Spotify Musician Collaboration Network Analysis

Terry Zhang

1 Introduction

"No man is an island" was coined by Elizabethan era poet John Donne. This truism about the nature of humanity being an interconnected one is the core of human progress. Collaborative relationships have existed for as long as our species has, yet only recently could we begin to understand and study them. An interconnected network of collaborations exists in many forms, such as scientific citation networks (Newman, 2001b), international trade networks (Xu & Cheng, 2016), and musician collaboration networks (Topirceanu, Barina, & Udrescu, 2014; Budner & Grahl, 2016). One of the earliest studies of collaborative networks is the Erdős number (Grossman & Ion, 1995) which quantified the number of collaborators and separation a researcher has to the influential mathematician Paul Erdős. Such a project demonstrates an early example of desire to better understand the networked nature of collaborations.

This paper will specifically focus on the collaborations between musical artists from a statistical perspective. Given the lack of readily available data, a Python script was developed to easily construct a weighted network of musical collaborations. Using the popular music streaming service Spotify's API, novel weighted networks of artist collaborations are generated by snowball sampling (see appendix A). The sampling method extracted collaboration networks for two distinct genres of music, rap/hip hop and electronic music, for analysis.

2 Background

Collaborative networks have been studied in several forms. The most notable examples are the scientific collaboration networks (Newman, 2001b; Barrat, Barthelemy, Pastor-Satorras, & Vespignani, 2004; Ebadi & Schiffauerova, 2015) and movie actor collaborations using the International Movie Database (IMDB) dataset (Newman, 2001a; Topirceanu et al., 2014). The study of novel networks reveals beneficial findings for better understanding the dynamics of society as a whole. For example, the scientific collaboration network confirms Milgram's small world hypothesis where the relative shortest path within a network is small relative to the network size (Newman, 2001a). Only recently have collaboration networks in a musical context been studied (Topirceanu et al., 2014; Budner & Grahl, 2016; Oliveira, Silva, Seufitelli, Lacerda, & Moro, 2020). This is likely due to the availability of network data on musical collaborations until recent years.

Several characteristics of collaborative networks have been identified by previous studies. In the scientific collaboration network, the degree distribution of a number of collaborators, i.e. degree, appears to follow a power-law with exponential cut-off distribution (Newman, 2001c). The power-law distribution of degree was first identified as an alternative hypotheses against the Erdős–Rényi random network by Barabasi (Barabási & Albert, 1999). Most importantly, the scale-free network appears to be more common in real networks, but far from universal (Amaral & Scala, n.d.; Holme, 2019). For example, in the movie actor collaborations network, the degree distribution of actors appears to follow a Gaussian decay which indicates a lack of scale-freeness (Amaral & Scala, n.d.). The scale-free property is said to be a result of preferential attachment, where well connected nodes are more likely to receive a new edge. A weighted driven attachment theory for weighted networks was suggested by Barrat, Barthélemy, and Vespignani (2004). For weighted networks, a new tie of certain weight has a probability of an impact on the existing weights in the network. The resulting behaviour of this hypothesis is a scale-free behaviour of the node strength distribution. It is thus important to establish the topological features against these hypotheses for novel networks such as the one in this study. To that end, the research questions of this paper are:

RQ1: What is the underlying distribution of these novel networks? Specifically in terms of node degree and strength.

RQ2: What are the differences between the two genre subnetworks?

3 Data

The data used in this study focuses on two contrasting genres of music, rap and hip hop (henceforth rap) and electronic. The chosen data source for this study is the popular music streaming service Spotify. Using Spotify's application programming interface (API), structured data objects for albums, artists, and tracks were extracted (Spotify, n.d.). Collaborative relationships are identified as co-appearing artists on track credits. In addition to collaborative relationships, artist characteristics of number of followers and Spotify's popularity measure is also extracted.

As with most networks in reality, the full size of the collaborative network is likely unquantifiable and infeasible for analysis and a subnetwork through sampling is necessary. Snowball sampling arose as an ideal technique given the data structure returned by the API. Snowball sampling has been used in both social and biological researches to generate subnetworks feasible for analysis (Bernard et al., 2010; Elliott, Leicht, Whitmore, Reinert, & Reed-Tsochas, 2018).

The intuition for snowball sampling is that the Spotify database holds all collaborative relationships between artists in their track information. By walking through the relationships, we can extract a subnetwork of collaborators. Snowball sampling was implemented as a Python script taking an input seed node as an artist's unique identifier. The script begins by extracting collaborative relationships using co-appearing artists on each of the seed artist's tracks and adding all collaborators to a queue of nodes to visit next. Each pair of artists appearing on a track together represents one instance of collaboration. In networks terms, each artist is a node and a collaborative instance is an edge. The script will snowball sample the database within the predetermined number of hops.

A hop limit of 200 direct visits was set in consideration of the large potential network of collaborators returned and the associated computation time of analysis algorithms. After cleaning the data, the script extracted collaborative relationships of $N_R = 3615$ rap and $N_E = 1449$ electronic artists as nodes with $E_R = 11005$ for rap and $E_E = 1821$ for electronic collaborative relationships as edges. Edge weights were calculated as the total number of collaborative instances between two artists.

4 Results

Table 1 provides the summary statistics of the rap and electronic music collaborations network. The statistics and their implications are discussed in this section.

4.1 Degree

Degree is conventionally used as a basic measure of a network's topology (Opsahl, Agneessens, & Skvoretz, 2010). The degree k_i of a node i is the number of links adjacent to it (Freeman, 1978). In terms of collaborations, the degree is the number of collaborators an artist has worked with. This measure only considers binary relationships - whether a collaboration is present or not.

The summary statistics for average degree and strength presented in Table 1 shows a marked difference between the two genres' collaborative networks. On average, an artist in the rap network has about 6 collaborators and 2.5 collaborators in

Table 1: Summary statistics of the rap and electronic collaboration networks.

	Rap	Electronic
total artists (N)	3615	1449
total collaborations (E)	11005	1821
average followers	359865.9	136926.8
average popularity	30.7	28.6
avgerage degree	6.1	2.5
average strength	11.3	4.3
size of giant component	3461	990
as a percentage	95.7%	68.3%
clustering	0.5	0.4
mean distance	3	4
mean weighted distance	4.5	6.6
max. distance	7	12

the electronic network. This conveys that artists within the rap network collaborate at a considerably higher level than within the electronic community.

Social networks, such as this collaboration network, have been proposed as having a scale-free distribution of degrees (Barabási & Albert, 1999). A scale-free network has a degree distribution that follows a power law:

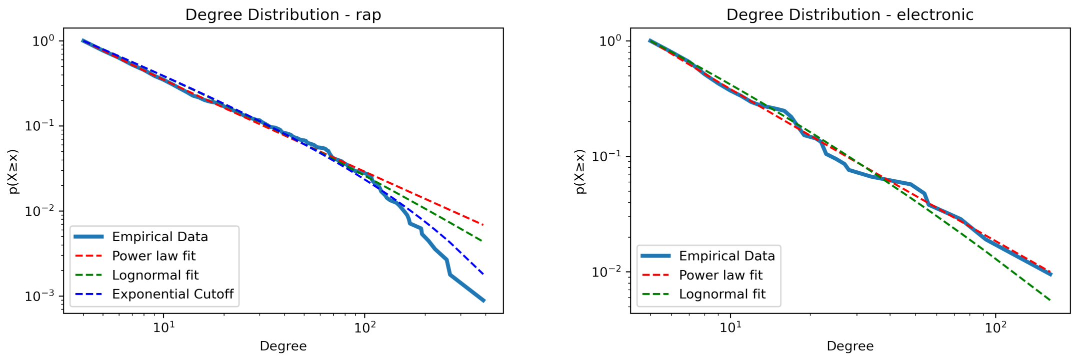
$$P(k_i) \approx k^{-\alpha} \tag{1}$$

with α being a parameter known as the *scaling parameter* and often lies in the range $2 < \alpha < 3$ (Clauset, Shalizi, & Newman, 2009).

To test the scale-free network hypothesis, I applied three different approaches. Beginning with a visual analysis of the complementary cumulative distribution function (CCDF) on a log-log plot as suggested by Newman (2005). Following, I tested the scale-free hypothesis non-parametrically using Monte Carlo simulation against the Barabasi-Albert model as proposed by Elliott et al. (2018). Lastly, the hypothesis was tested statistically using goodness-of-fit and likelihood ratio tests implemented in the Python PowerLaw package by Clauset et al. (2009). The last method will also provide an estimate for the tuning parameter α for each network.

4.1.1 Degree Distribution

The degree distribution of the electronic and rap music collaborative networks is presented in figure 1. These distributions were plotted on log-log plots along with potential fits. A straight line on the log-log plot suggests power law, however, such a conclusion cannot be drawn so simply. There are linear decay patterns that appear



(a) Rap (b) Electronic

Figure 1: Degree distributions of rap and electronic collaborative networks with theoretical fits. The distributions are plotted as CCDF on log-log scale. Rap appears to follow a truncated power-law fit whereas electronic follows power law closely.

visually similar to power law and may mislead conclusions (Clauset et al., 2009). By plotting the theoretical fits of distributions, the visual inspection of power-law behaviour is less dubious.

Upon visual inspection of the rap network, there appears to be a close fit with power-law until 10^2 degree. After which, the distribution begins decaying at a rate greater than an exponential cut-off. Most interestingly are that the decay does not appear to follow the curvature present in exponential cut-off and rather appears to decay linearly. Figure 2 presents a closer inspection of the decaying tail and it appears to follow the power-law fit well. The electronic network appears to follow a power-law distribution strongly. The linearly decaying tail is not present as the electronic network decays uniformly throughout the degrees.

4.1.2 Degree Power Law Test

The statistical analyses for the presence of power-law applied in this study were proposed by Clauset et al. (2009). The parameter α of power-law fit was estimated by MLE. However, MLE alone cannot convey the appropriateness of a power-law fit. Therefore, the goodness-of-fit must be evaluated. The Kolmogorov-Smirnov (KS) test was carried out to measure the goodness-of-fit of a power-law distribution on the empirical degree distribution. KS statistics provides the distance between an empirical distribution function and a theoretical distribution function. To obtain a p-value for a power-law fit, Clauset et al. proposed a Monte Carlo process measuring the empirical KS statistics against those of randomly generated dataset. Using this process, the empirical KS statistics were evaluated against the KS statistics of 1,000 randomly generated synthetic data. The proportion of simulated KS statistics

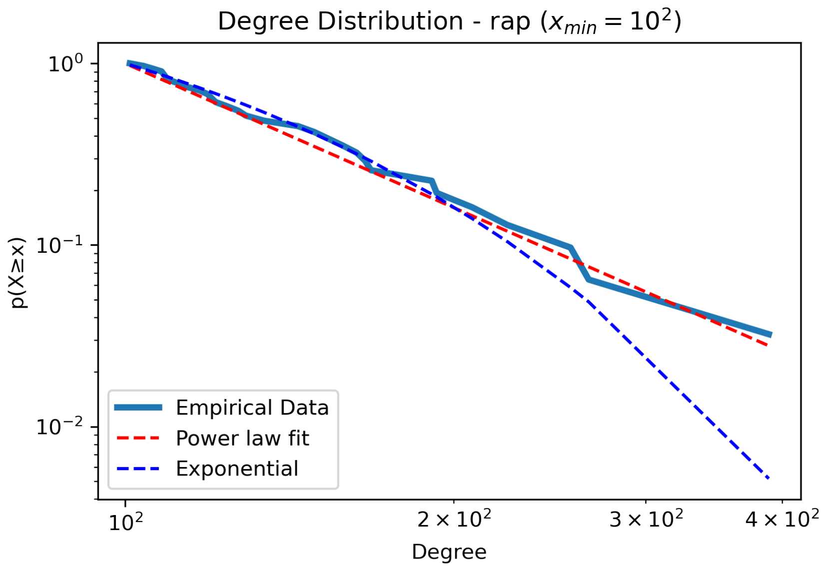


Figure 2: Degree distribution of the rap collaborative network with a defined x_{min}. The x_{min} was set at the start of the decaying tail of the full distribution in fig 1.

Table 2: Statistical test results of power law behaviour for degree and strength distribution. The $\hat{\alpha}$ value has the standard deviation of the decimal places inside the brackets. Power law p-value represents the results of the Monte Carlo Kolmogorov-Smirnov goodness-of-fit test for the given distribution and power law as proposed by Clauset et al. (2009). Statistically significant results are bolded.

			power law	log-normal		exponential		cut-off		
		$\hat{\alpha}$	p	LR	p	LR	p	LR	p	support
k_i	rap	2.06(3)	**0.83**	-2.07	0.11	436.98	**0.00**	-5.34	**0.00**	with cut-off
	rap$_{tail}$	-	**0.56**	-0.50	0.51	-0.27	0.84	-0.54	0.30	moderate
	elec	2.29(13)	**0.79**	-0.36	0.58	24.85	**0.03**	-0.47	0.33	moderate
s_i	rap	1.75(2)	0.00	-26.35	**0.00**	1639.13	**0.00**	-28.89	**0.00**	none
	elec	2.09(4)	0.03	-2.25	0.17	382.88	**0.00**	-2.50	**0.03**	none

greater than the observed KS statics is reported as the p-value in table 2. Given that a $p > 0.1$ is considered significant for this test, power-law remains an appropriate fit for both collaborative networks and the tail of the rap network.

While the KS Monte Carlo test and p-values begin to provide an indication for the goodness-of-fit, there are other possible distributions that may be of a better fit (Clauset et al., 2009). To rule out such potential and further solidify the power-law fit, the likelihood ratio (LR) is calculated for three other theoretical distributions (Table 2). LR statistics calculates the goodness-of-fit between two competing distributions. The sign of the LR shows which distribution is favoured. In this study, a negative sign favours the alternative distribution and positive for power law. For LR test, $p < 0.1$ is considered significant. The results from these tests see the exponential distribution ruled out for both the rap and electronic network. The LR ratio results of the rap network also confirm the presence of a cut-off as seen visually in figure 1. Overall, the power-law can be deemed to be an appropriate generalization of the degree distributions of the rap and electronic music collaboration network.

4.2 Strength

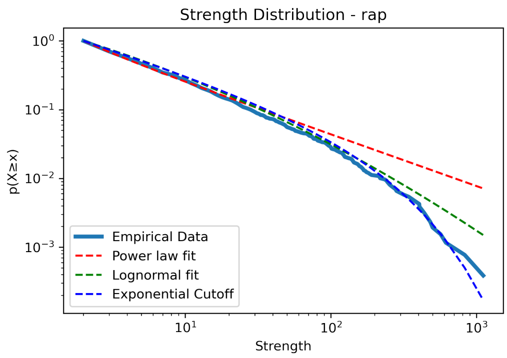
The measure of degree was extended to a weighted degree measure by (Newman, 2001a; Barrat, Barthélemy, & Vespignani, 2004) weighted networks, known as strength. It is the sum of all edge weights adjacent to node i:

$$S_i = \sum_{j \in \Pi(i)} w_{ij} \tag{2}$$

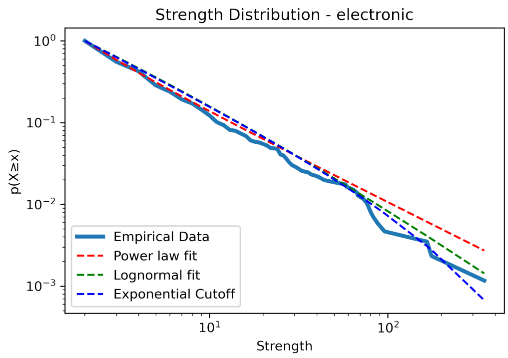
In the context of this study, strength can be understood as an artist's total collaborative instances. The average strength of nodes in the rap network is 11.3 and 4.3 in the electronic network table 1.

4.2.1 Strength Distribution

The strength distribution of the two networks does not clearly follow a power law. The visual comparison of the rap network's empirical CCDF seems to reveal a distribution that follows low normal initially and then shifting into an exponential decay. The distribution of the electronic network does not appear to readily follow the two potential theoretical distributions. The visual inspection of the distributions is inconclusive for power law.



(a) Rap



(b) Electronic

Figure 3: Strength distributions of rap and electronic collaborative networks with theoretical fits. The distributions are plotted as CCDF on log-log scale. Rap appears to follow a log-normal distribution and becoming a exponential cut off. Electronic strength distribution does not appear to follow a clear theoretical distribution.

4.2.2 Strength Power Law Test

To determine whether the network is scale-free in strength distribution, the methods described in the previous section are again carried out. The results for the statistical tests are presented in table 2. The p-value for the Monte Carlo goodness-of-fit test of power-law on both distributions fails to support the power law as an appropriate fit. Testing alternative distributions for the rap network provides evidence supporting the log-normal and exponential cut-off distribution. These statistics corroborate with the visual inspection of the distribution. The alternative distribution results for the electronic network appear to provide some evidence for the power-law with exponential cut-off distribution. However, given the lack of support for a power law, this result is likely untrustworthy.

4.3 Betweenness Centrality

The betweenness centrality (BC) is a popular measure for indexing the nodes within a network. BC was formalized by Freeman as the number of shortest paths that pass through a node (1978). A node of high centrality has many shortest paths running through it, which gives it *control* over the flow of information in the network. However, this measure in its naive form does not generalize well for weighted networks. The calculation of betweenness relies on finding the shortest path between the nodes. The shortest path algorithm, in its naive form, minimizes the number of nodes visited and holds the assumption that each node visit is equally costly. Thus, calculating BC using the naive approach overlooks the important edge

Table 3: Ranking of rap and electronic artists based on betweenness centrality. The shortest path definition by Opsahl et al. was employed with varying levels of α tested.

	rank	$\alpha = 0$	$\alpha = 0.5$	$\alpha = 1$	$\alpha = 1.5$
Rap	1	Gucci Mane (0.122)	Gucci Mane (0.243)	Gucci Mane (0.334)	Gucci Mane (0.358)
	2	Trae Tha Truth (0.08)	Rick Ross (0.165)	Rick Ross (0.197)	Rick Ross (0.25)
	3	Waka Flocka Flame (0.077)	Waka Flocka Flame (0.097)	Lil Wayne (0.173)	Lil Wayne (0.217)
	4	Snoop Dogg (0.071)	Snoop Dogg (0.092)	Waka Flocka Flame (0.133)	Snoop Dogg (0.191)
	5	Rick Ross (0.061)	Lil Wayne (0.092)	French Montana (0.125)	Waka Flocka Flame (0.173)
	6	Too $hort (0.061)	French Montana (0.087)	Snoop Dogg (0.123)	French Montana (0.16)
	7	T.I. (0.051)	Trae Tha Truth (0.085)	Birdman (0.101)	Birdman (0.144)
	8	French Montana (0.046)	Too $hort (0.06)	Wiz Khalifa (0.087)	Wiz Khalifa (0.133)
	9	Boosie Badazz (0.044)	Zaytoven (0.058)	Zaytoven (0.085)	Jeezy (0.124)
	10	The Game (0.041)	T.I. (0.057)	Trae Tha Truth (0.082)	T.I. (0.117)
Elec.	1	deadmau5 (0.231)	Kaskade (0.242)	Kaskade (0.249)	Kaskade (0.295)
	2	Kaskade (0.217)	deadmau5 (0.23)	deadmau5 (0.231)	deadmau5 (0.232)
	3	Tiësto (0.163)	Tiësto (0.178)	Tiësto (0.204)	Diplo (0.21)
	4	Mr. Bill (0.082)	Mr. Bill (0.083)	Diplo (0.089)	Tiësto (0.206)
	5	Diplo (0.069)	Diplo (0.078)	Mr. Bill (0.086)	Skrillex (0.205)
	6	JES (0.046)	Skrillex (0.048)	Skrillex (0.07)	Mr. Bill (0.09)
	7	Dillon Francis (0.04)	Tinlicker (0.042)	Tinlicker (0.043)	Tinlicker (0.043)
	8	Tinlicker (0.04)	JES (0.041)	Allure (0.042)	Allure (0.042)
	9	Wolfgang Gartner (0.031)	EDX (0.038)	JES (0.042)	JES (0.042)
	10	Chris Lorenzo (0.031)	Wolfgang Gartner (0.032)	EDX (0.04)	EDX (0.04)

characteristics.

Dijkstra's algorithm seeks to overcome the shortcomings of the naive shortest path calculation by including weighted edges into its calculation (Opsahl et al., 2010). As edge weight often represents strengths, Dijkstra proposes including their inverse as costs. Dijkstra's algorithm for shortest path is:

$$d^w(i,j) = \min\left(\frac{1}{w_{ih}} + \ldots + \frac{1}{w_{hj}}\right) \tag{3}$$

where d^w is the shortest path based on costs between node i and j. Dijkstra's algorithm, however, does not consider the number of intermediary nodes as a cost.

Dijkstra's definition of shortest path algorithm is most recently expanded by Opsahl et al. (2010) to include a tuning parameter which introduces the ability to account for intermediary nodes as a cost. Opsahl et al. defines the shortest path algorithm as:

$$d^{w\alpha}(i,j) = \min\left(\frac{1}{(w_{ih})^{\alpha}} + \ldots + \frac{1}{(w_{hj})^{\alpha}}\right) \tag{4}$$

where α is a positive tuning parameter that controls the algorithm's cost function between edge weights and the number of intermediary nodes. Using this definition of the shortest path for BC calculation, I tested varying levels of α to identify any noteworthy patterns. When $\alpha < 1$ shorter paths of the least cost are favoured. With $\alpha > 1$, stronger ties are favoured and the number of nodes is neglected.

In the context of the artist collaborative network, edge weights are seen as the strength of collaboration between artists and the number of intermediary nodes is seen as the number of collaborators an artist has had collaborative relationships with. In table 3, artists are ranked by BC calculated with varying values of α. Within the rap network, Gucci Mane ($k_i = 389$, $s_i = 1112$) is the most central artist across all values of α indicating a high number of collaborators and strong ties overall. Conversely, the ranking of Trae Tha Truth ($k_i = 256$, $s_i = 829$) drops steadily as α increases indicating a great number of collaborators but few of those are of great strength. Several artists such as Waka Flocka Flame ($k_i = 266$, $s_i = 566$) and Snoop Dogg ($k_i = 224$, $s_i = 482$) remain relatively stable through the αs indicating a comparable number of collaborators and collaborative strengths. Interestingly, Lil Wayne ($k_i = 168$, $s_i = 499$) rises considerably into the top 10 from outside of the ranking when $\alpha > 0$. Upon checking the underlying data, he has considerably fewer collaborators but much stronger ties with them.

The fluctuation of ranking by BC at varying values of α in the electronic network is less apparent at $\alpha > 0$. The top artists remained relatively stable. deadmau5 ($k_i = 56$, $s_i = 96$) and Kaskade ($k_i = 92$, $s_i = 176$) switched positions as α grew greater than 0. Seeing that deadmau5 had a higher position at $\alpha = 0$, this can mean he is connected to artists who are also of high centrality giving him an advantage in forming shortest paths. Kaskade rose because of his relatively stronger collaborative strength. Diplo ($k_i = 74$, $s_i = 166$) rises in ranks as more emphasis is placed on tie strength given his higher collaborative ties.

5 Discussion and Conclusion

This paper implemented a novel approach to apply snowball sampling of the underlying collaborative network of musical artists using Spotify's API. This method simplifies investigations into a collaborative network that has received considerably lesser attention. The collaborative network for rap and electronic genre was extracted for analysis. The network summary statistics of the two networks in table 1shows a marked difference between the two networks. The rap network on average appears to be a more popular genre with considerably more artists and followers. From a network perspective, the rap network also appears to be better connected with shorter average paths and higher degrees. Examining the two networks from a sociability perspective as proposed by Topirceanu et al. shows the rap network to be more sociable than the electronic music network. A more social network is characterized by a greater average degree, shorter average path length, higher clustering coefficient and shorter diameter. Indeed, these observations are corroborated by the classification of the rap network's well-established collaborations (Oliveira et al., 2020). To suggest somewhat speculatively, it is possible that the sociability of the network has an effect on the popularity of the genre.

To test the degree and strength distribution of these two novel collaboration networks, a power-law test was carried out heuristically and statistically. The degree distribution of both of the networks demonstrated a scale-free distribution. However, this does not necessarily mean they are scale-free networks as they are sampled from a real network. Most collaborative networks in previous studies demonstrate a power law with exponential cut-off property (Newman, 2001c, 2001a; Amaral & Scala, n.d.). Therefore, the sample drawn in this paper may not be large enough to capture the exponential cut-off effect. The distribution clearly rejects a scale-free hypothesis and suggests that a more complex mechanism of weighting is occurring. Scale-free networks are often rare in social networks and mostly exist in a weak form thus suggesting that collaboration networks is complex social networks that cannot be modelled using a Barabasi-Albert model (Holme, 2019).

The main limitation of this study is the sampling method as I drew one large snowball sample of 200 hops. However, future study may explore the network using a high number of small 1 or 2-hop snowball samples suggested by Elliott et al. (2018). Alternatively, a larger network may be constructed using the sampling module with different seed artists and constructing a cross-genre study of collaborations.

References

Amaral, L. A. N., & Scala, A. (n.d.). Classes of small-world networks. , 4. doi: 10.1073/pnas.200327197

Barabási, A.-L., & Albert, R. (1999, October). Emergence of Scaling in Random Networks. *Science*, *286*(5439), 509–512. doi: 10.1126/science.286.5439.509

Barrat, A., Barthelemy, M., Pastor-Satorras, R., & Vespignani, A. (2004, March). The architecture of complex weighted networks. *Proceedings of the National Academy of Sciences*, *101*(11), 3747–3752. doi: 10.1073/pnas.0400087101

Barrat, A., Barthélemy, M., & Vespignani, A. (2004, June). Weighted Evolving Networks: Coupling Topology and Weight Dynamics. *Phys. Rev. Lett.*, *92*(22), 228701. doi: 10.1103/PhysRevLett.92.228701

Bernard, H. R., Hallett, T., Iovita, A., Johnsen, E. C., Lyerla, R., McCarty, C., ... Stroup, D. F. (2010, December). Counting hard-to-count populations: The network scale-up method for public health. *Sexually Transmitted Infections*, *86*(Suppl 2), ii11-ii15. doi: 10.1136/sti.2010.044446

Budner, P., & Grahl, J. (2016, November). Collaboration Networks in the Music Industry. *arXiv:1611.00377 [physics]*.

Clauset, A., Shalizi, C. R., & Newman, M. E. J. (2009, November). Power-law distributions in empirical data. *SIAM Rev.*, *51*(4), 661–703. doi: 10.1137/070710111

Ebadi, A., & Schiffauerova, A. (2015, October). How to become an important player in scientific collaboration networks? *Journal of Informetrics*, *9*(4), 809–825. doi: 10.1016/j.joi.2015.08.002

Elliott, A., Leicht, E., Whitmore, A., Reinert, G., & Reed-Tsochas, F. (2018, January). A nonparametric significance test for sampled networks. *Bioinformatics*, *34*(1), 64–71. doi: 10.1093/bioinformatics/btx419

Freeman, L. C. (1978, January). Centrality in social networks conceptual clarification. *Social Networks*, *1*(3), 215–239. doi: 10.1016/0378-8733(78)90021-7

Grossman, J. W., & Ion, P. D. F. (1995). On a Portion of the Well-Known Collaboration Graph. *Congressus Numerantium*, *108*, 129–131.

Holme, P. (2019, March). Rare and everywhere: Perspectives on scale-free networks. *Nat Commun*, *10*. doi: 10.1038/s41467-019-09038-8

Newman, M. E. J. (2001a, June). Scientific collaboration networks. II. Shortest paths, weighted networks, and centrality. *Phys. Rev. E*, *64*(1), 016132. doi: 10.1103/PhysRevE.64.016132

Newman, M. E. J. (2001b, June). Scientific collaboration networks. I. Network construction and fundamental results. *Phys. Rev. E*, *64*(1), 016131. doi: 10.1103/PhysRevE.64.016131

Newman, M. E. J. (2001c, January). The structure of scientific collaboration networks. *PNAS*, *98*(2), 404–409. doi: 10.1073/pnas.98.2.404

Newman, M. E. J. (2005, September). Power laws, Pareto distributions and Zipf's law. *Contemporary Physics*, *46*(5), 323–351. doi: 10.1080/00107510500052444

Oliveira, G. P., Silva, M. O., Seufitelli, D. B., Lacerda, A., & Moro, M. M. (2020). DETECTING COLLABORATION PROFILES IN SUCCESS-BASED MUSIC GENRE NETWORKS. , 7.

Opsahl, T., Agneessens, F., & Skvoretz, J. (2010, July). Node centrality in weighted networks: Generalizing degree and shortest paths. *Social Networks*, *32*(3), 245–251. doi: 10.1016/j.socnet.2010.03.006

Spotify. (n.d.). *Documentation — Spotify for Developers.* https://developer.spotify.com/documentation/.

Topirceanu, A., Barina, G., & Udrescu, M. (2014, September). MuSeNet: Collaboration in the Music Artists Industry. In *2014 European Network Intelligence Conference* (pp. 89–94). doi: 10.1109/ENIC.2014.10

Xu, H., & Cheng, L. (2016, April). The QAP weighted network analysis method and its application in international services trade. *Physica A: Statistical Mechanics and its Applications*, *448*, 91–101. doi: 10.1016/j.physa.2015.12.094

A collab_net Module

```python
import json
import pickle as p
import sqlite3
from collections import Counter
from itertools import combinations

import networkx as nx
import numpy as np
import pandas as pd
import spotipy
from spotipy.oauth2 import SpotifyClientCredentials, SpotifyOAuth

from spotify_secret import keys

SPOTIFY_CLIENT_ID = keys["SPOTIFY_CLIENT_ID"]
SPOTIPY_CLIENT_SECRET = keys['SPOTIPY_CLIENT_SECRET']
SPOTIPY_REDIRECT_URI = keys['SPOTIPY_REDIRECT_URI']


auth_manager = SpotifyClientCredentials(client_id=SPOTIFY_CLIENT_ID,
                                        client_secret=SPOTIPY_CLIENT_SECRET,
                                        requests_timeout=120)


class collabNet:

    def __init__(self, start_id, limit=50,
                 save_attr=True, attr_filename='artist_attr.json',
                 save_collabs=True, collabs_filename='collabs.csv'):
        sp = spotipy.Spotify(auth_manager=auth_manager)
        self.limit = limit
        self.seen = list()
        self.unseen = list()
        self.unseen.append(start_id)
        self.albums = list()
        self.seen_name = list()
        self.seen_artists_uri = set()
        self.seen_tracks = set()
        self.collabs = list()
        self.artist_attr = dict()
        self.weighted_edge_list = list()

        i = 1
        for uri in self.unseen:
            if uri in self.seen:
                continue

            try:
                results = sp.artist_albums(uri, album_type=['album', 'single'])
            except:
```

```
                self.unseen.remove(uri)
                continue

            artist_name = sp.artist(uri)['name']
            print(f"{i}. currently on: {artist_name}")
            self.seen_name.append(artist_name)
            i += 1

            simp_albums = results['items']
            while results['next']:
                results = sp.next(results)
                # print(results[items])
                simp_albums.extend(results['items'])

            album_uris = []
            for album in simp_albums:
                album_uris.append(album['uri'].replace("spotify:album:", ''))

            n = 5
            album_uri_chunks = [album_uris[i:i + n]
                                for i in range(0, len(album_uris), n)]

            for chunk in album_uri_chunks:
                self.albums.extend(sp.albums(chunk)['albums'])

            tracks = list()
            for album in self.albums:
                if album == None:
                    continue
                else:
                    tracks.extend(album['tracks']['items'])

            for track in tracks:
                artists = track['artists']
                track_name = track['name']
                if track_name in self.seen_tracks:
                    continue

                track_artists = [art['name'] for art in artists]
                track_collabs = list(combinations(track_artists, 2))
                self.collabs.extend(track_collabs)
                self.seen_tracks.add(track_name)

                for art in artists:
                    collab_uri = art['uri'].replace("spotify:artist:", '')
                    self.seen_artists_uri.add(collab_uri)

                    if (collab_uri == uri or
                        collab_uri in self.seen):
                        continue
                    else:

```

```python
                        self.unseen.append(collab_uri)

            self.seen.append(uri)
            self.unseen.remove(uri)
            if len(self.unseen) < 1 or len(self.seen) >= self.limit:
                break
            else:
                continue


        print("Getting artists attributes...")
        n = 50
        collab_artist_uris = list(self.seen_artists_uri)
        artist_uri_chunks = [collab_artist_uris[i:i + n]
                             for i in range(0, len(collab_artist_uris), n)]

        collab_artists = list()
        for artist_uri_chunk in artist_uri_chunks:
            track_artists = sp.artists(artist_uri_chunk)['artists']
            collab_artists.extend(track_artists)

        artist_attr_temp = {art['name']: {'followers': art['followers']['total'],
                                          'popularity': art['popularity'],
                                          'genres': art['genres']} for art in collab_artists}
        self.artist_attr.update(artist_attr_temp)
        self.make_edge_list()
        print('Done!')

        if save_attr == True:
            self.save_attr_json(attr_filename)
        if save_collabs == True:
            self.save_collab_csv(collabs_filename)


    def make_edge_list(self):
        collab_counts = Counter(self.collabs)

        for count in collab_counts:
            self.weighted_edge_list.append(count + (collab_counts[count], ))


    def save_attr_json(self, filename='artist_attr.json'):
        with open(filename, 'w') as f:
            json.dump(self.artist_attr, f)
        print('Artist attribute saved as JSON')


    def save_collab_csv(self, filename='collabs.csv'):
        column_names = ['artist', 'collab', 'weight']

```

```
        data = self.weighted_edge_list

        collab_df = pd.DataFrame(data, columns=column_names)
        collab_df.to_csv(filename)

        print("Collaborations saved to CSV")



# drake = "3TVXtAsR1Inumwj472S9r4"
# # deadmau5 = "2CIMQHirSU0MQqyYHq0eOx"

# collab_net = collabNet(drake, limit=2)

# print('done')
```