Music Analysis using Spotify dataset

Adithya S Oasis, Abishai Ebenezer, and Dhruv Sharma
PES university
adithya.oasis@gmail.com, abishai.ebenezer.m@gmail.com, dhruvsharma2600@gmail.com

INTRODUCTION

Music is an art form, and a cultural activity, whose medium is sound. Music is a part of our daily lives and Studies have shown that when people listen to music, their emotions fluctuate, and the effect is to change their behavior (Orr et al., 1998). Studies have shown that different languages, tempos, tones, and sound levels of music can cause different effects on emotions, mental activities, and physical reactions. General definitions of music include common elements such as pitch, rhythm, dynamics (loudness and softness), and the sonic qualities of timbre and texture. Different styles or types of music may emphasize, de-emphasize or omit some of these elements. Music is performed with a vast range of instruments and vocal techniques ranging from singing to rapping; there are solely instrumental pieces, solely vocal pieces and pieces that combine singing and instruments. Our data analysis involves exploring the different aspects of a song and to compare them. Lyrics are words that make up a song usually consisting of verses and choruses. We have also used topic modelling so that we could use the lyrics in our data analysis. The aim of our project is to apply concepts that we have learnt in Data Analytics to a dataset [9] which contains various music parameters for each song and to appreciate the power of Data Analytics to discover useful information, even from unstructured data. In most of our tasks, we have tried to group songs together using various techniques in data analytics. This grouping forms the main basis for our data analysis.

PREVIOUS WORK

(**Abishai**) [1] deals with unsupervised learning of text profiles from unstructured text data. I have used various data analysis and machine learning techniques to create meaningful insights into the music dataset. In this paper, the authors have proposed a method of unsupervised learning of text profiles of the music. Also, they have suggested a formal method for analyzing the data of the profiles and use them to find similar artists. To accomplish this task, the technique of natural language processing(NLP) is used. As a part of the NLP, various algorithms like N-grams, Part of speech tagging, noun phrase extraction etc are used. These are simple noise reduction techniques that are used for a better search of similar artists/songs. Using the mentioned algorithms in the paper (like TF-IDF etc) , I have sought to categorize and group the songs.

(**Adithya**) [2] checks on the methods of creating appropriate scenarios for tracking music similarities and thereby using k-nn to find similar songs to recommend. They used mahalanobis distance as the metric to find and recommend the similar songs. Instead of using the covariance matrix they tested by using a dynamic matrix which was to learn with algorithms(eg- LDA, RCA, NCA). After testing by using the trained matrix from the different algorithms, they found that the difference in performance wasn't very drastic compared to just using a covariance matrix. They also found out that whitening or normalizing the data was found to have the largest impact which I have therefore used in creating the model for recommending the songs.

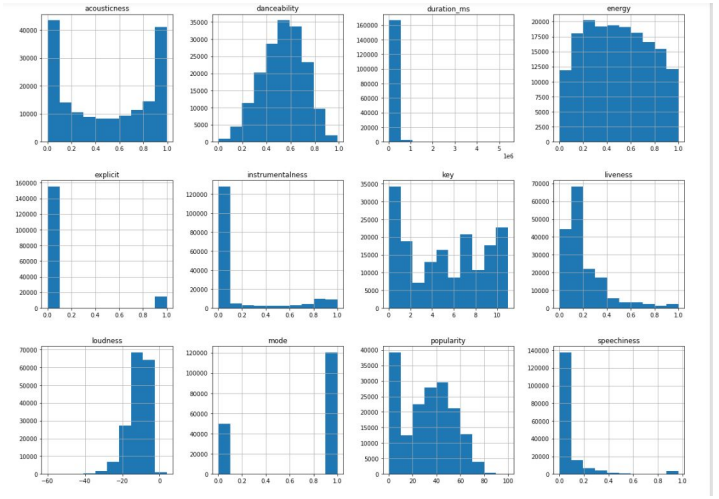
(**Dhruv**) [3] deals with genre classification & prediction based on various music parameters Like popularity,danceability , etc .We want to find similar music and classify and be able to predict the genre based on similarity.For this, they used several classification methods such as Knn, j48 decision tree, SVM, and many more.They tried to use a Knn classifier

combined with SMBGT[10] for classification of music.They also tried to improve the model by using, instead of just k-NN, a combination of several diverse and independent trained classifiers to try to improve the accuracy. I tried to use the above methods of knn, cosine similarity for genre prediction/classification.

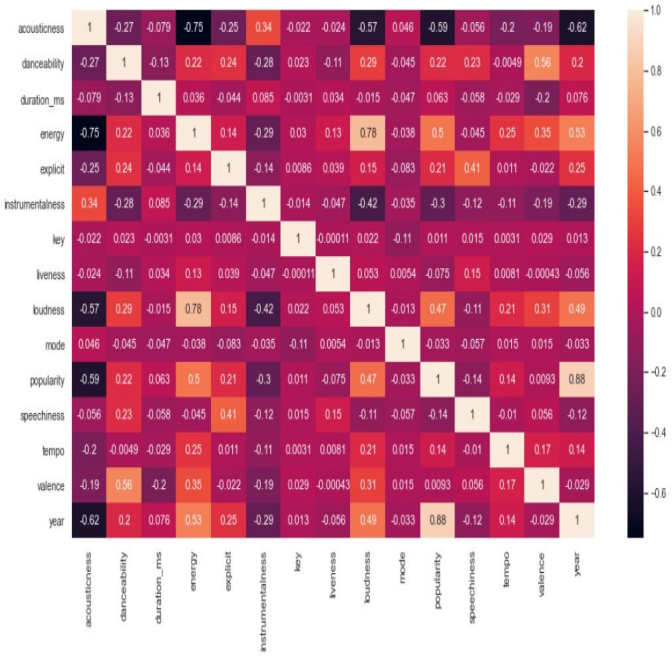
Exploratory Data Analysis

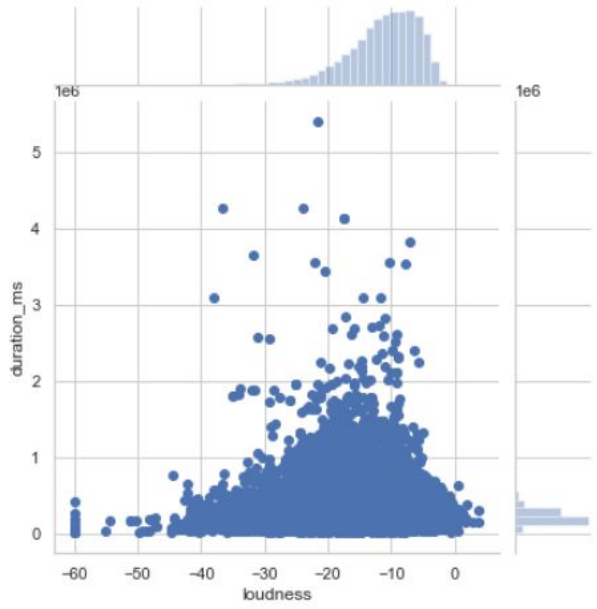
The dataset contains 169909 rows and 19 columns. There are no null values present in the original dataset.

```
In [3]: dataset.shape
Out[3]: (169909, 19)

In [4]: dataset.dtypes
Out[4]: acousticness         float64
        artists               object
        danceability         float64
        duration_ms            int64
        energy               float64
        explicit               int64
        id                    object
        instrumentalness     float64
        key                    int64
        liveness             float64
        loudness             float64
        mode                   int64
        name                  object
        popularity             int64
        release_date          object
        speechiness          float64
        tempo                float64
        valence              float64
        year                   int64
        dtype: object

In [7]: dataset.isnull().sum()
Out[7]: acousticness         0
        artists              0
        danceability         0
        duration_ms          0
        energy               0
        explicit             0
        id                   0
        instrumentalness     0
        key                  0
        liveness             0
        loudness             0
        mode                 0
        name                 0
        popularity           0
        release_date         0
        speechiness          0
        tempo                0
        valence              0
        year                 0
        dtype: int64
```



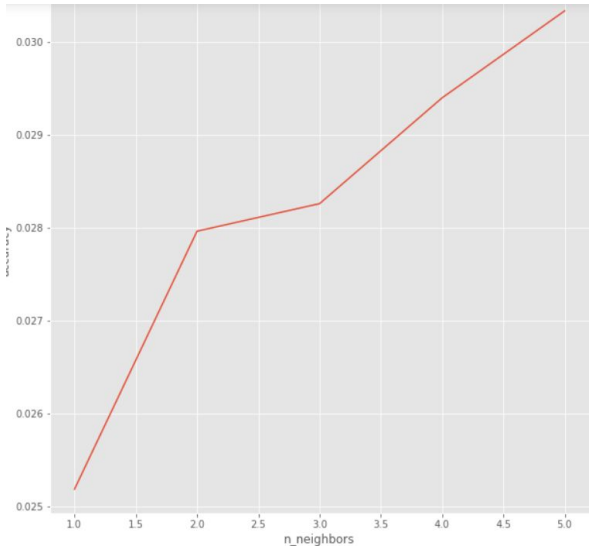
From the below map, we find that loudness and energy are highly correlated i.e they have high positive correlation coefficient = 0.8. We also observed that speechiness and popularity have a correlation coefficient almost equal to 0, which could infer that speechiness is not related to popularity. We also observed that energy and acousticness have high correlation . Below is also a scatter plot of loudness against duration(in milliseconds).



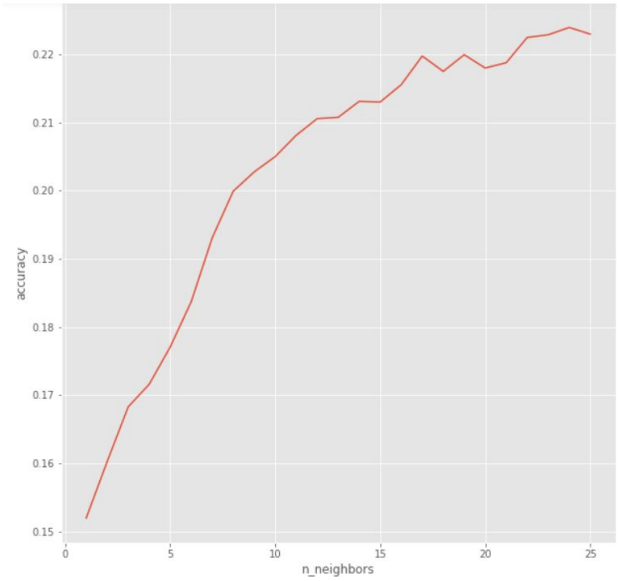


Individual Tasks

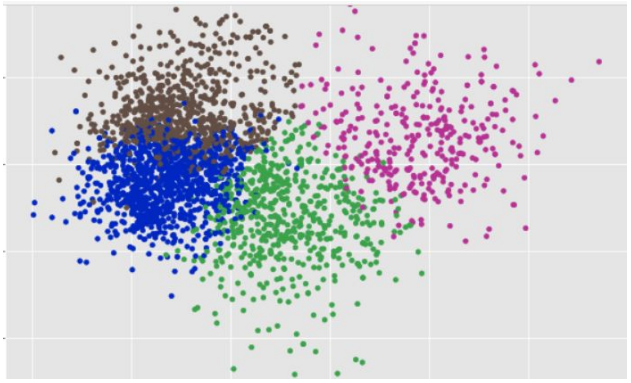
Dhruv Sharma- Genre classification - Our dataset had rows with missing genre values.So, to input the missing values I tried genre classification.For genre classification, I used 2 methods:- K nearest neighbour and K means clustering to input the missing values. K-nn is a supervised form of machine learning while k means clustering being unsupervised.First I tried using Knn to input the missing neighbours but that ended up giving a very low accuracy of approximately 0.03.



I then did knn classification for the 30 most popular genres,as there were too many genres and I thought only using the 30 most popular genres might give a better accuracy. I also increased the number of neighbours.The accuracy I got after training & testing the model on 30 of the most popular genres was:-



As we can see, the accuracy has greatly improved, so that means my method worked, but still is very less - 0.22. However, the accuracy couldn't improve any more.This means knn is not a good method of classification of genres due to its low accuracy. I also tried using cosine similarity but rejected that too because its accuracy was the worst (0.004). I then decided to use k-means clustering for classifying genres. After plotting clusters for different values of K and calculating silhouette scores[8] ,and doing t and anova tests, I found k=4 was the best number of clusters to take.So, the clusters look like this:- (silhouette score =0.162) k=4.



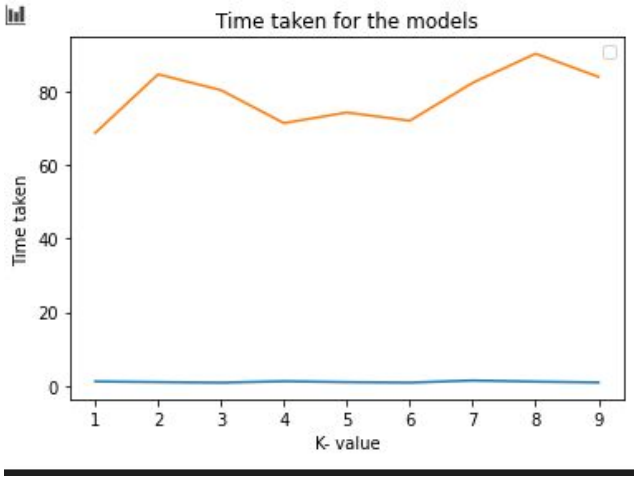
Hence, we classified all our songs in 4 genres based on parameters such as popularity, acousticness, danceability, and more.

Adithya S Oasis - Song recommendation system - I created a recommendation system that takes in a song and gives out a list of songs that are similar to the inputted one. Each song's parameters has been used to map every song into an Euclidean space where K-nn[6] was used to get the closest 'n' songs to give out as the recommendations.

I used PCA[7] to reduce the numeric columns to 11 components which are then used as attributes for the Euclidean space.

A lot of people would prefer songs from the same artists and hence in the given model, we can toggle whether the user would prefer if the songs having the same artist would have a higher weightage or not.

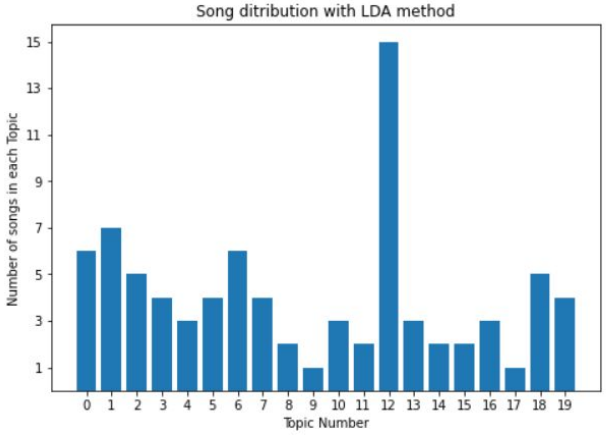
Using mahalanobis[2] as the distance metric increased the time to get the results by quite a bit and considering the recommendation for every song requires the check on the whole database, it was necessary to use a less computationally intensive metric for which i have used the euclidean distance which gave a better time taken to performance.



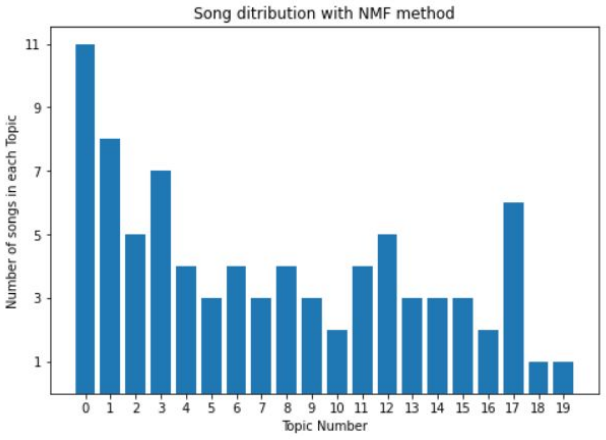
The above graph represents the time taken to run as K-value increases. Blue line representing the model with Euclidean distance and Red being the time taken with Mahalanobis.

After finding the similarity metric of every song compared to the inputted song, the results are sorted in accordance to the metric and popularity. From there the top K songs are returned as the recommendations.

Abishai Ebenezer M - Topic Modelling and neural networks. To achieve Topic modelling I used 2 methods - Non-Negative Matrix Factorization(NMF)[4] and Latent Dirichlet Allocation (LDA)[5]. These forms of Topic Modelling use unsupervised form of machine learning. Both these algorithms run through the text, and return a set of possible topics that the text could belong to along with their probabilities. This could give us an idea of what topic a song belongs to. The results are shown below for the first 200 songs in the dataset.



In the LDA method, we see that most of the songs out of the first 200 belong to topic 12. The top 15 words belonging to topic 12 are - 'got', 'glory', 'number', 'machine', 'rudy', 'bobby', 'waterhouse', 'said', 'way', 'long', 'time', 'just', 'like', 'randy', and 'lawrence'.



In the NMF method, we see that most of the songs belong to topic 0. The top 15 words for topic 0 are 'cuatro', 'qué', 'va', 'siento', 'mañana', 'dolor', 'sol', 'el', 'su', 'deja', 'cuando', 'ay', 'machete', 'tu', and 'instrumental'. By these methods

we see that we can use Topic Modelling to group songs together based on similar topics.

I also wrote an artificial neural network (PredictingPopularity) to predict the popularity of a song based on 9 different parameters such as acousticness, danceability, duration, energy, instrumentalness, liveness, loudness, speechiness and tempo. I decided to reward the model during training if it predicted the popularity of the song within a range of 10 popularity points. Doing such a training helped improve the accuracy of the model to 33.49% on the training dataset and 33.77% in the testing dataset. By this we prove that it is not particularly feasible to predict the popularity of a song just based on the above mentioned parameters. Popularity of a song might be more related to other factors such as genre of the song, artist, date of release etc.

Conclusion

In this project, we learnt how data analytics can be applied to analyze and compare music using varying approaches. We were also able to explore the different aspects of a song like lyrics, loudness, tempo, duration, speechiness etc. These parameters gave us great insights into how music can be explored from different aspects and how to use these parameters in doing our data analysis. We learnt how to apply the various concepts we learnt in class like dimensionality reduction, topic modelling, text classification, k-means clustering, similarity measures, knn, distance measures etc, to understand our data better and find similar occurrences based on these parameters. Along the way, we were able to learn the significance of various algorithms in machine learning and how to fit these different algorithms together to achieve our final desired result. To take this project forward, we aim to create better algorithms to predict popularity and even create songs of our own that could suit these parameters.

References:

[1] - http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.131.1237&rep=rep1&type=pdf

[2] https://www.slaney.org/malcolm/yahoo/Slaney2008-MusicSimilarityMetricsISMIR.pdf

[3] https://www.researchgate.net/publication/268525089_Music_Data_Analysis_A_State-of-the-art_Survey

[4] https://www.cc.gatech.edu/~hpark/papers/nmf_book_chapter.pdf

[5] LDA - https://ai.stanford.edu/~ang/papers/nips01-lda.pdf

[6]K-NN https://en.wikipedia.org/wiki/K-nearest_neighbors_algorithm

[7]PCA https://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html

[8]Silhouette-score https://en.wikipedia.org/wiki/Silhouette_(clustering)

[9]Database https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks

[10]SMBGT https://www.semanticscholar.org/paper/Genre-classification-of-symbolic-music-with-SMBGT-Kotsifakos-Kotsifakos/5600aa7117a3f4254ab98a5874225014e7941879