Department of Computer Science

2018/2019

One Million Songs Dataset Project

The Individual Report

Supervisor:

Dr. Michael Bane

By:

Yuefeng, Liang - 201350099

1. Team Description

Our team name is Team 1 and our team members are me (Liang Yuefeng), Ismail Al Jahdhami, Carlos Arinto. In this project, we chose Carlos as the team leader. I am with Carlos were in charge of the main code part of the project. Ismail was responsible for the meeting recording, literature review and the small part of analysis.
Carlos mainly did the prediction model and feature selection, and I focused on the data transformation and cleaning data part and the main analysis part of the project. Ismail did the association of the main dataset with another small dataset that has song genre and tried to analyse the relationship between music types and song popularity.

2. Introduction

2.1. Project Description

With high volumes of data increasing in the music field, varying from textual data to audio and visual data, many organizations and individual around the world concern about how to deal with these data.
This project, titled "One Million Songs Dataset", performed analysis on songs' metadata. The aims were finding some patterns in the data and build a tool to use song popularity score to draw conclusions about the data that could be of interest to some workers or investors in the musical industry.
The objectives of this project were:
1.Research on previous literatures and use features that other people have considered worth analyzing
2.Determine whether popularity is a useful feature to model as well as the reasons why/why not.
3.Based on the outcomes of both previous objectives, carry out our own analysis of patterns in the data (whether it means trying to replicate the discoveries made or attempt to find something new)
4.Build a big data pipeline to implement objectives 1, 2 and 3. This pipeline will need to perform appropriately with the scaling of the data.

2.2. Background

In terms of analysis music metadata, there were some previous researches found some useful points, which we did some review on these literatures.
Pham et al (2015), found that predicting song popularity is important in keeping businesses

competitive within the growing music industry. Their research was based on collecting some features to predict popularity using different classification and regression algorithms. Thus, according to this study, we could obtain some points about how to predict the song popularity [1].
Mohamed Nasreldin et al (2018), determine whether the popularity of a specific song is predictable. Their study aimed to find answers to three key questions:
1.Find out the features for popular songs
2.What could be the largest impact on song's success
3.Can old songs be used to predict the popularity of new songs
They found the important features are related to Artist information (familiarities, hotness and artist identifications) and technical information like tempo, mode, and loudness. In addition, there are also some missing values for songs hotness feature in their dataset, so they filled by connecting to Top 100 songs website [2].
We found both literatures are useful for our project as they used the features that helped to find and predict the popularity of the songs.

3. Outcomes

3.1. Project

After the process of converting and cleaning the data, we got the cleaned dataset.
Then, we used the prediction model and extracted the feature importance. This allowed us to decide we should focus on these features on popularity as they were better to predict. Thus, we got four features to analysis (Popularity, Artist Familiarity, Artist Hotness, Loudness).

Feature (showing top 6)	Importance
ArtistFamiliarity	0.3710
ArtistHotness	0.2037
Loudness	0.1009

Figure1. Feature importance

Then we implemented the general trend analysis of the selected numeric features of songs over past 20 years. (We found that the data in past 20 years occupied 80% of the whole sample dataset, so we decided to analysis these data).
Below is a simple plot produced in Python.matplotlib that shows the loudness of songs have been decreasing and in what seems to be a nonlinear pattern over time.

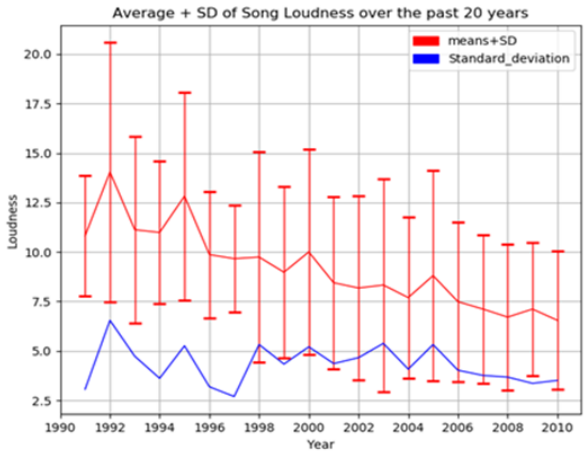


Figure2. General trend analysis of loudness over past 20 years

This seemed to be the only one pattern found in analysis of these four features as the other three graphs show some fluctuation lines over time.

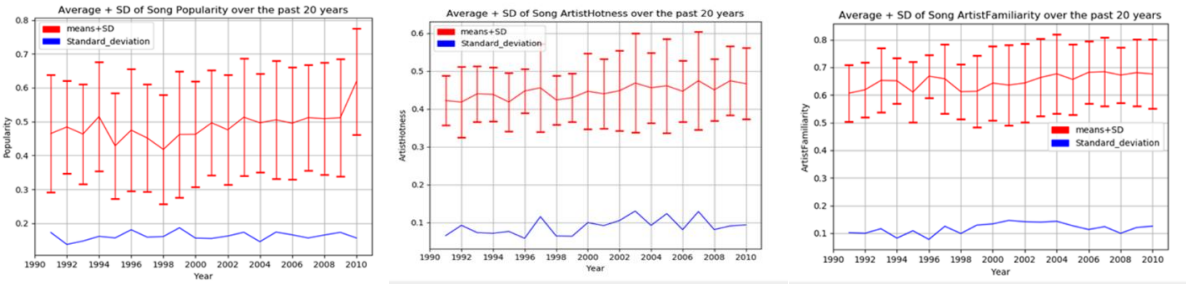


Figure3(a, b, c). General trend analysis of other three features over past 20 years

Then, we implemented the analysis of correlation between the song popularity and artist location and the song genre respectively. We tried to find out if these two features related to the song popularity. However, according to the output graphs, there may be no correlation between popularity and the artist location or the song genres. The lines are fluctuating over time in two graphs, so there seem to be no pattern.

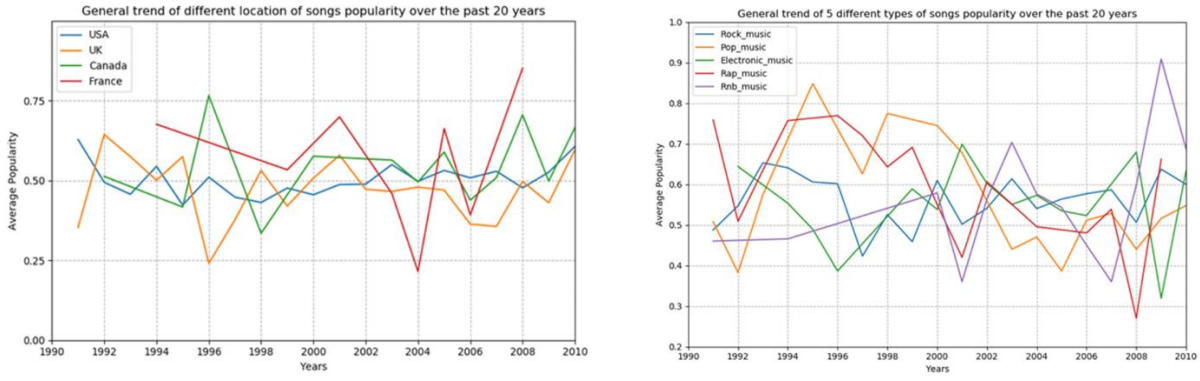
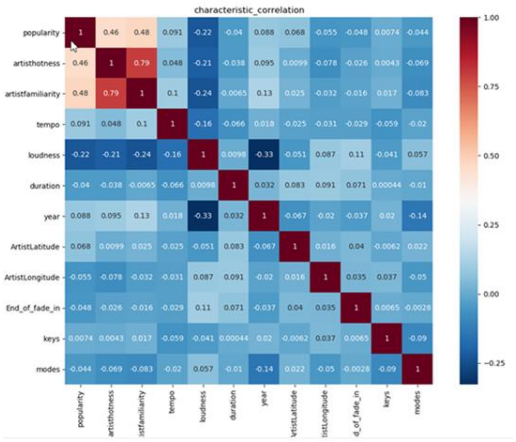
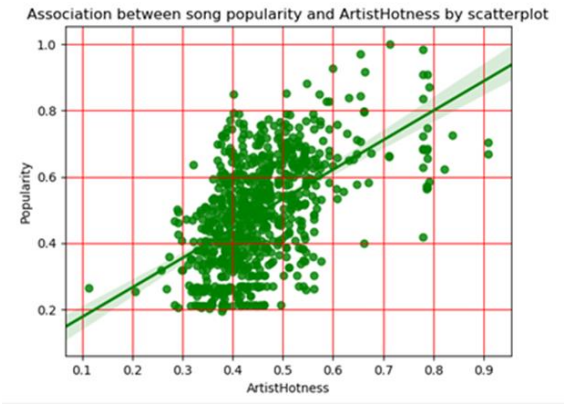
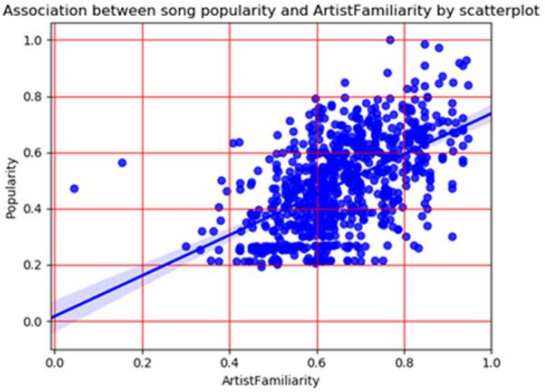


Figure4 (a, b). Correlation analysis of two features with song popularity

In the third part of analysis, we applied the correlation analysis between all features to get a heatmap. Then, we visualized the linear relationships determined by regression using a method in Seaborn library [3].







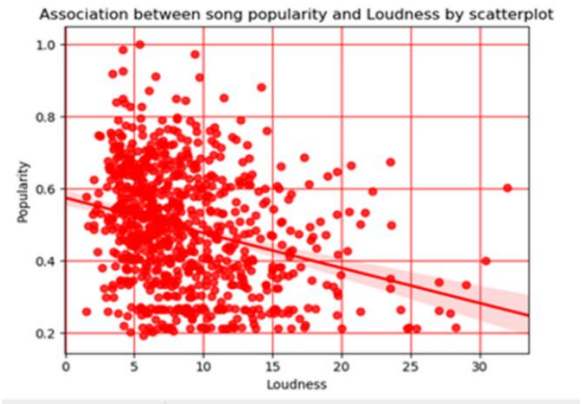


Figure5 (a, b, c, d) The correlation heatmap and three scatter plots

These three graphs show us the regression lines, and we could obtain the conclusions about the linear relationship between these features. It seems to be with the artist hotness or artist familiarity increasing, the song popularity increased, while when the song loudness increasing, the popularity decreased.

As for the prediction model, we used two classifiers, Multilayer Perceptron and Random Forest, and the ratio of split data was 7:3. That ratio means we used 70% of data to train and 30% of data for testing.
Finally, we implemented some prediction methods to our features and tried to find out the most accurate one to use. According to the RSME, we think the Random Forest method is the most accurate one.

Popularity

Regression Methods	RMSE (100 runs average)
Random Forest	0.13983
Decision Tree	0.14110
Gradient-boosted	0.15258
Linear	0.14110

Year

Regression Methods	RMSE (100 runs average)
Random Forest	10.3418
Decision Tree	10.9350
Gradient-boosted	10.7310
Linear	10.8888

Figure6. Methods for prediction model

After the prediction model working, we could obtain the best features and the ability to predict years and song popularity based on other features. These best features were used in the analysis part of our project, for now we could get three best features related to the popularity.

2.2. Individual

2.2.1. Data transformation and Cleaning

As for this project, I started with the converting and cleaning the data. Before this process, the source data was in .h5 format. I used an algorithm to read each data file (each .h5 file included the metadata of only one song) and extracted the columns that what we need. Then, I converted it into a csv line and output to a csv file.

```
## Extraction ## transforming, loading
songH5File = open_h5_file_read(currentPath4)
song = Song(str(get_song_id(songH5File)))

##Transformation - ##
song.artistID = str(get_artist_id(songH5File))
song.albumID = str(get_release_7digitalid(songH5File))
song.albumName = str(get_release(songH5File))
```

Figure7. Codes for read .h5 file and extract columns

I tried to compare the pythonic way and the pyspark way. The pythonic method means that I cleaned the data after I extracted the feature columns from hdf5 file. This method only used pythonic code in the whole process.
As for the pyspark method, after I got the csv file, I load it into spark dataframe and use the dataframe to clean the data. This method gave me the higher performance, so I chose to use this one to implement the cleaning process.

```
The average time of pyspark method:
convert:
458s

clean:
1s


The average time of pythonic method:
convert and clean:
620s
```

Figure8. Performance comparison

In the sample dataset and subset provided, there are rows which contain missing data and values that contain symbols such as punctuation signs and noise such as the existence of a "b'" before some of the song and artist names. I just deleted these rows. Also, as for the artist location column, I need to load a csv file included most cities names and countries names around the world, then converted the different ways in which locations were expressed into the corresponding country.

A	B	C	D	E	F	G	H	I	J	K	L
SongNum	AlbumName	Title	ArtistName	ArtistLocation	ArtistHotness	ArtistFamiliarity	Popularity	Duration	Tempo	Loudness	Year
1	b'Fear Itself'	bI Didn't M	b'Casual'	b'California - LA'	0.401997543	0.581793766	0.60211999	218.9318	92.198	11.197	0
2	b'Dimensions'	b'Soul Dec	b'The Box Tops'	b'Memphis, TN'	0.417499645	0.630630038	nan	148.0355	121.274	9.843	1969
3	b'Las Numero 1	b'Amor De	b'Sonora Santaner	b''	0.343428378	0.487356791	nan	177.4755	100.07	9.689	0
4	b'Friend Or Foe'	b'Somethi	b'Adam Ant'	b'London, England'	0.454231157	0.630382334	nan	233.4036	119.293	9.013	1982
5	b'Muertos Vivos	b'Face the	b'Gob'	b''	0.401723686	0.651045661	0.60450074	209.6061	129.738	4.501	2007

Figure9. Before cleaning

A	B	C	D	E	F	G	H	I	J	K	L	M
SongNum	AlbumID	AlbumName	Title	ArtistName	ArtistLocation	ArtistHotness	ArtistFamiliarity	Popularity	Duration	Tempo	Loudness	Year
16	135122	Outskirt	Floating	Blue Rodeo	Canada	0.44793548	0.636423645	0.405115722	491.1277	119.826	8.576	1987
24	223365	Miss Machine	Setting Fi	The Dillinger E	USA	0.541888972	0.839962768	0.666527846	207.7775	166.862	4.264	2004
25	652784	Sue Thompson -	James (Ho	SUE THOMPS	USA	0.306242264	0.435415818	0.495293621	124.8649	137.522	12.332	1985
33	643400	Best of The Shar	Twist and	The Shangri-L	USA	0.418216518	0.640807102	0.443291312	164.8061	130.1	10.922	1964
41	709879	The Palest Grey	Spin	Scarlet's Rema	USA	0.393121934	0.527865076	0.450992312	198.7391	115.061	7.469	2007
42	706005	A Match & Some	Burning In	The Suicide M	USA	0.467538444	0.668674533	0.528782481	95.68608	115.887	2.022	2003

Figure10. After cleaning

As for loading data to spark dataframe, I compared the performance of running the code in Dawson for reading from CSV or PARQUET, the results showed that it was 6 times faster to read from PARQUET file than from CSV. Thus, I used PARQUET format to save our cleaned data.

	CSV File	Parquet File
Average (10 runs)	0.83856859	0.13541603

Figure11. Performance comparison of reading different format files

2.2.2. Analysing Part

In the analysis part, I used dataframe in pyspark to extract the four selected features columns and applied plot method in Python.matplotlib to draw the trend analysis graphs.

As for the analysis of the correlation between popularity and artist location, I extracted the location column and the popularity column, then put them into two lists. Finally, I used these two lists to draw the analysis graphs by applied plot method in Python.matplotlib.

Also, I implemented the heatmap of the correlation matrix after calculated the matrix by using corr() method in python.pandas library. Then, I did the analysis of correlation between popularity and artist hotness or artist familiarity or loudness by using regplot in Seaborn library, which is a Python data visualization library based on matplotlib.

All these graphs are mentioned in the project outcome part.

2.3. Members' contributions

Carlos was responsible for building models for selecting features and predicting features. He also compared different regression methods and finally decided to use Random Forest regression method. Besides, he helped me with thinking out an algorithm to read all the .h5 file to get all data.

At the same time, Ismail was responsible for the literatures review and Meeting record. In the analysis part, as there is no song genre feature in the main dataset, he linked the main dataset with another small dataset, which has the genre feature. Then, he implemented the analysis of the correlation between music genre and popular music.

4. Evaluation

Strengths:

1. In this project, after cleaning process, we have a totally clean dataset and we could save the dataset into any general format.
2. Then, we built a model to predict some features, like popularity and years.

3. We found a trend pattern in the dataset, which is the loudness of the song decreasing over time. It could be of interest to a third party in the musical industry.

Weaknesses:

1. The performance of the prediction model is low, it took a long time to run the model.
2. About the cleaning data part, we just applied remove process to clean the data which means lose information in the cleaning process.
3. This project using Pyspark, thus the users may need the knowledge of python and spark.
4. It may take long time to run these codes in personal computer as the volumes of the data is high.

went well

1. Team worked well together and good communication with team members.
2. Everyone completed their assigned work on time.

what could have been done better

1. We may make the presentation better if we have more time to prepare.
2. If we research more knowledge of data science before the project, we would save the time in analysis part.
3. If everyone in the team try best to do the project, I think it would be done better.

what you, personally, have learned during the project

1. How to work with others to finish a project.
2. Some knowledge of data science, data statistics and data analysis (Descriptive and Inferential Statistics)
3. How to detect code bug error by myself and search for solution on the internet

5. Suggestions for future developments.

Firstly, as for the missing value, we could use some classifiers to fill in the empty value, but the classifiers must be good and accurate.

As for the performance of prediction model, we need to find out a more efficient model or try to improve the performance our owe model.

We could try to associate more dataset with the main dataset so that we could have more features to analyse.

We need to improve the understand of data science and some concepts in data science, we could do more practice or do some course about big data or read some data science blogs on the internet.

6. Professional Issues

PUBLIC INTEREST

In this project, we have due regard for privacy and security. The resource we used are all open source and our project is not going to infringe of the legitimate rights of third parties. Otherwise, we conduct our professional activities without discrimination on the grounds of sex, marital status, nationality, colour, race, ethnic origin, religion, age or disability, or of any other condition or requirement. As our team members come from different countries, so we did consider this rule in our activities.

PROFESSIONAL COMPETENCE AND INTEGRITY

We only undertake to do work or provide a service that is within our professional competence. This project is in big data filed, and all we have done are data analysis and data prediction. Also, we still need to develop the professional knowledge of data science, skills and competence on a continuing basis, maintaining awareness of technological developments, procedures, and standards in big data field. Moreover, we respect and value alternative viewpoints, accept honest criticisms in our project and we would avoid injuring others, their property, reputation, or employment by false or malicious or negligent action or inaction.

DUTY TO RELEVANT AUTHORITY

In our project, we seek to avoid any situation that may give rise to a conflict of interest between our team and the relevant authority. In fact, our project is a developing project, and we did not obtain any profit from it now. We did not disclose or authorise to be disclosed the confidential information, and we also did not use it for personal gain or to benefit a third party. We did not use any confidential information in our project. As for the project, we did not misrepresent or withhold information on the performance of systems or services. All our source code or outcome will be put into github.

DUTY TO THE PROFESSION

We accept our personal duty to uphold the reputation of the profession and good standing of BCS, The Chartered Institute for IT, and not take any action which could bring the profession into disrepute. Also, we will act with integrity and respect in our professional relationships with all members of BCS.

7. Bibliography

[1] Pham, J et al. (2015), 'Predicting Song Popularity', (online) Available at: http://cs229.stanford.edu/proj2015/140_report.pdf , [Accessed on: 24 February 2019].

[2] Mohamed, N et al. (2018), 'Song Popularity Predictor', (online) Available at: https://towardsdatascience.com/song-popularity-predictor-1ef69735e380 , [Accessed on: 24 February 2019].

[3] Seaborn library. (online) Available at: https://seaborn.pydata.org/index.html, [Accessed on: 27 February 2019]