Comparison of Network Analytics and Significance Analysis on Spotify Artist Feature Collaboration Network

Learning From Networks - Final report

Fabio Cociancich, Luca Fantin, Alessandro Lincetto

Master Degree in Computer Engineering - University of Padova

I. Motivation

This project analyzes the artist collaboration network on Spotify to identify musical trends. The goal is to study artist popularity and their connections using centrality and clustering metrics. The statistical significance of the metrics calculated on the Spotify network is evaluated against random networks, for an in-depth understanding of the dynamics of artistic collaboration. The analysis was performed using the NetworkX and NetworKit packages. The results provide a basis for further research.

II. Dataset

For this project, we used the Spotify Artist Feature Collaboration Network from Kaggle [1]. This dataset consists of a graph where nodes correspond to artists and edges connect artists who have collaborated on at least one song. It has 156,422 nodes, which include around 20,000 artists who appear in the Spotify weekly charts and around 136,000 artists who had at least one feature with the chart artists, and 300,386 edges between them. Out of the information included with the nodes, the ones we used to analyze our results are the following:

- artist popularity, expressed as an integer number between 0 and 100 (100 corresponding to the most popular artist on the service), according to the Spotify API;
- list of genres, according to the Spotify API.

III. Measures

The measures considered for these analyses are both graph- and node-level graph metrics. At the graph level, we compute the number of connected components and the clustering coefficient of the graph, both global and average. At the node level, instead, we have the local clustering coefficients and a series of centrality measures. Alongside those presented during the lectures (degree, closeness, betweenness, PageRank), we also considered the *eigenvector centrality*, which is built on the intuition that a node is important if it is connected to other important nodes. Given a graph $G = (V, E)$, let us define $\mathbf{x} \in \mathbb{R}^{|V|}$ the vector of the centrality values for all nodes in G, A the adjacency matrix of G and $\lambda \neq 0$ a constant. For any node i we can write:

$$x_i = \frac{1}{\lambda} \sum_{j=1}^{|V|} A_{i,j} x_j \quad \rightarrow \quad Ax = \lambda x$$

The mathematical representation of the intuition can thus be reformulated as finding the eigenvector of the adjacency matrix corresponding to the eigenvalue λ; such vector includes the values of the eigenvector centrality for all nodes. This centrality measure has been studied extensively [2] [3] [4], also in the context of social media network analysis [5], including Spotify [6] [7].

IV. Significance analysis framework

The significance analysis we performed is composed by the statistical testing framework and the random graph model chosen. For the latter, we chose the *Holme-Kim model* [8]. The generation of a random graph starts from a number of nodes smaller than the desired graph size and no edges, iteratively adds new nodes and connects them with already existing ones with a distribution that favours nodes with an already high degree. The resulting graphs show a power-law distribution of the degrees: the probability of seeing a node with a certain degree decreases exponentially as the degree increases. Such characteristic is observed in many real-world networks and is captured by this model more accurately compared to the Erdős-Rényi model. Furthermore, this model produces graphs with tunable global clustering coefficients by creating additional edges: once a newly created node v is connected to an existing one w, a new edge is created between v and one of the neighbours of w with a certain probability.

The null hypothesis considered in this project states that the metric values computed on the real graph well conform to the distribution determined by the generated random graphs, which implies our Spotify dataset does not have any significant feature that can explain the values we compute. Our statistical testing procedures consists of two steps. Any statistical test assumes that the considered population has a known distribution, most often a Gaussian one. Because of the specificity of our random graph model, the first step computes how similar the distribution of the metric we are considering is compared to a Gaussian distribution, through a *normality test*. We used Shapiro-Wilk test [9], since it is considered the most powerful normality test available [10]. The second step then checks the validity of our null hypothesis by computing the probability that the random distribution of the metric generates a value greater or lower than the real value, also called p-value.

V. Code

All the code developed for this project, together with results files, dataset files and more, can be found in our GitHub repository [11]. The central script is `main.py`, which computes any combination of the measures presented in section III on various graphs, depending on the command line arguments provided. The script can work on the entire dataset, subgraphs taken from the dataset with respect to certain genres or popularity thresholds, and random graphs generated with the Holme-Kim model [8] with parameters specified through the command line arguments. The libraries NetworkX [12] and NetworKit [13] are used to represent the graphs, compute the metrics and generate the random graphs.

VI. Experimental setup

All computations on graphs have been performed on the CAPRI High-Performance Computing (HPC) cluster [14]. The hardware capabilities and the presence of the SLURM job scheduler system allowed us to perform heavy computations in a feasible time frame. This system features the following hardware:

- 16 Intel(R) Xeon(R) Gold 6130 @ 2.10GHz CPUs
- 6 TB DDR4 RAM
- 2 NVIDIA Tesla P100 16GB GPUs
- 40 TB of disk space

On the other side, the analysis of the data computed by the cluster has been performed on our local machines. They all employ AMD Ryzen 5/7 CPUs and RAMs ranging from 8 GBs and 24 GBs.

VII. Data analysis on whole dataset

We started this analysis by computing all available metrics on the whole dataset. All graph-level metric computations took around one minute and used between 240 and 300 MBs of RAM. We have similar values for the node-level metric computations, especially degree centrality, eigenvector centrality and local clustering coefficients, but there were some notable exceptions. Closeness and betweenness are the only centralities that require significant operations on graphs (computing distances and shortest paths), thus their execution times are considerably larger than the others: the former took more than 20 hours and the latter more than 4 days. On the other hand, computing the PageRank centralities used around 500 MBs of RAM. This is to be expected, as NetworkX computes it via the power method, which relies heavily on matrix computations.

The centrality measures were compared between each other and against their average by ranking the artists with respect to each measure, as shown in table I. We can see that different centrality definitions highlight different characteristics of different nodes. In the top 5 positions for degree and PageRank centralities we can find classical composers such as Bach and Sibelius, whose works have been performed by numerous orchestras; Traditional, which is a generic tag used on Spotify to mark traditional songs coming from the folklore of any culture in the world and thus do not have a specific author; Mc Gw and Mc MN, two Brazilian producers whose many connections are a combination of other songs sampling their work and working within specific genres known online [15]. In general, the importance given to these artists by these measures is not reflected in the real music world. Instead, closeness, betweenness and eigenvector centrality place at the top artists with a huge mainstream presence and success: rappers like such as Snoop Dogg and Gucci Mane and DJs/producers like David Guetta, Steve Aoki and Diplo. These artists are also those that have the highest average ranking across all centralities.

We have also created a visual representation of the distribution of centrality measures, represented by figure 1. The values for each measure have been sorted in descending order and scaled to 1. The resulting plot shows how most centrality measures have a distribution with many nodes having a low number of collaborations and a few nodes having a very high number. Closeness centrality is the only measure with a clearly different distribution, with most values concentrated withing a narrow range, suggesting that many artists are relatively close in the network, while a small portion are significantly distant from the rest of the graph.

Finally, we created a similar plot for the local clustering coefficients, reported in figure 2. We see a much more irregular distribution, with almost 8,000 artists with a coefficient equal to 1. These artists are involved in every possible triangle with their neighbours, suggesting their importance in their specific neighbourhood or connected component. If we further analyze this set of artists, we discover a mean populartiy value of around 23.5 with a standard deviation of around 15, and no genres indicated for more than 5,000 of them. Thus, these artists are generally less popular, less likely to have large neighbourhoods and more likely to be involved in all possible triangles with their neighbourhood.

VIII. Data analysis on subgraphs

The analysis of the node-level metrics on subgraphs about several genres reveals that there are significant variability among various types of music. First, we plotted the distributions of these measures through boxplots, some of which are displayed in figure 3. Betweenness shows a non-uniform distribution; some genres have concentrated values, suggesting a more homogeneous "bridge" role, while others are more dispersed. Similarly, closeness does not exhibit a uniform distribution, with some genres being closer to each other and others more distant, indicating heterogeneity in proximity between genres. The clustering coefficients generally shows high values, but with a wide distribution, highlighting that while there is a tendency to form clusters, it is not uniform. The degree centrality presents a heterogeneous distribution: some genres have more connections, while others have fewer. The eigenvector shows a variable distribution, with some genres more concentrated at specific values. PageRank, on the other hand, displays a more concentrated distribution, with values less dispersed compared to the other metrics. In general, closeness centrality values appear to have a much wider spread across most genres compared to all other metrics.

We also computed the correlation matrix between all node-level metrics across all considered subgraphs, reported in figure 4. It reveals that some metrics, such as betweenness, degree, PageRank, and eigenvector, are strongly correlated with each other, while the clustering coefficient appears less correlated, suggesting that the tendency to form clusters depends on distinct factors.

IX. Data analysis on random graphs

For the significance analysis we considered the real graph and a selection of subgraphs. Because of the NetworkX implementation of the Holme-Kim, these graphs had to have more edges than nodes and a global clustering coefficient lower than 0.3 to be accurately resembled by the random graphs. For each of these graphs, 200 random graphs were generated, with the same number of nodes, a similar number of edges and a comparable global clustering coefficient. The last part was easy for graphs with a coefficient under 0.1, but reaching values above that threshold became almost impossible. On these generated graphs, we computed all available graph-level metrics and the maximum and average value of the closeness and eigenvector centralities, in order to represent both distributions of the node rankings with respect to centrality values. The closeness centrality was not computed for the larger graphs, corresponding to the entire dataset and the pop genre subgraph, due to time constraints.

A. *Normality test*

For each set of random graphs generated, all values of the metrics were tested against a Gaussian distribution using the Shapiro-Wilk test. The result, reported in table II, was that most of the distributions could be assimilated to a Gaussian one. For the other ones, plotting the histogram of the values revealed a distribution graphically resembling a Gaussian curve. Some examples of such histograms are reported in figure 5. Thus, we decided to take into consideration all metrics for all sets of graphs for the next phase of statistical testing.

B. *p-value computation*

For each metric listed in this section, we took its value from the real (sub)graph and computed the probabilities that the Gaussian distribution corresponding to the random graphs give a value higher or lower than the real one. The results, reported in table III, show that we cannot accept our null hypothesis, thus we can say that our (sub)graphs have particular features that affect the metric values in some significant way. In fact, for almost all graphs, if the real ones were generated by the same distribution as the random graphs, we should expect to see higher maximum and average values for the considered centralities, a higher average clustering coefficient and a lower global clustering coefficient. The latter results confirms the difficulties in replicating the value computed on the real graphs through the random graph model. The results for the other metrics could be due to the more fragmented nature of the real graph, which has a large number of connected components and less edges.

X. Future work

Our analysis on the whole dataset and its genre subgraphs is mainly concerned with the node-level metrics. Future extensions of our work could be comparing the graph-level metrics of these graphs to study how network dynamic change within the whole network between different genres.

Another open field is using the other features included in the dataset, like the number of followers and the number of chart hits of each artist, for the same analyses presented in this report. This also includes using the popularity level more extensively than what has been reported here.

Contributions

Out of the work presented in this report, Luca Fantin wrote the code for creating and analyzing the random graphs and refactored the code into the final version. Fabio Cociancich wrote the scripts to create subgraphs based on genre, artists, popularity threshold and percentages of top popular artists, to compute the metrics of a graph, he analyzed them and saved the results in the csv files. Alessandro Lincetto wrote the first scripts for generating random graphs with the NetworkX methods, creating graphs from the CSV files of the dataset and computing the graphs metrics; he also performed the metric computations for some subgraphs. Each member wrote the report section related to their work. The percentage of work done can thus be estimated to be 40%, 30%, 30% respectively.

#	*1*	*2*	*3*	*4*	*5*
Degree	Johann Sebastian Bach	Traditional	Mc Gw	MC MN	Jean Sibelius
Closeness	R3HAB	Snoop Dogg	Diplo	David Guetta	Tiësto
Betweenness	Snoop Dogg	Traditional	R3HAB	Diplo	Johann Sebastian Bach
PageRank	Johann Sebastian Bach	Traditional	Jean Sibelius	Mc Gw	MC MN
Eigenvector	Farruko	French Montana	Gucci Mane	Ty Dolla $ign	Lil Wayne
Average rank	Snoop Dogg (5.8)	Gucci Mane (11.6)	David Guetta (18.8)	Steve Aoki (19.0)	Diplo (20.8)

TABLE I: Top 5 artists in the whole dataset, according to our centrality measures. The "average rank" row also reports the value for each artist.

Reference graph	*Average cc*	*Global cc*	*Approximate global cc*	*Maximum eigenvector*	*Average eigenvector*	*Maximum closeness*	*Average closeness*
House subgraph	✗	✓	✓	✓	✓	✓	✓
Pop subgraph	✗	✗	✗	✓	✓		
Rap subgraph	✗	✗	✗	✓	✓	✓	✗
Whole dataset	✗	✗	✗	✓	✓		
Top 10% popularity subgraph	✗	✗	✗	✓	✓	✗	✗
Trap subgraph	✗	✗	✗	✓	✓	✓	✗

TABLE II: Results of the Shapiro-Wilk normality tests for all considered graphs. The "reference graph" is the graph to which the random graphs used in the analysis refer to. "cc" stands for "clustering coefficient".

Reference graph	*Average cc*	*Global cc*	*Approximate global cc*	*Maximum eigenvector*	*Average eigenvector*	*Maximum closeness*	*Average closeness*
House subgraph	>	<	<	>	>	>	>
Pop subgraph	>	<	<	>	>		
Rap subgraph	>	<	<	>	>	>	>
Whole dataset	>	<	<	>	<		
Top 10% popularity subgraph	>	<	<	>	>	>	>
Trap subgraph	>	<	<	>	>	>	>

TABLE III: Results of the p-value computations: the contents of the cells represent how we should expect the metric values to be, compared to the values computed on the real (sub)graphs, if we were to accept our null hypothesis. "cc" stands for "clustering coefficient".

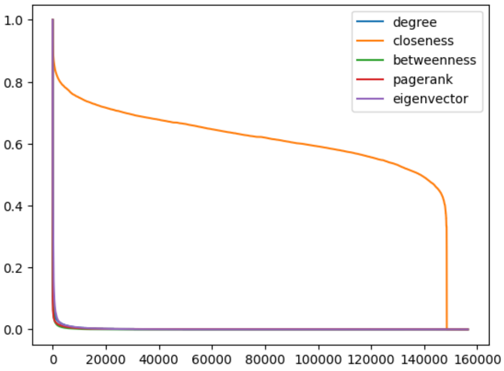


Fig. 1: Distribution of the centrality measures on the whole dataset. All values have been sorted in descending order and normalized.

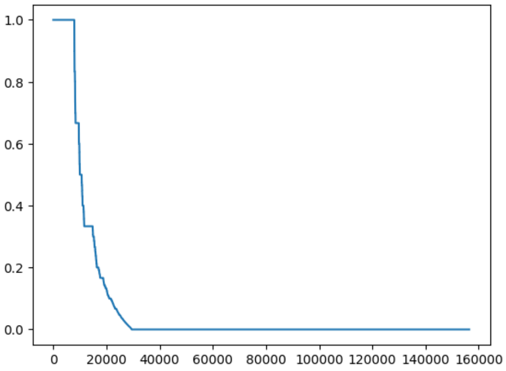
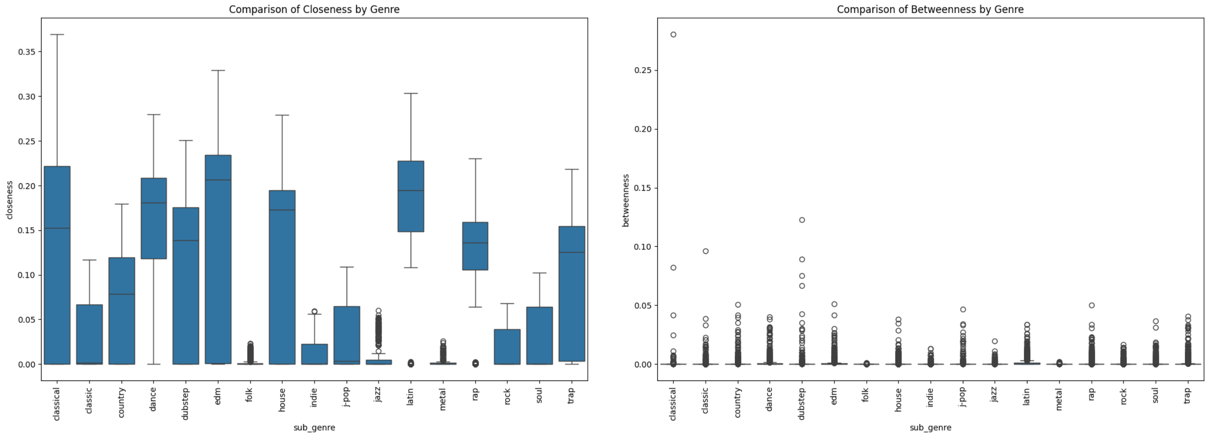


Fig. 2: Distribution of the local clustering coefficients on the whole dataset. All values have been sorted in descending order.



(a) Closeness centrality

(b) Betweenness centrality

Fig. 3: Boxplots of the distribution of some centrality measures on several genre subgraphs.

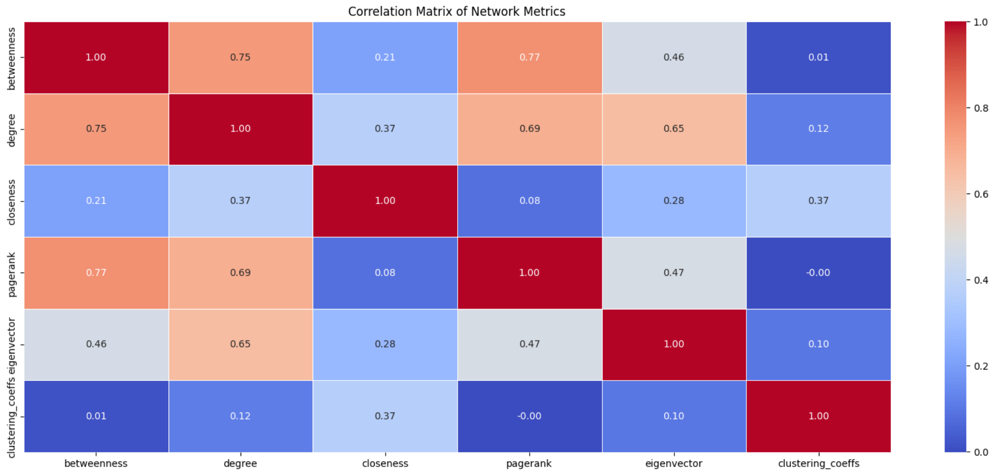
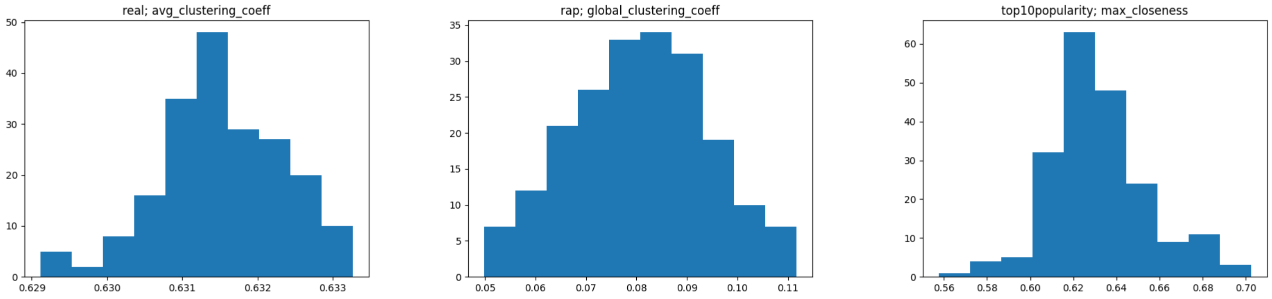


Fig. 4: Correlation matrix between all node-level metrics computed on all genre subgraphs.



(a) Whole dataset, average clustering coefficient

(b) Rap subgraph, global clustering coefficient

(c) Top 10% popularity subgraph, maximum closeness

Fig. 5: Example of histograms for the metrics computed on random graphs that do not have a Gaussian distribution according to the Shapiro-Wilk test.

References

[1] Julian Freyberg. *Spotify Artist Feature Collaboration Network*. URL: https://www.kaggle.com/datasets/jfreyberg/spotify-artist-feature-collaboration-network.

[2] Phillip Bonacich. “Some unique properties of eigenvector centrality”. In: *Social Networks* 29.4 (2007), pp. 555–564. ISSN: 0378-8733. DOI: https://doi.org/10.1016/j.socnet.2007.04.002.

[3] Stephen Borgatti, Kathleen Carley, and David Krackhardt. “On the Robustness of Centrality Measures Under Conditions of Imperfect Data”. In: *Social Networks* 28 (May 2006), pp. 124–136. DOI: 10.1016/j.socnet.2005.05.001.

[4] Leo Spizzirri. “Justification and application of eigenvector centrality”. In: *Algebra in Geography: Eigenvectors of Network* (2011). URL: https://sites.math.washington.edu/~morrow/336_11/papers/leo.pdf.

[5] Warih Maharani, Adiwijaya, and Alfian Akbar Gozali. “Degree centrality and eigenvector centrality in twitter”. In: *2014 8th International Conference on Telecommunication Systems Services and Applications (TSSA)*. 2014, pp. 1–5. DOI: 10.1109/TSSA.2014.7065911.

[6] Tobin South, Matthew Roughan, and Lewis Mitchell. “Popularity and centrality in Spotify networks: critical transitions in eigenvector centrality”. In: *Journal of Complex Networks* 8.6 (Mar. 2021), cnaa050. ISSN: 2051-1329. DOI: 10.1093/comnet/cnaa050. URL: https://doi.org/10.1093/comnet/cnaa050.

[7] Tobin South. “Network analysis of the Spotify artist collaboration graph”. In: *Australian Mathematical Sciences Institute* (2018), pp. 1–12. URL: https://vrs.amsi.org.au/wp-content/uploads/sites/84/2018/04/tobin_south_vrs-report.pdf.

[8] Petter Holme and Beom Jun Kim. “Growing scale-free networks with tunable clustering”. In: *Physical Review E* 65.2 (Jan. 2002). ISSN: 1095-3787. DOI: 10.1103/physreve.65.026107. URL: http://dx.doi.org/10.1103/PhysRevE.65.026107.

[9] S. S. Shapiro and M. B. Wilk. “An Analysis of Variance Test for Normality (Complete Samples)”. In: *Biometrika* 52.3/4 (1965), pp. 591–611. ISSN: 00063444, 14643510. URL: http://www.jstor.org/stable/2333709 (visited on 12/17/2024).

[10] Nornadiah Mohd Razali and Bee Yap. “Power Comparisons of Shapiro-Wilk, Kolmogorov-Smirnov, Lilliefors and Anderson-Darling Tests”. In: *J. Stat. Model. Analytics* 2 (Jan. 2011).

[11] Fabio Cociancich, Luca Fantin, and Alessandro Lincetto. *lfn_project*. URL: https://github.com/fantinluca/lfn_project/.

[12] *NetworkX documentation*. URL: https://networkx.org/.

[13] *NetworKit*. URL: https://networkit.github.io/.

[14] University of Padova Strategic Research Infrastructure Grant 2017. *CAPRI: Calcolo ad Alte Prestazioni per la Ricerca e l’Innovazione*. URL: https://capri.dei.unipd.it/.

[15] Billboard. *Mc Gw Is on 3,000 Songs Already This Year — And He’s Not Slowing Down*. 2024. URL: https://www.billboard.com/music/features/mc-gw-interview-1235800921/.