Capstone Project

Data Scientist Nanodegree

November 16, 2019

Definition

Project Overview

Understand user decision is important for providing services in industry. In this project, I explored a small subset of user data from Sparkify, which is a music streaming service.

Problem Statement

The goal for this project is to understand the main features that contribute to user service cancellation, and potentially provide suggestion for improving user service experience. I'd like to understand following questions:

- How long does it take for user to cancel the service?
- What are the top songs paid user like?
- What is the main factor that contribute to a free user subscribing the service?
- What is the main factor that contribute to a paid user cancel service?

Metrics

Metrics used in this project include r2 score, f1 score and accuracy.

R2 score, which is also known as coefficient of determination, is defined as following:

$$r2 = 1 - \frac{SS_{res}}{SS_{tot}}$$

Where SS_{tot} is the total sum of square error, and SS_{res} is the sum of square residual errors (1).

F1 score is a binary classification score that measuring the test accuracy. It has a expression as following (2):

$$F1 = 2 \cdot \frac{precision \cdot recall}{precision - recall}$$

Accuracy is a ratio of number of correct predictions to the total number of input samples, shown as following (3):

$$Accuracy = \frac{Number\ of\ Correct\ Predictions}{Total\ Numer\ of\ Predictions}$$

Analysis

Data Exploration

The Sparkify dataset has 286500 records and each records has 18 attributes, listed as following:

- **artist**: string -> artist name

- **auth**: string -> cancelled/guest/loggedin/loggedout
- **firstName**: string -> user name
- **gender**: string -> user gender
- **itemInSession**: long -> log count in a given session
- **lastName**: string -> user name
- **length**: double -> song's length in seconds
- **level**: string -> subscription level; 2 categories (free and paid)
- **location**: string -> user location
- **method**: string -> http request method; 2 categories (GET and PUT)
- **page**: string -> type of interaction (page accessed in the music streaming app); 22 categories (NextSong, Home, Login, Cancellation Confirmation, etc.)
- **registration**: long -> user's registration timestamp
- **sessionId**: long -> session to which the log belongs to
- **song**: string -> song name
- **status**: long -> http status code; 3 categories (200, 307 and 404)
- **ts**: long -> (bigint) timestamp of a given log
- **userAgent**: string -> browser or tool for using service
- **userId**: string -> user identifier

For this project, I selected 13 question related features, including **artist**, **auth**, **gender**, **itemInSession**, **length**, **level**, **location**, **page**, **registration**, **sessionId**, **song**, **userId** and **ts**.

Exploratory Visualization

The plot below shows distribution of when a user plays a song or cancels the service. The difference in the distribution is obvious. Thus the cancelation is mostly happened in morning around 7 am and in evening around 19 pm, while most user listen to songs at 16 pm. This is helpful to have a glimpse to users' service cancellation services.

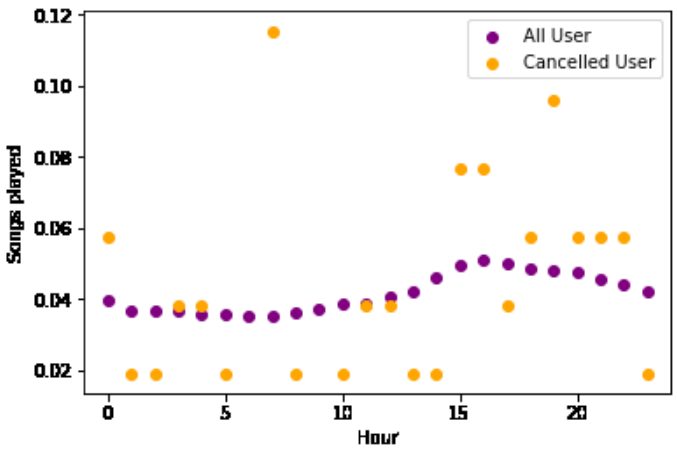

Figure 1. A plot showing distribution for time that user play songs or cancel services.

Algorithms and Techniques

PySpark library is used for performing exploratory data analysis at scale, building machine learning pipelines.

Methodology

Data Preprocessing

The dataset is preprocessed to remove missing values in sessionId and userId, remove duplicated records. The dataset records number drop from 286,500 to 278154.

Implementation

The implementation for this project including 3 folds, including models, metrics and data handling.

Models include lasso linear regression models and general linear model, which are implemented from pyspark.ml.regression module.

Metrics including r2 score (), f1 score and accuracy socre. These are implemented from pyspark.ml.evaluation module.

The data handling is through pyspark.sql module, which is an API for python to handle big data through Spark.

Refinement

For lasso linear regression, the regularization parameter is set to 0, the elastic network parameter is set to 1 for implementing l1 penalty for overfitting.

For general linear model, default set of parameters are implemented.

Results

Question 1. How long does it take for user to cancel the service?

The average time interval for a user to subscribe and then cancel is 57 days. Most unsubscription happens around 50 days. Only few user unsubscribe after 188 days (Figure 2).

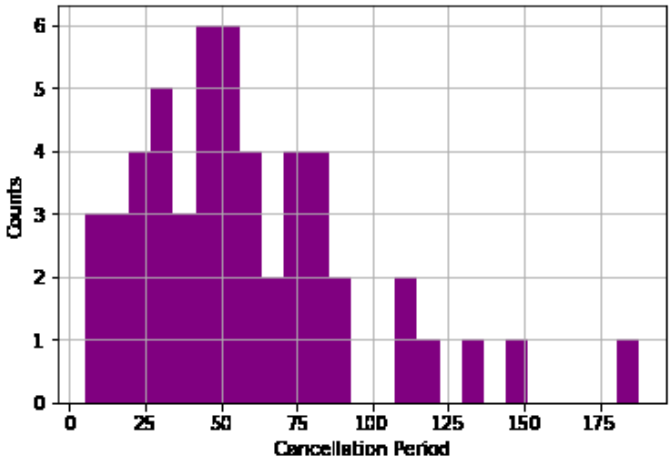

Figure 2. Time period (days) distribution before user unsubscribed from the service.

Question 2: What are the top songs paid user like?

Top songs subscribed user like are:

- You're The One
- Undo
- Revelry

Question 3: What is the main factor that contribute to a free user subscribing the service?

For exploring this question, I created 13 features, and created linear regression models for understand the most beneficial or deteriorating factors that impact user's subscription behavior. The label for the model is created from the most recent "**level**" attribute of the dataset. And the 13 features are list bellow:

- **aritstCount** – the total number of artist the user has listened to.
- **gender** – the user gender.
- **addfriendCount** – the total number for a user adding friends.
- **addplylistCount** – the total number for a user adding a song to a play list.
- **errorCount** – the total number of users encountering error messages.
- **downgradeCount** – the total number of downgrade.
- **helpCount** – the total number for a user visiting help page.
- **SettingsCount** – the total number for a user visiting setting page.
- **ThumbDownCount** – the total number of thumb down.
- **ThumbUpCount** – the total number of thumb up.
- **upgradeCount** – the total number of upgrade.
- **sessionsCount** – the total number of sessions for a user.
- **songsCoun** – the total number of songs a user has listened to.

I build 2 regression model for solving the question. For lasso regression, I got following set of coefficients:

feature	coef
aritstCount	1.799450
addfriendCount	0.340525
ThumbUpCount	0.259553
downgradeCount	0.199955
upgradeCount	0.028441
gender	0.016902
errorCount	-0.001053
addplylistCount	-0.011718
SettingsCount	-0.023103
helpCount	-0.062746
ThumbDownCount	-0.070367
songsCount	-0.599392
sessionsCount	-1.630034

Table 1. Lasso regression coefficients for predicting currently paid/free user.

The lasso regression model apply l1 penalty to the number of features, so as to reduce overfitting. The r2 score for this model is 0.79, the f1 score is 0.77, and accuracy score is 0.76. Thus this is a meaningful model.

By reading Table 1., we find that the number of artists listened by users, the number of adding friends and the number of thumb up events are 3 factors that most positively impact user's subscription to the music service. This may imply the variety of song database and a friendly community are most important for users making decision to switch to a paid service.

A general linear regression model is also created for solving this question, and I got similar results (4). This model has better statistics, as the f1 score and accuracy score are 0.81, indication a consistent prediction for validation set.

Question 4: What is the main factor that contribute to a paid user cancel service?

Similar lasso regression was done for the same feature set, except that the label changed to reflect the "Cancellation Conformation" behavior. The r2 score for this model is 0.13, meaning this is not a very reliable model. The possible reason for this to happen might due to the bias in the dataset, since only a tiny part of total population cancelled their service, thus there is not enough information to create accurate prediction. Using a larger and more balanced dataset might give us better insight for user cancellation behaviors.

Though the model is not reliable, we could see that thumb up count, adding friends and adding songs to playlists negatively impact the cancellation decision, we could infer that a friendly community is a key to keep customers happy.

Conclusion

In this project, I created linear regression model to answer music service subscription related questions. The analysis is based on a small set of the Sparkify data. The data handling is mostly done by pyspark, which is a python API for Spark handling big data, thus the code would be potentially beneficial for analyzing large scale data. In this analysis, a big variety of songs plus friendly community are shown to have positive impact for user subscribing to the music streaming service.

Supplementary Documents

The GitHub repository for this project can be found in this link:

https://github.com/kikyo91/DS7-Sparkify

Reference

(1) https://en.wikipedia.org/wiki/Coefficient_of_determination

(2) https://en.wikipedia.org/wiki/F1_score

(3) https://towardsdatascience.com/metrics-to-evaluate-your-machine-learning-algorithm-f10ba6e38234

(4) https://github.com/kikyo91/DS7-Sparkify