# Data Mining Final Project

Aastha Ghimire, Grace Jenkins, Hannah Jones, Wenying Shen

April 2021

## 1 Introduction

For this project we used a Spotify dataset to analyze popularity scores of tracks based on audio features. We used a variety of supervised and unsupervised methods with varying success to explore the popularity and genre of the data.

### 1.1 Data

The Spotify data was uploaded on Kaggle by Yamac Eren Ay who scrapped this data using Spotify Web API, Spotipy. It is updated weekly and out of the 15 versions, we are using version 11. He includes six total tables. We use three tables: `data.csv`, `data_w_genres_o.csv`, `data_by_genres.csv`. The data table contains several thousand tracks, their artist (or artists), release date, the numerical values of tracks' audio characteristics, and popularity score. The data with genres table contains various audio features of genres which are aggregated by calculating mean for numerical audio features and mode for categorical audio features. It is not clear exactly how the data is sampled, but the author mentions that the data is randomly selected by Spotipy (Spotify Web API) and limits pulling 2000 songs per year.

One of the variables we are interested in predicting, 'popularity', is an integer value between 0 and 100, 100 being the most popular song. It is based on a Spotify algorithm that weighs the total number of streams, and those streams' recency [2]. The popularity changes overtime, meaning depending on when data is scrapped, the popularity could be different. For example, White Christmas by Bing Crosby will have a higher popularity score around the holidays. According to Spotify's Web API Reference, duplicate tracks (e.g. the same track from a single and an album) are rated independently, which is something we accounted for during data cleaning.

We began data exploring and modeling with two specific questions listed in our mind .

1. Can we predict popularity based on the auditory components of the song and the release year?
2. Will grouping of the songs correctly predict genre? Are there visible groups based on genre?

## 2 Data Exploration

To begin our analysis, we looked at the correlation plot of various audio features, specially correlation of popularity with other features as shown in Figure 1. As shown in correlation plot, popularity is comparatively highly correlated with the years a song has existed, acousticness, and energy.

### 2.1 Popularity Score

As previously mentioned, there can be multiple copies of the same song, released at different times as a single, on an EP (Extended Play Record), and then an album. We found that the same song with the

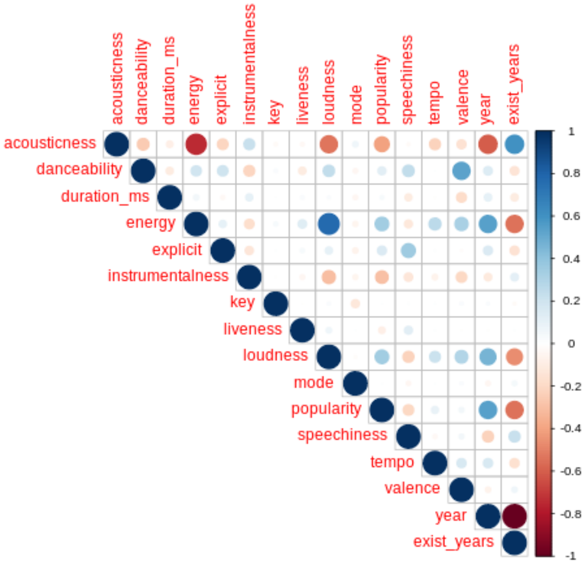

Figure 1: Correlation of audio features

same track characteristics would have different popularity scores due to different releases of the same song. To mitigate an effect on our regression predictions, we decided to remove any song duplicates. We chose to include a song with all its original characteristics but with the highest popularity score.

## 2.2 Genres

Spotify has over 2900 unique genres ranging from 8-bit to Canadian pop (hello Justin Bieber!). A majority of artists are categorized under multiple genres as well. for example, Unknown Mortal Orchestra, can be classified as ‘chillwave’, ‘indie rock’, ‘indie soul’, ‘kiwi rock’, ‘neo-psychedelic’, ‘new rave’, ‘portland hip hop’, and ‘psychedelic pop’.

We used the `data_w_genres_o.csv` to extract genre from each artists and join this with the dataset with all the songs. However, an artist can be given one or more of thousands of unique genre types. In order to be able to use genre for our analyses, we combined various genres according to our music knowledge and expertise. For example, we combined unique genres such as salsa, bachata, rumba, etc. into one genre that we called ‘Latin’. In doing so, we reduced the number of genres to eighteen.

We then looked a the correlation of these eighteen genres in order to ascertain whether more consolidation was necessary. The correlation plot can be seen in Figure 2. Based on these correlation, we further consolidated genres. For example, hip hop and rap were combined into one genre. We then joined these genres with our database that contains all the songs by the artist(s) on the song.

One interesting thing we came across during genre analysis is that Ghanian pop is the genre with the most danceability (describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity.)

Out of the consolidated genres, we found their distribution as shown in Figure 3, where each square represents 1000 tracks.

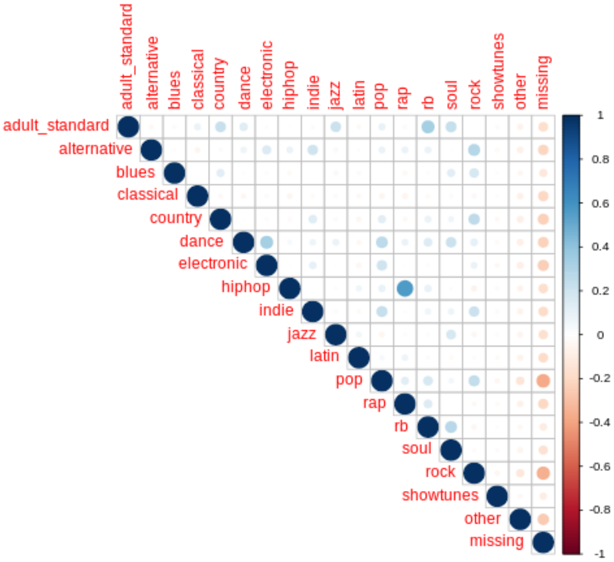

Figure 2: Correlation of genres

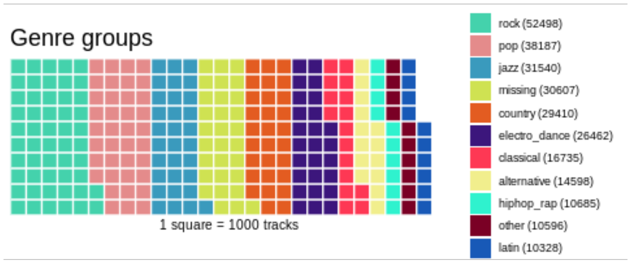

Figure 3: Total of Genres

# 3 Regression on Popularity

## 3.1 The Model

In predicting popularity, we used a linear regression model. We also tried regularization methods such as ridge, enet, and LASSO, as well as PCR. However, our number of parameters is much smaller than the number of training examples and it seems that the linear model does not suffer from over-fitting, as shown below. Indeed, the regularization methods return essentially the same coefficients as the original linear model. Similarly, PCR does not improve performance. We believe this is because it takes about half of the 14 parameters.

## 3.2 Grouping Data

As noted earlier, the data has a strange artifact regarding popularity by year. We believe that this artifact either has to do with how Spotify calculates popularity or how the Spotify Python API pulls the songs from Spotify. In any case, we decided to split the data along this artifact and see if our regression results improved along the split. We split the data into two groups, ”Above Artifact” and ”Below Artifact” according to Figure 4.

In Figure 5 we see that indeed the RMSE of the linear model decreases when we split and perform regression on the different parts of the data.

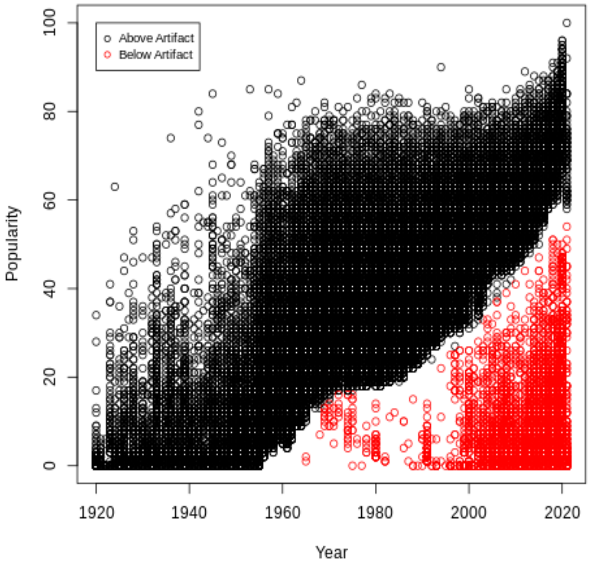

Figure 4: Splits for Data

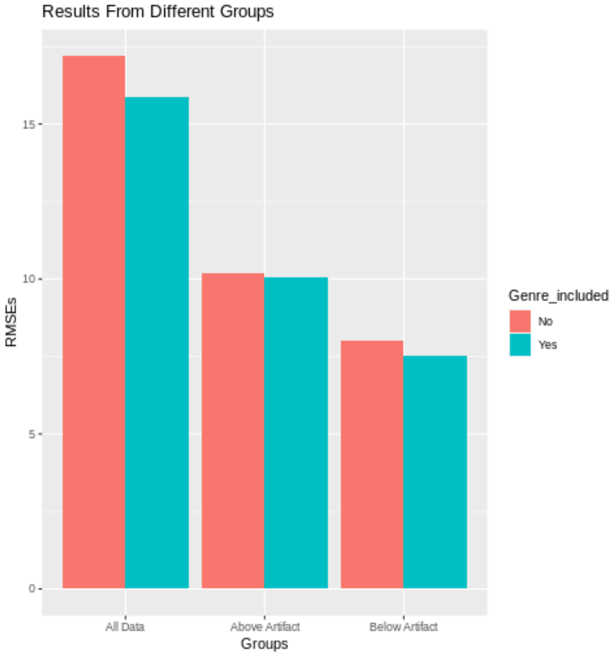

Figure 5: Regression Results for Different Groups, Including and not including genre

## 3.3 Including Genre

As described earlier, we consolidated the genres for each artist based on our knowledge of music as well as the correlation between certain genres. We coded these 10 additional columns as vectors of 0s and 1s if the song was in that genre. We added these categorical parameters to the linear model. Figure 5 shows the effect on the model of adding these categorical variables. It does not seem that adding these variables leads to over-fitting, and indeed including the genre categorical variable seems to somewhat improve results. However, for both above and below artifacts, the model is not significantly improved.

The Above Artifact group contains the majority of the data and gives better regression results. Table 2 reports the coefficients for this regression both with and without genre. As dance lovers, we were excited to see that danceability has the largest positive coefficient when predicting popularity. The most negative coefficient was acousticness, which tracks well with what we know to be popular types of music. In terms of genre, rock and latin were the genres that had the highest positive coefficient.

| Feature | Coeff. without Genre | Coeff with Genre |
|---|---|---|
| acousticness | -3.92 | -3.00 |
| danceability | 5.46 | 4.86 |
| duration_ms | -8.91e-07 | -9.10e-07 |
| energy | -1.30 | -1.60 |
| explicit | -5.76e-02 | 0.866 |
| instrumentalness | -2.81 | -1.85 |
| key | -7.88e-04 | 8.97e-04 |
| liveness | -3.14 | -3.21 |
| loudness | 0.102 | 8.88e-02 |
| mode | -0.294 | -0.299 |
| speechiness | -5.71 | -3.33 |
| tempo | 5.92e-03 | 3.19e-03 |
| valence | -1.51 | -1.91 |
| exist_years | -0.709 | -0.704 |
| hiphop_rap |  | 7.53e-03 |
| electro_dance |  | 9.23e-02 |
| jazz |  | 1.91 |
| alternative |  | -2.21 |
| rock |  | 2.74 |
| other |  | -0.874 |
| pop |  | 0.901 |
| country |  | -0.906 |
| classical |  | -1.81 |
| latin |  | 2.35 |
| missing |  | -1.65 |

# 4 Predicting Genre

## 4.1 LDA

One way that we decided to look at genre is by using LDA to model the binary classification of whether or not a certain song is in a genre. We decide to perform this analysis on pop and rock. We performed LDA using all the musical features of the song and the number of years a song has existed.

### 4.1.1 Pop

We conducted LDA on the 'pop' genre that we consolidated. 23% of songs fall under this category. The model predicted whether or not a song was in the pop genre with 76% out of sample accuracy. The confusion matrix is shown below.

|  | Actual:0 | Actual:1 |
|---|---|---|
| Pred:0 | 93271 | 3824 |
| Pred:1 | 26114 | 4520 |

Table 1: Confusion Matrix for LDA on Pop

We looked at the predictions graphed with respect to different features and probability. This can be seen in Figure 8. As expected, higher popularity consistently yields predictions in the pop category.

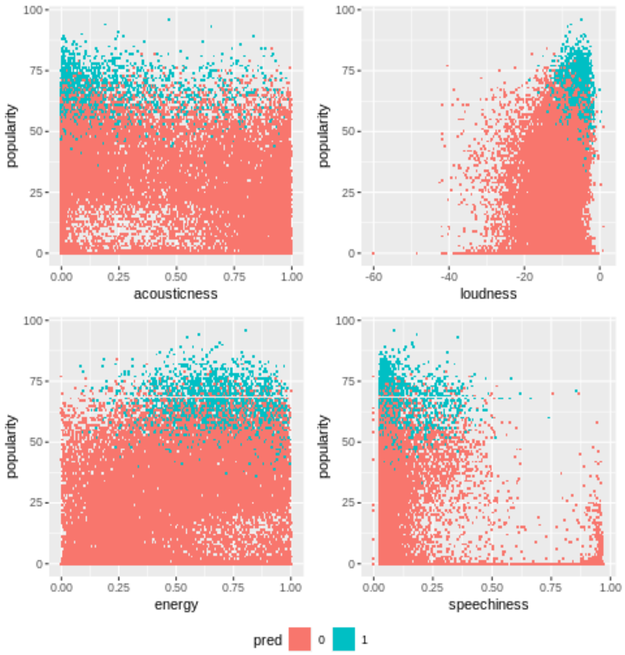

Figure 6: LDA for Pop

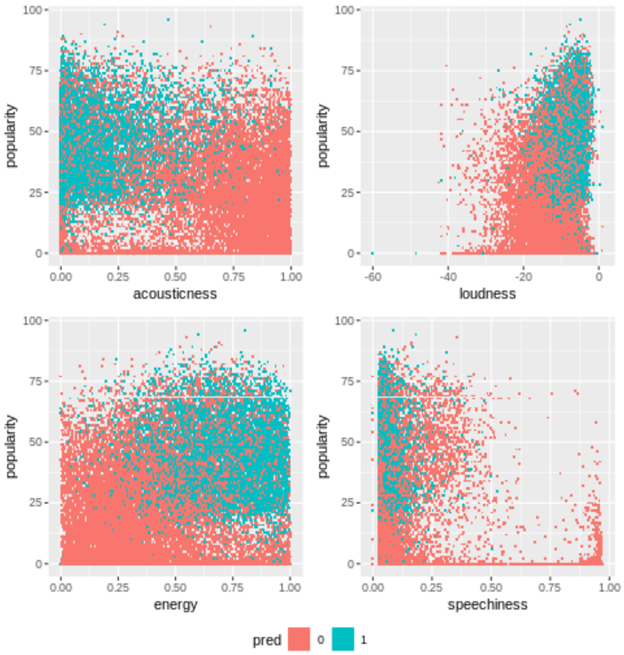

Figure 7: LDA for Rock

### 4.1.2 Rock

We did the same LDA on our constructed Rock genres. 32% of the songs fall under this aggregated category. The overall out of sample accuracy for LDA was also 76%. The confusion matrix is below.

| | Actual: 0 | Actual:1 |
|---|---|---|
| Pred:0 | 73818 | 11844 |
| Pred:1 | 18601 | 23466 |

Table 2: Confusion Matrix for LDA on Rock

## 4.2 K-Means Clustering

We decide to take on an unsupervised clustering approach because we are interested in seeing if the tracks cluster by genre. As mentioned data cleaning, the hundreds of unique genres are grouped into ten over arching genres.

K-means clustering was done using the `kmeans` function in R. The input for the function is all the numerical values, besides the classification of genre. Clustering is an unsupervised technique, which is why the classification of genre columns was removed. Initially, we tried to create clusters with 11 centers, which would mimic the number of overarching genres we have. There wasn't much of a correlation with 10 clusters, as the groups were extremely different sizes, see table, and the total within sum of squares was 4.988644e+14, or 33.84336 taking the natural log.

Finding that the data doesn't naturally group its self into genres mimics the findings from WM undergrads Ethan Shelburne '21, Clare Heinbaugh '23 and Stuart Thomas '21. They found that genre does not predict

|  | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 |
|---|---|---|---|---|---|---|---|---|---|---|---|
| `kmeans` centers | 24595 | 21885 | 20937 | 18580 | 17473 | 15015 | 14640 | 12790 | 9802 | 3535 | 189 |
| Genre Amount | 52498 | 38187 | 31540 | 30607 | 29410 | 26462 | 16735 | 14598 | 10685 | 10596 | 10328 |

or correlate with musical elements, and decided to create their own specifications for ”genre” based on more aural characteristics [1]. With our outcome, we set out to discover if the data clusters naturally a different way. Below is a graph of the total within sum of squares for 1 to 50 centers.

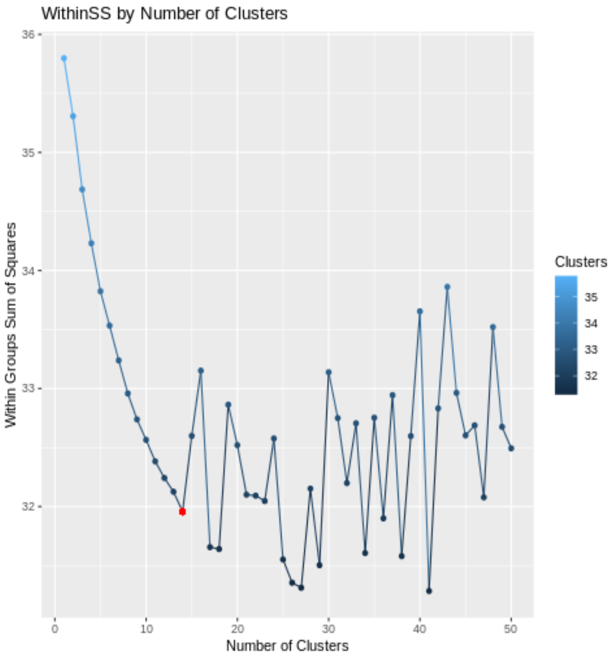

Figure 8: Cluster Within SS

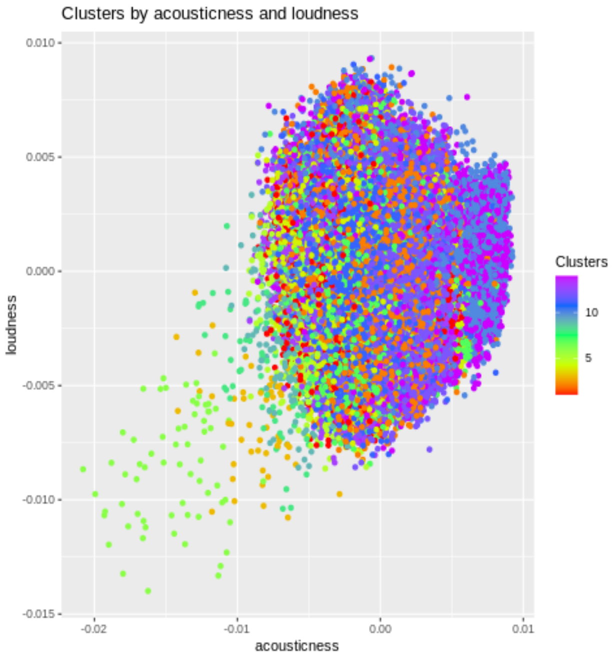

Figure 9: Cluster Groups

There really isn't an ”elbow” on the graph, but we found that 14 minimized the total within sum of squares before a large jump in the error. The number of centers is set to 14 for further analysis. It was apparent that for each group, it is hard to gauge if there are any clusters. Below is a graph of `acousticness` and `loudness`.

Over the 14 cluster groups, we assigned the highest average genre value to that cluster's category. When this happens, seven clusters belong to the rock genre, one to pop (a size of 61 tracks), three are classical and three are missing. Only four of the original 11 genres are present! It is difficult to say what groups the pop cluster, but one strong element that is consistent within the group is 39 of the 61 tracks belong to ‘arab folk’, ‘belly dance’, and ‘classic arab pop’.

# 5 Conclusion

Music streaming has gain popularity since 2008, the year Spotify was founded. We were interested in exploring how a song's audio features may predict it's popularity or to determine what genre a track is. Using many regression methods and clustering on a Kaggle dataset we discovered it's extremely difficult to understand, through Spotify's eyes, how music is classified.

We found that the regression worked the best for the data above the artifact and the highest coefficient was danceability. The music did not cluster by genre but it was also unclear how it did group.

## References

[1] *Sonic  iconic: WM team wins top honors at international data analytics competition for model on musical influence*. Accessed: 05/04/2021. URL: `https://www.wm.edu/news/stories/2021/sonic-iconic-wm-team-wins-top-honors-at-international-data-analytics-competition-for-model-on-musical-influence.php?utm_source=wmdigest&utm_medium=email&utm_campaign=news`.

[2] *Web API Reference*. Accessed: 05/04/2021. URL: `https://developer.spotify.com/documentation/web-api/reference/#endpoint-get-several-tracks`.