Songs Genre

Yoann Ayoub and Louise Lizé

September 12, 2024

Abstract

This document presents an analysis of the 'Songs Genre' dataset. The latter comes from Spotify and its package spotifyr, created by Charlie Thompson, Josiah Parry, Donal Phipps, and Tom Wolff. This document presents the various analyses performed on this dataset and the results obtained.

1 Data Overview

Our dataset consists of 23 variables and over 30,000 entries. It includes approximately 5,000 tracks from 6 playlist genres: Edm, Latin, Pop, R&B, Rap, and Rock.

We have categorized the variables into four groups:

1. Qualitative identification variables: for the *tracks*, we have trackID, trackName, and trackArtist; for the *albums*, we have trackAlbumID, trackAlbumName, and trackAlbumReleaseDate; and for the *playlists*, we have playlistName, playlistID, playlistGenre, and playlistSubgenre.
2. Quantitative characterization variables:
   - *key* : represents the keys, 0 = C, 1 = C#/Db, 2 = D, etc. -1 indicates an undetected key.
   - *mode*: 1 for major, 0 for minor.
   - *loudness*: overall sound level in decibels.
   - *tempo*: the tempo in beats per minute.
   - *durationMs*: the duration in milliseconds.
3. Quantitative track description variables (based on measurements and algorithms, with values ranging between 0 and 1):
   - *danceability*: describes how "danceable" a track is.
   - *energy*: measures the intensity and activity of a track. For example, metal would be close to 1, while Bach's preludes would be close to 0.
   - *speechiness*: detects the presence of spoken words in the track. For instance, poetry is close to 1, while highly sung genres like Rock are close to 0.
   - *acousticness*: as the name suggests, it measures how acoustic the track is.
   - *instrumentalness*: predicts whether a track is "non-vocal." Onomatopoeias like "ooh" and "aah" are treated as "instrumental," while rap or spoken words are classified as "vocal."
   - *liveness*: detects the presence of an audience in the recording.
   - *valence*: describes the positivity conveyed by a track. 1 for a happy track, and 0 for a sad one.
4. Finally, the quantitative variable *popularity*, ranging from 0 to 100, represents the popularity of a track.

2 Descriptive Data Analysis

2.1 Data Cleaning

The initial analyses allowed us to clean and make modifications to our dataset. By studying the *duration* variable, we noticed that some tracks are very short. Tracks with a duration of less than 1 minute were therefore removed. To facilitate our analyses, we also split the variable corresponding to the date into a year variable and a month variable. We decided to use the year as a quantitative variable later on, as it allows us to obtain much more interesting results. However, it will be important to be cautious when using it in certain cases, for example, if we want to predict labels for new music (after 2020). We also normalized our quantitative variables using the *MinMaxScaler* function from sklearn.

2.2 Quantitative Data Analysis

Initially, we created boxplots for the different quantitative variables. We observed some initial differences between genres. For example, the variable *speechiness* (figure 7), where the values for the Rap genre seem more dispersed and higher than for other genres. This seems logical since it indicates that Rap has the most spoken words.

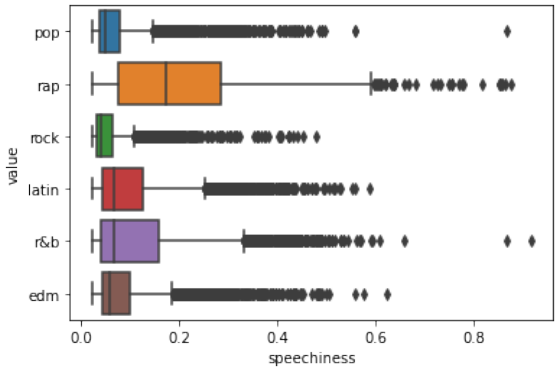

Figure 1: Boxplot - speechiness by genre

Through this type of analysis, Edm seems to be the genre that is easiest to distinguish from the others. Its boxplots often stand out. For instance, Edm music appears to have a lower *acousticness* than the other genres. It has a tempo that is much less dispersed compared to others, revolving around 125 beats per minute. It also seems to be the least popular and generally has a lower valence than the others. The boxplots for *trackAlbumReleaseYear* are also very different depending on the genre. Rock music shows a large dispersion (1980-2010) compared to the others, which generally start from the 2000s (R&B, Rap) or even the 2010s (Pop, Latin, Edm).

We attempted to perform the same analyses based on song popularity. Apart from very popular songs (>80/100), which are mostly concentrated around the year 2019, we did not notice significant differences in the variables based on popularity.

We can also highlight the initial correlations between our variables by generating a heatmap. We observed a relationship between *loudness*, *energy*, and *acousticness*. These correlations could be interpreted as follows: an energetic track has a high sound level, and an acoustic track seems to have very little energy and low sound intensity, which seems quite logical. To further the study, we analyzed correlations within the different genres. This allowed us to distinguish the Rap and Edm genres, where the number of correlated variables is greater and the correlations are stronger. For Edm, we notably found an interesting correlation between popularity and instrumentalness-duration.

2.3 Analysis of Qualitative Data

We recall that the dataset is constructed by collecting music from playlists. Therefore, about 25% of the tracks in our dataset appear in multiple playlists and are thus listed more than once.

One qualitative variable that caught our attention is the *trackArtist* variable. Initially, we studied the number of tracks present in our dataset by artist. Fifty percent of the artists have more than six tracks, which we consider the "main" artists. The others will be referred to as "small" artists. We also find an almost 50-50 distribution between the number of tracks produced by a main artist and a small artist. The number of tracks per artist in the dataset goes up to 153, and about 75% of artists have fewer than 17 tracks.

We then wondered if the popularity of a track depends on whether the artist is a main artist or not. To explore this, we used the *Chi-Square Test of Independence* covered in SY02. With the p-value being very close to 0, we can reject the null hypothesis: it seems that these two variables are not independent. This is particularly noticeable for very popular tracks (>80/100), as indicated by the expected frequency table from this test. Small artists actually have only 107 very popular tracks, while the expected number was around 707, and main artists have 1,233 very popular tracks, though the expected result was 633. We repeated the test, replacing the popularity variable with genres, but we did not observe similarly interesting results.

We also wondered whether the main artists had a preferred music genre. The 864 main artists have an average of 2.25 genres, and one-third of them have only one. Thus, we are curious whether taking into account the artist and the style of their repertoire might help in predicting the genre of a track.

We also expected to find differences in the *trackAlbumReleaseMonth* variable depending on the genres, such as more Latin music during the summer, but there does not seem to be any notable impact.

Next, we wanted to study the *playlistName* variable with the goal of identifying the genre of a track based on the name of its playlist. This helped us understand that the *playlistGenre* variable, as the name suggests, refers to the genre of the playlist the track belongs to, and not necessarily the genre of the track itself. We then noticed that some of our tracks belong to multiple

playlist genres at the same time. We were probably influenced by the title "Song genres" or by reading the article mentioned on the tidyTuesday GitLab, which does not clarify this small ambiguity [2].

2.4 Conclusion of the Exploratory Analysis

This part has helped us define our lines of analysis. We will first study the genre of a track using the *playlist_genre* variable, which gives the genre of the playlist in which the track is found. Then, in a second step, we will attempt to predict the genre of a playlist based on the tracks that make it up.

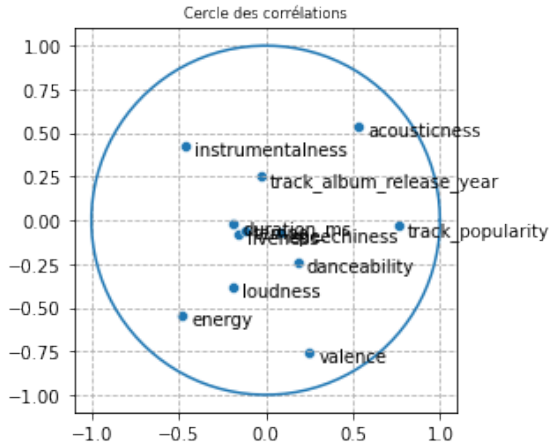

Figure 2: Correlation circle - 1st factorial plane

3 Playlist Genre of a Track

It is important to note that in this section, when we refer to the genre of a track, we are actually referring to the genre of the playlist in which the track is found.

First, we will present the application of unsupervised methods on our dataset. The direction of our research does not really lend itself to these methods, but we still wanted to apply them to try to learn more about our dataset. In the second part, we will study supervised methods.

3.1 Unsupervised Methods

3.1.1 Principal Component Analysis (PCA)

PCA on our dataset would allow us to reduce the number of quantitative variables into 6 main axes of inertia, representing just over 80% of the information. The representation of our data according to these axes did not allow us to differentiate tracks by their genre. However, it helped to better understand the correlation between our variables. The correlation circle based on axes 1 and 2, which retains almost 40% of the information, is shown in figure 7. The main observations we can make in this first plane are that *trackPopularity* contributes greatly to the first axis, *acousticness* is somewhat negatively correlated with *valence* and *energy*, and *instrumentalness* and *energy* are also quite negatively correlated. This seems consistent with what was observed during the exploratory analysis.

3.1.2 Clustering Algorithm

Given the large volume of data, hierarchical clustering (CAH) does not seem well-suited for our study. Therefore, the K-MEANS algorithm was preferred. The number of clusters chosen was 6, to try to highlight the 6 music genres. Unfortunately, we did not observe a clear separation. However, some data still showed differences between genres. For example, one of the results returned by KMeans is that 57% of the tracks in cluster 1 are Edm, 57% of those in cluster 3 are Rock, and this same group contains almost 0% Edm tracks. This aligns with our initial observations, suggesting that Edm and Rock seemed to be easier to separate from the other genres.

3.2 Supervised Methods

In this section, the chosen labels for the different methods will be our 6 music genres. We remind you that our dataset has a fairly balanced distribution across the genres. Since our goal is to predict genres, we will use classification methods.

3.2.1 K-Nearest Neighbors Method (KNN)

This method allows us to obtain a score of 0.51, which might seem like a rather average result. It's important to take into account Scikit's warning that reminds us that in the multiclass case, the score can sometimes be "harsh." In fact, this means that we move from having about a 1 in 6 chance of guessing a track's genre to a 1 in 2 chance. The score obtained here will serve as our reference score, which we will try to improve further.

To achieve a more conclusive result, we reduced the number of labels to 2, comparing two music genres at a time. A comparison with Rock, for example, gives very good results (up to 0.91), as well as with Edm and Rap,

as we had expected. Across all two-genre comparisons, we find an average score above 0.80, which is quite satisfying. We also tried a comparison of three genres (Edm, Rap, and Rock), which gave a still-satisfactory average score of 0.81. When adding Latin in a comparison of four genres, we obtained a score of 0.7.

3.2.2 Other Classifiers: LDA, QDA, Naive Bayes, 1-VS-1, 1-VS-REST

Next, we tried other classifiers in hopes of improving the score obtained with KNN.

First, we wanted to check whether our classes followed a multidimensional normal distribution. By performing the Shapiro-Wilk test on each class, the null hypothesis was rejected. Therefore, it is highly unlikely that these data follow a normal distribution. After some research, it appears that having a large dataset still allows for conclusive results.

We repeated the use of the various classifiers about ten times. They gave results very similar to those obtained with KNN, with a score close to 0.5 on the 6 genres and similar average scores when comparing two genres.

3.2.3 Decision Tree and Random Forest

We then applied the binary decision tree method to our dataset. The average score obtained with this method for classifying the 6 genres is not very good: 0.45 when constructing the entire tree and 0.4 when limiting it to the 6 nodes shown in figure 7. Additionally, the impurity criterion for each class in this figure is high, with a Gini coefficient between 0.5 and 0.8. So, although the result is far from satisfactory, the main advantage of this method is that it highlights the variables that are most important for classification. Figure 7 shows the simplest possible tree, with each genre being the majority in a group. It should be noted that we had to force such a structure using the *max_ depth* and *min_ samples_ leaf* arguments of sklearn's DecisionTreeClassifier, and this figure is based on the dtreeviz package [1].

We observe that the most important variable for determining a track's genre is its release year, with a *feature_ importance_* of 0.36. We note that some genres are easier to characterize than others: Rock by an older release year and low danceability, Rap by high speechiness, and Edm by a high tempo. The other three music genres seem more difficult to classify since they are more evenly distributed among the various classes.

To make the most of binary trees and aim for the best

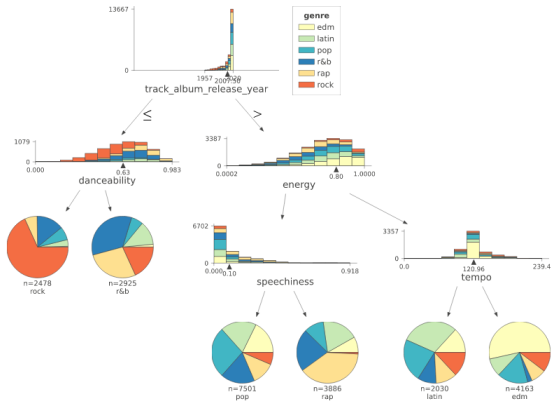

Figure 3: Decision tree

possible score, we applied the Random Forest method to our dataset. Using this method, we obtained a score of 0.57 for classifying the 6 genres, which represents the best score obtained on the dataset containing all 32,000 tracks.

3.2.4 Study Without Ambiguous Tracks

There are tracks that appear multiple times in different playlists. For some of them, all the playlists are of the same genre. For others, the playlists belong to different genres (for the same track). We will consider that if the same track appears in playlists of different genres, then the genre of that track is ambiguous. We built a new dataset by removing all tracks with an ambiguous genre. By applying the various supervised methods to the resulting 23,000 data points, we observed a significant improvement in the score for each method. For the classification of the 6 genres, the improvement ranged from 5% for KNN, with a score of 0.56, to 7% for Random Forest, with a score of 0.64.

3.2.5 Study on Main Artists

As mentioned in the exploratory analysis, tracks by the same artist tend to belong to 2 or 3 playlist genres on average. We tried to use this information in our supervised methods. We created a new dataset by keeping only the 13,000 tracks belonging to main artists (artists with more than 6 tracks in the dataset). By applying the various classifiers without the information on the artist's genre repertoire, we found an average score of 0.55. Taking this information into account raised the score to 0.77. To achieve this, we first split the dataset into two-thirds training data and one-third test data. Then we built a frequency table of genres by artist us-

ing the training data. Finally, we added the 6 frequency columns to the training and test data.

4 Analysis of Playlist Genres

We followed the same reasoning as in the previous section. To analyze the genre of a playlist based on the tracks it contains, we averaged the values for each variable across all tracks in the playlist. We also attempted to take variance into account, but this did not yield conclusive results. We then focused solely on the average per variable. This reduced our data to 449 entries, each corresponding to a playlist.

4.1 Unsupervised Methods

4.1.1 Principal Component Analysis (PCA)

PCA allows us to summarize 80% of the information using 5 main axes of inertia. The first plane represents only 50% of the information. Nonetheless, we can see a slight distinction according to genres in this first plane (see figure **??**).

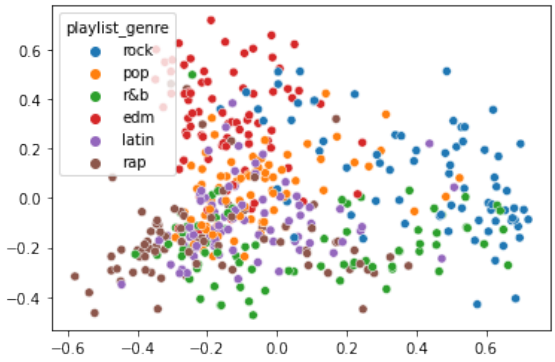

Figure 4: PCA Representation in the first plane

4.1.2 Clustering Algorithm

On average, applying K-means gives us a Rand index score of 0.26, which is not very good. Upon closer inspection, however, some results seem rather satisfactory. For example, one result returned by KMeans shows that 70% of *latin* playlists were classified in cluster 0, 70% of *rock* playlists in cluster 3, and even 80% of *edm* playlists in cluster 5. We expected a similar result for *rap*, but the algorithm mainly classified playlists of this genre into two distinct clusters (50% in cluster 4 and 35% in cluster 0). Some genres never appear in certain clusters.

4.2 Supervised Methods

4.2.1 Classification

We recorded the scores of the methods using various classifiers 50 times (see figure 7). The score tends toward 0.70.

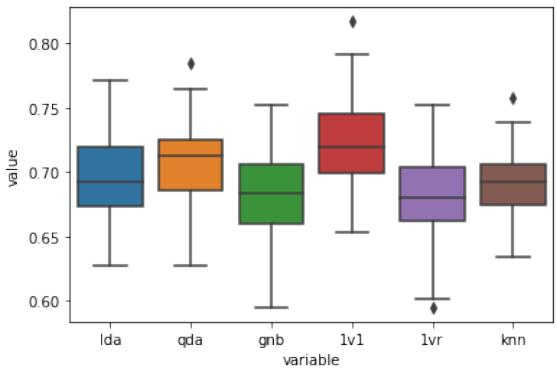

Figure 5: Comparison of different classifiers

We also generated the corresponding confusion matrices. Figure 7 shows an example of one of the best matrices obtained for the One VS One classifier. Given the high scores on the diagonal and values close to zero elsewhere, we are very satisfied with the results.

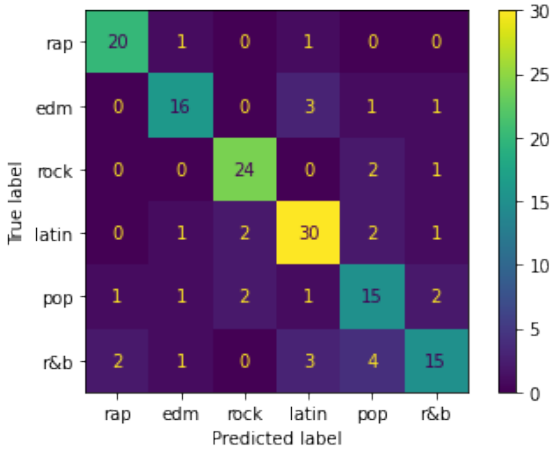

Figure 6: Confusion Matrix - One VS One

4.2.2 Decision Tree on Playlists

Finally, we applied decision tree methods in the same way as in section 3.2.3. Once again, the average score was lower than that of the previous classifiers: 0.65 for the 6 genres using the entire tree and 0.6 when limited to the first 6 nodes (see figure 7). However, these scores are significantly better than those obtained with the track dataset, as indicated by the impurity criterion, with Gini coefficients between 0.3 and 0.6 for the various classes. We also observe that some classes mainly contain a single genre, particularly for Rock (80%), Latin (78%), and Rap (76%). It is worth noting that this tree was easier to build than the previous one since it was naturally constructed by the algorithm when keeping only the first 6 nodes.

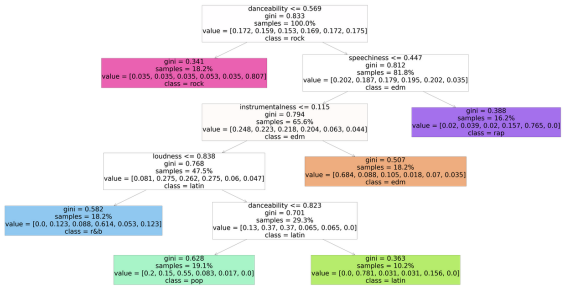

Figure 7: Decision Tree on Playlists

It is also interesting to note that different variables are used when applying this method to playlists versus tracks. Here, the release year, tempo, and energy are replaced by *loudness* and *instrumentalness*. We can summarize the main characteristics of playlists for each genre in table **??**.

Table 1: Main Characteristics of Each Genre

Genre	Main Characteristic
Rock	Difficult to dance to
Rap	Many spoken words
Edm	Very instrumental
R&B	Low sound volume
Latin	Very easy to dance to
Pop	Everything else

Finally, we applied the Random Forest method, which gave us a score of 0.74. This method is also the most effective for this dataset, as the score is slightly higher than those obtained with other classifiers.

5 Conclusion

There is no binary answer to the prediction questions we posed at the beginning, but we can say that the various methods used consistently yield better results than random predictions. Some characteristics discovered during exploratory analysis helped us make predictions on smaller datasets using specific criteria. With these modifications, we achieved excellent prediction scores, especially considering we aimed to classify data into 6 labels.

This dataset was very interesting to analyze, and we believe we have learned a lot from it. There are certainly many more insights to be gained. Another approach would be to explore whether we can predict the popularity of a track.

References

[1] parrt. dtreeviz : Decision tree visualization. Technical report.

[2] K. Pavlik. Understanding + classifying genres using spotify audio features. Technical report.