Project report on popularity prediction of songs in Spotify and Youtube

Omar Chafik, Santiago Lema

July 5, 2023

1 Introduction

Throughout history, music has held a profound place in human society, transcending cultural boundaries and serving as a fundamental means of expression and engagement. From ancient tribal rituals to modern-day streaming platforms, music has always captivated our hearts and souls. It has the unique ability to evoke emotions, connect people, and create shared experiences.

In today's digital age, the music industry has witnessed (and continues to witness) a deep transformation, where engagement and popularity on digital platforms play a crucial role in an artist's success. With millions of songs available on these platforms, understanding the factors that drive popularity and listeners' willingness to engage with a song is of great interest to artists, record labels, and other stakeholders in the music industry.

This project aims to delve into the realm of music popularity prediction, focusing on songs available on Spotify and YouTube, analyzing the relation between music features and users preferences. The goal will be pursued by developing a classification model capable of predicting the popularity segment of a song, based on selected features. This entails constructing a comprehensive dataset containing various features and attributes of songs from the mentioned platforms. Leveraging such machine learning techniques, we will explore and compare multiple classification models to identify the most effective approach. Furthermore, we will fine-tune the selected models to extract the best possible results.

The motivation for this problem statement is to understand better what makes a song successful on different platforms and to potentially use this information to inform marketing strategies for artists and record labels.

Also, this is an exciting opportunity to merge music, data science and industry insights to gain a better understanding of the factors which give life to this ancient form of art.

The ground set of data for this project was obtained from a pre-built dataset available in Kaggle called *Spotify and Youtube*, authored by Salvatore Rastelli[1]. The data was collected in February of 2023 so it is relatively recent to the production of this document.

However, we are aware that new content was released in the meantime so we considered using more recent data for testing the model.

2 Data

Diving deeper into the dataset, it contains 20.718 songs with the following features:

Attribute	**Unique values**	**Description**
Artist	2079	Name of the artist
Url_spotify	2079	Spotify URL for the song
Track	17841	Name of the track
Album	11937	Name of the album
Album_type	3	Type of the album
Uri	18862	URI of the song
Danceability	898	Measure of the song's danceability
Energy	1268	Measure of the song's energy
Key	12	Key of the song
Loudness	9417	Loudness level of the song
Speechiness	1303	Measure of the song's speechiness
Acousticness	3138	Measure of the song's acousticness
Instrumentalness	4012	Measure of the song's instrumentalness
Liveness	1536	Measure of the song's liveness
Valence	1293	Measure of the song's valence
Tempo	15024	Tempo of the song
Duration_ms	14690	Duration of the song in milliseconds
Url_youtube	18154	YouTube URL for the song
Title	18146	Title of the YouTube video
Channel	6714	YouTube channel of the uploader
Views	19245	Number of views on YouTube
Likes	17939	Number of likes on YouTube
Comments	10485	Number of comments on YouTube
Description	17395	Description of the YouTube video
Licensed	2	Indicates if the song is licensed
official_video	2	Indicates if it's an official video
Stream	18461	Indicates if the song is available for streaming

Table 1: Attributes of the dataset

The features of highest interest for the purpose of this project are the numerical ones, but we will also use some of the categorical. Some of these features refer to specific terminology used in the music industry to describe characteristics of a song. The following list, took from Spotify API's Documentation [2] provides a brief explanation of these terms:

- *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. A value of 0.0 is least danceable and 1.0 is most danceable.

- *Energy*: is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.

- *Key*: the key the track is in. Integers map to pitches using standard Pitch Class notation. E.g. 0 = C, 1 = C# , 2 = D, and so on. If no key was detected, the value is -1.

- *Loudness*: the overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track and are useful for comparing relative loudness of tracks. Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude). Values typically range between -60 and 0 db.

- *Speechiness*: detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.

- *Acousticness*: a confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.

- *Instrumentalness*: predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.

- *Liveness*: detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.

- *Valence*: a measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).

- *Tempo*: the overall estimated tempo of a track in beats per minute (BPM). In musical terminology, tempo is the speed or pace of a given piece and derives directly from the average beat duration.

The histograms observed in Figure 1 show the distribution of the main numerical attributes in the dataset will help to get a better understanding of the initial state of the data.

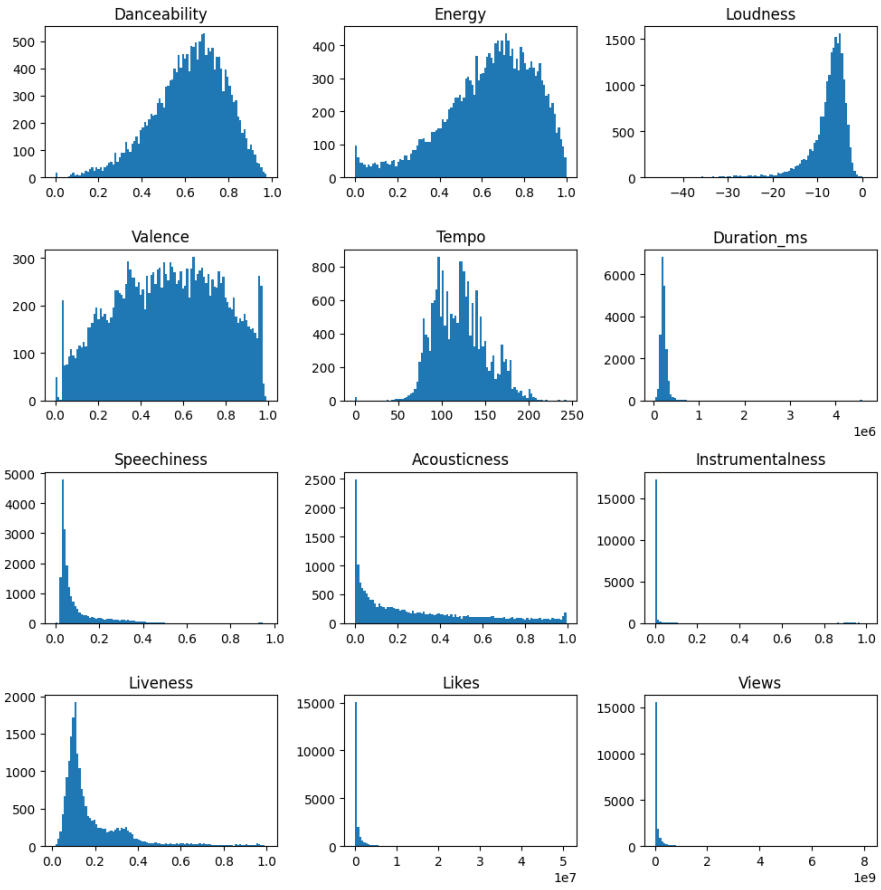

Figure 1: Histograms of main numerical attributes

As it can be seen, there is no specific target variable in the dataset. Earlier, when the goal was stated (predicting the popularity of a song), the actual definition of popularity was not mentioned. As a matter of fact, it is not easy to define the popularity of a song in a particular digital platform. Because of this subjectivity of the concept which is trying to be predicted, we will mention a series of definitions that were considered to eventually justify a more objective definition of the target variable *popularity*.

In the English language, according to the Cambridge Dictionary [3], popularity is "the fact that something or someone is liked, enjoyed, or supported by many people".

In the Collins Dictionary [4], the definition that can be found is that "something that is popular is enjoyed or liked by a lot of people".

In a sociologic context, paraphrasing Wikipedia [5], popularity, derived from the Latin term *popularis* meaning "common," has evolved to signify the "fact or condition of being well liked by the people". Although popularity is often attributed to individuals, it is fundamentally a social phenomenon that can only be comprehended within the context of groups. Popularity is a collective perception, and individuals gauge the consensus of a group's sentiments when assessing popularity of an individual or object. The degree of advocacy and positive opinions expressed by a group directly influences the attention and level of popularity something or someone attains.

However, popularity extends beyond individuals and can be attributed to various objects such as songs, movies, websites, activities, soaps, foods, and more. These objects collectively form popular culture, representing the prevailing preferences within society. Essentially, anything, whether human or non-human, has the potential to be regarded as popular.

Given these definitions, there are more basis to argue that popularity can exist as an attribute of a song if it helps measuring the degree of advocacy and positive opinions expressed by a group of people. In this case, the group of people will be the users of the digital platforms Spotify and YouTube, and the appreciation will be measured by the number of likes and views. In order to detail this process, first we can analyze these two variables further as they became now more relevant.

In the previous histograms it could be noticed a peculiar distribution for the likes and views. They can be better appreciated in the following rug plots.

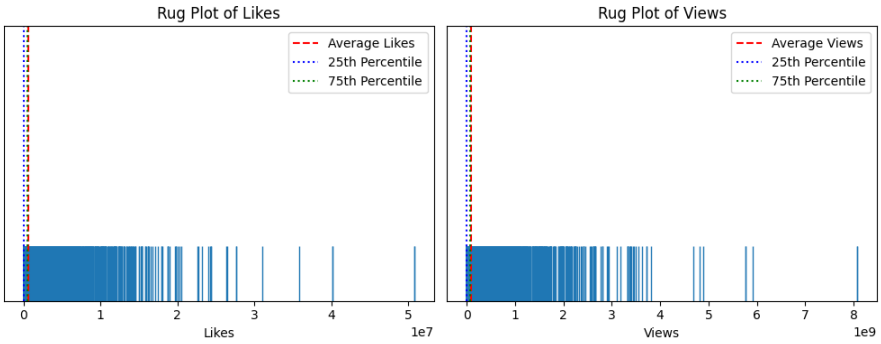

Figure 2: Distribution of likes and views

It is undoubtedly a very skewed distribution, with a very high concentration of values in the lower part of the range. This is a very common behavior in social media platforms, where the majority of the content is not popular and only a few pieces of content become viral.

This is also known as the *long tail* phenomenon, which is a characteristic of power law distributions [6].

If we take a look at the statistics of these two variables, we can see that the mean is

much higher than the median, which is a clear indicator of the skewness of the distribution, and also the last quartile is distributed in a much broader range of values.

Attribute	Likes	Views
Count	20177.0	20177.0
Mean	6.633×10^5	9.418×10^7
Std	1.789×10^6	2.750×10^8
Min	0.0	0.0
25%	21581.0	1835998.0
50%	124481.0	14542507.0
75%	522148.0	70572997.0
Max	5.078×10^7	8.079×10^9

Table 2: Basic Statistics for Likes and Views

To assess the popularity of songs given these attributes, a third attribute called "Popularity score" was developed. This score is calculated by weighting the likes and views. Comments were excluded based on the understanding that, according to established definitions of popularity, only positive comments would contribute to the popularity score. Conducting sentiment analysis on the comments was beyond the scope of this project and not feasible due to data limitations.

The weights initially assigned to likes and views were 0.7 and 0.3, respectively. This choice of weights was subjective and based on the interpretations of the definitions mentioned earlier. According to these definitions, popularity is primarily associated with liking or enjoying something, rather than the sheer quantity of views or spread of the content.

It is important to note that these weightings were determined based on a subjective decision and may vary depending on the context or specific goals of future analyses.

This score served as the basis for creating the target variable "popularity." The target variable was transformed into a categorical value that defines distinct zones or neighborhoods of popularity. Initially, the popularity variable was categorized into four values: "Low," "Moderate," "High," and "Very high." These categories corresponded to specific thresholds based on the distribution of the popularity score. The thresholds were set as follows: the lowest 30% of popularity scores fell into the "Low" category, popularity scores between 30

This approach allowed for the creation of discrete popularity levels, enabling a more nuanced analysis of the songs' popularity within distinct tiers. It should be noted that these threshold values and the number of categories were determined based on the specific distribution of the popularity score in the dataset. Alternative categorizations could be explored depending on the specific requirements of future analyses or applications.

Popularity category	Values count
Low	6053
Moderate	8071
High	4035
Very high	2018

Table 3: Popularity count by category

To ensure a focused and effective data preprocessing phase, certain attributes were removed from the dataset. Initially, categorical attributes that were not planned to be used in the classification model were eliminated. These included the artist name, URLs, URIs, track numbers, and album names. As these attributes did not contribute directly to the classification task, their removal helped streamline the dataset and reduce unnecessary complexity.

Furthermore, the attributes related to likes, comments, and views were also excluded from the dataset. These attributes were expected to exhibit high correlation with the popularity score, as they are potential indicators or direct factors contributing to the determination of popularity. By removing these attributes, we aimed to avoid issues of multicollinearity and prevent the classification model from relying too heavily on features that directly represent the target variable.

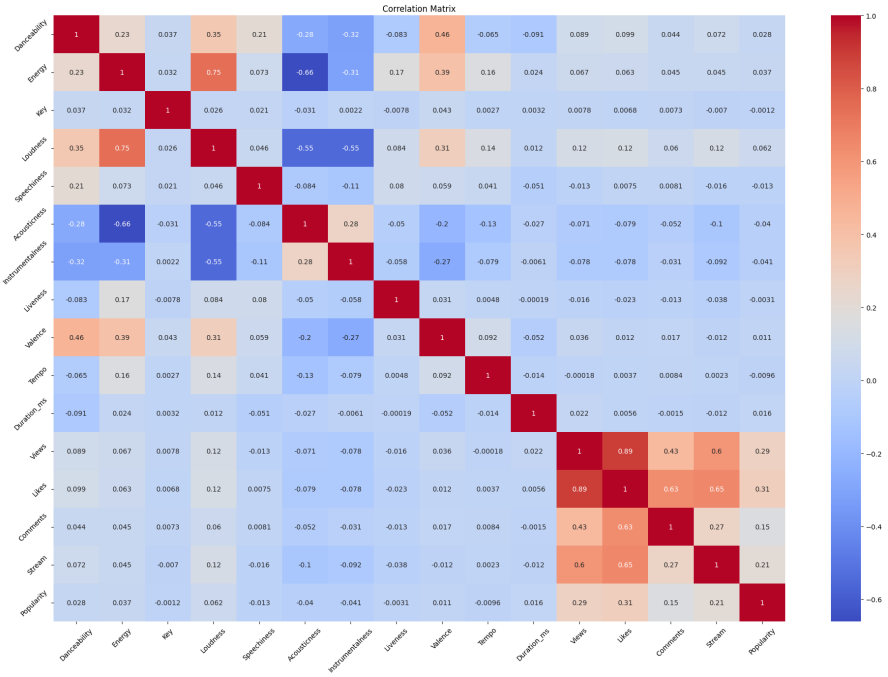

Figure 3: Correlation matrix

This preprocessing step allowed us to create a more refined dataset, focusing on the most relevant attributes for the classification model. By eliminating unnecessary categorical attributes and features highly correlated with the target variable, we aimed to enhance the model's performance and interpretability.

3 Method

In tackling the classification problem, we carefully selected several models to implement and subsequently compare their performance. The chosen models for this task include Support Vector Machine (SVM), Random Forest, and K-Nearest Neighbors (KNN).

Each of these models offers distinct advantages that make them suitable for our problem:

- Support Vector Machine (SVM): SVM is a powerful and versatile classification algorithm known for its ability to handle complex datasets and find optimal decision boundaries. It is particularly effective when dealing with high-dimensional data and can handle both linear and non-linear classification problems. SVM aims to maximize the margin between different classes, thereby promoting robustness and generalizability in the classification process.

- Random Forest: Random Forest is an ensemble learning method that combines multiple decision trees to make predictions. It excels in handling large and diverse datasets while avoiding overfitting. By using bootstrap sampling and feature randomization, Random Forest mitigates the risk of variance and bias, providing robust and accurate predictions. It is particularly useful when dealing with complex relationships and feature interactions.

- K-Nearest Neighbors (KNN): KNN is a non-parametric algorithm that classifies data points based on their proximity to other labeled instances. It is an intuitive and easy-to-understand method that requires no training phase. KNN works well with datasets where instances with similar features tend to have the same label. It is also effective when the decision boundaries are irregular or nonlinear. KNN allows for flexibility in defining the value of K, the number of nearest neighbors considered during classification.

After running all these, we will compare the results and choose the best model to fine-tune it and get the best possible results. However, before running the models, additional preprocessing steps were performed. For categorical variables, encoding techniques were applied. The target variable was encoded using label encoding, while the remaining categorical variables were encoded using one-hot encoding.

Label encoding was chosen for the target variable as it is a categorical variable with ordinal levels. Label encoding assigns a unique numerical label to each category, thereby preserving the ordinal relationship between the categories. This encoding allows the classification models to interpret the target variable correctly during training and prediction.

On the other hand, one-hot encoding was used for the remaining categorical variables. One-hot encoding creates binary dummy variables for each category, representing the presence or absence of that category in a given observation. This encoding scheme is suitable for non-ordinal categorical variables as it prevents the models from assigning any arbitrary ordinal relationship among the categories.

Regarding the numerical values, a normalization technique was applied specifically for the KNN model. Normalization ensures that all numerical features have a similar scale, which is crucial for distance-based algorithms like KNN. By scaling the numerical values to a standard range (e.g., between 0 and 1), the KNN algorithm can give equal importance to each feature during classification, preventing features with larger scales from dominating the distance calculations.

Overall, by encoding the categorical variables and applying normalization to the numerical values, the data was prepared appropriately for the selected models. These preprocessing steps aim to enhance the models' performance and ensure they can effectively learn from the data, leading to more accurate predictions of song popularity.

Almost ready to go through the models, the dataset was split into an 80/20 ratio, with 80% of the data used for training the models and 20% held out for testing purposes. This serves to evaluate the model's performance on unseen data. The selected ratio 80/20 is meant to strike a balance between having sufficient data for training the models and having a reasonable amount of data for evaluation. With 80% of the data allocated for training, the models can learn patterns and relationships from a substantial portion of the dataset, allowing them to capture the underlying trends in the data.

The remaining 20% of the data, reserved for testing, serves as an independent sample to evaluate the models' performance. By assessing the models on unseen data, we can measure their ability to generalize and make accurate predictions on new instances. This helps us estimate how well the models will perform when applied to real-world scenarios or new data points.

4 Results and conclusion

References

[1] Rasetri, S. (2020). Spotify and YouTube. Retrieved from https://www.kaggle.com/datasets/salvatorerastelli/spotify-and-youtube (Accessed: July 5, 2023).

[2] Spotify Developer API. Retrieved from https://developer.spotify.com/documentation/web-api/reference/get-audio-features (Accessed: July 5, 2023).

[3] Cambridge Dictionary. Retrieved from https://dictionary.cambridge.org/dictionary/english/popularity (Accessed: July 5, 2023).

[4] Collins Dictionary. Retrieved from https://www.collinsdictionary.com/dictionary/english/popularity (Accessed: July 5, 2023).

[5] Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Popularity (Accessed: July 5, 2023).

[6] Wikipedia. Retrieved from https://en.wikipedia.org/wiki/Long_tail (Accessed: July 5, 2023).