Report of Music play amount prediction

Xinyi Wang

Abstract: This contest is based on historical data users of Ali music, recording the number of times each song is listened. The contest expects contestants to discover artist that is about to be popular due to the prediction of number of times of different artist's songs in each period.

Keywords: discover popular prediction

1. Introduction

After 7 years of development, Ali music has now a storage of million pieces of music. Tens of millions of users participate actively on the platform each day. This huge storage of music resources plays an important role in guiding us to have a better recognition of music trends. In the contest, I'll try to get a basic recognition of procedures in data mining. Choose different features and try different models to get a good result.

2.Analyzing and Coding

2.1. Data cleaning

My first step is data cleaning, considering user listening the same song in a short period (an hour) for many times click farming or just doing other things with the music on. Meanwhile, I also consider user continuously downloading or collecting music in a short time as abnormal operation, thus delete the corresponding rows.

pseudo-code:

```
if (user playing the same song in a short time) or (downloading or collecting many time in a short time):
delete the corresponding row (only leaving the first record)
```

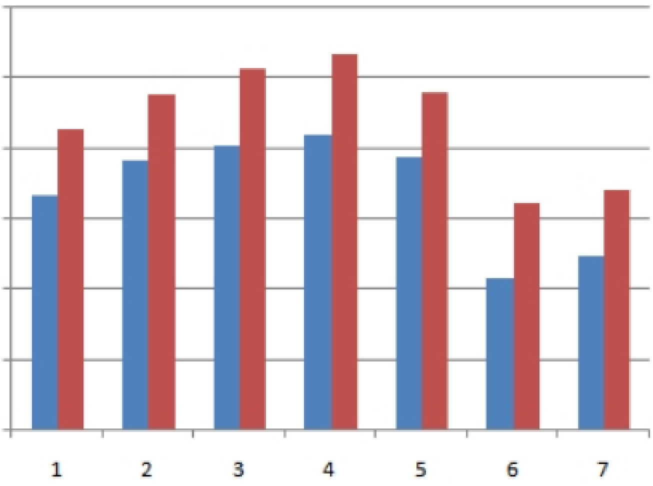
2.2. Features engineering

2.2.1.Old features

I analyzed the dataset on artist-song perspective, therefore I merged the user-action form and the artist-song form on the column ‘song_id’. Then I kept the original features ‘song_id’, ‘artist_id’, ’Ds’, ‘published_time’, ’song_init_plays’, ‘language’ ,’gender’, and also summed up ‘play’, ‘collect’ and ‘download’ numbers grouped by song_id. These are the original features I made use of, among which ‘Ds’ and ‘published time’ had negative effects on predict results, it also surprises me that ‘language’ and ’ gender’ contribute only a little to the final predict result. I’ve also Created some new features which I’ll mention below.

2.2.2.new features

(1)‘IsWeekend’ , ‘isMonday’, ‘isTueday’, ‘isWednesday’, ‘isThursday’, ‘isFriday’



The graph above shows artist’s played total amount each day in a week for 2 months, indicating the play amount of music perhaps follow a rule, which suggests during weekends, people have more spare time and have other freer ways to relax themselves, thus resulting in a low play amount in music playing. As for weekdays, we can see from the graph, from Monday to Thursday with the increment of working time, people’s stress accumulates and tend to listen to music more, while on Friday after finishing a week’s work people might feel relaxed and are free to try other entertainments. Therefore I consider one week as a cycle and created these features(however I found except for the feature isWeekend, whether to keep other features nearly don’t make any difference).

(2)‘dayAfterPublished’ and ‘has published recently’

I consider these two features a scale to measure the current heat of the song, both two features slightly raised the score of predict results.

2.3. Model selection

Tested linear regression, svm, random forest and GDBT, from which I chose random forest and GDBT, final model=random forest*0.5+GDBT*0.5, raising the score about 50 points.

2.4. Other small skills used to raise score

$$\sigma_{i,j} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left((S_{i,j,k} - T_{j,k})/(T_{j,k})\right)^2}$$

$$\phi_j = \sqrt{\sum_{k=1}^{N} T_{j,k}}$$

$$F_i = \sum\nolimits_{j\in W} (1 - \sigma_{i,j}) * \phi_j$$

As we can conclude from the picture above, two types of artists are important in our prediction:

(1) artists with large amount of play

$$\phi_j = \sqrt{\sum_{k=1}^{N} T_{j,k}}$$

(2) artist play amount with strange distributions which is hard to predict

$$\sigma_{i,j} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}((S_{i,j,k} - T_{j,k})/(T_{j,k}))^2}$$

Separate these two kinds of artist and predict their songs respectively will result in a small increment in your final score.

2.5 Test results

After testing changing features and different models my test result change between 5500-5600

3. Ideals didn’t finish due to the time limit

(1) Separate songs into two groups (hot songs or not) according to the total play amount in the first four month and build models for them respectively. (first classify an then do the regression）

(2) Create a new feature for song ‘is currently popular’. (haven’t yet got the ideal how to divide songs into these two areas)

(3)Is it possible to use incremental prediction for this problem? If it’s ok then we can make use of many sequential features such as the predict result of yesterday’s play amount, predicted play amount’s mean value, and predicted play amount’s derivative，yesterday’s output would become today’s input. I’ve used this way in artist-artist perspective and received quite good result for most artist(I guess the result will be better if using artist-song sight)， however some artists’ play amount follows no periodicity and will change abruptly. The incremental model receives a very bad result for these artists got -2400 score for the worst one). So, is it possible to classify artists into regular ones and irregular ones and apply incremental model (which seems to have a quite good effect on regular artists) for regular artists?

(4) use K-means first to cluster songs into different groups and to get Cluster Feature

3. Questions

(1) Is there a better way to do feature engineering other than creating features explicitly, I found it that feature engineering cost most of my time and require much math and expertise.

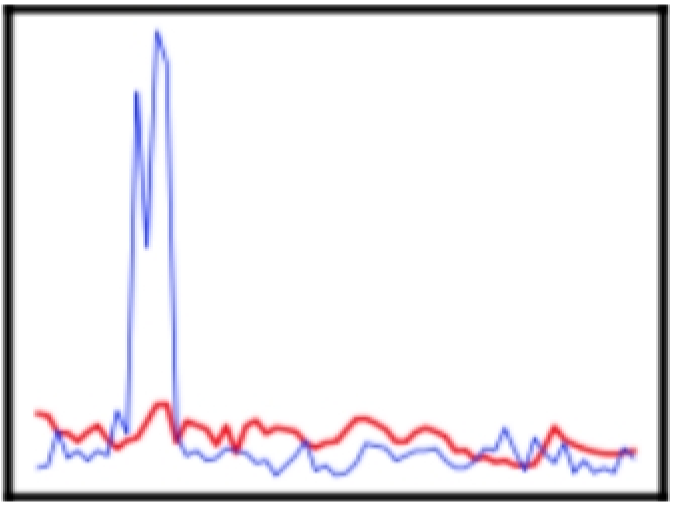
(2) How to do data cleaning, to choose which data to delete manually or is there a rule to follow (such as a relationship between the d-value of record vale and mean value, and the variance of the whole dataset)?

(3) I found that song play amount has a cycle of one week, however, creating features isMonday, …, isFriday or not doesn’t make difference, why?

(4) As we can see the grading formula is as below：

$$\sigma_{i,j} = \sqrt{\frac{1}{N}\sum_{k=1}^{N}\left((S_{i,j,k} - T_{j,k})/(T_{j,k})\right)^2}$$

We can conclude that when the real play amount of an artist in a day is small while your predict amount is big, the error would be very large. For artist with artist id “2b7fedeea967becd9408b896de8ff903”



Whose play amount stays low for most of the time. However, in the end of June his play amount bounced up to maximum value, resulting in a great error. How to deal with these peak value which have great effect on result？(smoothing?)

(5) Is there a way to deal with the dataset to make it subject to normal distribution？

4. Future work

Refer to others' work and find out the reason for my bad result, then find the answers to my questions, realize the ideals I mentioned above and improve my model to get a comparatively high score.

5. Conclusion

After finishing this work, I barely had a basic understanding of the procedure of data mining. It really surprises me that feature engineering would be so troublesome and tiring. Besides I also learnt that we shall first analyze the data and find pattern in the dataset with the help of graphs, after that shall we build our features。This work also taught me more feature don't always lead to better results, features such as 'Ds', 'published time' had negative results when I'm predicting. In general, although I didn't get a good score for my model, I've learnt a lot through this progress, and I think I can learn more through realizing the unfinished work left behind.