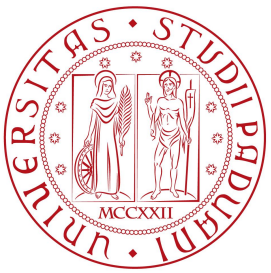
University of Padova
Department of Mathematics

Master Degree in Data Science



Song genre classification using Spotify and Genius data

Marco Ballarini, ID: 2096997
Laura Legrottaglie, ID: 2073222
Pietro Renna, ID: 2089068

Statistical methods for high dimensional data

Accademic Year 2023/2024

1 Introduction

The following work focuses on the classification of song genres using data from Spotify and Genius. The objective is to develop a model that can accurately categorize songs into specific genres (rock, pop, and rap). This model could be highly beneficial in enhancing and automating the process of labeling songs in playlists on digital music platforms.
The dataset has been collected using songs having english lyrics from several Spotify playlists of different years. Specifically, audio feature predictors have been obtained from Spotify API in Table 5 and lyrics have been collected through Genius API.
The incorporation of text data into this analysis raises two critical questions: firstly, the extent to which lyrics and audio features play a significant role in predicting a song's genre, and secondly, if specific words and themes persistently recur more to a particular genre than in others. In addition to these interpretive objectives, the last aim is to evaluate and compare the predictive capabilities of different models, focusing on their efficacy in classifying songs into genres.

2 Preprocessing

Since lyrics scraped from Genius are in an unstructured format, a set of preprocessing operations has been performed on raw data. Stopwords, special characters, numbers and some artifacts (such as "Lyrics", "Chorus", "Verse") have been removed. Then, both lemmatization and stemming have been performed on cleaned lyrics text. Furthermore, an audio description / commentary was retrieved instead of the lyrics for some songs and were, thus, removed manually.
The original dataset contains 2285 observations, successively splitted into train and test set (80% train and 20% test). A Document-Term Matrix, based on unigrams from the training set of lyrics, is constructed as foundational step in creating the TF-IDF matrix. To avoid data leakage, the TF-IDF matrix for the test set is created using just the words that appears in the train set. The resulting train set is composed by the words in the TF-IDF matrix along with the columns derived from Table 5.
Statistically speaking, the task is a high dimensional multiclass classification problem with 13554 predictors.

3 Models

3.1 Shrinkage methods

Several models from the elastic-net family ($\alpha = 0, 0.25, 0.5, 0.75, 1$) have been fitted and tuned using 10-fold Cross-Validation with user-supplied folds: the best α and the best regularization parameter λ are selected on each model using the exact same folds between all shrinkage models.
It is worth mentioning that all the models except the Ridge ($\alpha = 0$) have similar performances using multinomial deviance as measure. Since the overall accuracy is

a crucial metric to consider when choosing the best model in terms of prediction, a cross-validation is also performed using the misclassification error as benchmark. The results are almost the same as the CV considering multinomial deviance, so from now on the considered models are the one with multinomial deviance. The best resulting model is the Lasso ($\alpha = 1$, $\lambda_{min} = 0.023$ and $\lambda_{1se} = 0.030$).
In terms of coefficients interpretation, both the ones derived from the model with λ_{min} and the ones from λ_{1se} may be considered for prediction and interpretation purposes. However, the λ_{1se} solution is chosen since it has slightly higher performances with respect to test set data and gives more interpretability along with less complexity by selecting fewer predictors. Looking at Figure 1, it can be noticed that words coming from specific semantic areas influence the probability of a song being classified as pop. For instance, romantic terms like "love", "babi", "beauti" and "heart", in addition to all the words related to the dancing area such as "dancin" and "night", positively affect the probability of a song being pop. Conversely, words like "smoke" and "fuck" affect negatively this probability. The same analysis is conducted on the rap genre (Figure 2): in particular, words that come from drug/smoke/drink semantic area like "codein", "drink" and "smoke", multiple cursed words and slang terms like "homie", "skrrt" and "lil" are identified as significant by the model. Unlike the previous cases, regarding the rock genre (Figure 3), the model shrinks most of the terms to zero and it seems that lyrics information does not affect the probability of a song being rock: even if some words are found, there is no common context between them and they are not easily interpretable, leading to audio features being more relevant.
Regarding audio features, results show that for the pop class the *loudness* and *acousticness* increase the probability of a song to be pop, while the *key 2* (pitch D) and the *duration* (even though its coefficient is near zero) reduce this probability. For rap genre two important audio features are included among the relevant ones: *danceability* and *speechiness* are extremely positive related with this genre, instead *valence* is negative. Finally, rock songs are related to a positive *valence*, *energy* and *intrumentalness*. Additionally, there is a higher probability of a song being categorized as rock if it is in a major key, particularly with an E pitch class. Beyond these considerations, a rock song is negatively related to *danceability*, *loudness* and *acousticness* and so rock genre is the one with the largest number of significant audio features.

3.1.1 Graphical Lasso

An interesting perspective arises by looking at the dependency structure inside the words selected from the Lasso model: this is done in order to assess whether there are some words in a song, for a particular genre, conditioned to the presence of other words. To detect this type of dependency structure, a Graphical Lasso on words selected by Lasso has been performed: this choice is due to the fact that fitting the model on the whole dataset would be unfeasible from a computational point of view and because it is interesting to see if there are some hidden dependencies between words that Lasso model considers important.
Training the model on the coefficients selected by the λ_{1se} results in a poor structure that simply shows a subset of different fully connected coefficients. A better

structure is obtained when performing the model on the coefficients selected by the λ_{min}: it keeps only specific connections between some words.
Using coefficients selected with λ_{1se} results in a dependence fully connected structure only between "baby", "heart" and "love", so it seems that the presence of the word "heart" is related with the presence of words "baby" and "love", and viceversa.
Regarding rap coefficients, the model selects just "nigga", "lil", "fuck", "shit" and "bitch" and like before they are all connected to each other.
Another possibility is to consider words selected using λ_{min}: the results for pop genre is a bit richer; "baby", "heart", "love" and "deep" are all connected to each other but not with other words; "ignit" and "something" results connected in two pairs, just like "night" and "wide", "closer" and "beauty" and "shame" and "trouble".
Rap genre has a situation similar to the one obtained with λ_{1se}, having few more cursed word and a slang word, but the resulting graph is still fully connected.
For the rock genre, the resulting estimate of covariance matrix from Graphical Lasso is a diagonal matrix, so it is impossible to draw a network structure since all the words are connected only on their own: this is probably caused by Lasso selection, which has found out only few words as significant coefficients for rock and furthermore these are not related on each other.
The choice of the regularization hyperparameter is done according to the path provided by *gglassopath* function in R, and choosing the only one that returns a covariance matrix with elements also outside the diagonal.

3.2 XGBoost

XGBoost can be used in terms of both prediction performances and interpretability (with feature importance metrics).
There are several hyperparameters that could be tuned; after several attempts, it turns out that hyperparameters affecting the model performances the most are the *max depth* and the *learning rate*. A 10-fold Cross Validation with different values of these two hyperparameters has been performed, selecting the value of 12 for *max depth* and the value of 0.2 *learning rate*; other combinations of hyperparameters have been considered (α and λ regularization hyperparameters) but in the end it seems that they do not affect the performances of the model in a relevant way.
It is also interesting to take a look at the feature importance that XGBoost selects: both words and audio features are considered important to split the tree, and thus to discriminate the genre; many words that Lasso selects in the final model appear also here (like "love", "nigga", "fuck" etc.).

3.3 Neural Network

Neural networks are a powerful model that can be used to classify songs in combination with word embedding. In fact, using TF-IDF matrices as input for a neural network results in poor performances due to the sparsity and high dimensionality of the matrices which makes it difficult and computational inefficient for the model to establish meaningful patterns, as similar data points can appear far apart in high-dimensional space (curse of dimensionality). Therefore, a more reasonable and clever approach is word embedding, where each word is represented as a vector. Conversely

to TF-IDF matrices, word embedding captures semantic relationships between words (i.e, words that have similar context will have similar vector representation), providing at the same time a dense and much lower dimensional representation of the data.
Word2Vec is a 2-layer neural network that can be used to generate word embedding given a text corpus. The specific chosen architecture is Continuous Bag of Words (CBOW), where the model predicts a target word from a context window (5 words are chosen in this case). The weights connecting the input layer to a specific layer of the 2-layer neural network, called *embedding layer*, after been fitted on the training corpus, can be used to create a 100-dimensional vector for each word. To obtain a fixed-length vector for each lyric irrespective to its original length, then, the mean of all the words' vectors in the preprocessed song is computed. The final input of the NN model is the combination of the embeddings of the lyrics and the audio features. Cross Validation has been performed to select the best model that results in a 2 hidden layers neural network with 10 neurons each with *learning rate=0.001* and ADAM as optimizer. It should be highlighted that this model is not suitable for interpretability purposes, as it does not provide information about the predictors influence. Consequently, its application will be confined to the final prediction comparisons.

4 Conclusions

The analysis have investigated the influence of lyrics terms in distinguishing the song genres. The models giving an interpretation of the coefficients (Lasso in Section 3.1 and XGBoost in Section 3.2) have shown reasonable results for pop and rap, highlighting specific words coherent and related for the two genres. On the other hand, it seems that both models are not able to discriminate rock genre using words. Furthermore, the influence of numerical features has been discussed for each genre.
In order to evaluate prediction ability, metrics that will be considered in order to evaluate models in terms of performances are: overall accuracy, sensitivity, specificity, F1 score and precision. The reason of using these metrics is to have a broader overview of models predicting behaviour.
The model with the highest Accuracy is XGBoost, even though Lasso and Neural Networks have almost the same value as can be seen in Table 1. Instead, considering the classification metrics, each model has a specific behavior depending on genre as illustrated in Table 2, Table 3 and Table 4.

Table 1: Accuracy of Models

Model	**Accuracy**
Lasso (λ_{1se})	0.709
Neural Network	0.779
XGBoost	0.781

Table 2: Metrics for Class: pop

Model	**Sensitivity**	**Specificity**	**Precision**	**F1 Score**
Lasso (λ_{1se})	0.7754	0.7116	0.5377	0.6350
Neural Network	0.7391	0.8150	0.6335	0.6823
XGBoost	0.8116	0.7837	0.6188	0.7022

Table 3: Metrics for Class: rap

Model	**Sensitivity**	**Specificity**	**Precision**	**F1 Score**
Lasso (λ_{1se})	0.8529	0.9252	0.8286	0.8406
Neural Network	0.8015	0.9533	0.8790	0.8385
XGBoost	0.6397	1.0000	1.0000	0.7803

Table 4: Metrics for Class: rock

Model	**Sensitivity**	**Specificity**	**Precision**	**F1 Score**
Lasso (λ_{1se})	0.5519	0.9380	0.8559	0.6711
Neural Network	0.7923	0.9015	0.8430	0.8169
XGBoost	0.8634	0.8869	0.8360	0.8495

Table 5: Description of Spotify Features

Feature	**Type**	**Description**
acousticness	number [float]	Confidence measure of a track being acoustic.
danceability	number [float]	Describes how suitable a track is for dancing.
duration_ms	integer	The duration of the track in milliseconds.
energy	number [float]	Represents a perceptual measure of intensity and activity.
instrumentalness	number [float]	Predicts whether a track contains no vocals.
key	factor	The key the track is in using standard Pitch Class notation.
liveness	number [float]	Detects the presence of an audience in the recording.
loudness	number [float]	The overall loudness of a track in decibels (dB).
mode	factor	Indicates the modality (major or minor) of a track.
speechiness	number [float]	Detects the presence of spoken words in a track.
tempo	number [float]	The overall estimated tempo of a track in beats per minute (BPM).
valence	number [float]	Measure describing the musical positiveness conveyed by a track.
genre	factor	Genre of the song (pop, rock, rap)

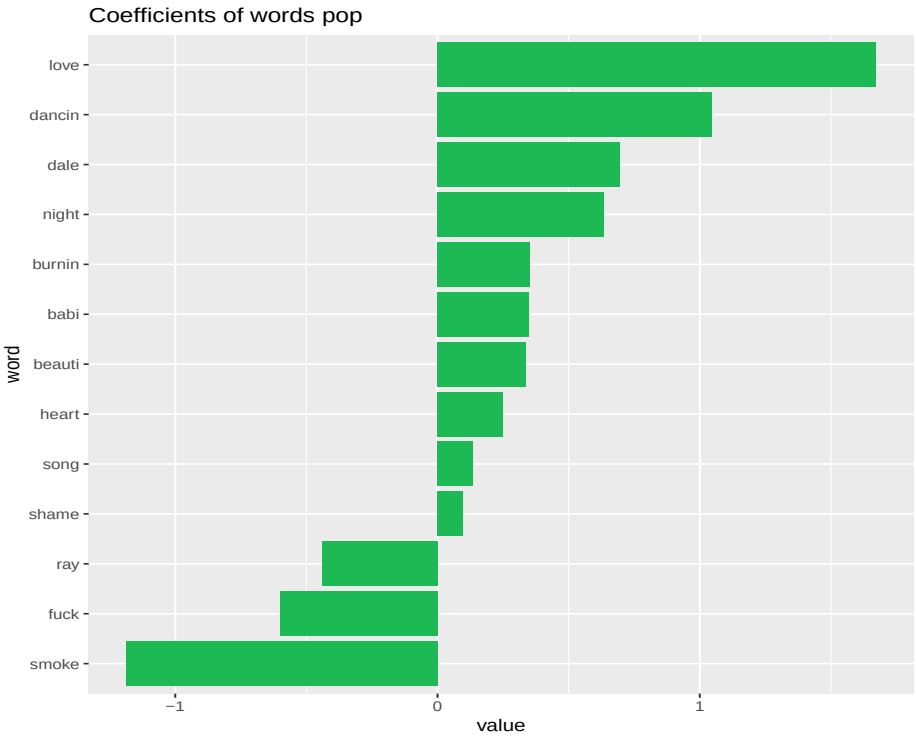


Figure 1: Coefficients of terms in pop genre selected by Lasso

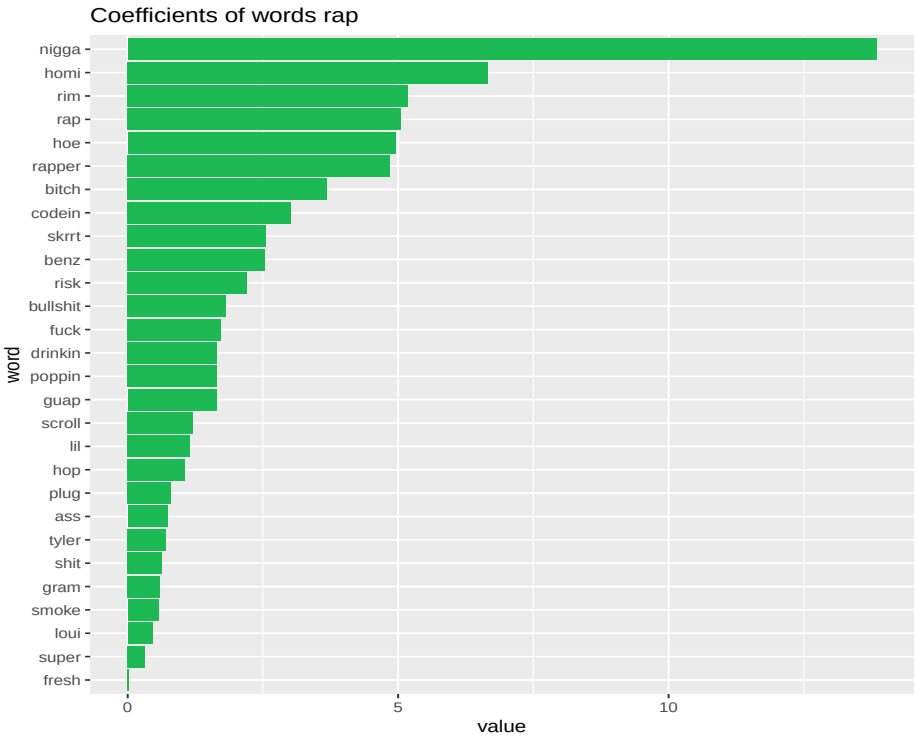


Figure 2: Coefficients of terms in rap genre selected by Lasso

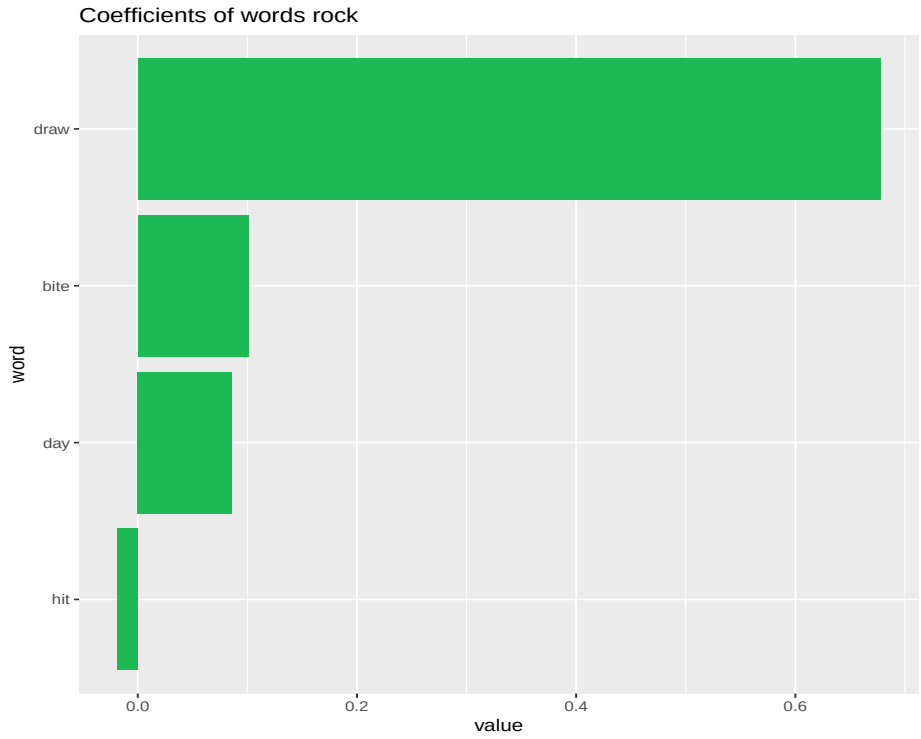


Figure 3: Coefficients of terms in rock genre selected by Lasso