Universitat Autònoma de Barcelona

Graphs and Network Analysiss
Project Report

ANALYSIS OF MUSIC NETWORKS

Students:
Júlia Garcia, 1630382
Josep Maria Rocafort, 1631378

8th may 2023

Contents

1 Introduction

This report presents the findings of a laboratory project conducted for the course "Graphs and Networks Analysis." The objective of this project was to analyze the data provided by Spotify in order to explore an artist and gain insights into their popularity, music genres, and listener demographics. This report aims to provide an interesting and comprehensive analysis of the collected data.

The structure of this report is as follows: First, we present the analysis of music networks, which is divided into sections corresponding to each Laboratory Session (1 to 4). In these sections, we discuss the results obtained from each session. Following that, we provide the results of Session 5, which was approached differently from the rest as it was not guided.

2 Analysis of music networks

2.1 Session 1: Data acquisition and storage

2.1.1 Graphs Order and Size

In this section, we implemented several functions so we could build different graphs and a dataset using Spotify data. The following plots correspond to two graphs of related artists starting with the artist Drake and exploring 200 artists using two different algorithms, BFS (breadth-first search) and DFS (depth-first search) respectively. Note: We explore until the generated graph has at least 200 nodes.

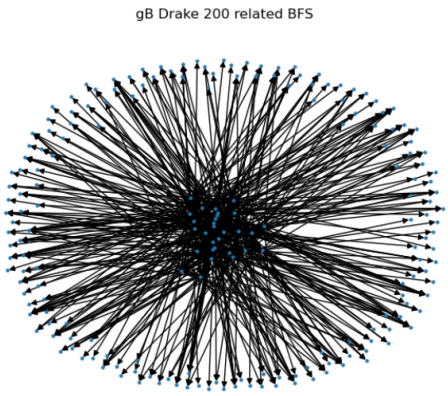


Figure 1: Graph obtained with BFS.

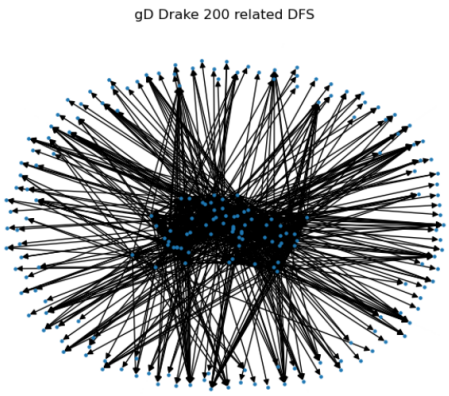


Figure 2: Graph obtained with DFS.

Although we end up with roughly the same number of nodes,(as we explicitelly coded for this) graphs (Gb and Gd) is different. Concretely, Gb has order =776 and Gd order =529 . The reason for the difference in the order of the two graphs is due to the strategies used to build them.
To build the first graph, the artists are explored level by level, meaning it explores all the neighbors of an artist before moving on to the next level. On the other hand, the strategy to build the second graph consists of exploring an artist's connections as deeply as possible before backtracking.

2.1.2 Degree Analysis

Indicate the minimum, maximum, and median of the in-degree and out-degree of the two graphs (gB and gD). Justify the obtained values.

```
gB:

indegree: median [3.0], max [34], min [0]
outdegree: median [0.0], max [20], min [0]

gD:

indegree: median [4.0], max [61], min [0]
outdegree: median [0.0], max [20], min [0]
```

At first we thought that these values didn't make sense and should be in reverse, meaning gB should have higher indegress as it should find more artists that are related to each other than gD which by the nature of deepth first search should find more and more different artists between them. Anyhow this hasn't been the case, actually our hypothesis is that the depth first search indeed has found more new nodes/artists but they all seem to still have connections in regards of being related to some group that was present in the search, thus this group having more connections.

2.1.3 Dataset Analysis

The number of artists in dataset D is 327. This includes the initial artist Drake and the additional artists explored in the graphs gB and gD. The number of artists should be between 200 and 400 because of the following scenarios: either all explored artists being unique (400) or some overlap resulting in a lower number of unique artists.
In the other hand, the number of unique songs obtained being 3131 is correct. Initially, it may seem that there would be 3270 songs (327 artists multiplied by 10 songs per artist) based on the assumption of 10 songs displayed for every artist in the graphs. However, we have to take into account the possibility of collaborations between artists that contributes to the lower count of unique songs.
The number of unique albums is the same as the number of the unique songs, meaning that all songs belong to different albums.

2.2 Session 2: Data preprocessing

2.2.1 Graphs Order and Size

The order and size of the four obtained undirected graphs are as follows:

- g'B: Order = 189, Size = 489
- g'D: Order = 196 , Size = 949
- gwB: Order = 251, Size = 628
- gwD: Order = 189, Size = 1777

2.2.2 Strategy for Obtaining gwB and gwD

The strategy used to obtain the graphs consisted of the following steps:

1 Load track data and filter artists based on gB and gD.

2 Compute mean audio features for the selected artists.

3 Create similarity graphs based on the audio features using a specified similarity metric.

4 Prune the similarity graphs by removing low-weight edge

2.2.3 Weakly and Strongly Connected Components in Directed Graphs

It is possible for the directed graphs obtained from the initial exploration of the crawler (gB and gD) to have more than one weakly connected component and one strongly connected component. This is because the crawler starts from a single seed artist and explores their related artists, which may lead to disconnected subgraphs. Additionally, the directed nature of the edges can create separate components when there are no reciprocal connections between artists.

2.2.4 Relationship between Directed and Undirected Graphs

The conversion from directed to undirected graphs merges the connected components and eliminates the directionality of the edges, resulting in a single connected component in the undirected graphs. In this case, we have obtained 2 connected components for g'B and 5 for g'D.

2.2.5 Largest Connected Component Size

The size of the largest connected component from g'B and g'D is as follows:

- Largest Connected Component in g'B: Size = 187
- Largest Connected Component in g'D: Size = 90

The comparison between the sizes of the largest connected components reveals which graph has a more extensive and interconnected network. In this case, the largest connected component was in the graph g'B.

2.3 Session 3: Data analysis

2.3.1 Number of shared nodes

Using the num_commo_nodes function, we determined the following:

- Number of nodes shared by gB (seed: Drake) and fB (seed: last crawled artist from DFS crawl): 174
- Number of nodes shared by gB (seed: Drake) and hB (seed: French Montana): 90

From the results, we can conclude that there is a substantial relationship between the artists associated with Drake and the last crawled artist from the BFS crawl. On the other hand, there also exists a connection between the artists associated with Drake and French Montana, although it may not be as strong as the connection with the last crawled artist.

2.3.2 Centrality analysis in g'B: degree and betweenness centrality

After calculating the centrality measures in graph g'B, we found that the number of nodes appearing in both the degree centrality and betweenness centrality rankings was 2. This indicates that these nodes play a crucial role in the graph by having a high number of direct connections (degree) and serving as important intermediaries in connecting other nodes (betweenness). The overlap between these two centrality measures highlights the significance of these nodes in maintaining the graph's structure and facilitating communication between different parts of the network.

2.3.3 Clique detection in g'B and g'D and comparison

We searched for cliques of size greater than or equal to the chosen minimum size in graphs g'B and g'D. To ensure a meaningful analysis, we selected the maximum value for the minimum clique size that generated at least 2 cliques. For graph g'B, the maximum value chosen for the minimum clique size was 7, resulting in the discovery of 7 cliques. In graph g'D, the maximum value chosen for the minimum clique size was 10, resulting in the discovery of 10 cliques.
Additionally, we calculated the total number of different nodes that were part of all these cliques. In graph g'B, this count was 20, representing the unique nodes found in the cliques. In graph g'D, the count was 26, indicating the total number of unique nodes found in the cliques.
In contrast to g'B, graph g'D displays a more complex internal structure with a higher level of interconnections among its nodes, indicating a greater degree of connectivity.

2.3.4 Analysis of the largest clique

Among the identified cliques, we focused on analyzing the clique with the maximum size. After examining the artists in this clique (J-Dawg, Big Pokey, Lil' Keke, Fat Pat...), we identified a defining characteristic that unites them.

- Duration: The average duration of the songs by these artists is around 250,000 seconds (approximately 4 minutes and 10 seconds). This suggests that they tend to create music with relatively longer durations.

- Popularity: The popularity values range from 25.5 to 45.7, indicating a moderate level of popularity among these artists.
- Danceability: The danceability values range from 0.603 to 0.814, indicating that their music tends to be suitable for dancing.
- Energy: The energy values range from 0.5077 to 0.7454, suggesting that their music carries a moderate to high level of energy.
- Loudness: The loudness values range from -10.5763 to -4.9954, indicating a relatively wide range of loudness levels in their songs.
- Speechiness: The speechiness values range from 0.19111 to 0.3244, suggesting that their music may contain a moderate amount of spoken words or lyrics.

From these observations, we can conclude that the artists in this clique tend to create moderately popular songs with longer durations, suitable for dancing, and carrying a moderate to high level of energy. The loudness levels vary, and their music may contain a moderate amount of spoken words or lyrics.

2.3.5 Community detection in gD and modularity

To detect communities within the graph gD, we employed both Louvain and Girvan-Newman algorithms.
Using the Louvain method, we obtained a modularity value of 0.70413 with 19 communities. This indicates a relatively strong community structure within the graph. On the other hand, using the Girvan-Newman method, we obtained a modularity value of 0.28211 with 2 communities. This suggests a weaker community structure compared to the previous method.
Considering the obtained modularity value, the first method returns a better partitioning of the graph into meaningful communities, where nodes within the same community are more densely connected and share similar characteristics or relationships. Therefore, the Louvain algorithm is more suitable for detecting communities in graph gD and providing insights into its internal structure and interconnections among nodes.

2.3.6 Advertising campaign cost and artist selection.

(a) Minimum cost for ensuring the ad is heard infinitely::

To ensure that a user who listens to music infinitely will hear the ad at some point, we need to consider the worst-case scenario where the user explores the entire graph before the ad is played. The minimum cost required would be equivalent to the number of unique artists in the graph multiplied by the cost per artist.

- For gB: 776 artists * 100 euros per artist = 77600 euros
- For gD: 529 artists * 100 euros per artist = 52900 euros

Justification: In the worst-case scenario, the user explores every artist in the graph before encountering the ad. By paying for all unique artists in the graph, we ensure that the ad will be played at some point during the user's infinite music-listening experience.

(b) Better spread of the ad with a budget of 400 euros:

To maximize the reach of the ad within a restricted budget of 400 euros, our objective is to select a diverse group of artists. By considering both popularity and danceability as criteria, we aim to target a broad audience and increase the likelihood of the ad reaching different listeners. To achieve this, we have sorted the nodes in each graph based on their popularity and danceability scores.

For gB:

Artist 1: Drake
Artist 2: Metro Boomin
Artist 3: 21 Savage
Artist 4: Kendrick Lamar

For gD:

Artist 1: Drake
Artist 2: 21 Savage
Artist 3: Future
Artist 4: J.Cole

2.3.7 Minimum hops to reach Travis Porter.

The minimum number of hops needed to reach Travis Porter depends on the specific connections and paths in the gB graph. To determine this value, we used the breadth-first search algorithm. The goal was to find the shortest path between the two artists in the graph. The results were as follows:

Minimum number of hops: 5

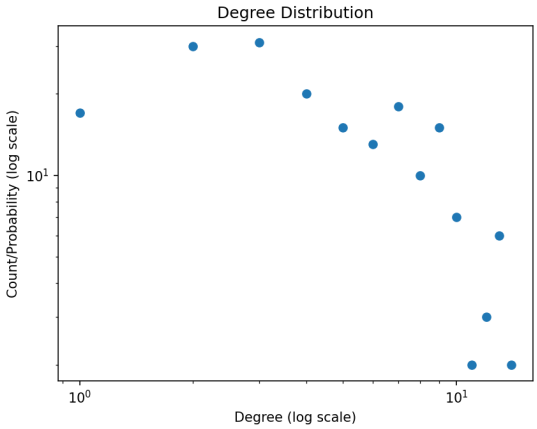
Artists to listen to before Travis Porter:

1. Young Dro
2. Rocko
3. Yo Gotti
4. Young Money
5. Soulja Boy

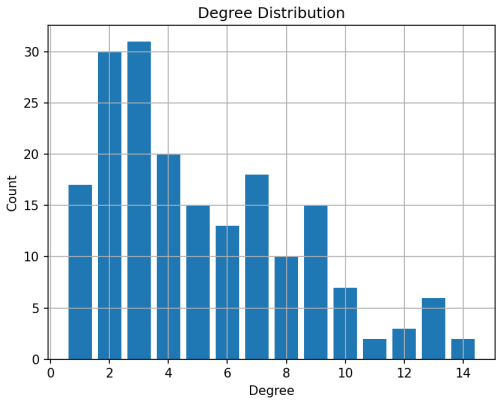
2.4 Session 4: Data visualization

2.4.1 Commenting on the results obtained in Exercise 4.

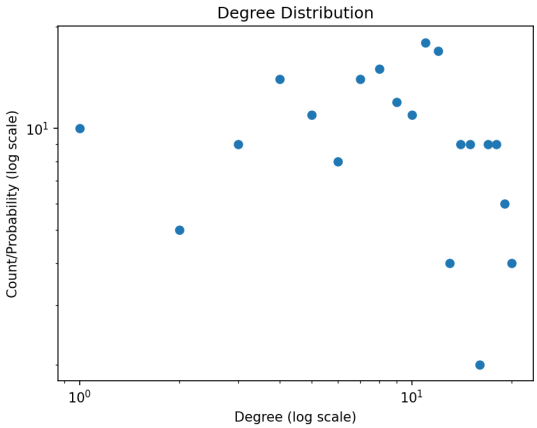
Question (a): The following figures represent the plots obtained by the degree distributions of the graphs:



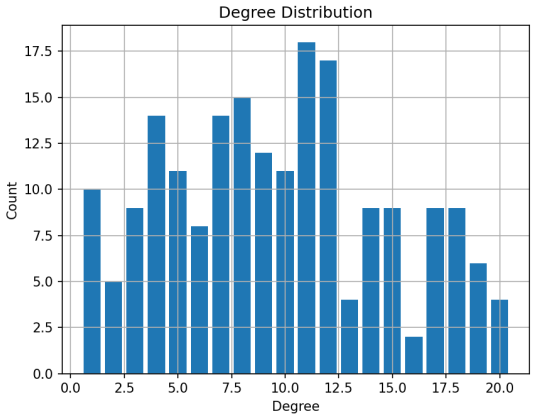
(a) Graph gB'.



(b) Bar plot gB'.



(c) Graph gD'.



(d) Bar plot gD'.

Figure 3: Degree Distribution Comparison

In conclusion, the analysis of the graphs obtained using BFS and DFS algorithms revealed distinct patterns in their degree distributions. The BFS graph exhibited a higher concentration of nodes with degrees ranging from 1 to 14, indicating a more interconnected and denser structure. On the other hand, the DFS graph displayed a more dispersed distribution, with degrees spanning from 1 to 20.
The contrasting degree distributions suggest that the BFS algorithm tends to explore neighboring nodes more extensively, resulting in a higher degree of connectivity among nodes. In contrast, the DFS algorithm explores deeper into the graph, potentially encountering nodes with lower degrees and contributing to a sparser degree distribution.

Question (b): After applying cosine similarity, it was determined that the artist with the highest similarity score to Drake from the graph is Future. Hence, we used the predefined function and obtain the following figure:

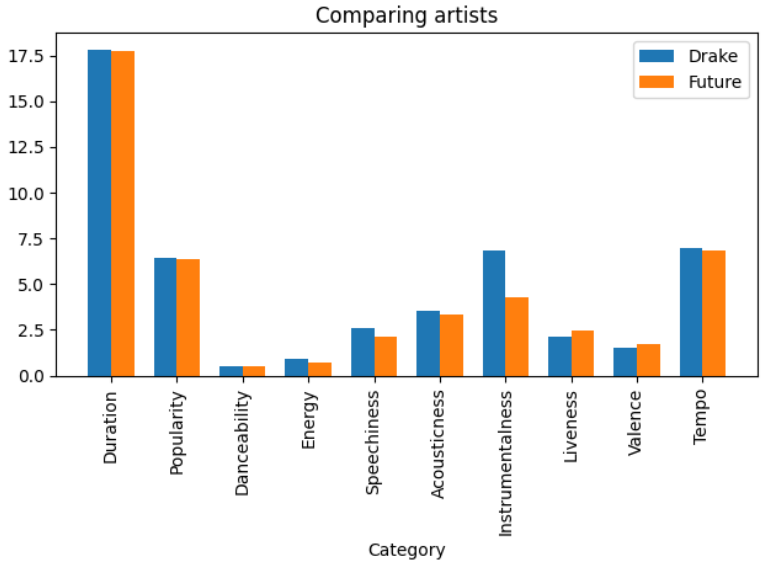


Figure 4: Compatring Drake and Future.

In the other hand, we determined that the less similar artist was Comethazine. However, it's important to note that we are comparing the nodes within the graph obtained from the BFS algorithm (gB) and not considering the entire dataset. This limitation explains the relatively lower differences observed in the degree distribution.

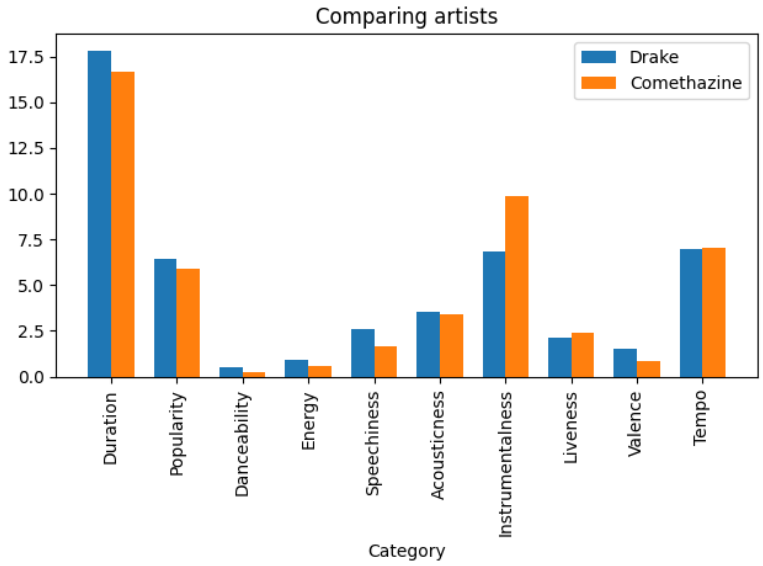
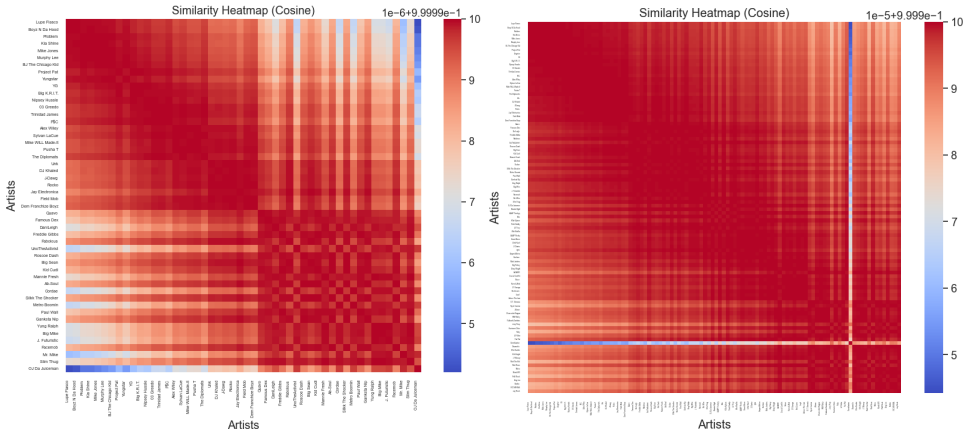


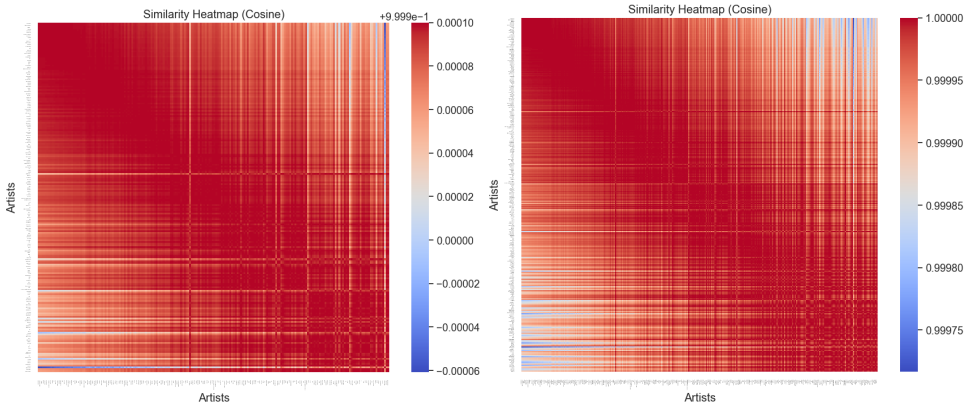
Figure 5: Compatring Drake and Comethazine.

Finally, we computed a heatmap showing the similarity between the artists in our dataset. We used the cosine similarity and obtained the following plots:



(a) Heatmap: 50 Artists.

(b) Heatmap: 100 Artists.



(c) Heatmap: 200 Artists.

(d) Heatmap: All Artists.

Figure 6: Heatmap Comparison

These heatmaps provide a visual representation of the similarity between the artists in our dataset. The color intensity in the heatmap indicates the level of similarity, with darker shades representing higher similarity.
We observe that the last artists in the dataset are generally less similar to the first ones. This implies that as we move towards the end of the dataset, there is a decrease in similarity between artists based on their audio features. However, it is important to note that overall, the dataset contains artists with relatively similar features.

2.4.2 Commenting on the Gephi visualizations.

Comparison between graphs gB and gD and their properties:

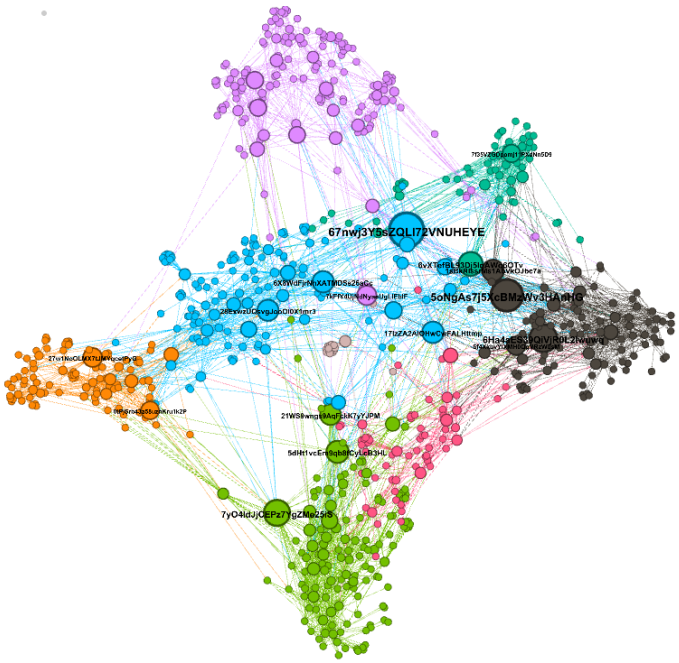


Figure 7: Graph gB

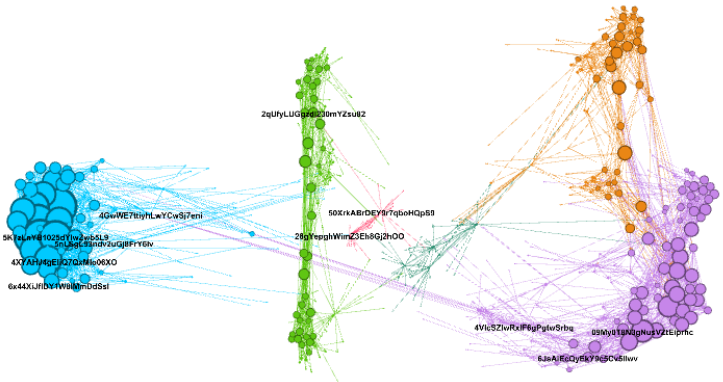


Figure 8: Graph gD

The previous graphs were generated using Gephi. In these graphs, the colors of the nodes represent the communities to which they belong, while the sizes of the nodes are proportional to their betweenness centrality. We have chosen to display only the ID of the artist with the highest betweenness centrality.

In the first graph visualization (gB), we detected that artists such as Wale, DJ Drama, MENSA and A$AP Mob have the highest betweeness centrality.This indicates that these

artists play a crucial role in connecting different parts of the network and facilitating the flow of information or interactions among other artists.
High betweenness centrality suggests that these artists have a significant influence on the communication and collaboration within the network. They act as important intermediaries or bridges between different groups or communities of artists, potentially serving as connectors or influencers in the music industry.

In the second graph (gD), we observed that the algorithm detected fewer communities compared to the first graph (gB). Interestingly, we noticed that most nodes with high betweenness centrality have been accumulated within one specific community, represented by the color blue.
This suggests that in the gD graph, there is a prominent community that acts as a central hub, with a high concentration of influential artists who play crucial roles in connecting and bridging different parts of the network. These artists, with their high betweenness centrality, are key figures in facilitating the flow of information and interactions within this community and potentially beyond.

3 Session 5: Personal Network Analysis Project

3.1 Objective of the Study

We want to determine how much the popularity value of an artist in spotify is determined by connection with other popular artists. Or maybe other patterns? We'll make some hypothesis and attempt to evaluate.

Hypotheses, collaborations between artists boost the popularity of one artist.

How do we aim to prove that? Get a graph where node values are the popularity score of an artist and edge weight the number of collaborations they have together.
Then measure correlation between edge weights and node values and also with centrality metrics.

3.2 Data Acquisition and Obtained Data

To obtain the data, we modified the original crawler from the Lab 1 but defined the following function, which given some artist looks at all of its songs and checks for collaborating artists. To do this we first had to get all albums and then from each album all the songs. Because doing this individually would have been an immense amount of API calls, notice how we tried to optimize the code by batching requests to have as less as possible.

We also gathered some extra information that we thought might be useful later on: duration of all songs of an artist accumulated, number of tracks with explicit language and finally number of tracks.

From this graph we calculated some metrics that thought useful in testing our hypothesis:

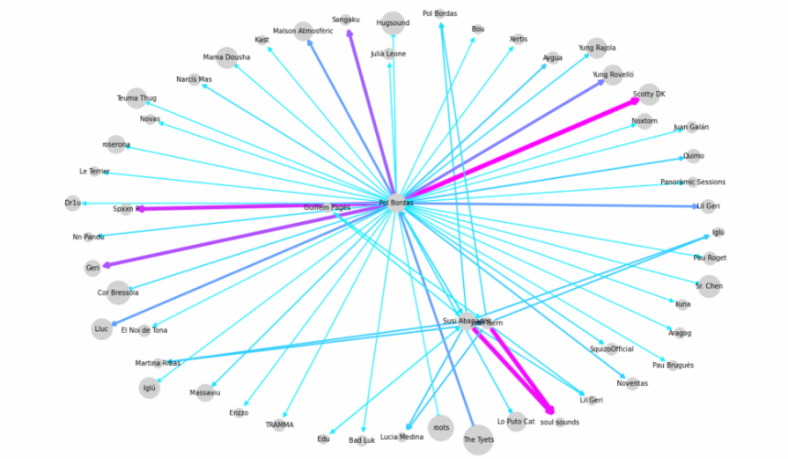


Figure 9: 5 nodes crawled graph

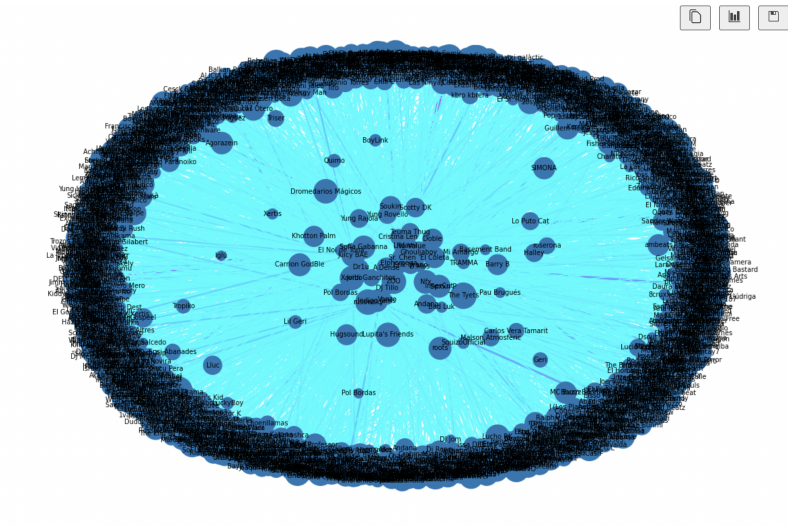


Figure 10: 100 nodes crawled graph

From this graph we calculated some metrics that thought useful in testing our hypothesis: in-degree,out-degree,closeness-centrality,betwennes-centrality,page-rank.In the following plot we can see these values plotted together with the spotify's popularity score.

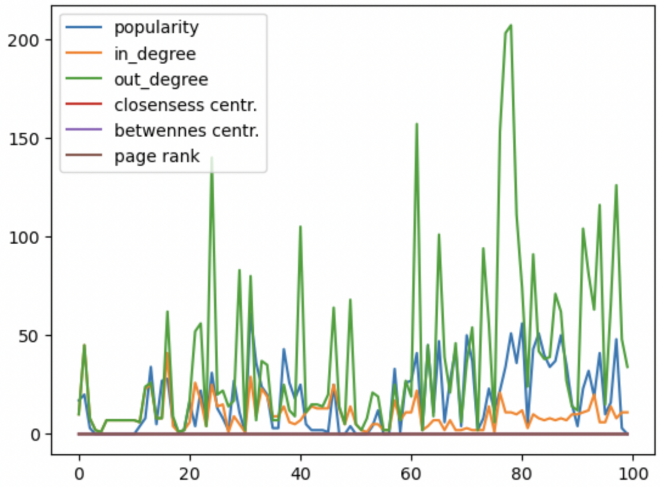


Figure 11: plot of node graph metrics

Here we have a dataframe with all info from all nodes in the graph:

3.2.1 API Endpoints:

These are the spotipy functions used:

```
sp.artist_albums(artist_id,limit=50,country="ES")
sp.album_tracks(album_id,limit=50)
sp.album_tracks(album_id,limit=50)
sp.artists(artists_batch)["artists"]
```

	n_follows	name	popularity	genres	durations	explicits	n_tracks	in_degree	out_degree	closeness_centr	betweenness_centr	page_rank
tEL7VkgfKQR	340	Susi Abanades	17	NaN	NaN	NaN	NaN	10	10	0.019544	0.004384	0.003159
'wOfzZmEqEl	986	Pol Bordas	20	'trap catala'	NaN	NaN	NaN	45	45	0.029631	0.013507	0.004817
vAJhAhhTpSX	20	Joan Isern	3	NaN	NaN	NaN	NaN	8	8	0.014088	0.000001	0.002709
gM1u4MQeak	0	Guillem Pagès	0	NaN	NaN	NaN	NaN	2	2	0.013834	0.000000	0.000557
C2N6eJCD4N	30	Edu	0	NaN	NaN	NaN	NaN	1	1	0.013792	0.000000	0.000493
...	...	...	...	...	...	...	...	...	...	...	...	...
/OBkcxlUaCu	2	Guillermo Cámbara	0	NaN	NaN	NaN	NaN	1	0	0.013864	0.000000	0.000431
PbfEFOb5jjt2	5	Cuellilargo	0	NaN	NaN	NaN	NaN	1	0	0.013864	0.000000	0.000431
EbTHyOmUvo	3972	Shawtyredd	1	NaN	NaN	NaN	NaN	1	0	0.013864	0.000000	0.000437
Jfjv1xNTd8m	293	Sleezy Hefe	6	NaN	NaN	NaN	NaN	1	0	0.013864	0.000000	0.000431
wpoWQLiUG	1	Zé gueretti	0	NaN	NaN	NaN	NaN	1	0	0.013864	0.000000	0.000431

Figure 12: dataframe with node data

3.3 Data Analysis

We calculated a cross correlation matrix with the features extracted plus the calculated from the graph and these were the results:

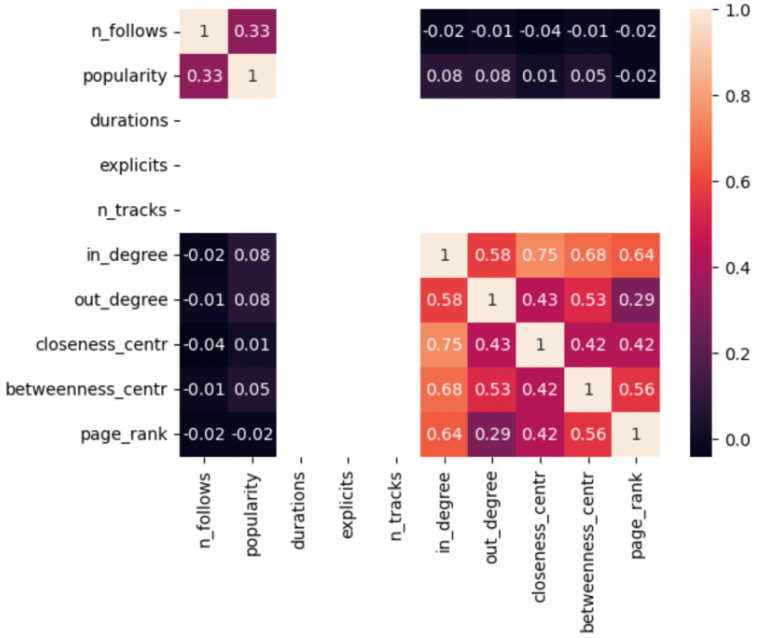


Figure 13: cross correlation matrix

3.4 Results and Findings

We basically discovered a correlation between number of followers and popularity score, which isn't really a surprise. We can also conclude that our hypotheses were false and so there wasn't at least (any linear) correlation between graph centrality metrics or indegree/outdegree and the popularity score.

Further work could be done attempting to make a model that predicts the popularity score based on the extracted data. Anyhow, we managed to answer the proposed hypotheses, there is no correlation between graph metrics and popularity score.