Evolution of clusters in music

Federico Inserra and Lovro Kalinić

January 2021

1 Introduction

In this project, we decided to study the communities that form in the musical world. In particular, we studied how artists' collaborations change over the years. Our idea was to see if an artist changes genre communities over time, for example if he collaborated with mainly pop artists in the first period of his career and with mainly hip hop artists afterwards.

To achieve this we used the dataset containing all the top hits from 1991 to 2020. Each of these songs is a collaboration between several artists. For each year from 1991 to 2020 we created a graph in which the nodes were the artists and two of them were connected by an arc if they collaborated on a song that year. Once the graphs were created, we ran algorithms to find communities within them. Our hypothesis was that artists who collaborate have similar genres and therefore different genres will correspond to different communities within the graph.

Having the communities for each year we could study how the artists evolve. We were able to see the trajectory of artist's most common genre over the years, perhaps going from rap to pop and then back to rap again. We could also see which artists collaborate the most and which artists change genres most frequently.

2 Dataset

In this project we were analyzing data from a Kaggle Spotify Dataset 1921-2020 dataset[1]. The entire dataset was obtained using the Spotify API and the Python library called *spotipy*. It contains more than 160 000 songs from 1921 to 2020. Each of the songs has large quantity of interesting information such as popularity (from 0 to 100), artists who made it, danceability (from 0 to 1), name, release date and so on. For our needs we have only taken songs from 1991 to 2020 in order to analyze more current artists. All of the songs we analyzed had at least two artists linked to it. Furthermore, we only chose songs with a popularity level of at least 10 to exclude songs uknown to the majority of the people.

[1] https://www.kaggle.com/yamaerenay/spotify-dataset-19212020-160k-tracks

Among all the information that the dataset provided us, we were only interested in the artists who collaborated on the song, the year of release and the genres of the artists who participated. Unfortunately, the last information was not present in this dataset, so we wrote a script that uses the *spotipy* module. For each artist within the dataset it retrieves the list of his genres (directly provided by Spotify) and stores them in a JSON file. Since Spotify API returns a list of sub-genres for each artist which is too specific for our purposes, we implemented a filter to transform these sub-genres into the one of 22 major genres. For example, the two sub-genres called Canadian pop and post-teen pop were both mapped to the more general pop genre. This allowed us to get a clearer idea of the genres of certain communities.

The graph on the Figure 1 represents the number of songs per year in our dataset. It can be easily seen how the number of collaboration songs has grown exponentially in recent years compared to previous years. This is probably also due to the ease with which new artists can nowadays make a name for themselves without having a big record companies behind them. This is mainly due to the explosion of music platforms such as Spotify, which allow music to be listened to in a much more convenient way than in the past, while also helping the user to discover new genres and artists.

3 Methodology

In order to prepare for the project, we wanted to analyze some of the most important papers with similar topic. We started with the Evolutionary clustering paper [2]. This paper was probably the first one to deal with the subject of evolutionary clustering. Although the paper was very interesting to read, it used different methodologies and approaches than ours. First of all because the paper focused more on building a framework that could operate *online* while in our case we built an algorithm that operates *offline*. The difference between *online* and *offline* in this case, as also explained in the paper, is that an algorithm that operates *online* knows only the data available up to time t and must operate blindly trying to classify the data that arrive at time $t+1$ on the basis of the information collected up to that moment. An algorithm that operates *offline* on the other hand already has all the data available, and this allows it to obtain results that are at least as good as those of the "online" algorithm.

Next papers which we analyzed were Adaptive Evolutionary Clustering[3] and Evolutionary Spectral Clustering by Incorporating Temporal Smoothness[4]. Both approaches taken were aimed at constructing "better" clusters that best reflect the data within them and always try to balance the trade-off between past and new data. These types of clusters are therefore in constant transformation, because they start from old data and transform to adapt to new data as they arrive.

[2] https://www.researchgate.net/publication/221654105_Evolutionary_clustering
[3] https://arxiv.org/pdf/1104.1990.pdf
[4] https://www.microsoft.com/en-us/research/wp-content/uploads/2017/01/evospe.pdf

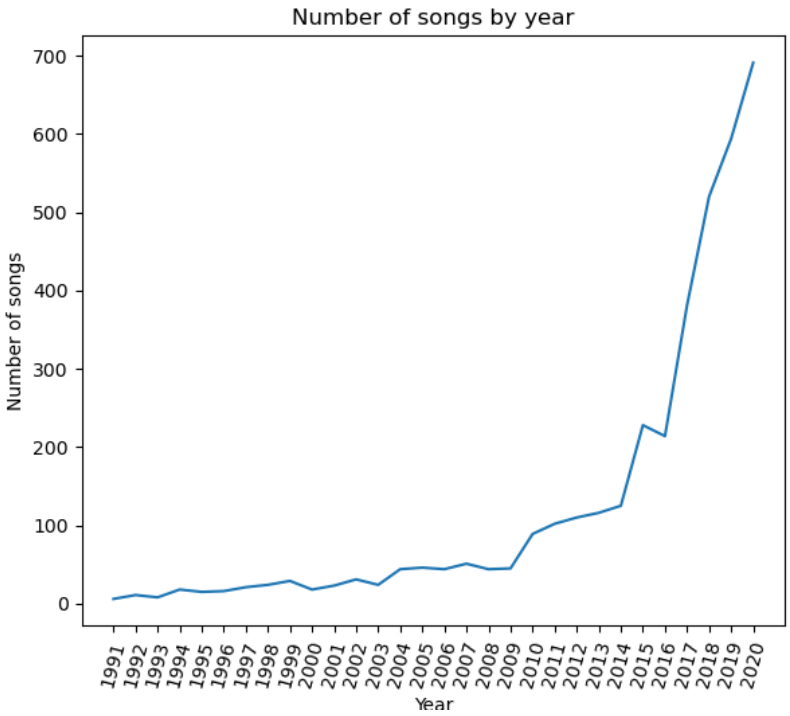

Figure 1: Number of songs by year of the dataset we used

Our approach is more oriented towards studying how these clusters evolve over time. In particular, what we study is how the entities within these clusters (in this case, the singers) evolve over time by changing communities to which they belong. In these terms our analysis should be consider "study on the evolution of clusters" rather than "evolutionary clustering". To implement our approach, we first divided all the songs according to the years they were released. Then we created a dictionary, whose keys were the years from 1991 to 2020 and each of them is associated with a list of songs released that year. Each song is represented by a song object which has attributes title, year and list of artists who collaborated on that song. At this point, we created a graph of collaborations for each year. The vertices of this graph are the artists who collaborated on at least one song in that year and two vertices are connected by an arc if they collaborated on a song in that year. Figure 2 is a visualisation of collaboration network of artists in the year 1991 and figure 3 in the year 2019.

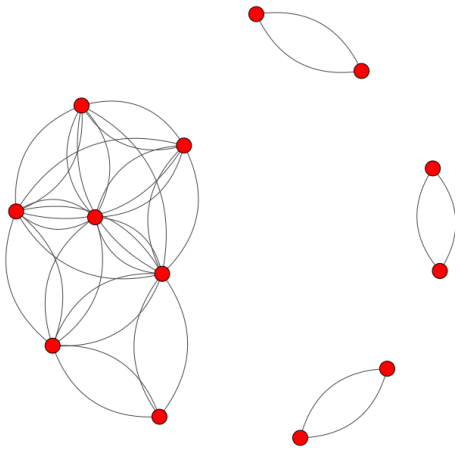

Figure 2: Graph of the year 1991

After constructing the graph we ran different community finding algorithms on these graphs to identify the different artist communities for each year. We analysed the communities returned by the algorithm, in particular studying which artists they were made up of. We then created a data structure in which each artist for each year is associated with a list of the genres of artists who were in the same community in that year. This data structure represents the history of each artist and how they have changed genre over the years. Using this structure we were able to represent the history of each artist on a chart and see how it has changed style over the years. For example, for the year 1991, the biggest community we can see in the figure 2 consists of artists Jerry Orbach, Angela Lansbury and five others. From the data we obtained from Spotify, the genre of Jerry Orbach's opus is Broadway and genres of Angela Lansbury's are called Hollywood, movie tunes and show tunes. So the list associated with this community consisted of all those genres.

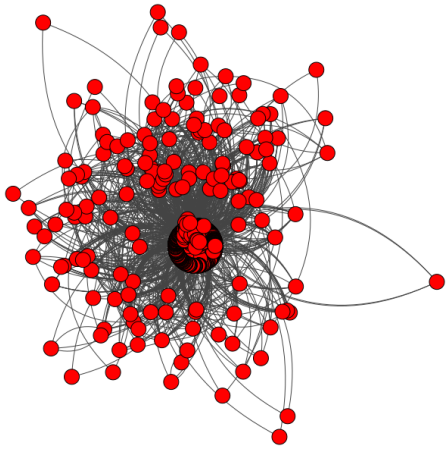

Figure 3: Graph of the year 2019

Having the list of genres, we found the most common one (or ones if the number of occurrences of these genres is the same) and declared it as the genre of the community. Then, for each artist, the timeline of his career evolution over genres, based on his collaborations, could be obtained.

There are many different community finding algorithms, so in order to decide which one to use for different networks we tried several of them and analysed their properties on the communities they generated in order to see which one was the most suitable for our purposes. We used the following metrics to evaluate different algorithms:

- *expansion* $\in (0, \infty)$ is defined as the ratio of edges leaving the community and the total number of vertices in the community.
- *conductance* $\in (0, 1)$ is a fraction of total edge volume that points outside the community.
- *modularity* $\in (0, 1)$ is the fraction of the edges that fall within the given groups minus the expected fraction if edges were distributed at random.
- *triangle participation ratio (TPR)* $\in (0, 1)$ is defined as a fraction of nodes in a community which belong to at least one community triad.

High values of TPR and modularity and low values of expansion and conductance are indicators that the algorithm produced quality communities. Considering all those metrics equally important, we used the following equation to grade them:

$$grade = expansion + conductance + (1 - modularity) + (1 - tpr)$$

where the algorithm with the best properties has the lowest grade. In the results section you can find the table that groups all the algorithms we analysed and their respective metrics.

4 Results

As mentioned in the previous section, for each year's graph we ran different community finding algorithms. We analyzed the results using different community measures and for each year we continued the analysis only with the one we found had the best properties.

Different algorithms showed the best properties for different years' networks. We can see that observing the tables 1 and 2. Bold algorithm names indicate best algorithms for those networks. We also wanted to see which of the algorithms had the best properties over the whole year interval. We calculated average metric values over the year and the results we obtained (which are displayed in table 3) was that algorithm *walktrap* was the most appropriate one.

The result of our algorithm was a timeline of the dominant genre over the active years of an artist. The first step was applying the community detection algorithm on networks of yearly collaborations. Each such network consisted of artists as vertices and collaborations between two artists as edges.

Table 1: Metrics of community algorithm results on the year 2001

	Expansion	Conductance	Modularity	TPR
walktrap	0.1928	0.0024	0.9933	0.6868
infomap	37.1566	0.8872	0.9933	0.1446
label.propagation	0.1928	0.0024	0.9933	0.6868
leading.eigenvector	1.9288	0.0250	0.9933	0.6868
multilevel	1.9288	0.0250	0.9933	0.6868

Table 2: Metrics of community algorithm results on the year 2011

	Expansion	Conductance	Modularity	TPR
walktrap	0.2483	0.0320	0.3981	0.4138
infomap	1.3241	0.1983	0.3981	0.3862
label.propagation	0.4689	0.0622	0.3980	0.3862
leading.eigenvector	0.1931	0.0247	0.3980	0.4068
multilevel	0.1931	0.0247	0.3980	0.4068

In the following figures (4-11), we can see the artists' stories. Each blue point represents the most common genre of collaboration artists in the active year of the artist we are analysing. Sometimes there are multiple genres for the same year, which means that there was the same quantity of those genres in collaborations.

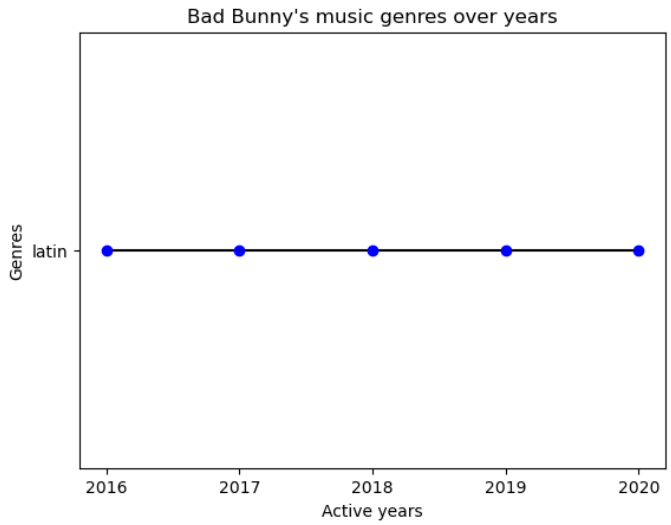

Figure 4: Evolution of Bad Bunny's most common genres

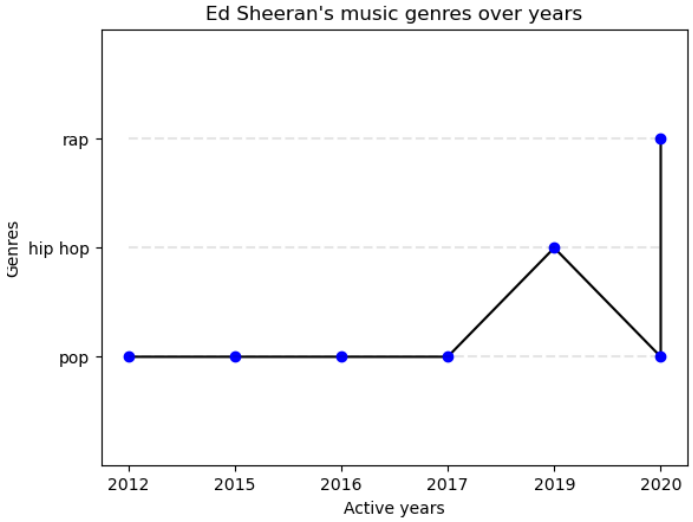

Figure 5: Evolution of Ed Sheeran's most common genres

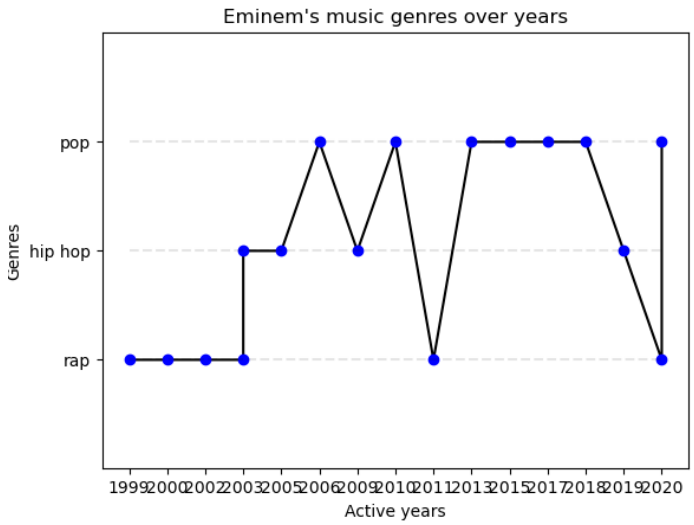

Figure 6: Evolution of Eminem's most common genres

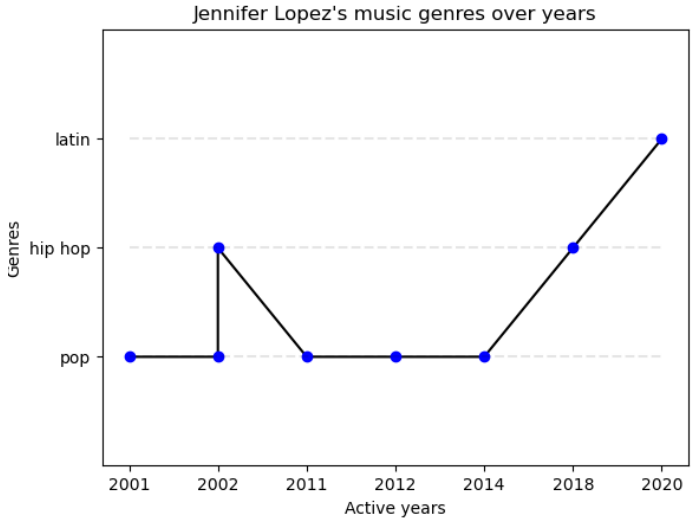

Figure 7: Evolution of Jennifer Lopez's most common genres

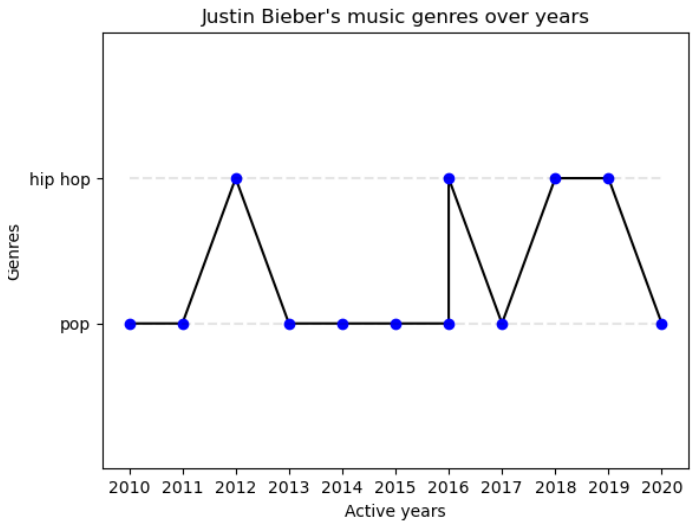

Figure 8: Evolution of Justin Bieber's most common genres9

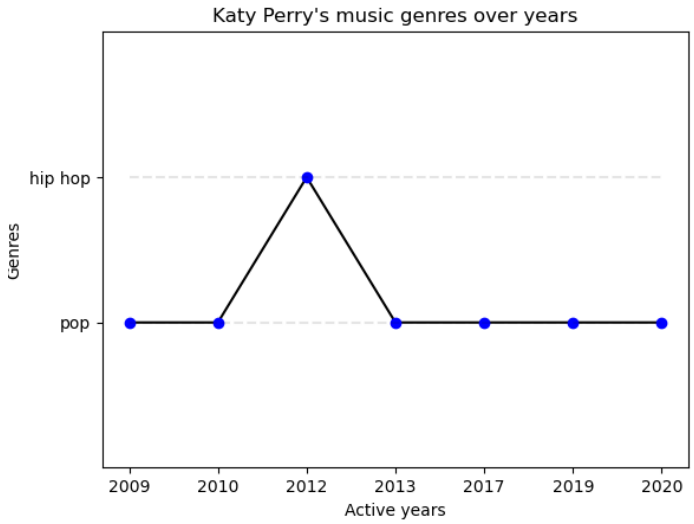

Figure 9: Evolution of Katy Perry's most common genres

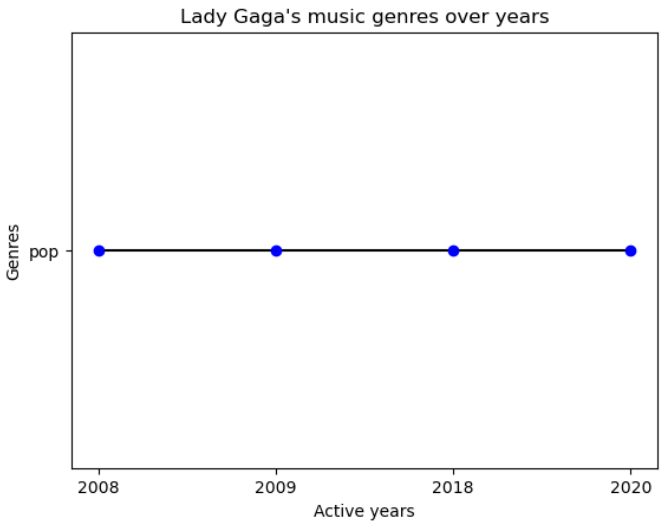

Figure 10: Evolution of Lady Gaga's most common genres

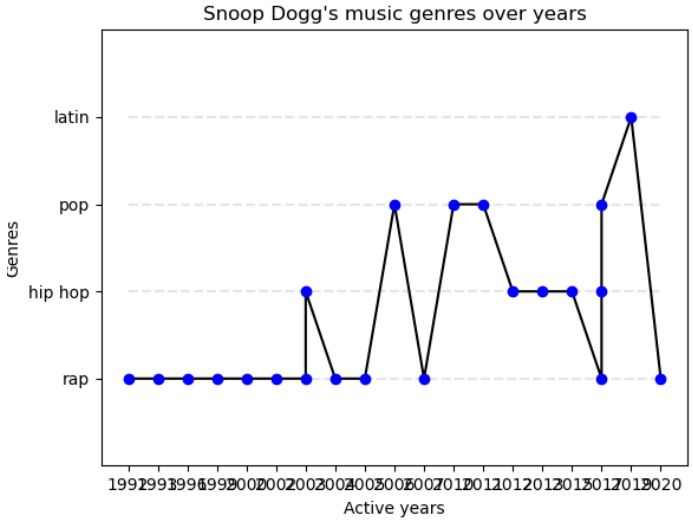

Figure 11: Evolution of Snoop Dogg's most common genres

Table 3: Average metrics of community algorithm results for all years

	Expansion	Conductance	Modularity	TPR	Grade
walktrap	0.1533	0.0150	0.6646	0.4415	1.0621
infomap	3.2214	0.2873	0.6646	0.1848	4.6676
label.propagation	0.2374	0.0222	0.6646	0.4361	1.1587
leading.eigenvector	0.2401	0.0183	0.6646	0.4380	1.1558
multilevel	0.2680	0.0217	0.6646	0.4420	1.1873

5 Conclusions

In this project we focused on the world of music. In particular, we have studied how artists have changed over time. We wanted to see which genre they most identified with, in each year over their careers.What we could conclude from different artists' stories is that most of them tend to try out different genres and styles as their careers are progressing (like for example, Snoop Dogg and Eminem). Few of them, like Lady Gaga and Bad Bunny, are loyal to their style and, from our results, they don't change it over years.

In general, we have noticed that the trend of artists in recent years is to do collaborations with artists, even of genres other than their own. For example, we can see that artists such as Snoop Dog, who belonged to the rap community from 1992 to 2002, have moved to different communities in the following years. Thanks to this project we have looked even more deeply into the communities that can be created in the real world, with a very concrete example being the artists on Spotify. We are sure that many more studies will be carried out on communities in graphs, especially nowadays when people are more and more connected and in more and more fields. A study like ours can then be applied in different areas to identify different behaviours and patterns.