Music Analysis and Classification with Similarity-Based Networks

Joel Benson, Trina Sudol

Washington University in St. Louis

December 14, 2019

Abstract

Thanks to modern technology and the internet, it is easier than ever to create, publish and access new music. As the amount of available music increases, the blending and creation of new musical genres occurs more and more frequently. We propose a new method of classifying music into genres using clustering techniques on similarity-based album networks. Our proposed method is to construct a similarity-based network according to track feature measures and group the tracks into genres using network clustering. In this paper, we will analyze the effectiveness of this method by comparing the classification resulting from our methods to the ground-truth genres the songs are labeled with. The data we will be using for this analysis is taken from the Free Music Archive, a library of free, legal song downloads paired with song metadata. The metadata contains descriptive labels for thousands of songs, including Echo Nest features such as 'danceability' and 'liveliness', as well as true genre labels. Referencing our Complex Network Analysis in Python textbook, we will use the Echo Nest features to construct a similarity-based network of the tracks. Then, we will first analyze a few genre sub-graphs to demonstrate the power of this method in learning about music and characterizing genres. Subsequently, we will use spectral clustering and modularity maximization on the network as a whole to obtain communities within this network, which we can interpret as a classification of the tracks into genres. Finally, since our data contains the ground-truth genres for each track, we will analyze the accuracy of our model in obtaining clusters of tracks similar to the true genre labels. We feel that though our results will provide interesting insight into how music is classified, they may be extended to a classification method of any domain.

Keywords— **Music Classification**, **Similarity-Based Networks**, **Network Clustering**

1 Introduction

Music classification plays an important role in the music industry, categorizing sounds in a way that allows listeners to learn important characteristics about music and make choices about whether certain things are worth listening to. The labelling of songs, albums, and artists with genre names fragments the landscape of music into communities that are much easier to navigate when listeners are looking for new music. However, there are many factors that go into the categorization of music into genres, some based on the music's social or geographic context, and some based on the musical content. Given the essential purpose of music classification for informing listeners of the 'type' of certain music, we argue that the classification of music should be done in such a way as to give listeners as much information about the content of music as possible. Our proposed method of classification is to use quantitative metrics of individual songs or albums to create a similarity-based network and use community-detection algorithms to divide the music into clusters. In the following pages we will discuss the process and results of implementing this method on a database of tracks, as well as how similarity-based networks can be used to understand more about music and analyze properties of current genre-based communities.

The implementation of the following can be found on Github:
`https://github.com/joelbenson/MusicNetworkAnalysis`

2 Data

The data used for this project was taken from the Free Music Archive, a library of legal audio downloads. The library contains thousands of tracks in addition to important metadata about each track. In addition to a name and ID number, each track contains quantitative metrics that were automatically generated with Echo Nest. Hoping to use as much information as possible, we included all of the available Echo Nest features in our analysis: acousticness, danceability, energy, instrumentalness, liveness, speechiness, tempo, and valence. A sample of the data is given here:

	echonest	echonest	echonest	echonest	echonest	echonest	echonest	echonest
	audio_features	audio_features	audio_features	audio_features	audio_features	audio_features	audio_features	audio_features
	acousticness	danceability	energy	instrumentalness	liveness	speechiness	tempo	valence
track_id								
2	0.4166752327	0.6758939853	0.6344762684	0.0106280683	0.1776465712	0.1593100648	165.9220000000	0.5766609880
3	0.3744077685	0.5286430621	0.8174611317	0.0018511032	0.1058799438	0.4618181276	126.9570000000	0.2692402421
5	0.0435668989	0.7455658702	0.7014699916	0.0006967990	0.3731433124	0.1245953419	100.2600000000	0.6216612236
10	0.9516699648	0.6581786543	0.9245251615	0.9654270154	0.1154738842	0.0329852191	111.5620000000	0.9635898919

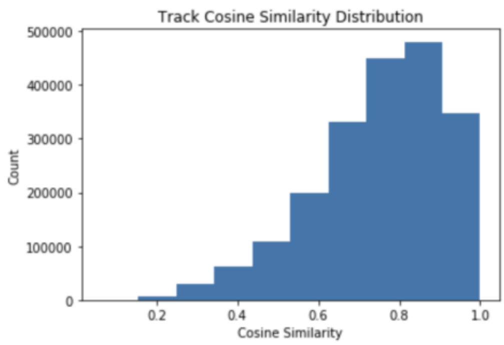
Note that each feature, except for tempo, falls on a scale from zero to one. Thus, we normalized the tempos of the dataset so all features would be treated equally in our similarity calculations. Furthermore, we also saw that some tracks contained the same track name. So, to narrow the size of our network for computational purposes while maintaining as great a variety of songs as possible, we removed duplicate tracks.

3 Network Construction

To construct our similarity-based network, we needed to make a decision regarding what measure to use for our similarity computations. Additionally, given we needed a sparse network to enable more efficient computation of communities, rather than constructing a weighted network we needed to choose a similarity threshold to determine when to connect tracks in our network.

First, after doing some research we decided to use cosine similarity to measure the similarity between tracks using the Echo Nest audio features. Treating the tracks like vectors in an 8-dimensional vector space, cosine similarity's dependence on direction rather than magnitude as well as its popular use in such cases as ours made it an appealing option. Then, with this similarity measure in mind, we computed the pairwise similarities and obtained distribution metrics to help determine an effective edge threshold:

```
Mean similarity: 0.7509836554532124
Standard Deviation: 0.15852765871813543
```



Unfortunately, as a result of the left skew, statistically significant values in the upper range fell out of the zero to one range of the similarities. So, rather than using significance as a threshold, we simply tuned the threshold to attain a fairly sparse network with a connected component encompassing as many of the songs as possible. A threshold of 0.95 gave us such a network, with a network density of 0.07 and only 6 tracks not included in the largest connected component. Thus, with this threshold and focusing only on the main connected component, we obtained our working network.

4 Initial Observations

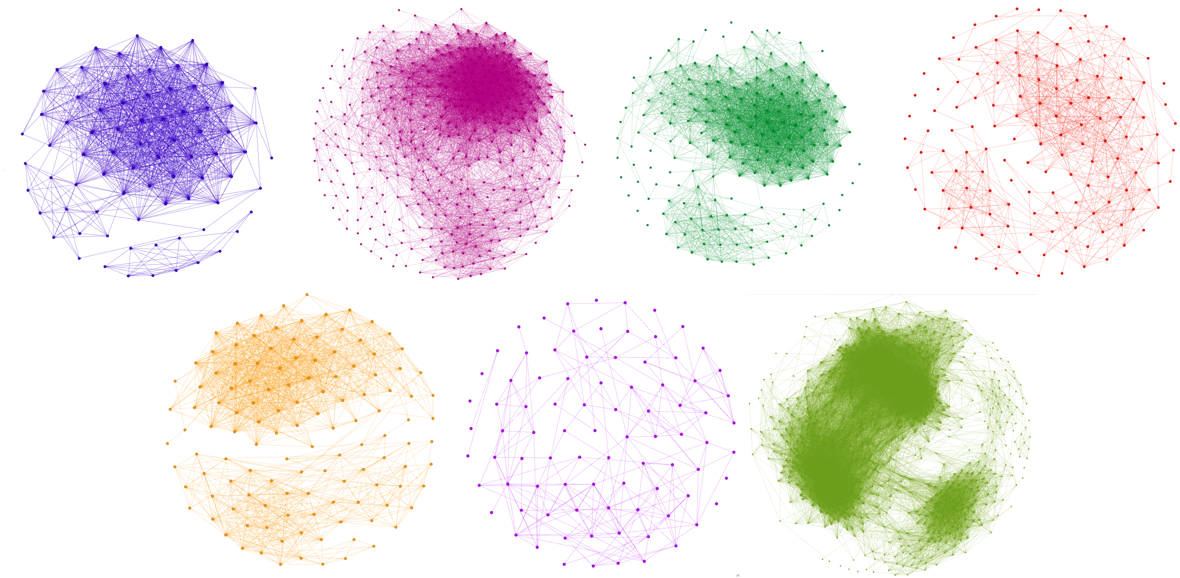
To help contextualize our discussion, we looked at some of the network's most similar tracks. The two most similar tracks were *The Bulgarian Secret Police M&M Torture Trick (Side 2)* by P.E.D. and *Heir to the Power Bear Fortune* by Argumentix. Considering rock is the most prevalent genre in our network, it is no surprise that that is the genre of both of these tracks. The tracks sound quite similar; both feature distorted vocals and an electric guitar. Both artists also abandon form and opt not to follow a typical song structure of verses and choruses. Instead, the tracks have multiple distinct verses separated by sudden, drastic changes. The main difference is *The Bulgarian Secret Police M&M Torture Trick* uses more traditional instrumentation, while *Heir* *to the Power Bear Fortune* also utilizes various sound effects, including industrial noises and the sounds of a train running. Overall, both tracks sound very similar, but "rock" is a very vague classification of them. They certainly do not resemble music from well-known rock bands, so a more specific genre classification, such as punk or alternative rock, would be helpful for describing them.

Excluding rock songs, the two most similar tracks were folk songs, *Scatter Ways* by Travelling Bell and I am Fine by Mary Halvorson and Jessica Pavone. Again, these tracks share many similarities. Both feature soft vocals, acoustic sounds, and minimal instrumentation, though the specific instruments used differ. Beyond these key traits characteristic of folk music, there is more variation between the tracks than between the two rock tracks. *Scatter Ways* is more upbeat and poppy, while *I am Fine* is more serious and experimental. The base in *I am Fine* makes it sound classical at times. The other main difference between the two is that *Scatter Ways* has vocals throughout the majority of the track, while *I am Fine* has long instrumental breaks. Overall, both tracks definitely sound like folk songs, but *Scatter Ways* and *I am Fine* could be more accurately classified as pop folk and classical folk, respectively. As we will discuss in our genre analysis in the next section, the folk genre in particular seems to be characterized by a few specific features that dominate their similarity measures.

In general, the most similar songs of our network shared the same genre, which suggests that genre is an effective means of classifying music. In the following sections, we will evaluate this claim more thoroughly through genre sub-network analysis and classification methods.

5 Genre Sub-Networks Analysis

Visualizations of Genre Sub-Networks:



From left to right: (top row) Classical, Electronic, Folk, Hip-Hop; (bottom row) Old-Time/Historic, Pop, Rock

Genre Sub-Network Statistics:

Genre	Classical	Electronic	Folk	Hip-Hop	Old-Time	Pop	Rock	Average
# of Nodes	79	412	184	139	124	87	845	267.14
# of Edges	1115	9586	3185	729	1454	226	34581	7268.
Average Degree	14.114	23.267	17.310	5.245	11.726	2.598	40.936	16.46
Diameter	5	10	9	11	7	15	9	9.43
Average Path Length	1.673	2.992	2.722	3.952	3.081	5.208	3.054	3.24
Density	0.362	0.113	0.189	0.076	0.191	0.060	0.097	0.155
Modularity	0.184	0.370	0.305	0.524	0.465	0.672	0.551	0.439
of WCC	2	10	13	9	1	7	6	6.9
Clustering Coefficient	0.811	0.609	0.714	0.604	0.729	0.570	0.625	0.666

Note: Of the 12 genres sub-networks, only the 7 largest were included. The other 5 were too small to produce meaningful results.

The Rock sub-network is by far the largest, with 845 nodes and 34,581 edges. With a density of 0.097, it is also one of the most sparse, denser only than Hip-Hop (0.076) and Pop (0.060). This already suggests a need for more specificity within our current genre classification system. The Rock sub-network's above average modularity score (0.551) and below average clustering coefficient (0.625) are further evidence that classifying a song as "Rock" is not specific enough to provide much information about the music, so new genre labels or sub-genres could be more effective.

Interestingly, the Classical sub-network is the smallest (79 nodes), but its density is 0.362, making it the most dense genre. This sub-network also has the lowest modularity (0.184) and diameter (5), suggesting that even the least similar songs in this genre are fairly similar. Furthermore, it has the highest clustering coefficient (0.811), which indicates the small-world effect. The presence of this phenomenon, combined with the fact that it has only 2 weakly connected components and the shortest average path length (1.673) of all the genres, confirm that songs in this genre are very similar to each other and Classical is an effective identifier of such music.

In our preliminary analysis, Folk seemed to be most distinctive genre. The Folk sub-network's above average density (0.189) and clustering coefficient (0.714) and its below average modularity (0.305) and average path length (3.952) support this claim. However, it also has the highest number of weakly connected components. This discrepancy implies that Folk music may have one or a few common qualities that are highly characteristic of the genre but differ greatly in other ways. Based on our initial analysis, it is likely that this defining characteristic is acousticness, though more detailed analysis of each feature would be necessary to confirm this. While this doesn't necessarily suggest an issue in our current genre classification system, it does reveal the complexity of classifying music and suggest that there may not be one conclusive method of defining genre.

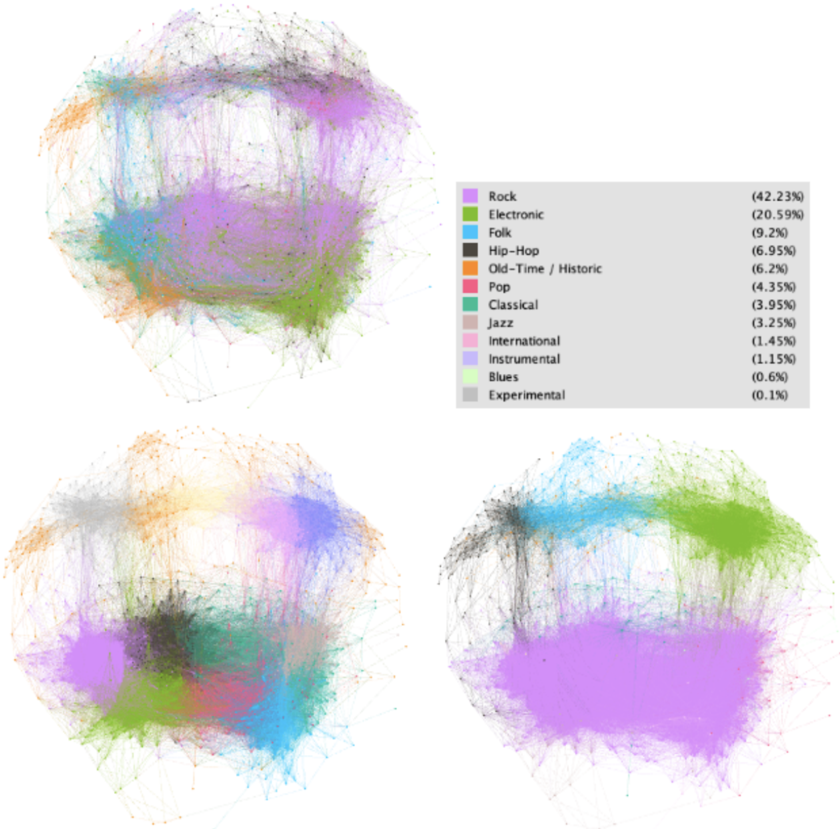
As noted above, the sparsest sub-networks are Hip-Hop and Pop. These two sub-networks also have the longest average path lengths (3.952 and 5.208, respectively) and largest diameters (11 and 15, respectively), which implies there is more variation, or a wider range of music, within these genres. That Hip-Hop and Pop also have above average modularity scores (0.524 and 0.672, respectively) and below average clustering coefficients (0.604 and 0.57, respectively) indicates that, like Rock, these genres could be more effective at classifying music if they were divided into sub-genres.

Analyzing each genre as its own sub-network is helpful for not only evaluating the current genre classification, but also proposes a method for improving it. Modifying our genre networks so their statistics resemble those of the Classical sub-network would result in genre groups that are far more descriptive and representative of the music within them. As such, network analysis and classification methods, which we discuss in the next section, could produce a more effective genre classification system.

6 Classification Methods

We will now proceed to focus again on the network as a whole and see how we can effectively classify tracks into genres. Again, our hope is to make our genre labels more meaningful in the sense that they give more information about the songs which they label. Our approach to labelling the tracks in such a way is to locate communities within our similarity-based network. Our assumption is that these communities represent groups of songs that are most similar to each other. So, our methods will be to use modularity maximization and spectral clustering on the network. Betweenness-based clustering was additionally considered, but its relative ineffectiveness and lack of inefficiency on such a large network led us to simply use the two methods. Additionally, for the purposes of comparing our labels to the true genre labels, we used these methods to find the same number of clusters as the true genres. In addition to its usefulness for comparing the methods, this may preserve the practical use of genre labels as a useful guide through the world of music rather than an overwhelming assortment of labels.

7 Results



Visualizations of clusters: The network on top is colored according to true genre labels in accordance with the legend on the right. The networks below on the left and right are colored according to the modularity maximization and spectral clustering methods respectively.

	True Genres	Modularity Maximization	Spectral Clustering
Modularity	0.741	0.919	0.588

Comparing the modularity scores of each clustering method to that of the true genres and to each other, it is clear that modularity maximization was the most effective clustering method for our similarity-based network. With a modularity of 0.919, it not only significantly outperformed spectral clustering, but also performed better than standard genre classification, which had a modularity score of 0.741. These results coincide with our findings from the genre sub-networks analysis and provide further evidence that our current genre classification system is not as effective as it could be. Spectral clustering, on the other hand, had a modularity score of 0.588, which was even lower than that of the true genres. One reason this may have happened is that within spectral clustering, there are different ways to actually implement the method. We chose to use the second smallest eigenvector to create our clusters, as that was the method we discussed in class. It is possible, however, that a different method, such as using multiple eigenvectors, could perform better. More detailed analysis of various spectral clustering methods specifically would be necessary to determine if spectral clustering could be used for effective genre classification.

The visualizations of our clusters coincide with our results. Modularity maximization produced the most distinct clusters. Although there are some regions where one genre dominates, the true genre clusters have much more overlap, which implies that genres may be helpful for characterizing some tracks, but there are many songs which do not fit into only one existing genre. Spectral clustering produced more distinct clusters than the true genres, but over half of the nodes are in one cluster. Classifying songs based on this clustering likely provides little information to listeners. Therefore, modularity maximization was the most effective method for classifying music and could be used to improve our current genre classification system.

8 Conclusion

In conclusion, we see that similarity-based networks provide us with a valuable tool for understanding the relationships between tracks and genres. We feel that these proposed methods of classification more effectively communicate the structure of the world of music and help listeners navigate this complex world. However, we do recognize the limitations in our method as we have implemented it and note that more research would be needed to make these methods truly useful and practical. First, we were limited in our descriptions of songs by the Echo Nest features which were provided in the Free Music Archive dataset. Should these classifications be useful, we feel they would need to take into account more subjective features such as emotional and lyrical content. Also, further research would need to be done into the effectiveness of various similarity measures or edge threshold values in obtaining a meaningful similarity-based music network. Despite these limitations, we feel that these results display the possible usefulness of these methods in the classification of music, and more generally, making complex networks of content more easily navigable by users.

References

M. Defferrard, K. Benzi, P. Vandergheynst and X. Bresson. "FMA: A Dataset for Music Analysis." *18th International Society for Music Information Retrieval Conference*. Sept. 2017. Accessed on Nov. 23, 2019. [Online]. Available: https://arxiv.org/abs/1612.01840

D. Zinoviev. "Similarity-Based Networks." *Complex Network Analysis in Python.* Raleigh, NC: Pragmatic Bookshelf, 2018, ch.14. Accessed on: Nov. 24, 2019. [Online] Complex Network Analysis in Python, Chapter 14 (Similarity-Based Networks)