Genre clustering of music reviews

Tudor Andrei Dumitrascu

January 18, 2021

1 Introduction

In this project two clustering methods will be compared and evaluated on a text dataset.

2 Dataset

The dataset consists of album reviews of various music albums from the Pitchfork website. Nolan Conaway (2017)

2.1 Preprocessing

In order to use the dataset some preprocessing steps were taken:

1. remove the \xa0 symbol When loading the csv file the xa0 whitespace symbol appeared as text and it had to be removed
2. remove symbols Removes all the symbols that are not alphanumeric or whitespaces, along with the dash ('-') and underscore ('_')
3. stemming This removes all the suffixes and prefixes of the words. There are multiple stemming algorithms, and the Snowball Stemmer was used in this case.

For a better result, lemmatization could have been used for the last step. This was the initial idea but was replaced with stemming due to the fact that the lemmatization of a entry (i.e. a review of an album) took too long, and considering the fact that the dataset contains 18k rows, it's not a feasible solution.

For the final preprocessing step, the whole dataset was processed using the TF-IDF algorithm.

Due to the unbalanced nature of the dataset. See the figure below.

In order to compensate for that the majority classes were downsampled.

Out of all the initial 9 classes, 4 most popular have been used. This was done by sampling trough the classes and keeping only some of the apparitions (i.e. max(rows, upper_limit)) and upper limit was set on some of them.

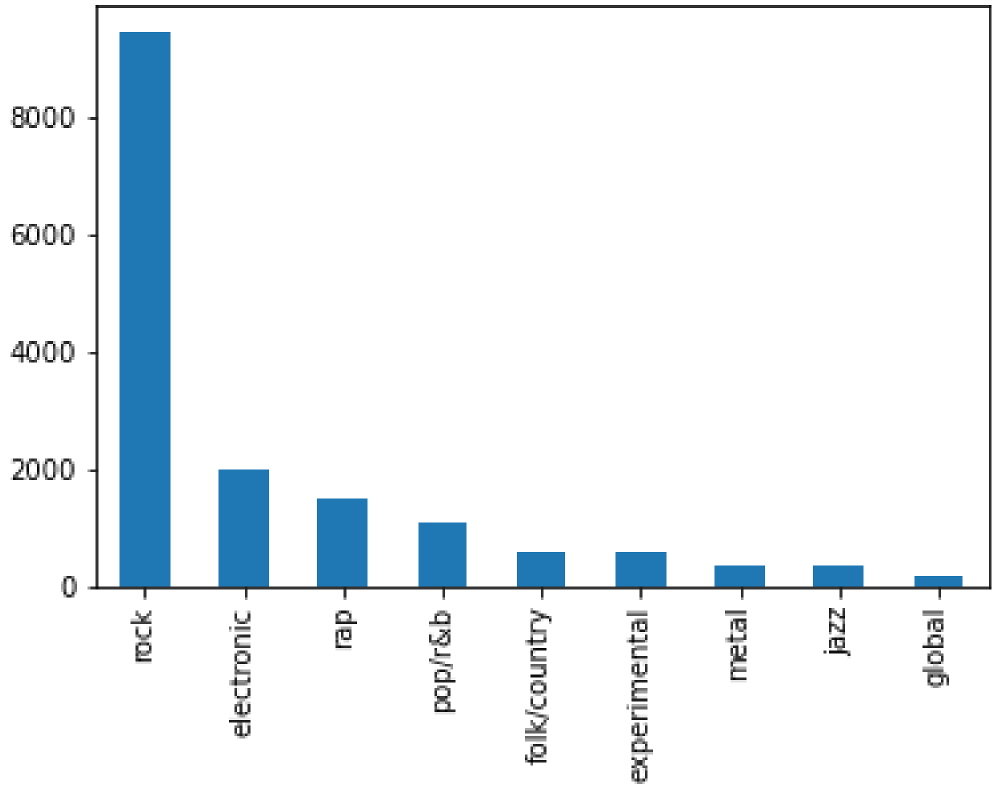

Figure 1: Genres Distribuition

3 Methods

In order to have a better overview of the performance of the clustering algorithms, a supervised method was used and also the probability of choosing at random were computed

3.1 Kmeans

3.2 DBSCAN

4 Evaluation and Results

In order to evaluate the models, a simple function that compares the true labels with the predicted labels is not enough as the label assignment is random by the clustering algorithms. Therefore, multiple metrics have been used:

- Completeness Score
- Homogeneity Score
- Silhouette Score
- Accuracy As stated previously the labels cannot be simply compared, but this problem can be overcome. By permutating the columns of the confusion matrix in order to obtain the maximum value on the main diagonal. This permutation process is quite consuming and it's replaced by the Hungarian algorithm Morbieu (2018) François Role (2019)

Method	Accuracy	Completeness	Homogeneity	Silhouette
Random prediction	11.1%			
RandomForest	62%			
Kmeans	62%	62%	62%	62%
- PCA	62%	62%	62%	62%
- t-SNE	62%	62%	62%	62%
DBSCAN				
- PCA	62%	62%	62%	62%
- t-SNE	62%	62%	62%	62%

5 Conclusion

References

François Role, Mohamed Nadif., Stanislas Morbieu. 2019. “CoClust: A Python Package for Co-Clustering.” 2019. https://smorbieu.gitlab.io/accuracy-from-classification-to-clustering-evaluation/.

Morbieu, Stanislas. 2018. “Accuracy: From Classification to Clustering Evaluation.” 2018. https://smorbieu.gitlab.io/accuracy-from-classification-to-clustering-evaluation/.

Nolan Conaway. 2017. “18,393 Pitchfork Reviews.” https://www.kaggle.com/nolanbconaway/pitchfork-data.