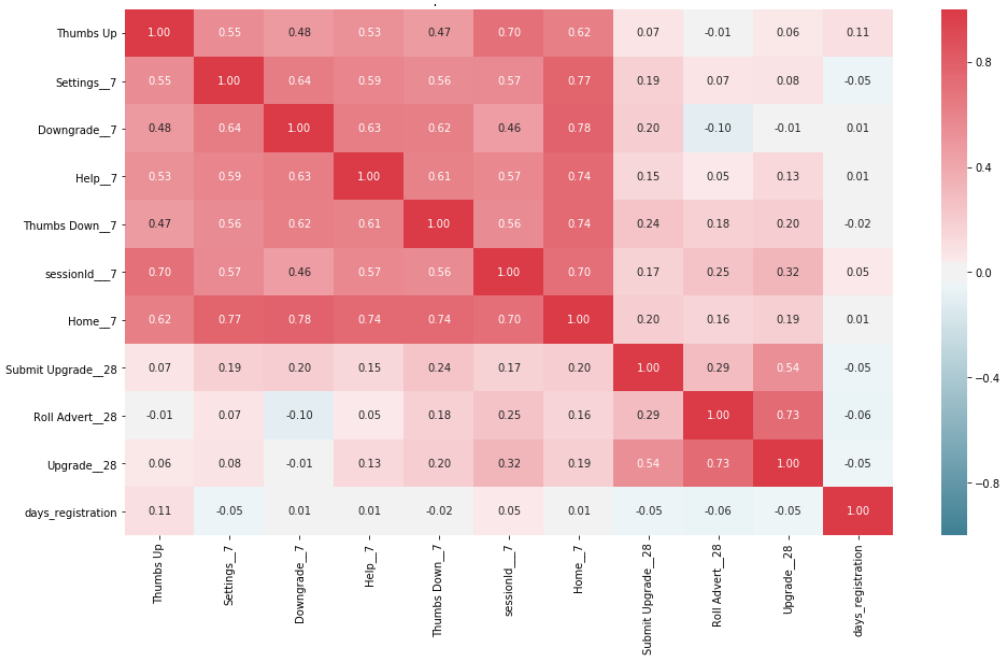


Figure 5 Correlation matrix (pearson correlation) for selected attributes for model



The 11 selected attributes for modelling were:

- Count of pages over:
  - 7 days (*Settings__7, Downgrade__7, Help__7, Thumbs Down__7, Home__7*)
  - 28 days (*Submit Upgrade__28, Roll Advert__28, Upgrade__28*)
  - All analyzed time frame (*Thumbs Up*)
- Count of sessions in the last 7 days (*sessionId___7*)
- Number of days since registration (*days_registration*)

It is worth noting that gender or subscription level do not have a high explanatory power in identifying churn, and were not selected for modelling.

4. Modelling

Achieved results

Two machine learning methods were applied and evaluated:

- Logistic regression.
- Multi-Layer Perceptron (MLP) with 3 layers, each containing the following nodes
  - 11 – inputs
  - 4
  - 2 – predicted labels

Since the churned users are a fairly small subset, F1 score was used as the optimization metric. In real-life, different weights would be assigned to different prediction categories and the model would be optimized with the given weights (e.g. TP = 100; FP = 10; FN = 50; TN = 1).

To maximize the F1 score, both the Logistic Regression and the Multi-Layer Perceptron used the returned probability that the user will churn. For the Logistic Regression, the threshold which maximized the F1 score was determined through a pyspark.ml function (using fMeasureByThreshold). For the MLP model the threshold was established by iterating through a series of thresholds and selecting the one which maximized the F1 score.

For both models model training and threshold selection was done on training data (350 users), and final model evaluation was done on test data (98 users).

The Logistic Regression provided an F1 score for test data of 0.76. The MLP model provided slightly better results, with an F1 score of 0.78.

Figure 6 Logistic regression: Normalized confusion matrix for test data

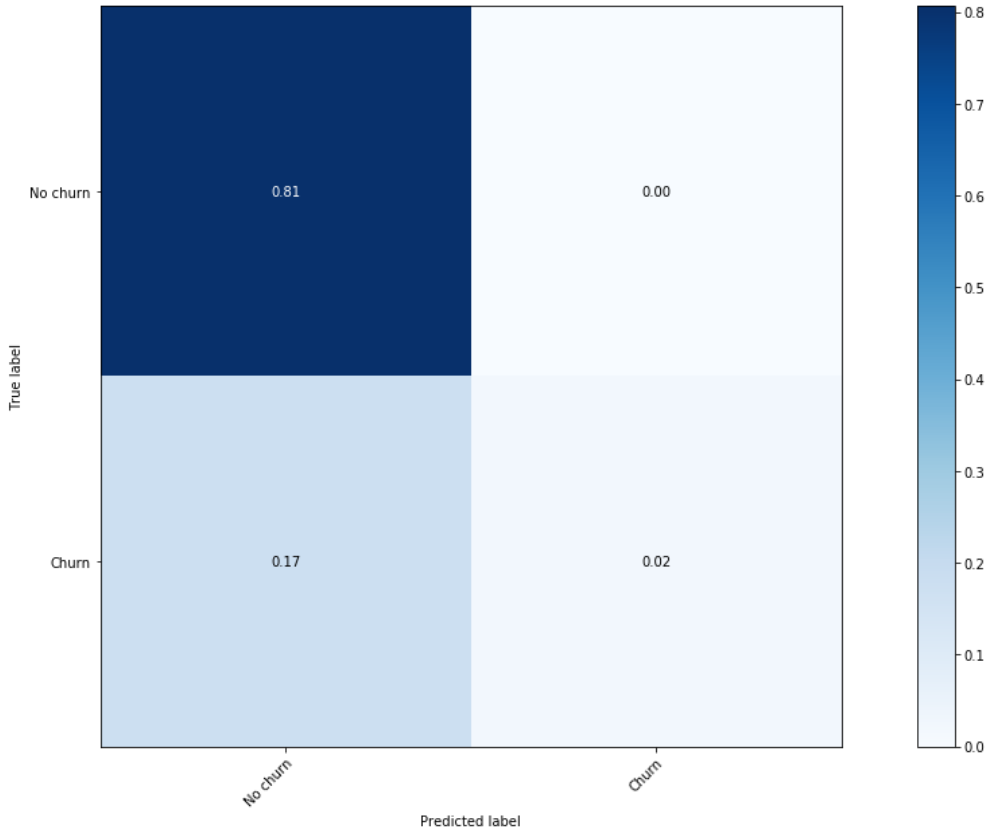
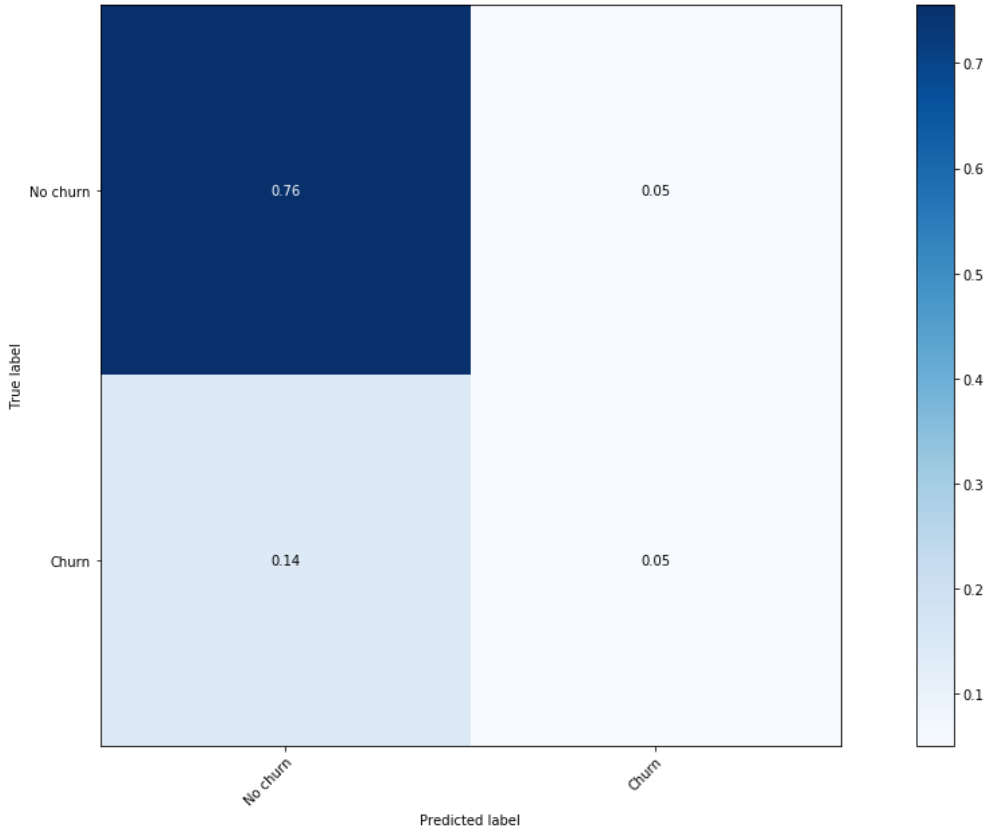


Figure 7 MLP: Normalized confusion matrix for test data



Future work

To improve the current results further ideas could be:

- Usage of a bigger data set (e.g. with more users and a bigger time-frame).
- Calculation of additional attributes from the log data and use these for churn modelling (e.g. including location, or the listened to artists or songs).
- Usage of more sophisticated models (e.g. Gaussian process classification)