Problem Chosen
D

2021
MCM/ICM
Summary Sheet

Team Control Number
2121741

This paper focuses on analyzing the influence of music through networks using illustrative visualizations, statistical analyses, and predictive modeling. In the first section, we analyze the inherent similarities of music within and between genres using a K-Means clustering algorithm and visualize the data using T-distributed Stochastic Neighbor Embedding dimensionality reduction. We continue by analyzing which genres have music most similar to each other (aside from themselves). To do so, we utilize a Pearson correlation matrix and plot the genres with significant correlation on a bar chart.

In the second section, we devise a novel method of calculating influence scores by combining a breadth-first search algorithm with an exponential decay function, and then plot the top influence scores on a bar chart. We also extract the timeframe of an influencer influencing a follower and create meaningful subnetwork visualizations using NetworkX.

In the third section, we analyze whether influencers have a significant impact on their followers by deploying a K-Nearest Neighbors algorithm. We observe that influencers should have similar music to their followers if there is true influence, and therefore they should be neighbors with one another when plotted in multi-dimensional space.

In the fourth section, we built a predictive model using XGBoost and extracted the most important & distinguishing features for genre prediction. We supplemented our feature importance analysis with alternative statistical testing.

In the fifth and final section, we explored trends in how songs characteristics have changed over time and connected them to real world events and explanations. We made observations about trends for all songs, and also specifically analyzed Rhythm and Blues (R&B) and its most influential artists.

Through these analyses, we hope to enlighten the reader on the influence of music through networks and how music has changed over time using a combination of math, computer science, and of course, music.

Table of Contents

Letter to the ICM--3

Problem Statement---4

Important Definitions---4

Problem Assumptions---4-5

Similarities Within and Between Genres---------------------------5-8

Are artists within genres more similar than artists between genres?-----------5-6

Are some genres related to others?---7-8

Measuring Influence--8-12

How do we develop a model to measure influence?-------------------------------8-10

After how many years were most artists influenced?---------------------------------11

Visualizing and Exploring a Subnetwork---11-12

Impact of Influencers---13-14

Influencer → Follower---13-14

Follower → Influencer--14

Feature Importance and Predictive Modeling--------------------15-17

Feature Importance & Predictive Modeling with XGBoost-------------------15-16

Feature Importance with Statistics--17

Changes Over Time---18-23

Conclusion--23

Packages Used--24

References--24

Appendix--24

Dear Integrative Collective Music Society,

Team 2121741 has conducted an in-depth exploration and analysis of data that was provided to us, and has gained a deeper, rich understanding of the influence of music through networks. First, we used the dataset, aggregated each artist by genre, and performed a K-Means clustering analysis-in doing so, we were able to confidently conclude that music within genres are statistically similar.

We also measured influence throughout the network at different depths by utilizing a breadth-first search algorithm. Depth 1 included an influencer's direct followers, Depth 2 included an influencer's direct followers and their followers, and so on up to Depth 10. We combined those values with an exponential decay function to determine an artist's total influence score. According to our model, the top 10 influencers in the dataset are: The Beatles, Bob Dylan, Chuck Berry, James Brown, Elvis Presley, The Rolling Stones, Little Richard, Jimi Hendrix, Hank Williams, Ray Charles. We also devised a novel method of visualizing subnetworks.

On top of numerical influence, we wanted to analyze whether influencers truly influence their followers. We did so by utilizing a K-Nearest Neighbors algorithm, in which we found that influencers have some but not much influence on their followers.

To add upon our in-depth analysis, we decided to create predictive models using XGBoost, a gradient-boosting decision tree library. Just from the numerical features of each song, our XGBoost model can correctly predict the genre of a song with 66% accuracy. Using this model (as well as some statistical measures), we also analyzed which features most distinguished certain genres, and found that instrumentalness was by far the most influential characteristic, followed by acousticness and speechiness.

And finally, we observed the change in music production, influence, and features changed over time and drew conclusions about how different historical events and musical revolutions influenced the music industry. We specifically focused on R&B and how its features have changed over the past century.

In this paper, we lay out our findings categorically by sections. We hope our analysis will help you better understand the influence of music. Our goal with this analysis was to provide a solid foundation for future research, which undeniably should include a larger dataset with more artists, more songs, and more genres. The code that we use for our analysis is very simple to follow and understand for future work, and will be provided in the Appendices.

Best Regards,
Team 2121741

Problem Statement

Music has the power to culturally, morally, and emotionally influence our society. The sound and messages artists release through their art form directly impact their followers and listeners in powerful ways (Huang, 2014). In an effort to better understand the impact of music on musical artists and society, we have been asked by the Integrative Collective Music (ICM) Society to analyze musical influence across time, genres, and artists. The primary focus of this paper is to capture revolutionary changes, extraneous circumstances, and influential artists using illustrative visualizations and analyses.

Important Definitions

Cluster: A group of data tightly packed together in some dimension.
Feature: Analogous to "characteristic". This notation is used interchangeably throughout the paper.
Primary artist features: ['danceability', 'energy', 'valence', 'tempo', 'loudness', 'mode', 'key', 'acousticness', 'instrumentalness', 'liveness', 'speechiness', 'duration_ms', 'popularity', 'count']
Z-score: A standard unit of measurement. Each sample x in our dataset is scaled down to z-scores using the formula $z = x - \mu / \sigma$, where μ is the mean and σ is the standard deviation.

Problem Assumptions

Assumption 1: The data given in the datasets "influence_data", "full_musics_data", "data_by_artist", and "data_by_year" is accurate.

Reasoning: The efficacy of our analysis is largely dependent on the accuracy of the data. Because we cannot use any other data in this paper, we must assume that the data provided to us is correct.

Assumption 2: Each feature in the dataset has equal weighting; normalizing the data using z-scores is sufficient for analysis.

Reasoning: There's no reasonable way to weigh certain features heavier than others.

Assumption 3: Two genres or artists being similar is defined by and only by the inherent characteristics in their music, and not by extraneous factors such as genre and song title.

Reasoning: The whole purpose of using these numerical features is to analyze whether there are inherent similarities between genres/artists.

Assumption 4: Drop song and artists in the genre “Unknown” and “Children’s”.

Reasoning: There are only 3 artists labeled under “Unknown” and 4 artists labeled under “Children’s”, compared to the 5653 total artists in the data.

Similarities Within and Between Genres

Are artists within genres more similar than artists between genres?

Observation: Per Assumption 3, if artists within a genre are more similar to each other than to artists outside of their genre, the characteristics of their music should be similar enough such that we can accurately cluster and distinguish music from one genre to another.

We utilize a **K-Means Clustering Algorithm** to group our data. This algorithm uses Expectation-Maximization to mathematically assign groupings and find the centroid of each cluster (Dabbura, 2018). We initialize the number of clusters equal to the number of genres in our dataset (18).

We also utilize **T-distributed Stochastic Neighbor Embedding (t-SNE)** to reduce the dimensionality of our data from 14 (length of primary artist features) to 2. Since our primary objective with dimensionality reduction is visualization (so not for predictive models), we choose t-SNE instead of the popular alternative Principal Components Analysis (Kapri, 2020).

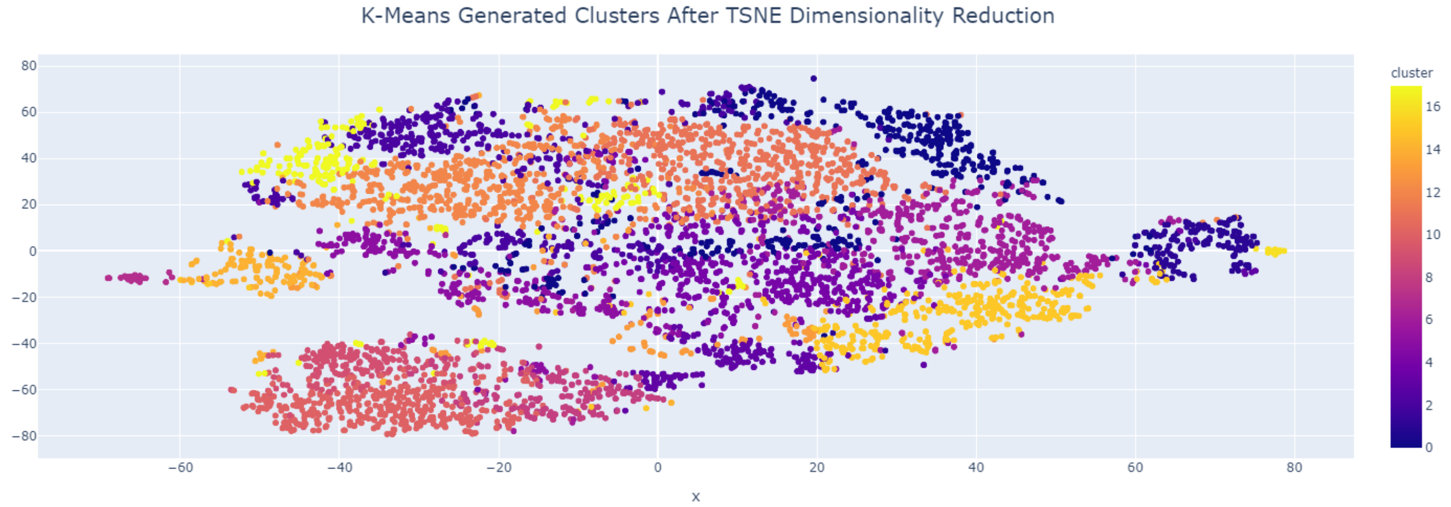
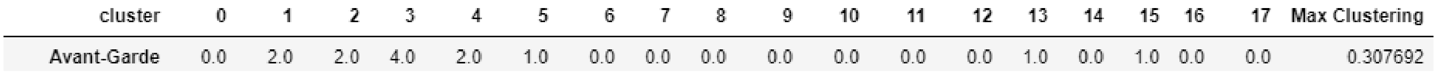


Fig 1. Each datapoint is a t-SNE reduced representation of an artist based on their primary characteristics. The colors represent different clusters generated by the K-Means algorithm.

To analyze whether the clustering algorithm was successfully able to separate different genres based on their features, we devise a test statistic called the **Maximum Clustering Coefficient (MCC)**. An example application is shown below:



cluster	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	Max Clustering
Avant-Garde	0.0	2.0	2.0	4.0	2.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	1.0	0.0	0.0	0.307692

Fig 2. The Maximum Clustering Coefficient is determined by the cluster that has the most artists divided by the total number of artists in that genre. Using Avant-Garde as an example, our MCC is $\frac{4}{13} \approx 0.307$.

We also deploy an A/B Test with 50 iterations to determine whether our results are significant.

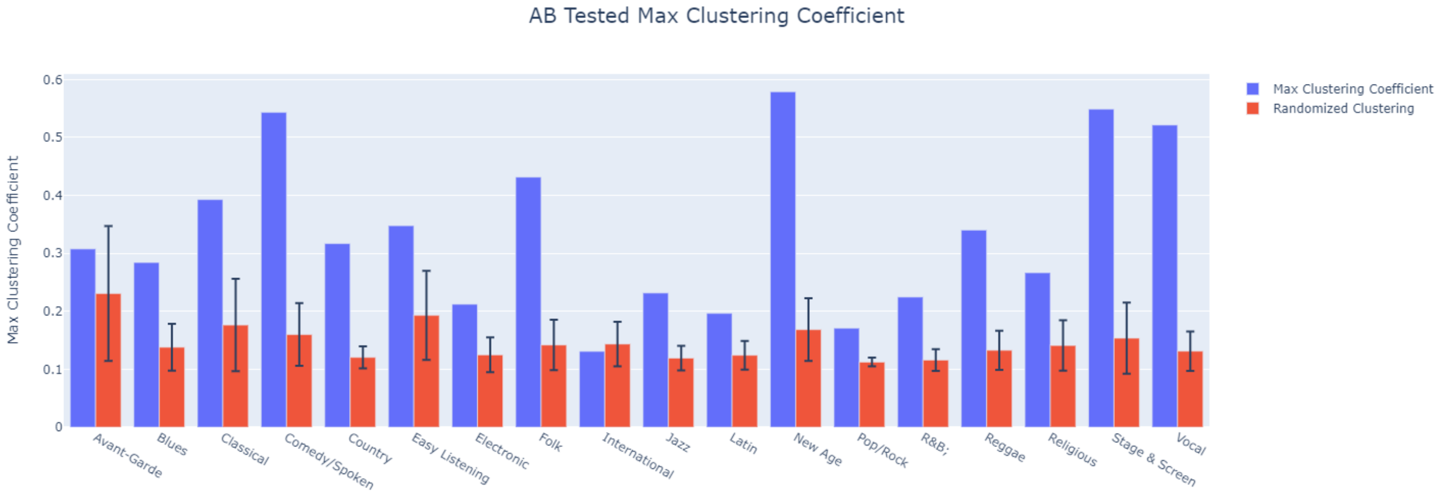


Fig 3. Actual MCC (blue) plotted against A/B tested MCC (red). Black whisker lines represent 95% confidence interval (2 STD away from mean).

It is quite obvious that our clustering results are statistically significant, as aside from Avant-Garde, the p-value for all genres are < 0.05 . Thus, we can confidently conclude that artists within one genre are more similar than artists within another genre, since the K-Means clustering algorithm was able to, for the most part, accurately cluster the data.

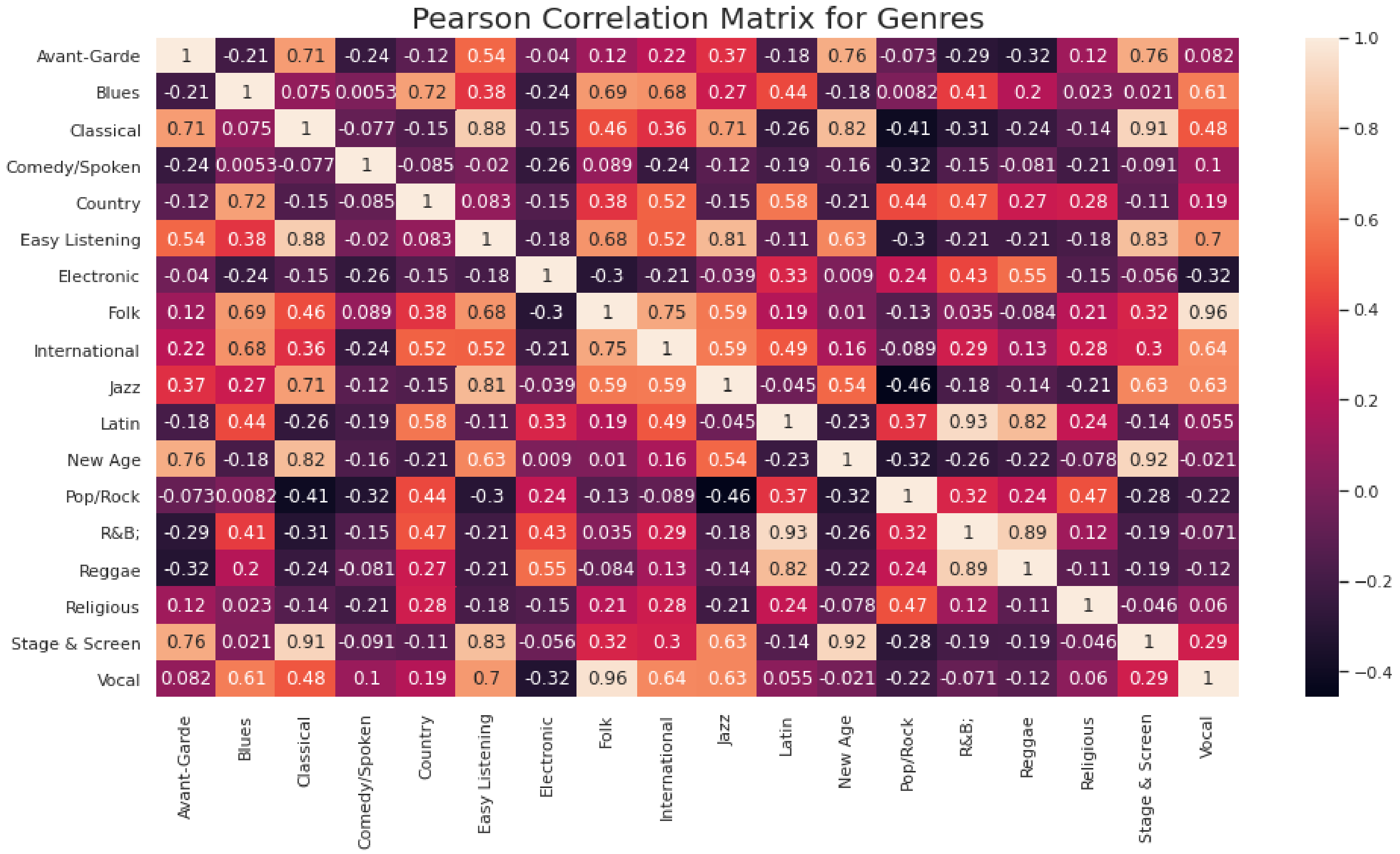
We also answered this question using two separate approaches: genre clustering and KNN classifying. Each of our performed analyses reached the same conclusion of statistical significance. For the sake of brevity, we do not include a formal write-up of our techniques and results in this paper (see Appendix A. for code).

Are some genres related to others?

To answer this question, we look for whether certain genres are correlated with each other. Our measure of correlation is the Pearson Coefficient, defined as:

$$r = \frac{\sum(x_i - \overline{x})(y_i - \overline{y})}{\sqrt{\sum(x_i - \overline{x})^2 \sum(y_i - \overline{y})^2}}$$

Shown below is a Pearson correlation matrix for each genre (excluding the dropped genres).



	Avant-Garde	Blues	Classical	Comedy/Spoken	Country	Easy Listening	Electronic	Folk	International	Jazz	Latin	New Age	Pop/Rock	R&B;	Reggae	Religious	Stage & Screen	Vocal
Avant-Garde	1	-0.21	0.71	-0.24	-0.12	0.54	-0.04	0.12	0.22	0.37	-0.18	0.76	-0.073	-0.29	-0.32	0.12	0.76	0.082
Blues	-0.21	1	0.075	0.0053	0.72	0.38	-0.24	0.69	0.68	0.27	0.44	-0.18	0.0082	0.41	0.2	0.023	0.021	0.61
Classical	0.71	0.075	1	-0.077	-0.15	0.88	-0.15	0.46	0.36	0.71	-0.26	0.82	-0.41	-0.31	-0.24	-0.14	0.91	0.48
Comedy/Spoken	-0.24	0.0053	-0.077	1	-0.085	-0.02	-0.26	0.089	-0.24	-0.12	-0.19	-0.16	-0.32	-0.15	-0.081	-0.21	-0.091	0.1
Country	-0.12	0.72	-0.15	-0.085	1	0.083	-0.15	0.38	0.52	-0.15	0.58	-0.21	0.44	0.47	0.27	0.28	-0.11	0.19
Easy Listening	0.54	0.38	0.88	-0.02	0.083	1	-0.18	0.68	0.52	0.81	-0.11	0.63	-0.3	-0.21	-0.21	-0.18	0.83	0.7
Electronic	-0.04	-0.24	-0.15	-0.26	-0.15	-0.18	1	-0.3	-0.21	-0.039	0.33	0.009	0.24	0.43	0.55	-0.15	-0.056	-0.32
Folk	0.12	0.69	0.46	0.089	0.38	0.68	-0.3	1	0.75	0.59	0.19	0.01	-0.13	0.035	-0.084	0.21	0.32	0.96
International	0.22	0.68	0.36	-0.24	0.52	0.52	-0.21	0.75	1	0.59	0.49	0.16	-0.089	0.29	0.13	0.28	0.3	0.64
Jazz	0.37	0.27	0.71	-0.12	-0.15	0.81	-0.039	0.59	0.59	1	-0.045	0.54	-0.46	-0.18	-0.14	-0.21	0.63	0.63
Latin	-0.18	0.44	-0.26	-0.19	0.58	-0.11	0.33	0.19	0.49	-0.045	1	-0.23	0.37	0.93	0.82	0.24	-0.14	0.055
New Age	0.76	-0.18	0.82	-0.16	-0.21	0.63	0.009	0.01	0.16	0.54	-0.23	1	-0.32	-0.26	-0.22	-0.078	0.92	-0.021
Pop/Rock	-0.073	0.0082	-0.41	-0.32	0.44	-0.3	0.24	-0.13	-0.089	-0.46	0.37	-0.32	1	0.32	0.24	0.47	-0.28	-0.22
R&B;	-0.29	0.41	-0.31	-0.15	0.47	-0.21	0.43	0.035	0.29	-0.18	0.93	-0.26	0.32	1	0.89	0.12	-0.19	-0.071
Reggae	-0.32	0.2	-0.24	-0.081	0.27	-0.21	0.55	-0.084	0.13	-0.14	0.82	-0.22	0.24	0.89	1	-0.11	-0.19	-0.12
Religious	0.12	0.023	-0.14	-0.21	0.28	-0.18	-0.15	0.21	0.28	-0.21	0.24	-0.078	0.47	0.12	-0.11	1	-0.046	0.06
Stage & Screen	0.76	0.021	0.91	-0.091	-0.11	0.83	-0.056	0.32	0.3	0.63	-0.14	0.92	-0.28	-0.19	-0.19	-0.046	1	0.29
Vocal	0.082	0.61	0.48	0.1	0.19	0.7	-0.32	0.96	0.64	0.63	0.055	-0.021	-0.22	-0.071	-0.12	0.06	0.29	1

Fig 4. A Pearson Coefficient Value > 0.7 or < -0.7 implies significant positive or negative correlation, respectively.

It is hard to see any meaningful correlation between genres using a matrix, so we extracted any significant correlation values and plotted them on a bar chart.

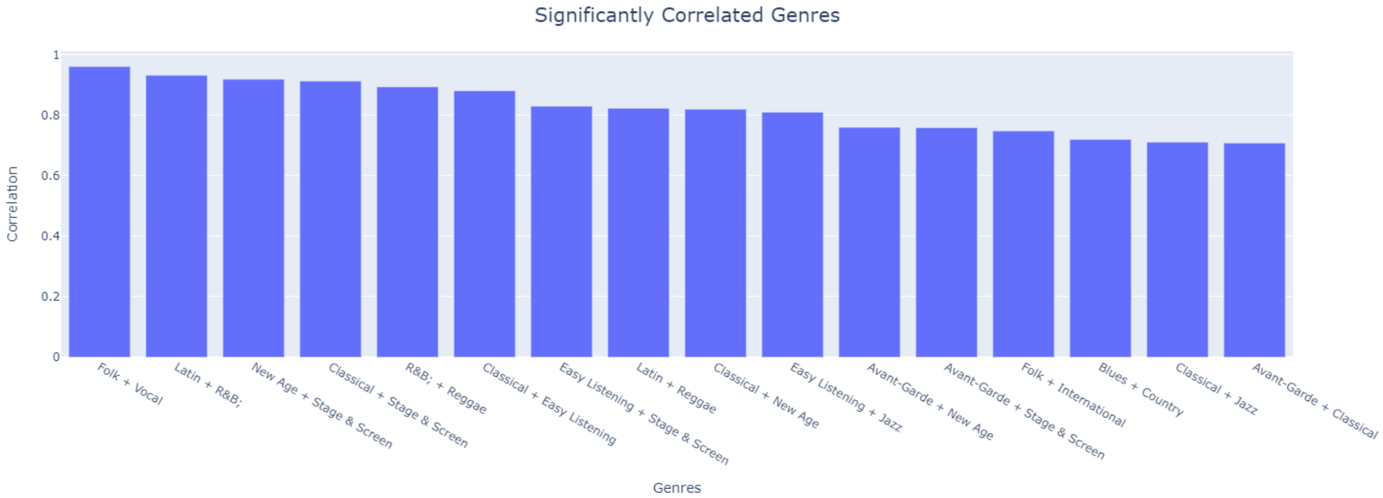


Fig 5. Folk and Local are the most correlated genres based on their features.

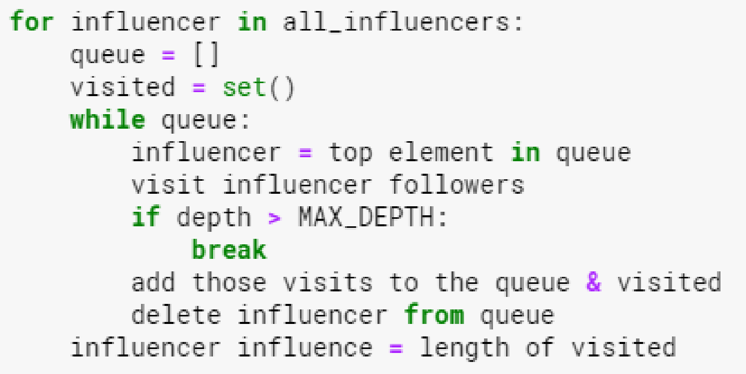
In essence, yes, some genres are related to each other, but most have low correlation (if at all). Note that our correlation is calculated only from the inherent numerical features of each genre.

Measuring Influence

How do we develop a model to measure influence?

Before we are able to measure influence, we must be able to store a graph of the network in a data structure. We do that with a simple dictionary, where each key represents an influencer, and each value represents that influencer's followers.

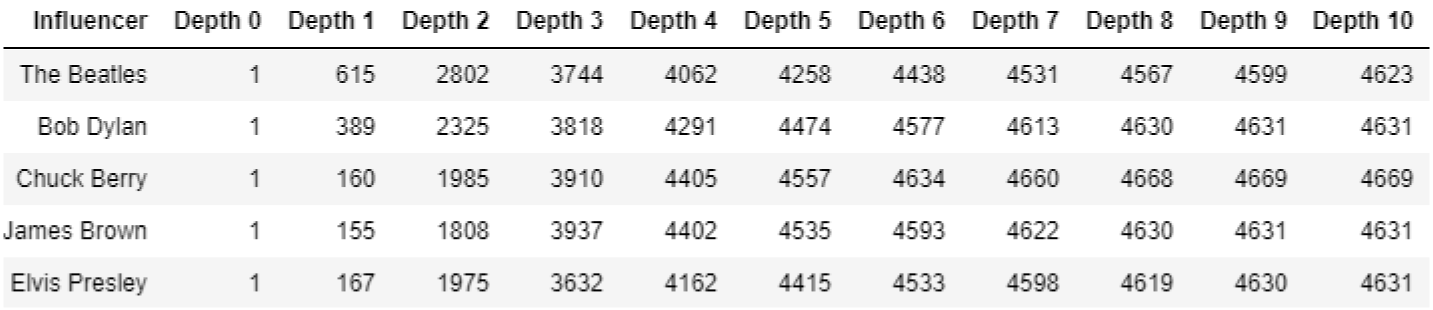
Using that graph, we can calculate influence by implementing a breadth-first search algorithm (BFS) which works as follows:



```
for influencer in all_influencers:
    queue = []
    visited = set()
    while queue:
        influencer = top element in queue
        visit influencer followers
        if depth > MAX_DEPTH:
            break
        add those visits to the queue & visited
        delete influencer from queue
    influencer influence = length of visited
```

Fig 6. BFS psuedo-code.

We use the parameter *d* to control the depth of our search. We iterate through *d* values from 0 to 10 and store them in a Pandas DataFrame as shown below:



Influencer	Depth 0	Depth 1	Depth 2	Depth 3	Depth 4	Depth 5	Depth 6	Depth 7	Depth 8	Depth 9	Depth 10
The Beatles	1	615	2802	3744	4062	4258	4438	4531	4567	4599	4623
Bob Dylan	1	389	2325	3818	4291	4474	4577	4613	4630	4631	4631
Chuck Berry	1	160	1985	3910	4405	4557	4634	4660	4668	4669	4669
James Brown	1	155	1808	3937	4402	4535	4593	4622	4630	4631	4631
Elvis Presley	1	167	1975	3632	4162	4415	4533	4598	4619	4630	4631

Fig 7. Depth 0 includes the artist themselves; Depth 1 includes their direct followers; Depth 2 includes their followers & followers' followers, etc.

We devised a metric for measuring influence called **Influence Score**, which weights closer, more direct connections heavier than further connections. Concretely, Influence Score is defined as follows:

$$i = \sum_{d=1}^{d=10} \frac{f_d - f_{d-1}}{FACTOR^{d-1}}$$

where f_d is the number of followers at depth *d* and *FACTOR* is some exponential smoothing constant (we set *FACTOR* = 2 for this paper).

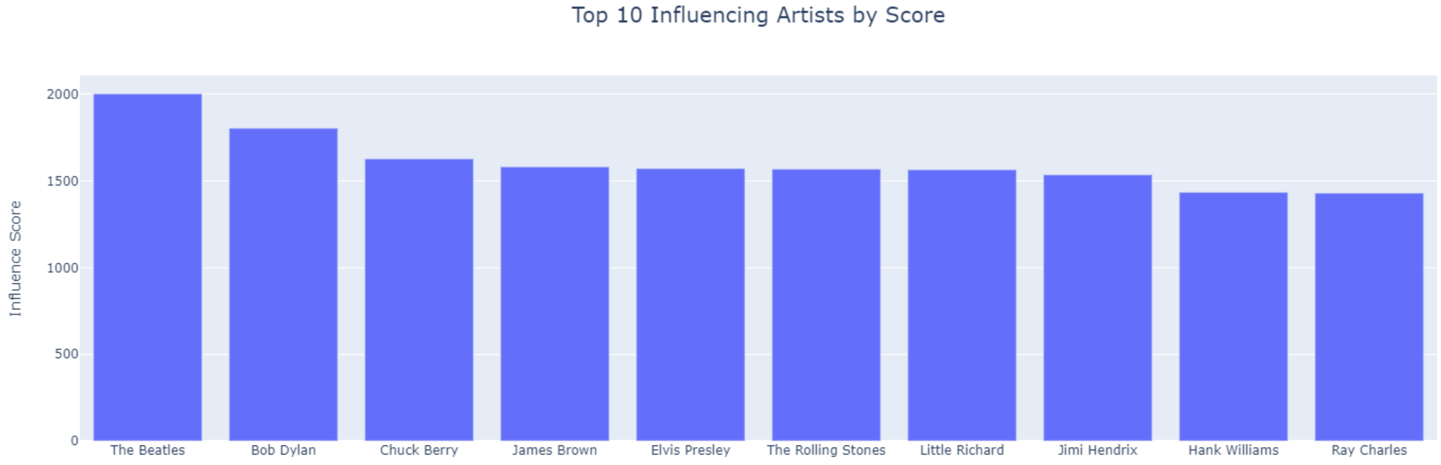


Fig 8. All of the artists on this list are big names (The Beatles, Elvis Presley!)

Here were some other visualizations we found particularly insightful:

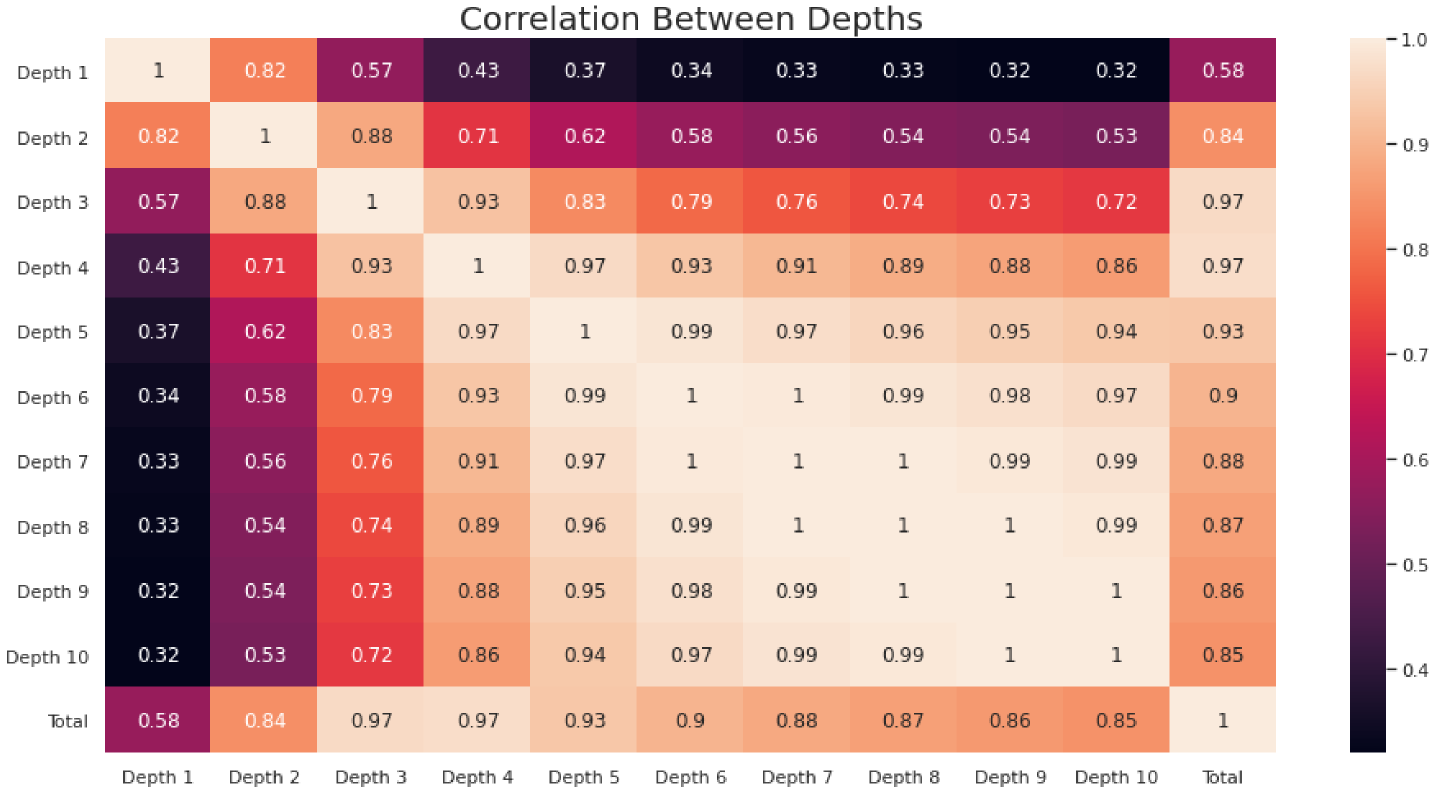


Fig 9. Correlation matrix for each depth and total score. The total number of people each artist influences seems to level out around Depth 4 (correlations are very similar to every subsequent depth).

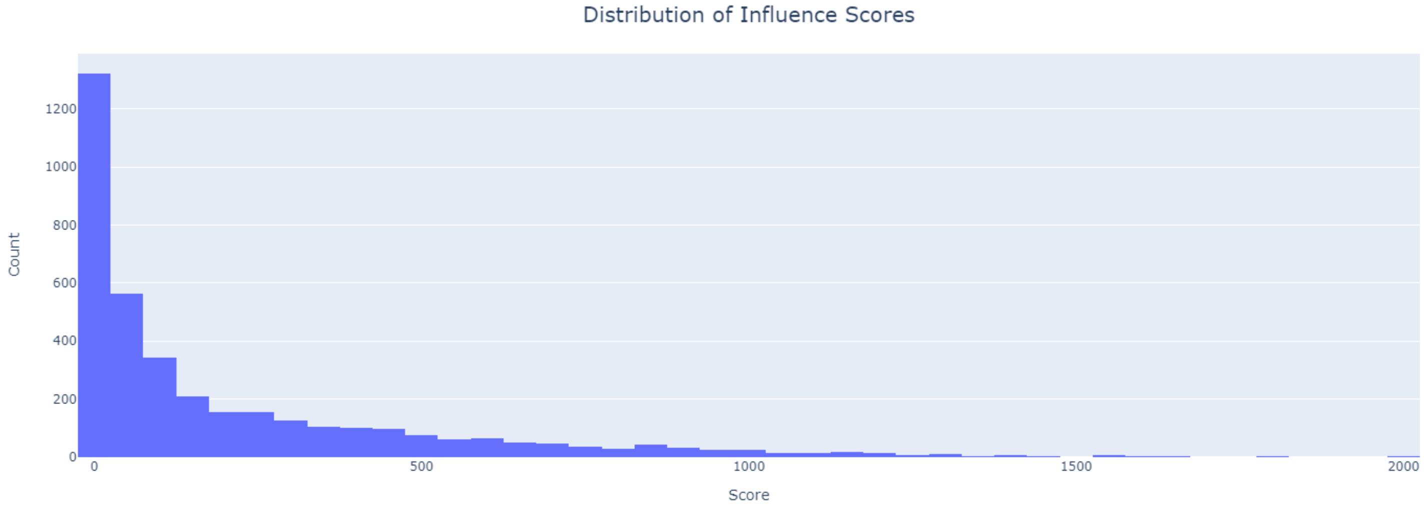


Fig 10. The distribution is clearly right-skewed and the mean influence score for all artists is ~209.

After how many years were most artists influenced?

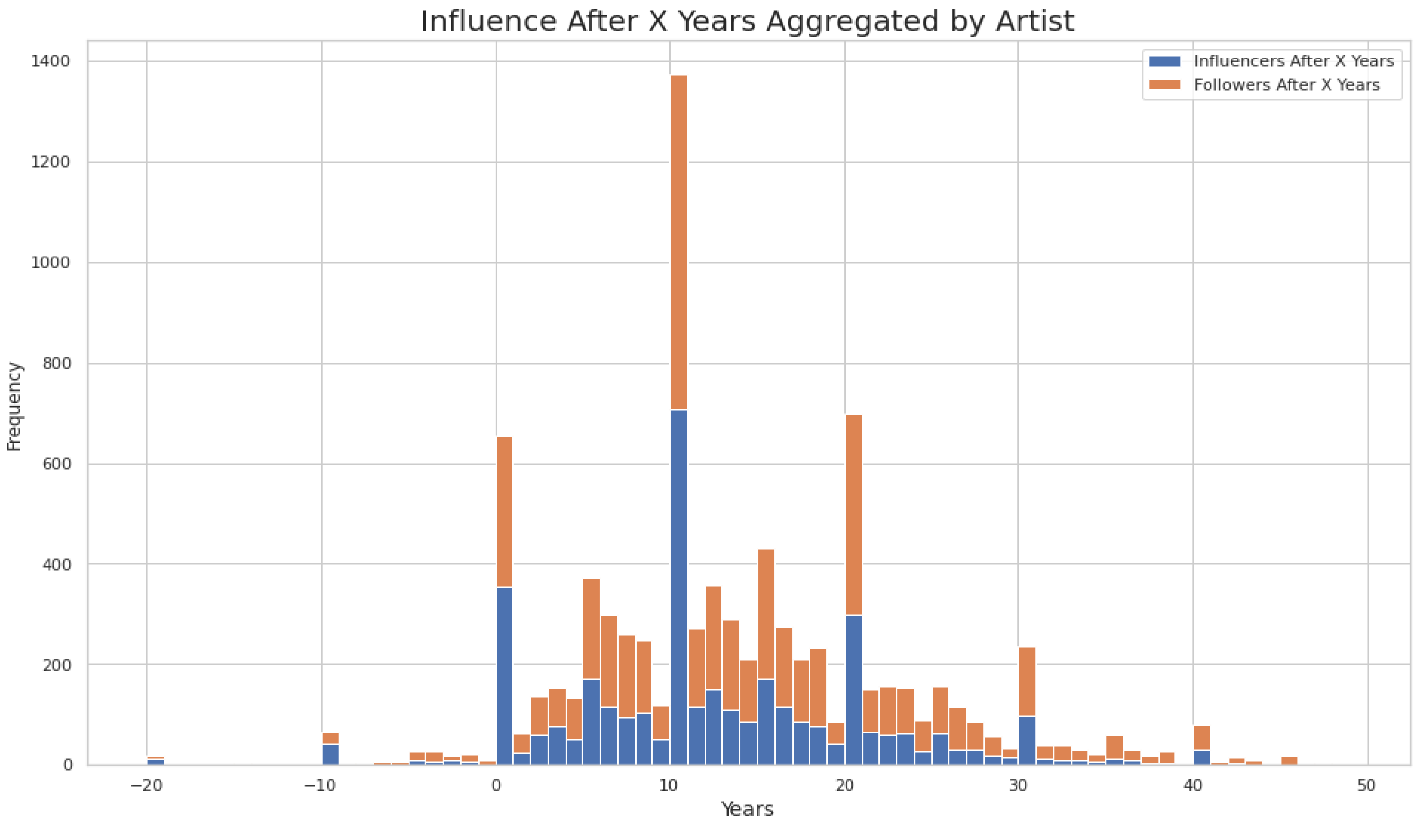


Fig 11. On average, influencers influence their followers after 12.5 years, whereas followers are influenced by their influencers after 14.7 years.

Visualizing and Exploring a Subnetwork

Although we can visualize a small portion of the network, it is hard to visualize the entire network in a meaningful way due to the sheer number of connections present. We present three visualizations using NetworkX below: one of a small subnetwork to Depth 10, one of a medium-sized subnetwork to Depth 10, and the final of the Beatles with Depth 1.

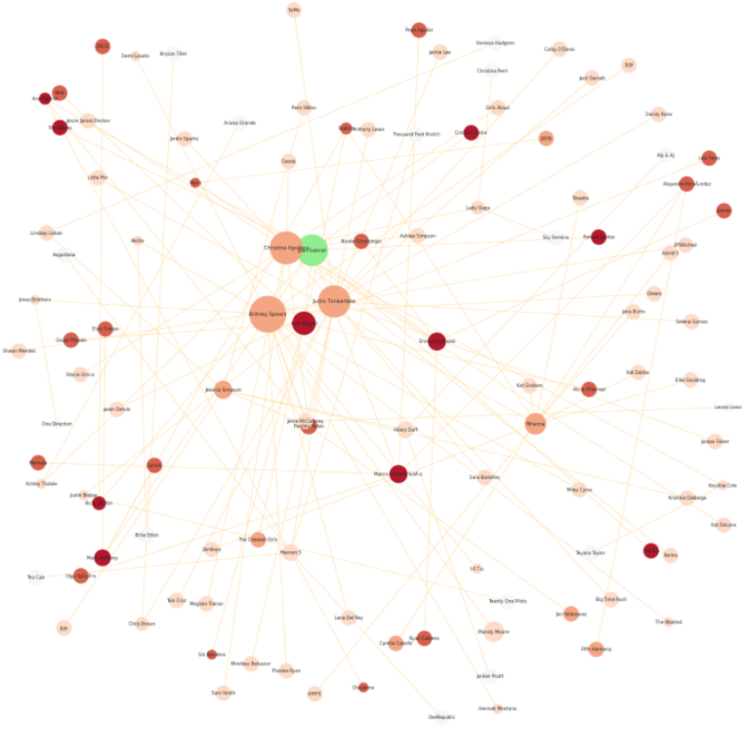


Fig 12. Small network (114 connections) - Juan Gabriel

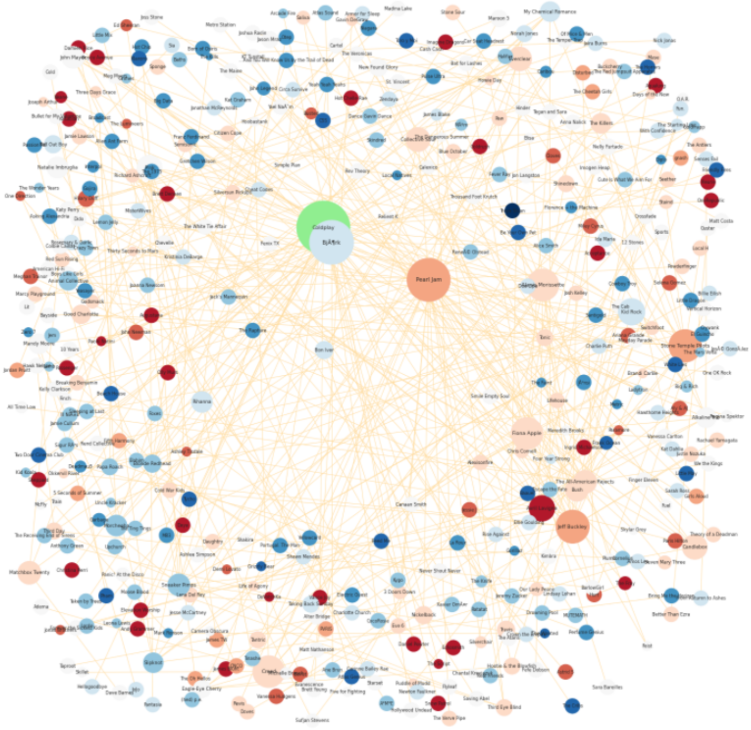


Fig 13. Medium-sized network (420 connections) - Coldplay

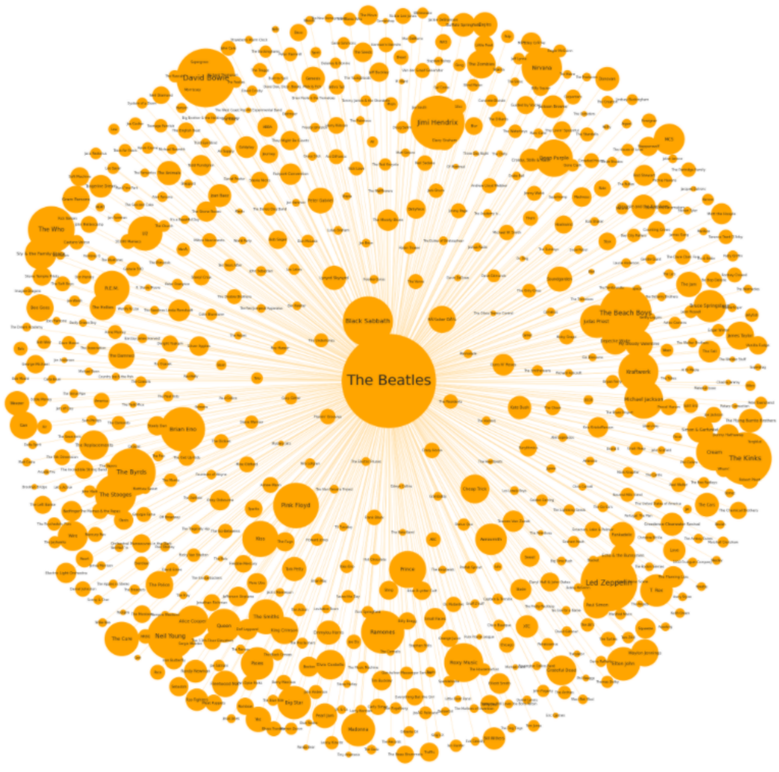


Fig 14. The Beatles Direct Followers (615 connections)

Impact of Influencers

Observation: The more similar influencers and their followers' music are to each other, the greater the impact that influencer had on the follower.

To analyze this potential similarity, we employ a **K-Nearest Neighbors** (KNN) algorithm, which finds the K artists with music most similar to another artist. First, we converted our features into Z-scores for a stable, standard unit. Then, we used Euclidean distance for our measure of similarity, which is defined as:

$$d(p,q) = \sum_{i=1}^{n} (q_i - p_i)^2$$

where p_i and q_i represent Euclidean vectors. We store the nearest neighbors as a set and find its intersection with our target set.

Influencer → Follower

For this scenario, we set our K-value to 20, which means we store each influencer's 20 closest neighbors (excluding itself) in a set. We find the intersection between that KNN-generated set and each influencer's followers and take the length of that intersection to extract the total amount of overlap.

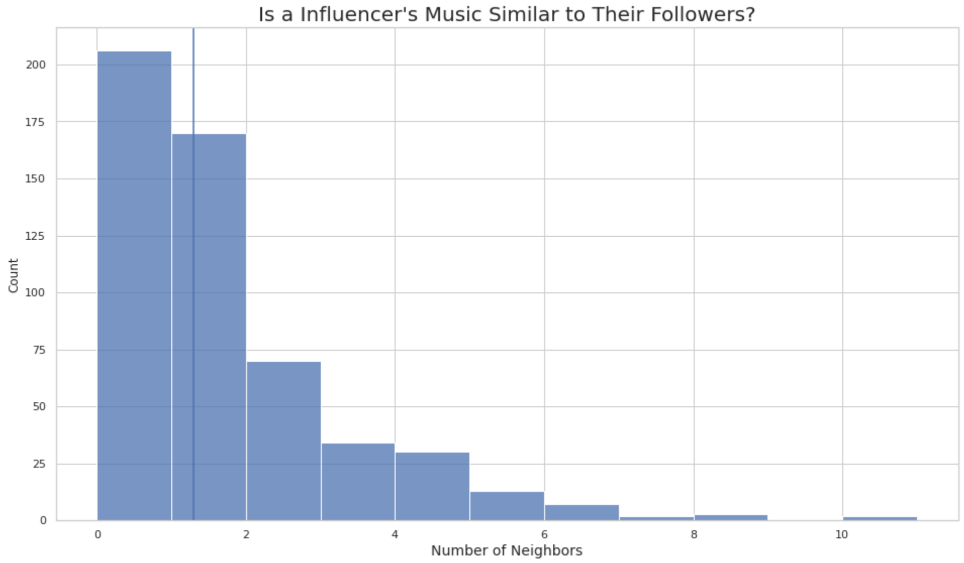


Fig 15. Bar graph of the intersection between an influencer's followers and the influencer's 20 most similar artists. The vertical line represents the mean.

We can see that for a majority of influencers, only a few of their followers are the people who have music most similar to them (Fig. 15). On average, only 1.3/20 artists (6.5%) overlap between an influencer's followers and an influencer's most similar artists.

Follower → Influencer

We then use a similar approach to analyze if a follower's music is similar to that of their influencers. Since each follower has generally less influencers than influencers have followers, we set our K-value to 10 here. The results of our analysis are shown below.

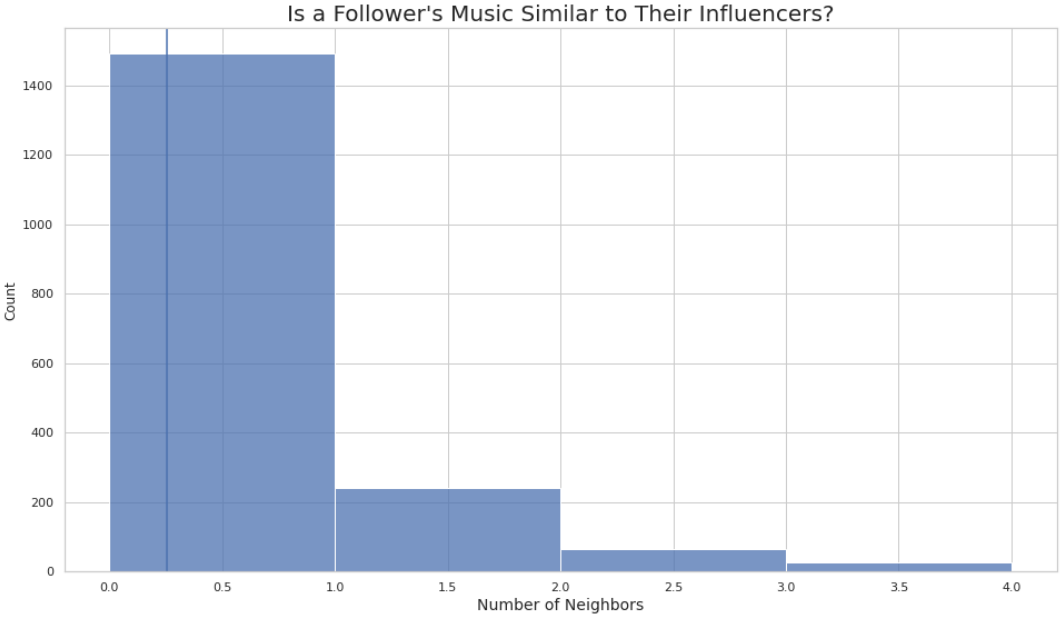


Fig 16. Bar graph of the intersection between a follower's influencers and the followers's 10 most similar artists. The vertical line represents the mean.

For a vast majority of followers, very few of a follower's influencers fall within the follower's most similar artists, similar to the analysis of influencer → follower (Fig. 16). On average, only 0.253/10 artists (2.5%) overlap between a follower's influencers and the follower's most similar artists.

From these two analyses, on average, influencers seem to influence their followers more than followers are influenced by their influencers. However, neither influencers nor followers have that similar music compared to their counterparts—the influencer-follower connection between an artist and another artist does not confidently result in similar music between the two.

Feature Importance & Predictive Modeling

Next, we analyzed which characteristics are most important for distinguishing between different genres. We did so using two methods: feature importance with a predictive model and feature importance using statistical analyses.

Feature Importance & Predictive Modeling with XGBoost

Our team uses XGBoost, an optimized distributed gradient boosting library, to build an accurate & efficient model which can classify a song's genres based on their numerical features. We used XGBoost's Cross Validation function and grid search to find optimal hyperparameters (see Appendix B for model & code)

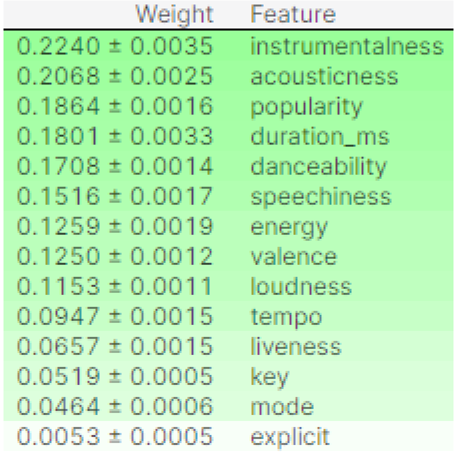
Hyperparameter name	Value
Learning rate (eta)	0.17
Max depth	10
Regularization lambda	1.5
Epochs (n estimators)	300

Table 1. Table of model hyperparameters.

Using our optimized model, we split our train/test data into 5 using Sklearn's KFold module, and tested the model on each fold. Our results are summarized below:

KFold CV Number	Accuracy	Multi Log Loss Change
1	66.646%	2.335 → 1.059
2	66.923%	2.330 → 1.053
3	66.194%	2.334 → 1.071
4	66.006%	2.334 → 1.063
5	66.082%	2.337 → 1.071

Table 2. 5-fold validation. Mean prediction accuracy is 66.370%.



Weight	Feature
0.2240 ± 0.0035	instrumentalness
0.2068 ± 0.0025	acousticness
0.1864 ± 0.0016	popularity
0.1801 ± 0.0033	duration_ms
0.1708 ± 0.0014	danceability
0.1516 ± 0.0017	speechiness
0.1259 ± 0.0019	energy
0.1250 ± 0.0012	valence
0.1153 ± 0.0011	loudness
0.0947 ± 0.0015	tempo
0.0657 ± 0.0015	liveness
0.0519 ± 0.0005	key
0.0464 ± 0.0006	mode
0.0053 ± 0.0005	explicit

Fig 17. Features near the top of the chart are most important according to the XGBoost model.

It is worth mentioning that there are by far more Pop/Rock songs than any other genre, which can lead to overfitting of the model. We performed the same analyses but without songs from the Pop/Rock genre, and the model's mean accuracy was ~61.5% and the feature importance values stayed identical.

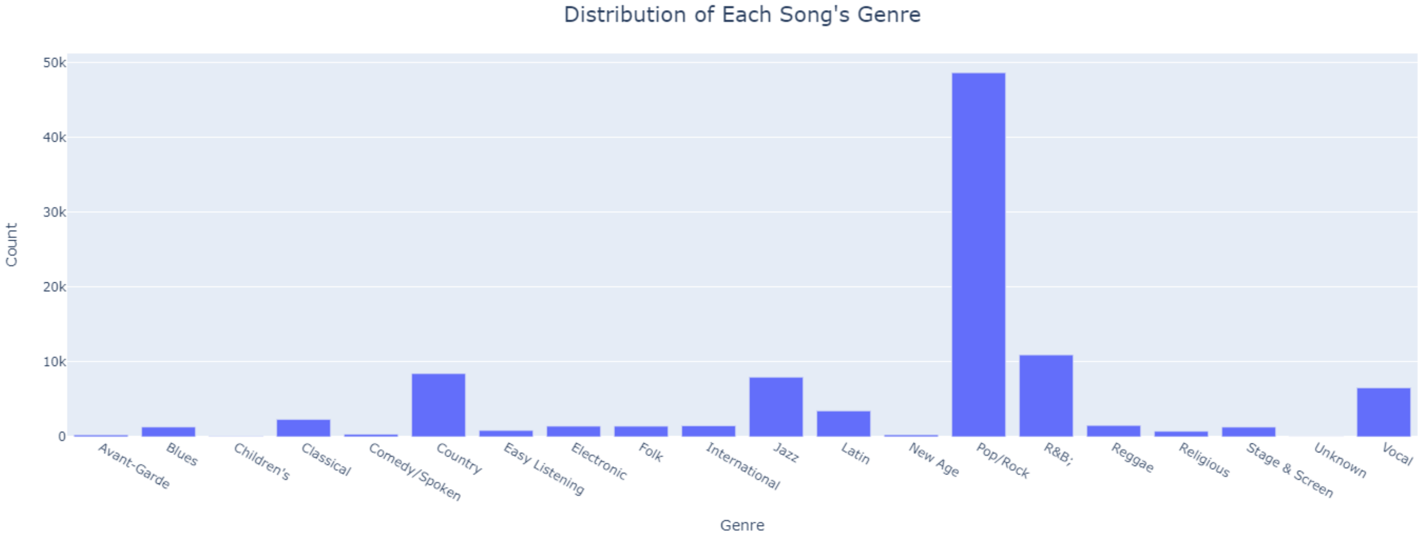


Fig 18. Pop/Rock is clearly the most popular genre of songs.

We also played around with the idea of implementing a neural network and were able to achieve accuracies of around ~62% without any optimization; however, we opt to end our discussion of predictive models here since they are not the main focus of our paper.

Feature Importance with Statistics

With this method, we aggregate the features like below, where we plot the average z-score value of the danceability value for each genre:

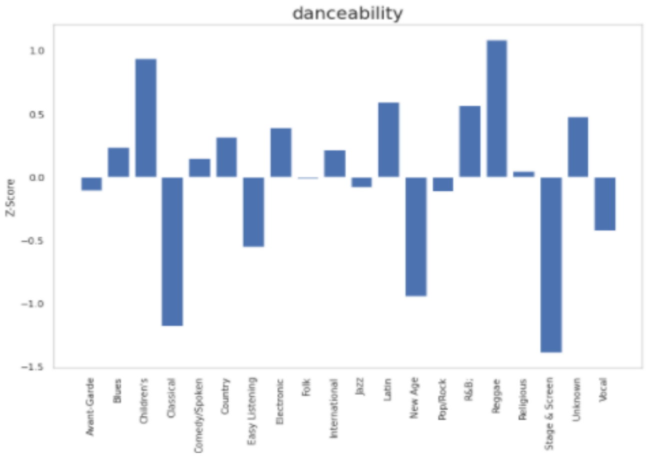


Fig 19. Average Z-score value of danceability for each genre.

We then measure the total standard deviation for the values in that data, and repeat this procedure for every feature. The feature with the highest standard deviation should be the most distinctive feature, and thus is most useful/important when distinguishing between genres.

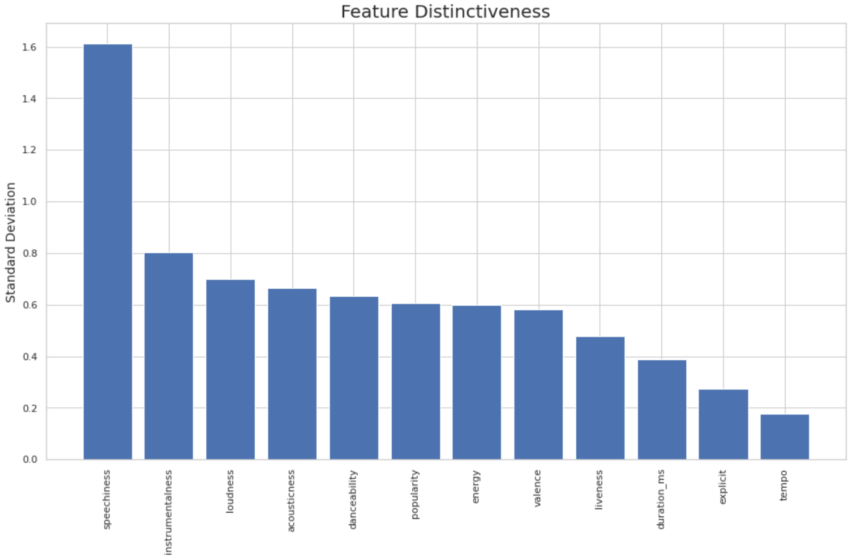


Fig 20. Speechiness is the most distinctive feature by far.

Instrumentalness (rank 1 & 2) is definitely the most distinctive feature, followed by speechiness (rank 6 & 1) and acousticness (rank 2 & 4). Using both analyses, explicit is not a very distinctive feature.

Changes Over Time

To first get an idea of how the music industry changes over time, we used the "full_music_data" data set to plot the number of songs released per year. This plot can be seen below.

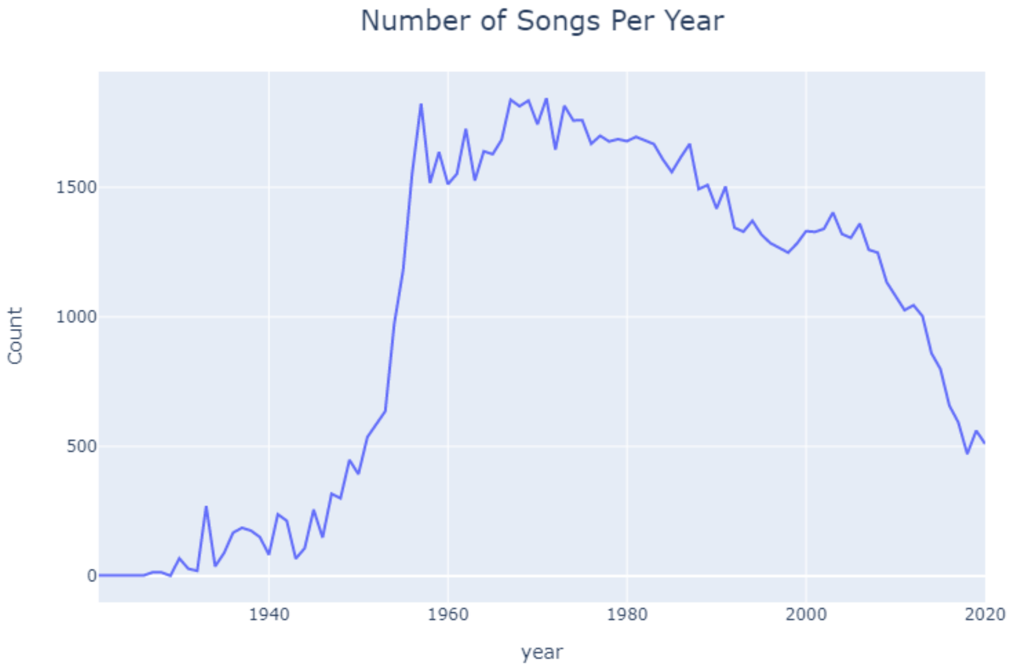


Fig 21. The number of songs in production drastically increased in the 1950s and 1960s.

We can extract some useful data regarding the historical events that might have affected the music industry and the influence of music (Fig. 21). 1939 marks the end of the Great Depression, which could explain this rapid growth of music production after 1940, as new economic growth could have fueled the music industry. The invention of the computer in 1943 could also explain this large growth in song production, as computers could be used to produce, advertise, and sell music more easily. The years with the highest number of songs were in the 1960s, which coincides with the invention of cassette tapes (Kendall, 2017). These would have provided more access to songs to more people, as well as facilitate the selling and production of music. The number of songs per year started to decline rapidly around 2008, which could be due to the Great Recession in '08.

Using the "influence_data" data set, we also created a plot of the number of followers present per year, which can be seen below.

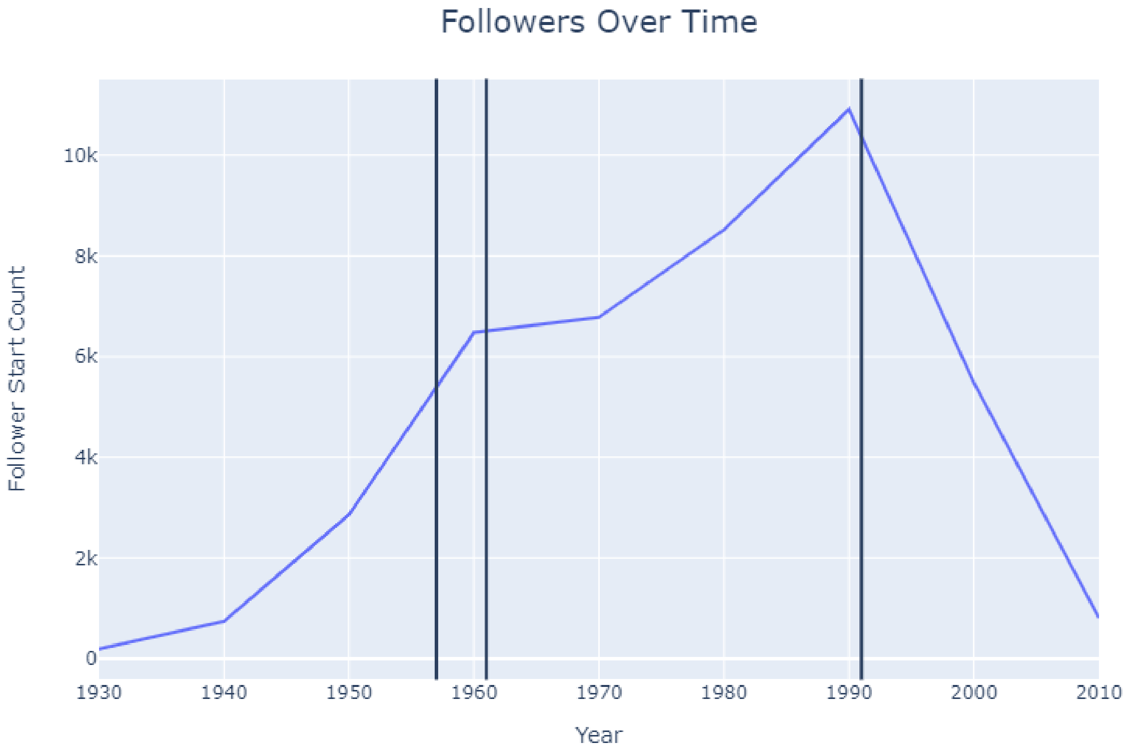


Fig 22. Line 1 - The Beatles' first release; Line 2 - Bob Dylan's first release; Line 3 - World Wide Web. We would like to note that the above graph overcounts the number of followers as it does not remove duplicates of followers (followers who say they are influenced by more than one influencer). The shape of the graph should be similar to if we only counted unique followers though.

The trend in the number of followers over time is roughly comparable to the number of songs produced every year (Fig. 22). The number of artists who start skyrocketed in the generation after The Beatles, Bob Dylan, and some of the other artists with large influence scores, which confirms the notion that our influence scores are somewhat accurate. Interestingly, the release of the World Wide Web is correlated with a sudden drop in follower count, which may be due to the advent of more original, contemporary music.

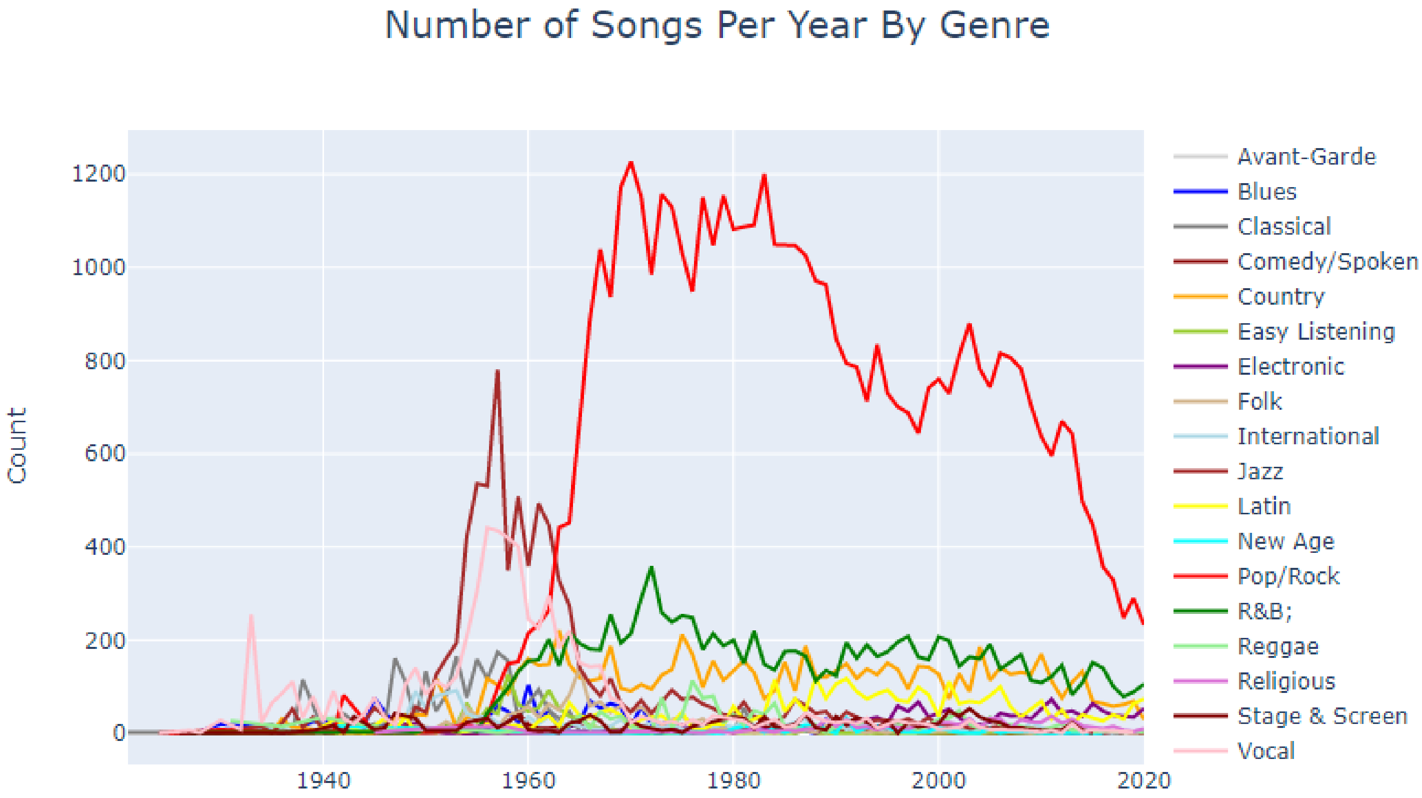


Fig 23. Jazz peaked in the 1950s, whereas Pop/Rock grew drastically in the 1960s.

The Pop/Rock genre was responsible for most of the growth in music from the 1960s and onwards (Fig. 23). Other genres of music declined while Pop/Rock grew. The Beatles, which we identified as the most influential artist, formed in 1957, which marks the rise of the Pop/Rock genre. The popularity and influentialness of the Beatles could have contributed to the rapid growth of Pop/Rock in the 1960s. Another observation from the data is that Jazz music rose and declined rapidly in popularity from about 1950 to 1970, and was actually the most published genre around 1955. This roughly marks the period when Bebop and cool Jazz were popular. The decline of Jazz from 1955 onwards also mirrors the growth of Pop/Rock.

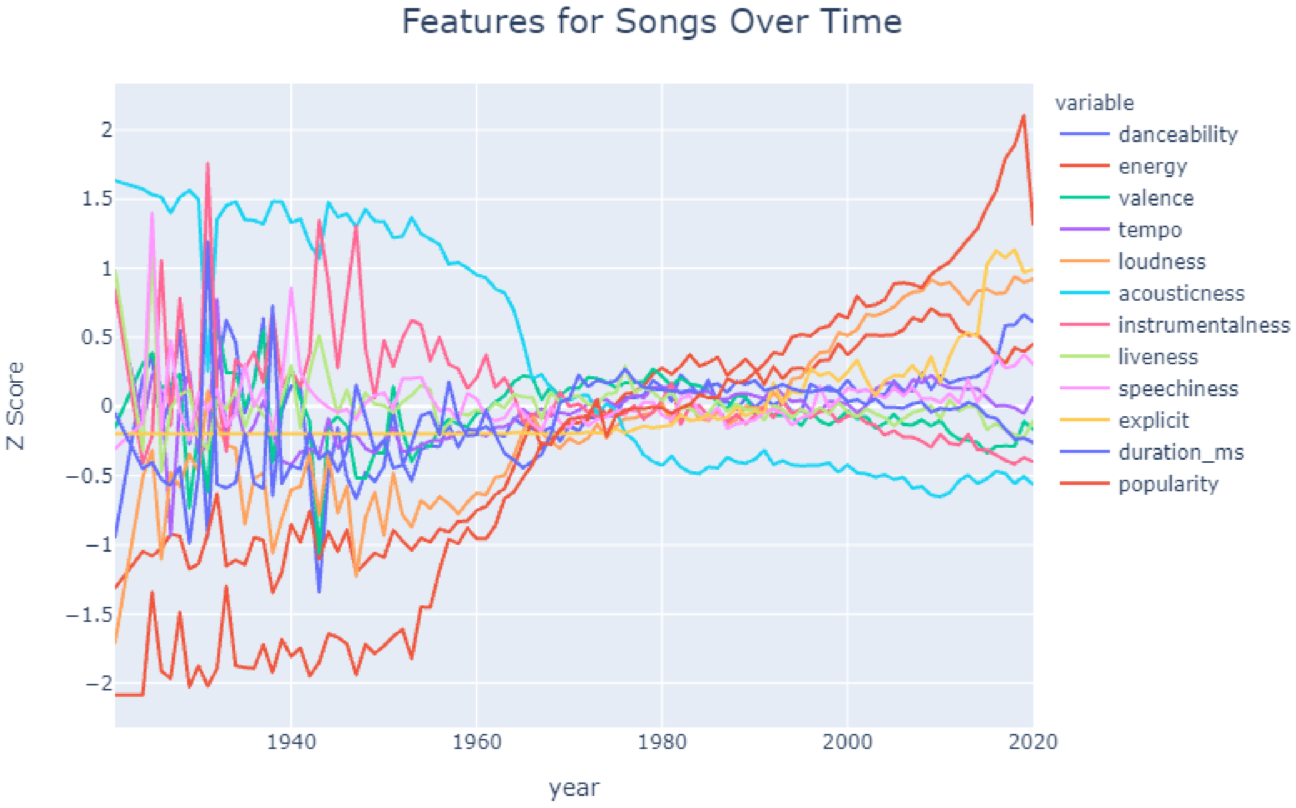


Fig 24. Features of songs over time.

Given the graphs of these features over time, we looked at important changes that indicated musical revolutions (Fig. 24).

The most apparent change of music was in the 1960s, where the acousticness of music dropped heavily. This identified one of the most important revolutions in music history, when electric guitars and amplification became a mainstay in popular music. Following this, acousticness continued to decline with the introduction of synthesizers and sampling.

Explicitness also rose over time, most notably in the past decade, and this actually indicated a revolution in how music reaches its listeners. In the age of radio, stations censored lyrics and generally discouraged explicitness. However, with the explosion of streaming services in the past few years, musicians that publish to these services are freer to use explicit language in their songs (Ross, 2017).

Loudness also followed a steady upwards trend, which indicated a musical phenomenon referred to as the “loudness war”. Throughout time, producers continued to drive up the loudness of their songs, as louder songs generally stood out more compared to quieter ones (NPR). In the 1990s, the arrival of digital signal processing allowed producers to drive up loudness even more, which correlated with increasing loudness on the graph.

Finally, we noticed that valence followed a downwards trend. Similarly we calculated the average mode each year, and that also followed a downwards trend. Since a mode of 0 indicated a minor key, and minor keys are generally less bright than major keys, we took this as another indicator of decreasing positivity. Though this pattern was interesting, it was difficult to connect it to any specific events, perhaps it indicated that people are generally more negative recently, or that sad music has become more trendy.

(Additionally, we noticed that popularity followed an upwards trend, but since the popularity metric was based on recent listeners, it seemed obvious that the more recent songs would be more popular.)

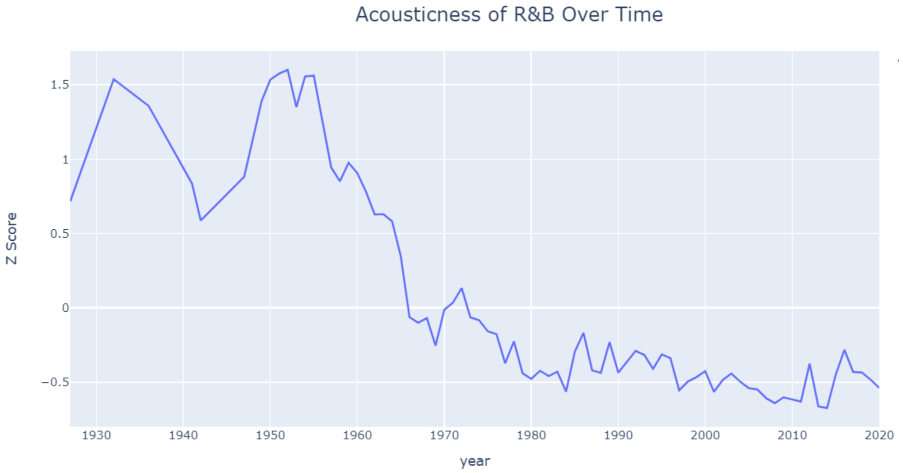


Fig 25. Acousticness of R&B over time. There is a clear downwards trend.

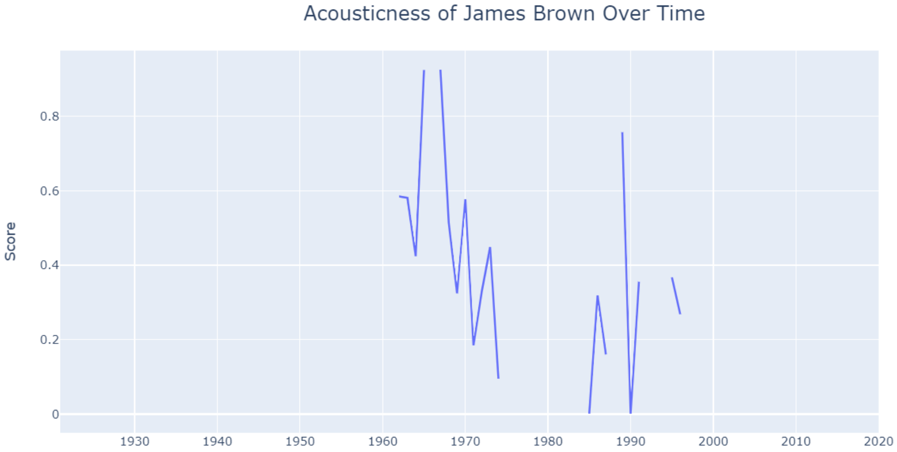


Fig 26. Average acousticness of James Brown's songs over time. Many years have missing data.

Finally, we analyzed genre and artist-specific trends (Fig, 25-26). We focused on R&B, and found a strong downwards trend in the average acousticness. Then, we filtered songs in R&B to only ones authored by James Brown, the most influential in the genre according to our metrics, and we found the same downwards trend. It was hard to establish whether he caused this trend though, especially considering the small amount of data we had for the artist.

Conclusion

From our K-Means clustering analyses, we find that artists are much more similar within genres than between genres. From our Pearson's Correlation matrix, we also find that certain genres are more closely related feature wise to each other than to other genres (Fig 5). Using a data network, we were able to model the influence of artists, and we found that the most influential artists were the Beatles.

Using a KNN algorithm, we were able to measure the similarity between influencers and their followers. We found that influencers seem to influence their followers more than followers are influenced by their influencers. However, in all there are not that many significant similarities between the music of influencers and their followers. We also determined the features that distinguish genres from each other. By using the standard deviation of the features across genres, we found that acousticness is the best feature to distinguish between genres.

By looking at the changes in song number, genre popularity, and song features over time, we were able to attribute some of these changes to certain historical events. For example, the rise of new technologies like the cassette tape and CD in the second half of the 20th century sparked rapid growth in music production, and the formation of the Beatles marked the rise of the pop/rock genre.

Additionally, we found that the popularization of the electric guitar led to a drop in the acousticness of songs, and the popularization of music streaming platforms led to higher explicitness in songs. Finally, we connected trends in genres to major influencers in those genres, for example, we observed that a decrease in the overall acousticness of R&B coincided with a decrease in the acousticness of James Brown's songs.

Packages Used

Our team utilized **Python** for this data science & modeling project, since it provides the simplest and most efficient libraries to visualize and analyze data. Listed below are some of the packages we used for each component of our analysis:

Visualizations	Analysis	Other
Plotly Matplotlib Seaborn Networkx	Sklearn (KMeans, StandardScaler, MinMaxScaler, TSNE) Scipy XGBoost Keras	Numpy Pandas

References

Dabbura, I. (2018). K-means Clustering: Algorithm, Applications, Evaluation Methods, and Drawbacks.
Huang, B. (2014). What Kind of Impact Does Our Music Really Make on Society?
Kapri, A. (2020). PCA vs LDA vs T-SNE — Let's Understand the difference between them!
Kendall, J. (2017). From Discs to Digital: The Odd History of Music Formats.
NPR (2009). The Loudness Wars: Why Music Sounds Worse.
Ross, E. (2017). Parental Advisory: How Songs With Explicit Lyrics Came to Dominate the Charts.

Appendix

A. https://www.kaggle.com/ironicninja/team-2121741-the-influence-of-music-code/
B. https://www.kaggle.com/ironicninja/the-influence-of-music-prediction-models/