Exploring Spotify Popularity Through Genre, Musical Features, and Artist-Level Trends

Data Science Project : COMP3125

Instructor: Fariba Khoshnasib-Zeinabad

Team Members: Dalton Crawford, Ian Babcock, Jing Pan

Date: 12/03/2025

Project Motivation

- **Why analyze Spotify?**
- **Research Questions**

  - Is the number of songs an artist releases associated with the popularity of their songs?

  - Which artists produce the highest proportion of popular songs?

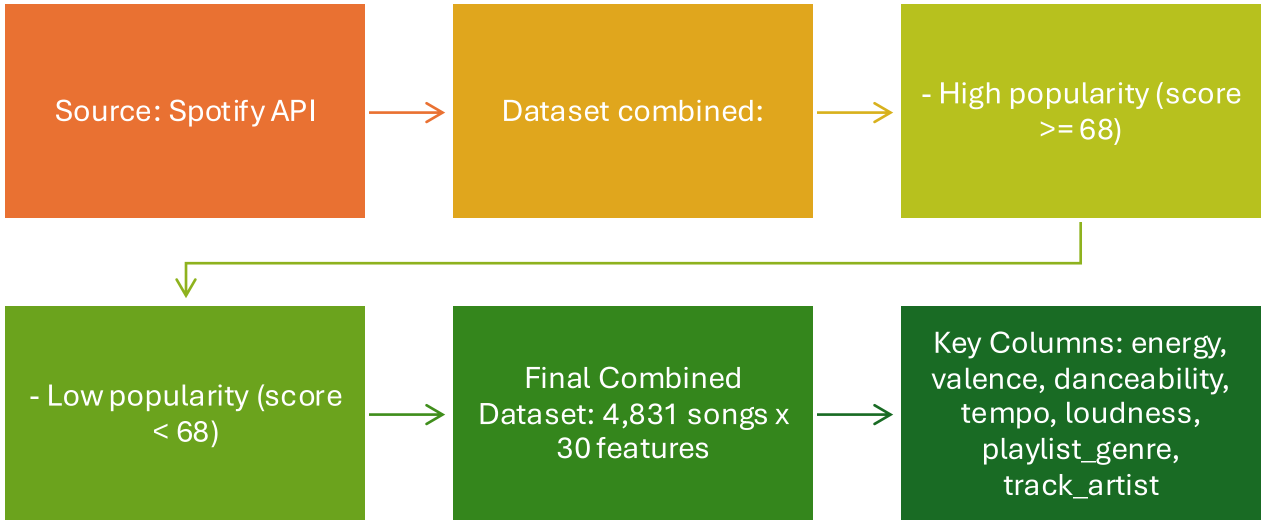
  - Compare genres and build predictive models using musical attributes

  - Are there any correlations in the data at all

  - Does genre reveal significant correlated traits

Dataset Overview and Cleaning

Dataset Overview




```
df_all.info()
```

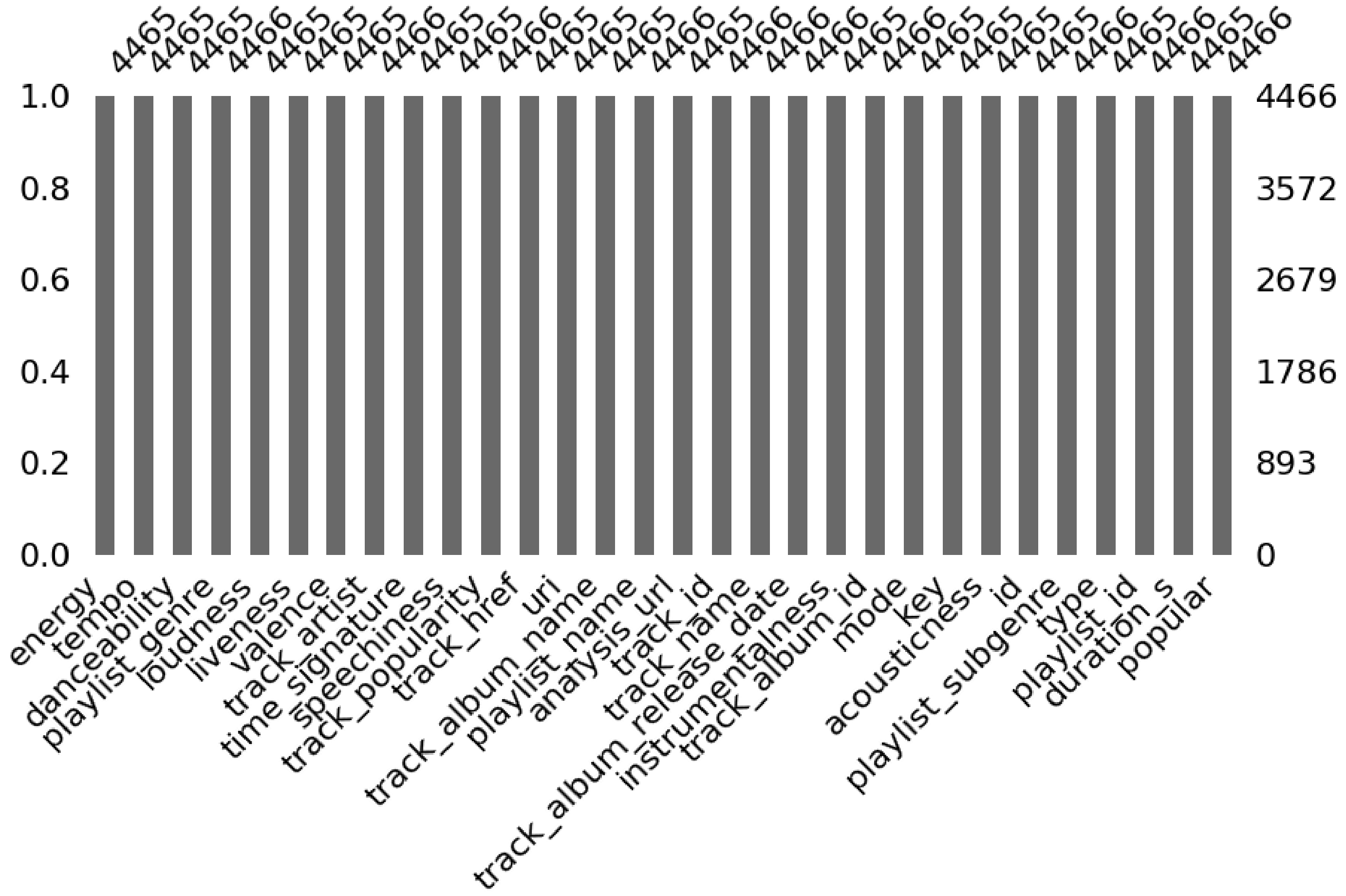
```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 4831 entries, 0 to 4830
Data columns (total 30 columns):
 #   Column                    Non-Null Count  Dtype
---  ------                    --------------  -----
 0   energy                    4830 non-null   float64
 1   tempo                     4830 non-null   float64
 2   danceability              4830 non-null   float64
 3   playlist_genre            4831 non-null   object
 4   loudness                  4830 non-null   float64
 5   liveness                  4830 non-null   float64
 6   valence                   4830 non-null   float64
 7   track_artist              4831 non-null   object
 8   time_signature            4830 non-null   float64
 9   speechiness               4830 non-null   float64
 10  track_popularity          4831 non-null   int64
 11  track_href                4830 non-null   object
 12  uri                       4830 non-null   object
 13  track_album_name          4830 non-null   object
 14  playlist_name             4831 non-null   object
 15  analysis_url              4830 non-null   object
 16  track_id                  4831 non-null   object
 17  track_name                4831 non-null   object
 18  track_album_release_date  4831 non-null   object
 19  instrumentalness          4830 non-null   float64
 20  track_album_id            4831 non-null   object
 21  mode                      4830 non-null   float64
 22  key                       4830 non-null   float64
 23  acousticness              4830 non-null   float64
 24  id                        4830 non-null   object
 25  playlist_subgenre         4831 non-null   object
 26  type                      4830 non-null   object
 27  playlist_id               4831 non-null   object
 28  duration_s                4830 non-null   float64
 29  popular                   4831 non-null   int64
dtypes: float64(13), int64(2), object(15)
memory usage: 1.1+ MB
```

Data Cleaning Pipeline: A Process Overview

1. Missing value inspection
2. Duplicate check (full-row & track-level)
3. Type conversion of musical features
4. Outlier detection using IQR

```python
# Visualization of missingness
import missingno as msno
msno.bar(df_all, figsize=(10,4))
```

```
<Axes: >
```



Missingness Bar Chart

- Most columns contain **no missing values (0)**
- A few features contain **only 1 missing value**
- Missing rate is below **0.05%**, so **no imputation was needed**

Duplicate check results

- No full-row duplicates detected.
- Track-level duplicate check (track_name + track_artist) also returned zero duplicates.
- Ensure each unique track appear only once

```
# Number of duplicate rows (full-row duplicates)
df_all.duplicated().sum()
```

```
np.int64(0)
```

```
# Duplicate detection based on track_name + track_artist
duplicates = df_all[df_all.duplicated(subset=['track_name', 'track_artist'], keep=False)]
duplicates.head()
```

energy	tempo	danceability	playlist_genre	loudness	liveness	valence	track_artist	time_signature	speechiness	...	track_album_id	mode	key	acousticness	id	playlist_subgenre	type	playlist_id	duration_s	popular

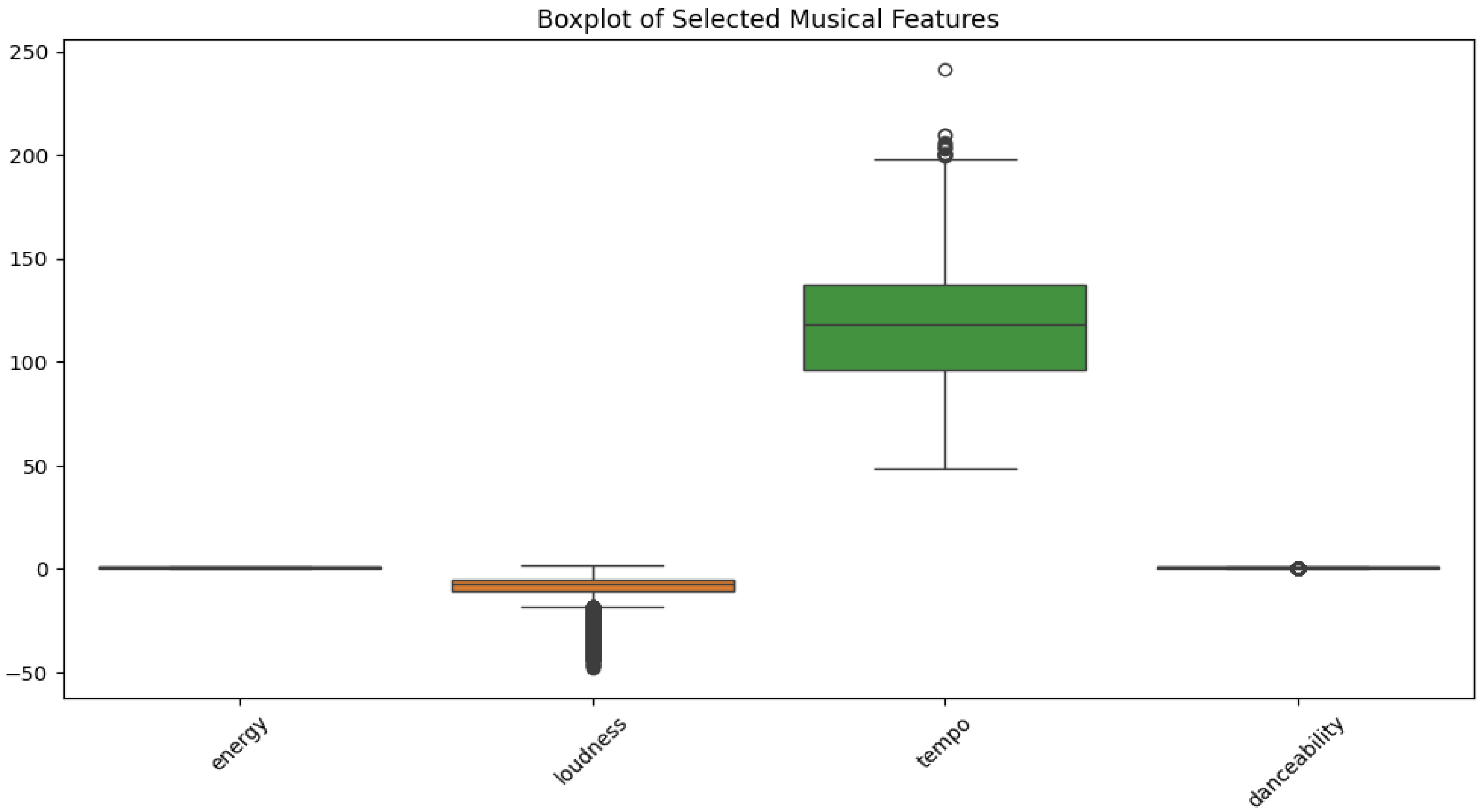
0 rows × 30 columns

+ Code + Text

```
# Remove duplicates based on track-level identifier
df_all = df_all.drop_duplicates(subset=['track_name', 'track_artist'])
```

Outlier Detection Summary

•Outliers were mainly observed in loudness and tempo, with a few in energy and danceability.
•Outliers represent stylistic differences rather than data errors.
•**So No removal applied to preserve differences in musical style**

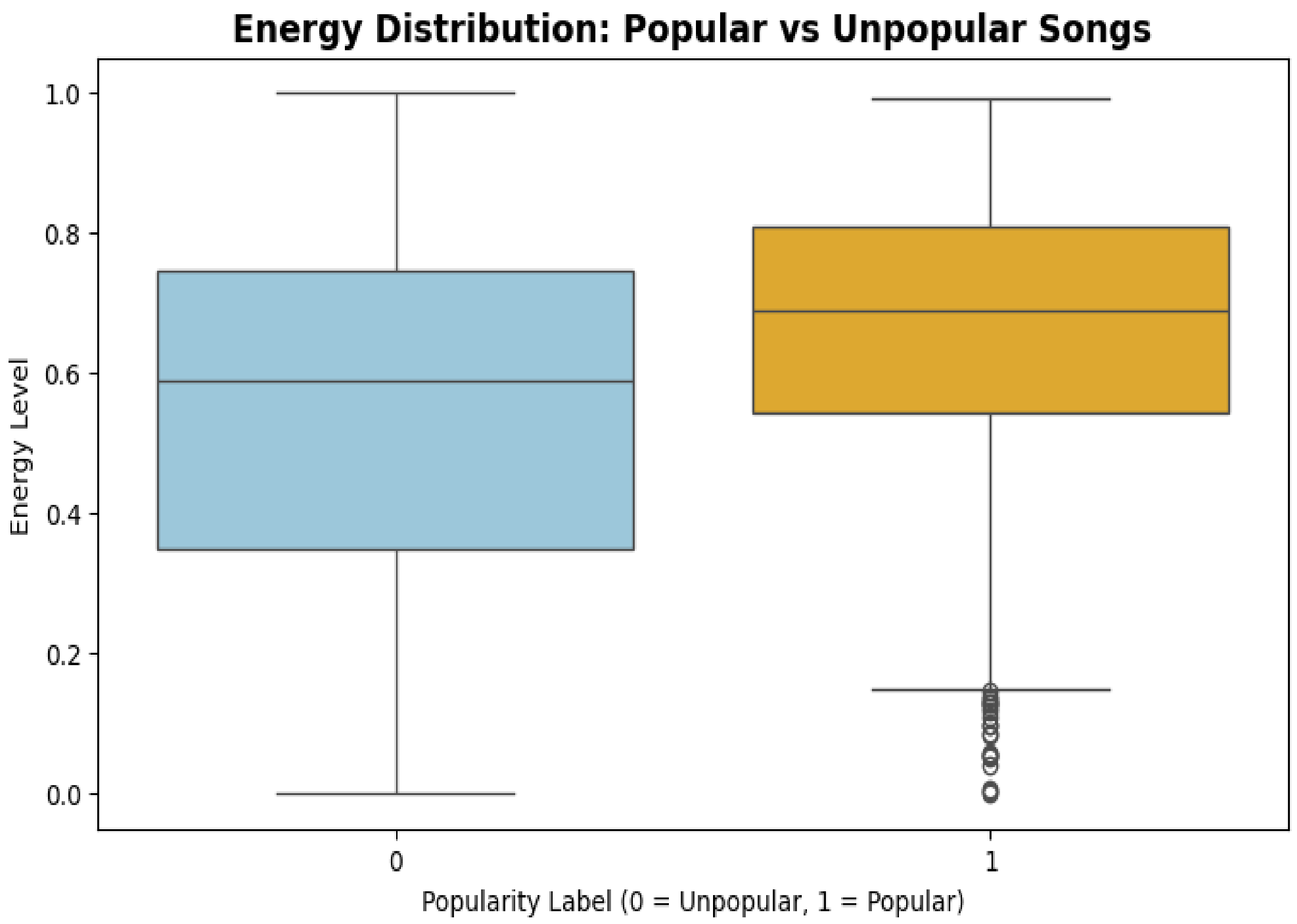


Distribution: Popular vs Unpopular

Binary Label Definitions:

1 = Popular (3,145 songs)

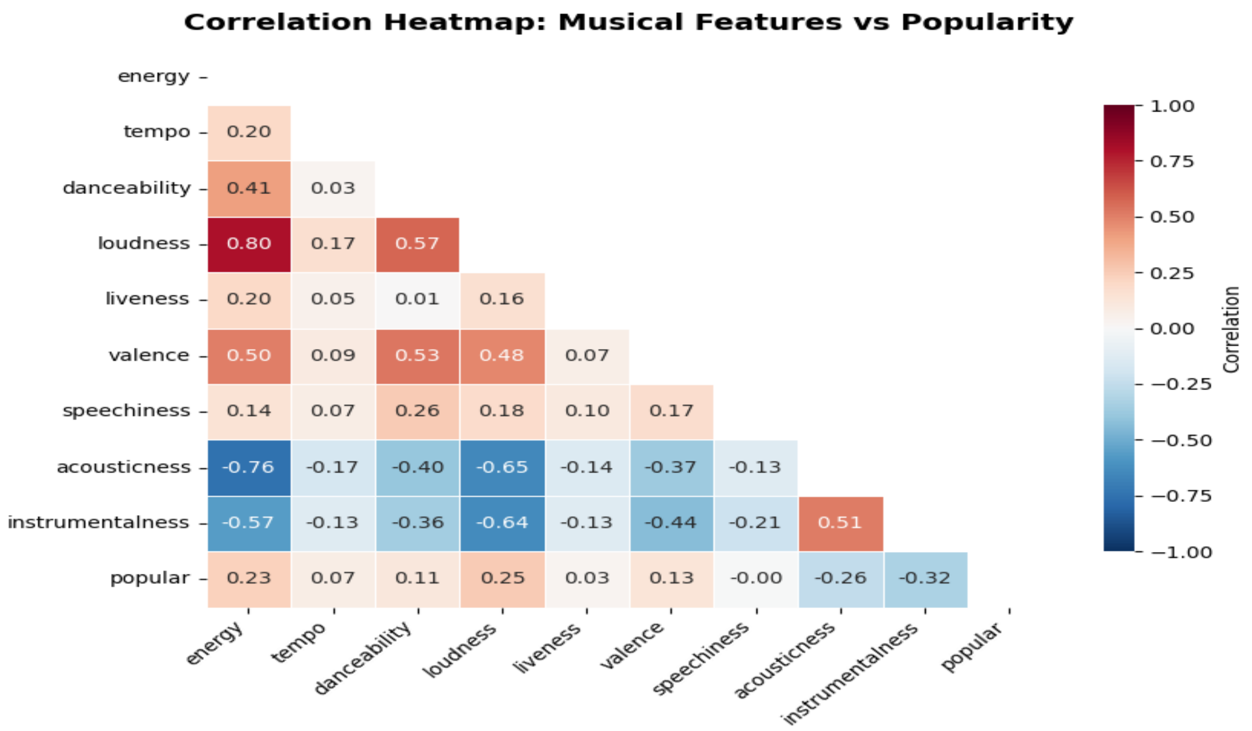
0 = Unpopular (1,686 songs)



Correlation Analysis

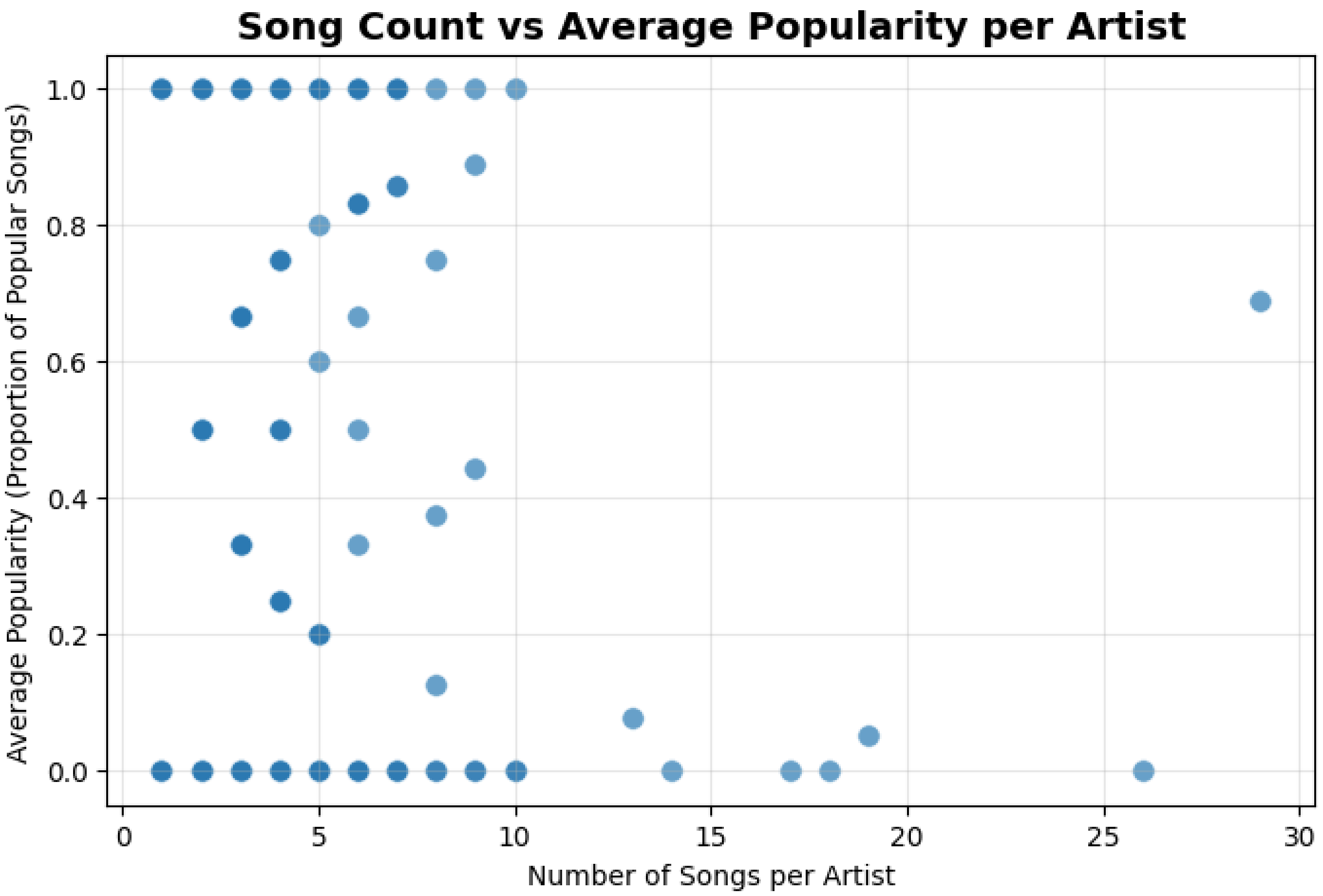
- Energy & loudness show the strongest positive correlations with popularity.
- Valence & danceability have weaker positive effects.
- Acousticness & instrumentalness correlate negatively with popularity.
- Energy & loudness are highly correlated with each other.

Feature	Correlation with Popularity
popular	1.000000
loudness	0.251485
energy	0.228412
valence	0.127455
danceability	0.112391
tempo	0.072527
liveness	0.029468
speechiness	-0.001311
acousticness	-0.256489
instrumentalness	-0.323456



Artist-Level Analysis

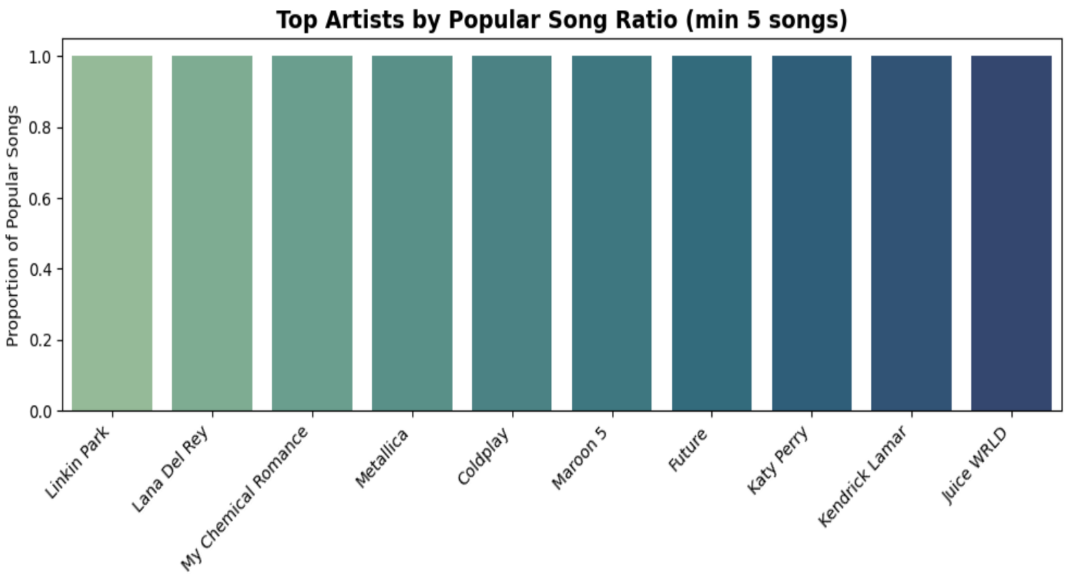
RQ1:
Is the number of songs an artist releases associated with the popularity of their songs?

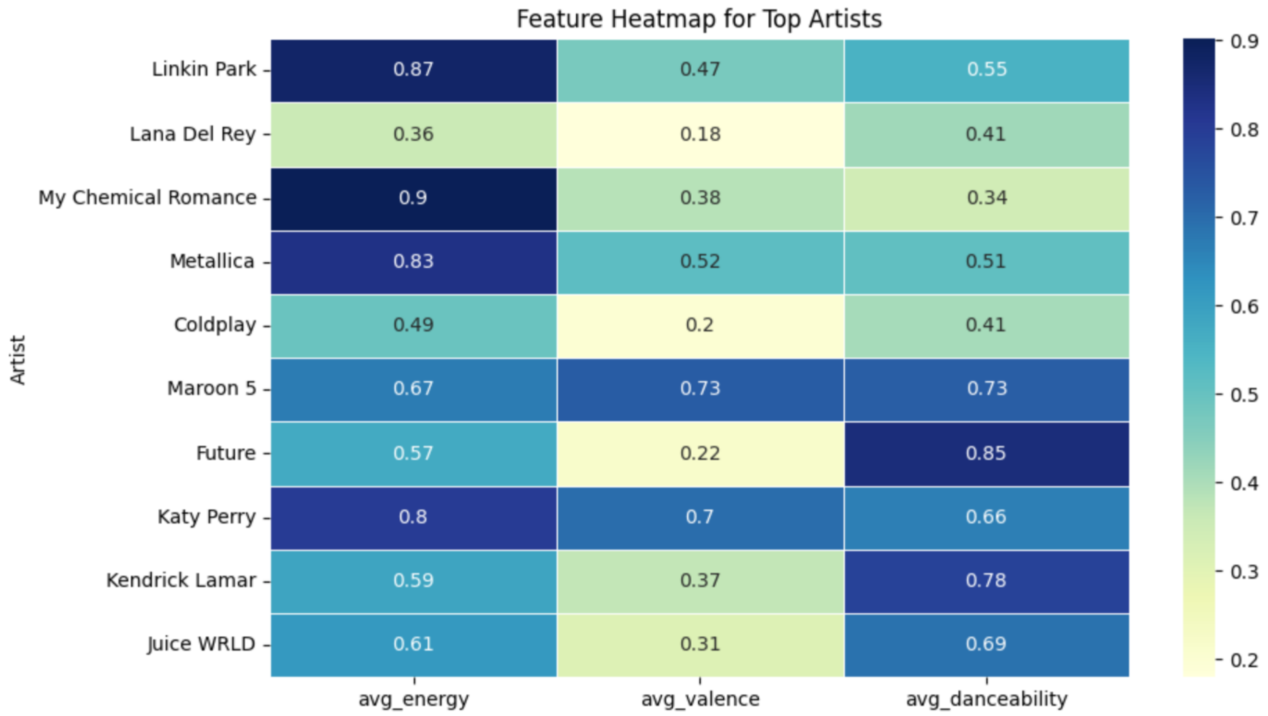


- **No strong relationship** between song count and average popularity *(r = 0.058)*.
- Artists with few tracks can still achieve **very high popularity**.
- **Quantity ≠ popularity**

RQ2:Which artists produce the highest proportion of popular songs?

- Used ≥5-track filter; several artists show 1.0 ratio—this reflects dataset labeling, not real-world popularity.
- Playlist genres ≠ artist genres → feature comparison instead of genre analysis.
- Heatmap: no single feature predicts popularity; artists succeed with different musical styles.
- Conclusion: popularity varies by how well artists express their style, not one feature.





Linear Regression

Linear Regression

Goal

Evaluate whether individual audio features can meaningfully predict a song's popularity.

Metrics

Predictors: **acousticness, instrumentalness, energy, valence, danceability, loudness**, and other audio features.

Target variable: **popularity score**

Insights

The model explains **only ~8% of the variance** in popularity ($R^2 \approx 0.085$).

No single audio feature strongly predicts popularity.

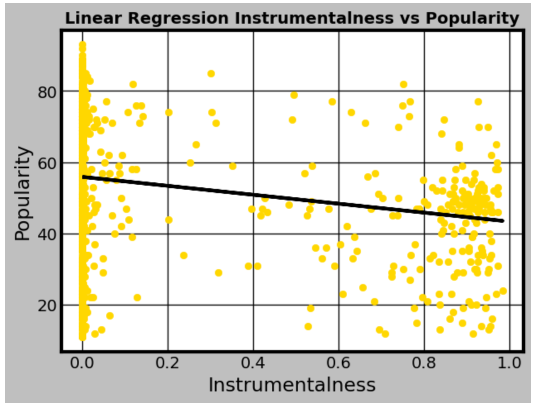
Popularity is influenced by **external factors** beyond audio characteristics (marketing, fanbase, exposure, playlisting).

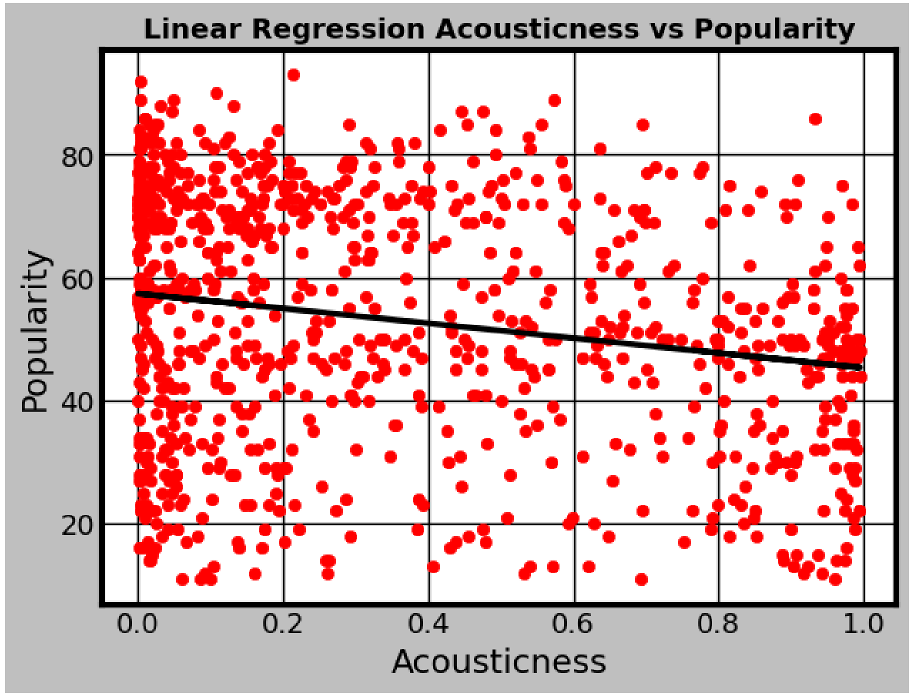
Result: linear regression has **very limited predictive power** for popularity.

Linear Regression Continued…

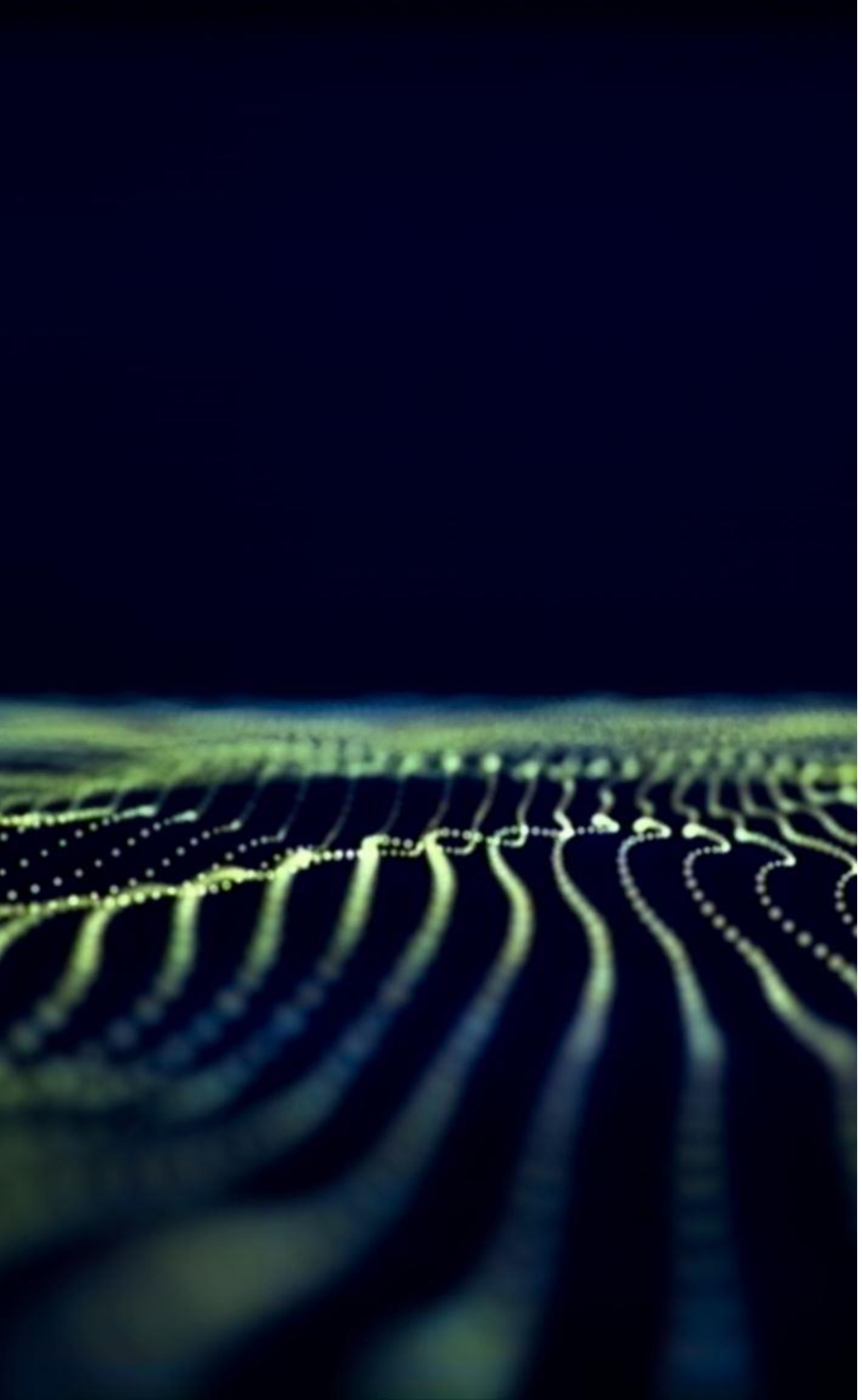
- The two biggest predictors from the features on popularity were acousticness and instrumentalness
- Deeper analysis still reveals that there is no clear linear correlation between any one of an audios features
- This is due to the basis of a songs popularity being much larger than the sum of its audio features
- Popularity is based more closely upon promotion, label, virality and artist recognition and number of other metrics that cannot be quantified

	seed27	seed19	seed34
energy	0.049126	0.042531	0.032821
tempo	0.002178	0.002304	0.005325
danceability	0.026880	0.014878	0.012935
loudness	0.053271	0.048631	0.041581
liveness	-0.001133	0.000892	-0.000552
valence	0.012468	0.002167	0.010384
speechiness	-0.001227	-0.000916	-0.003650
instrumentalness	0.075593	0.055098	0.055293
acousticness	0.055870	0.067329	0.047294
duration_s	-0.001885	0.000271	-0.001711



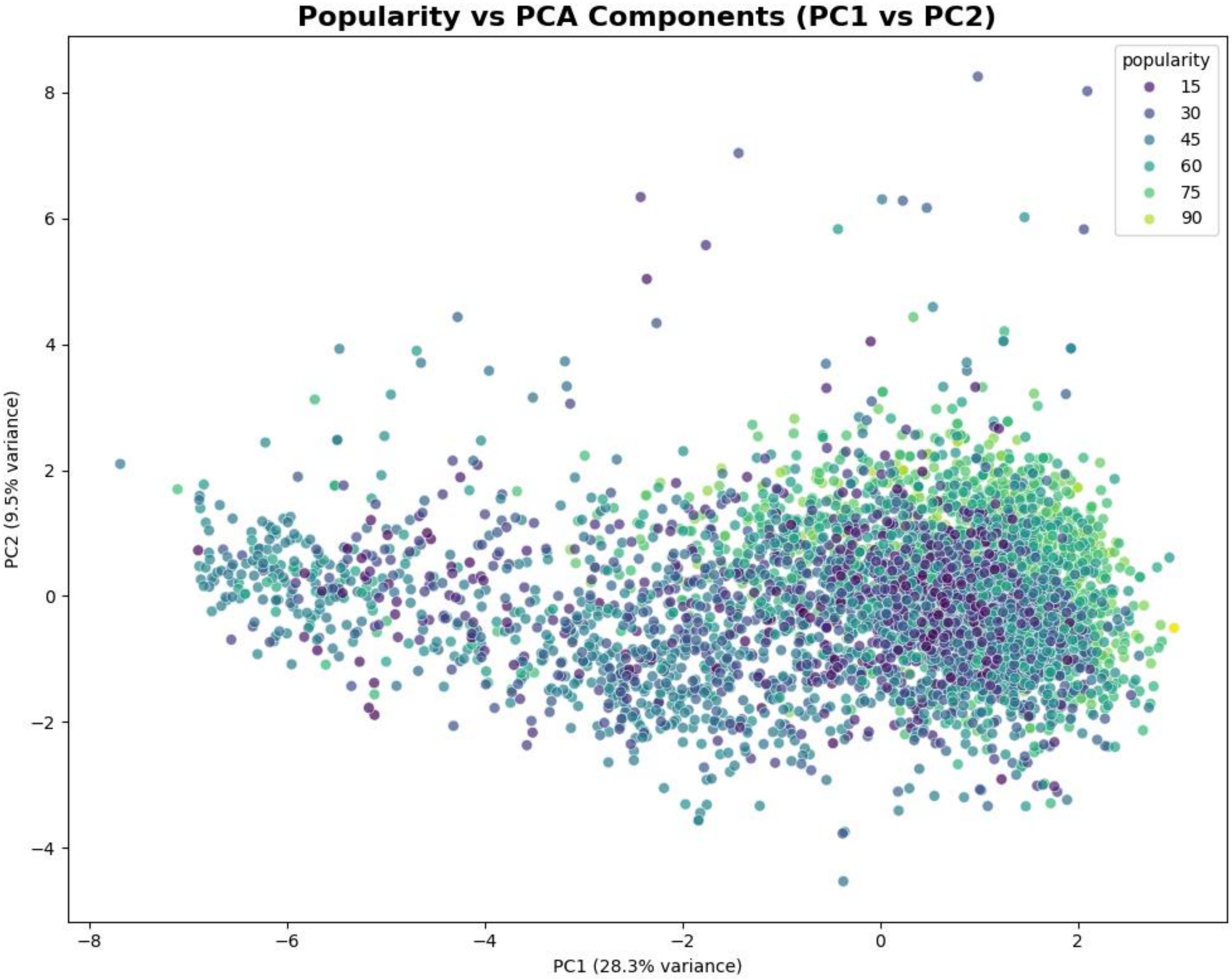


PCA Analysis



What is PCA analysis

- Raw Data
- Standardize Data
- Center Data by Distance
- Build Covariance Matrix
- Eigenvector Decomposition
- Principal Component Creation
- Feature Vector Projection


Popularity vs PCA Components (PC1 vs PC2)
popularity
15
30
45
60
75
90
PC2 (9.5% variance)
PC1 (28.3% variance)
8
6
4
2
0
−2
−4
−8
−6
−4
−2
0
2

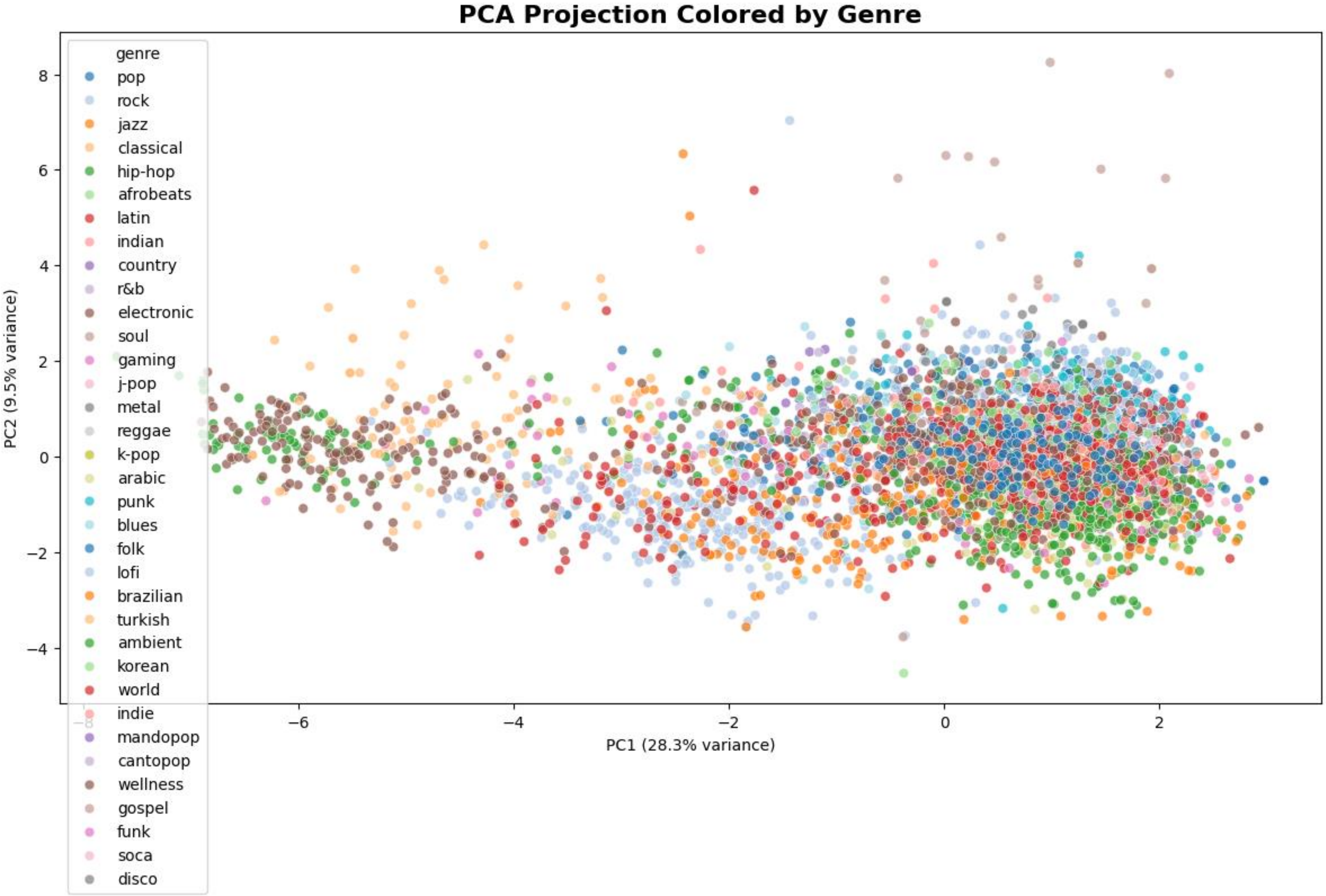
Genre-Level Trends

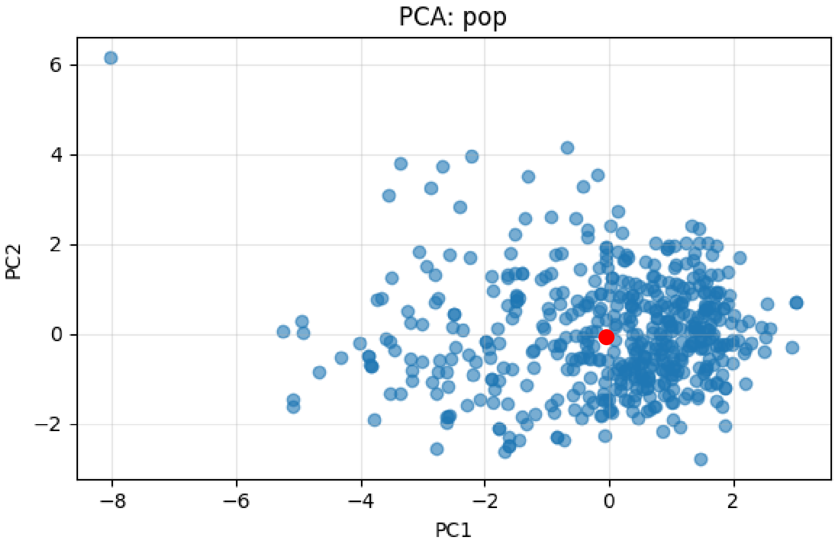
Comparing Average Popularity by Genre

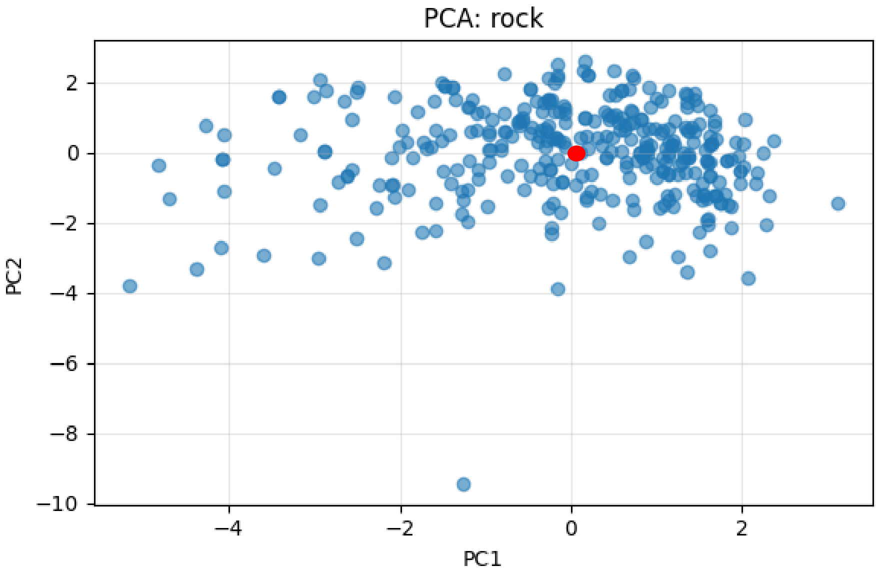
Example Comparisons: Pop vs. Rock vs. Hip-Hop

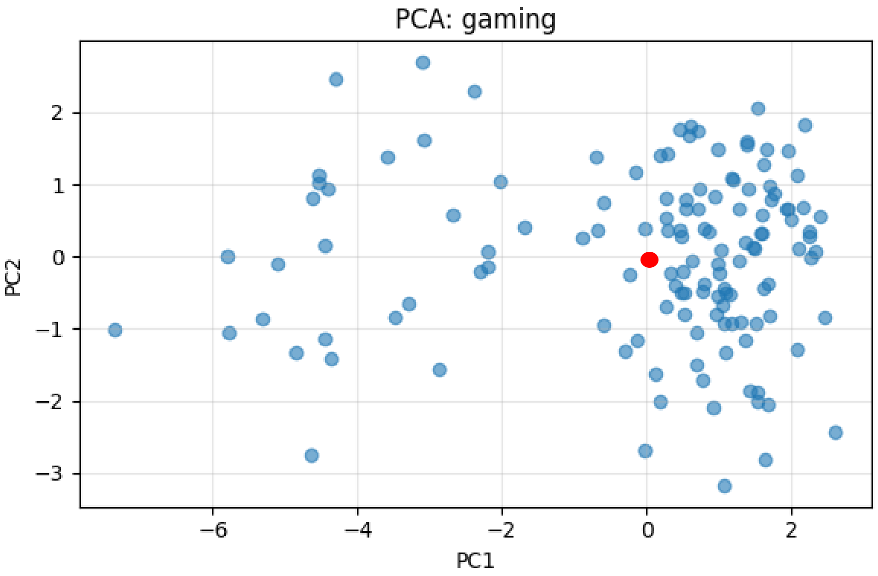
Popularity distribution by genre

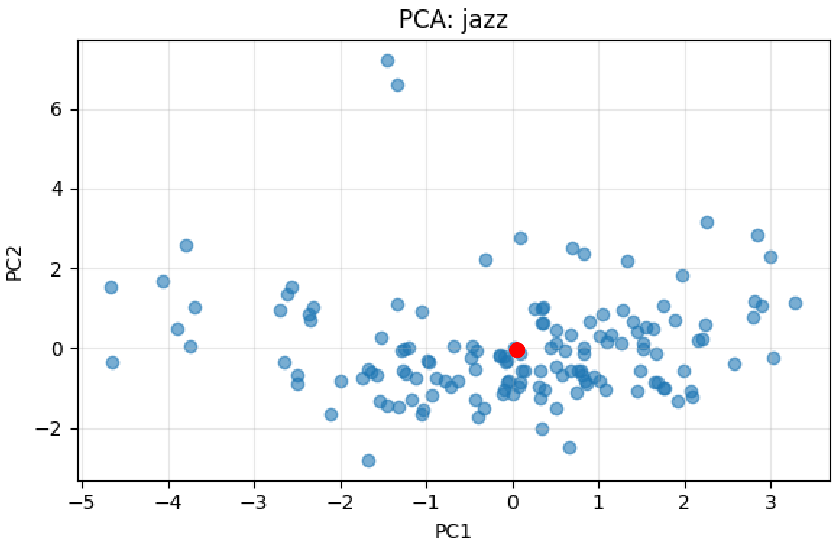
Addressing lack of cohesion in variance data resultant from PCA analysis.

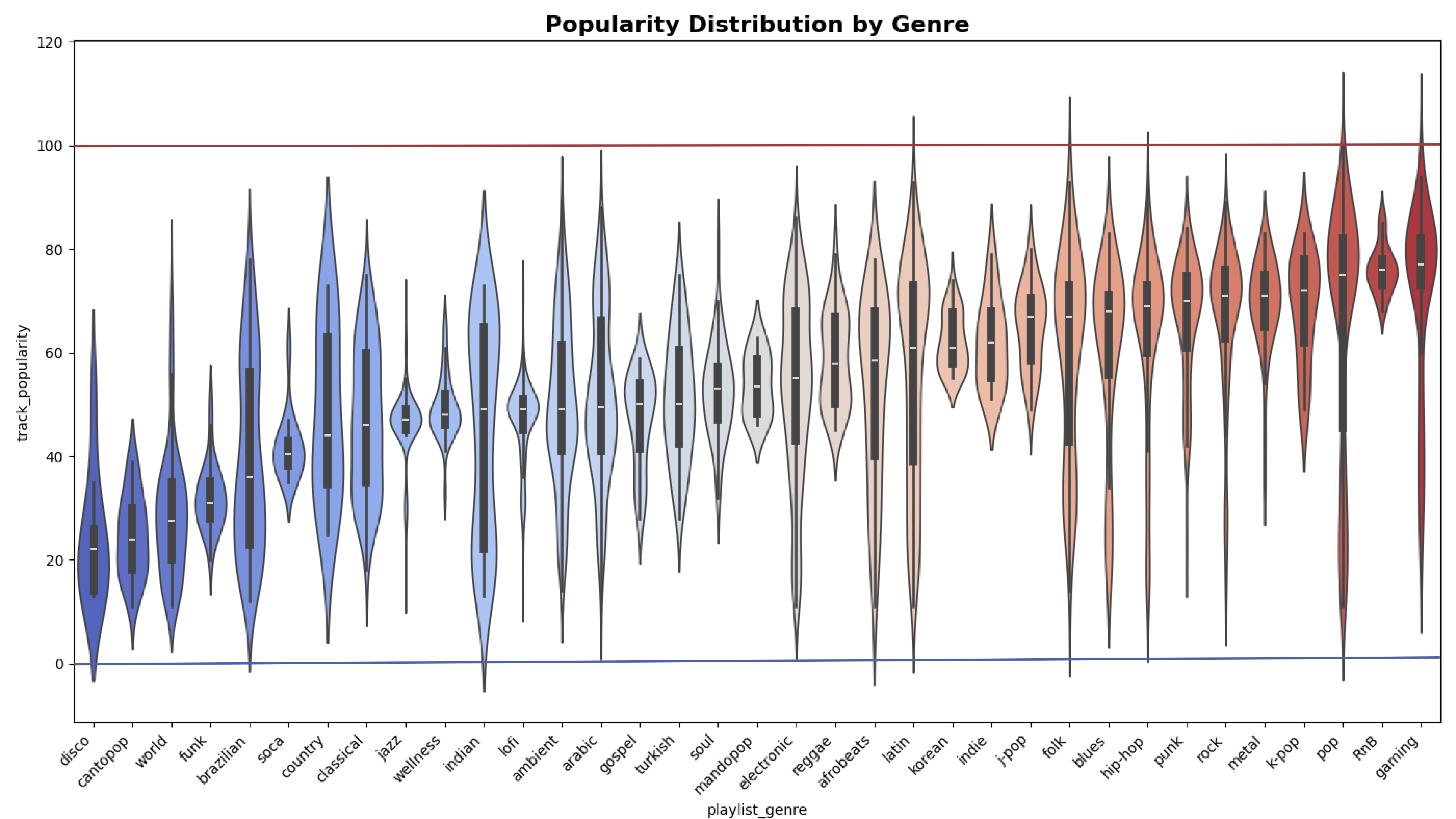

PCA Projection Colored by Genre
PC2 (9.5% variance)
PC1 (28.3% variance)
genre
pop
rock
jazz
classical
hip-hop
afrobeats
latin
indian
country
r&b
electronic
soul
gaming
j-pop
metal
reggae
k-pop
arabic
punk
blues
folk
lofi
brazilian
turkish
ambient
korean
world
indie
mandopop
cantopop
wellness
gospel
funk
soca
disco


PCA: pop
PC1
PC2


PCA: rock
PC1
PC2


PCA: gaming
PC1
PC2


PCA: jazz
PC1
PC2



Popularity Distribution by Genre
track_popularity
playlist_genre
120
100
80
60
40
20
0
disco
cantopop
world
funk
brazilian
soca
country
classical
jazz
wellness
indian
lofi
ambient
arabic
gospel
turkish
soul
mandopop
electronic
reggae
afrobeats
latin
korean
indie
j-pop
folk
blues
hip-hop
punk
rock
metal
k-pop
pop
RnB
gaming

Future Research

- Revise Song Based Analysis and inspect Spotify Methods
- Explore how artists' popularity changes over time using longitudinal data
- Using more quantifiable data such as sales and charting songs to measure popularity

Takeaways

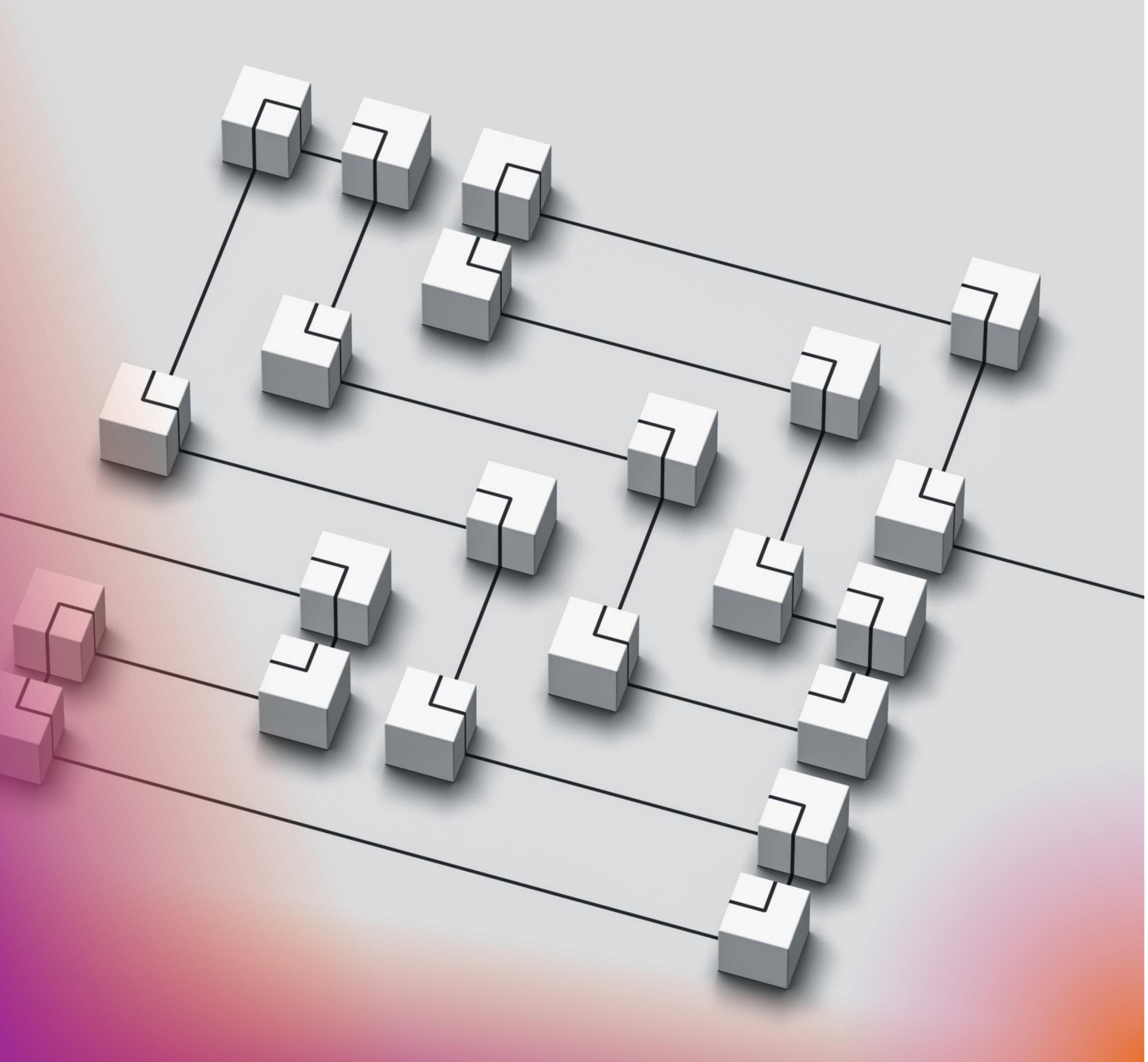
Music's audio features cannot predict popularity. It's much more dynamic than the sum of its features

- Clean and reliable data is the foundation.

- Popularity is heavily influenced by Artist Identity and Genre Diversity.

Genre does have a characteristic type, but it is very diffuse

Music cannot be easily reduced to static metrics, and instead require a deeper dynamic analysis



References

- Data Source:

  - Spotify Web API

- Tools & Libraries:

  - Python (Pandas, NumPy)

  - Visualization (Seaborn, Matplotlib)

  - Modeling (Scikit-learn)

- Project Scripts:

  - Team Data Cleaning and Modeling Notebooks

Thank You

Questions?

Thanks!

Backup 1: Popular Proportion

```python
# Combine Datasets
# Add a binary label for popularity: 1 = high popularity, 0 = low popularity
high_popularity['popular'] = 1
low_popularity['popular'] = 0
```

```python
# Calculate the average popularity ratio for each artist
artist_stats = df_all.groupby('track_artist').agg(
    song_count=('popular','size'),
    popularity_ratio=('popular','mean'),
    avg_energy=('energy','mean'),
    avg_valence=('valence','mean'),
    avg_danceability=('danceability','mean')
).sort_values('popularity_ratio', ascending=False)

# Visualization
```

popularity_ratio = mean(popular)

Because *popular* contains only 0s and 1s, the mean is equal to the proportion of popular songs for each artist.

Backup 2: Minimum Song Count Filter (≥ 5 Songs)

```python
# Count the most frequently appearing artists
artist_counts = df_all['track_artist'].value_counts().head(15)
display(artist_counts)

# Calculate the average popularity ratio for each artist
artist_stats = df_all.groupby('track_artist').agg(
    song_count=('popular','size'),
    popularity_ratio=('popular','mean'),
    avg_energy=('energy','mean'),
    avg_valence=('valence','mean'),
    avg_danceability=('danceability','mean')
).sort_values('popularity_ratio', ascending=False)

# Visualization
top_artists = artist_stats[artist_stats['song_count'] > 5].head(10)
top_artists['popularity_ratio'].plot(kind='bar', figsize=(10,4))
plt.title('Top Artists by Popular Song Ratio (min 5 songs)')
plt.ylabel('Proportion of Popular Songs')
plt.show()

display(top_artists)
```

	count
track_artist	
Bad Bunny	29
Ren Avel	26
Asake	19
LoFi Waiter	18
Seyi Vibez	17
Bnxn	14
Wizkid	13
Yume.Play	10
Linkin Park	10
Burna Boy	10
Zinoleesky	9
Red Hot Chili Peppers	9
Céline Dion	9
c152	9
Green Day	9

dtype: int64

Backup 3:
Explained Variance, Loadings, and Covariance Matrix

Explained variance ratio:
[0.2828076 0.09514005 0.08069049 0.07466738 0.07114451 0.06694801 0.06321044 0.06145504 0.05873256 0.04512505 0.03644486 0.03214738 0.02177648 0.00971015]

Principal components:
acousticness danceability duration_ms energy instrumentalness key liveness loudness mode speechiness tempo time_signature track_popularity valence
[[-0.39678917 0.32512212 0.04016307 0.43512448 -0.37720565 0.03137643 0.10233163 0.44605623 -0.06263254 0.14834262 0.11245358 0.14970781 0.15577856 0.32855364]
[-0.18931969 -0.38803357 0.5709981 0.16105669 -0.13399602 -0.13229378 0.08846692 0.03951755 0.31151537 -0.3934771 0.22320004 -0.18580108 0.20116041 -0.20023347]
[-0.00952265 -0.11868681 0.15827841 0.03171219 0.00882333 0.6972229 0.07063997 0.00288897 -0.60029014 -0.04400455 0.12011668 -0.2765175 0.08683075 -0.08862789]
[0.02074633 -0.1838349 -0.14415736 0.02716068 0.03897962 -0.09862778 0.7406232 -0.03820687 0.03732172 0.38777283 0.40757874 -0.05960801 -0.22192102 -0.11109803]
[0.06228418 0.08677893 -0.40636466 -0.08963855 -0.04632068 -0.07396589 -0.20971502 -0.03704197 0.14315551 0.08684863 0.40324834 -0.51953518 0.54670662 0.05480318]
[-0.00791111 0.02735314 0.12150039 0.03301361 0.0846522 -0.01022766 -0.49518311 0.00176745 -0.04005546 0.0365118 0.67986814 0.10509642 -0.49727295 0.08586654]
[-0.03289274 -0.11648521 -0.18548074 -0.04865746 0.05677778 0.45020569 -0.02097444 -0.05081558 0.31289861 0.00174328 0.21515702 0.659482 0.33798908 -0.21515886]
[0.06096796 -0.06881116 0.42993322 -0.14074767 -0.16494231 -0.1889713 -0.22628197 -0.07859726 -0.17115035 0.70357598 -0.05157366 0.11333411 0.27076712 -0.23123015]
[0.12050046 0.06182413 0.18921705 -0.05355999 -0.15596164 0.48815346 -0.03315587 -0.01941903 0.60562066 0.29952533 -0.16718618 -0.29381134 -0.2377146 0.22718946]
[0.41496922 0.19167588 0.34493342 -0.25101705 0.02198427 -0.04407569 0.24876109 -0.18146817 -0.07862836 -0.16091992 0.20170164 0.19497698 0.24049588 0.58584815]
[-0.23027265 0.56818612 0.26705594 0.01423687 0.67330999 0.03128878 0.10294154 0.03550787 0.11729252 0.05508042 0.03789602 -0.09946104 0.10015009 -0.21973696]
[-0.31268134 -0.46302767 -0.03219122 0.28395692 0.45750528 -0.01236887 -0.09861705 -0.25906571 -0.01334757 0.20809001 -0.11006625 0.00430736 0.1190436 0.50007938]
[0.62487078 -0.17891997 0.01470656 0.38432587 0.29843654 -0.01779103 -0.06326186 0.56748699 0.03301216 0.06742307 -0.03182516 0.00248477 0.0784138 -0.03706278]
[0.26358295 0.24383775 -0.00456598 0.67621211 -0.14796378 0.0032898 -0.00729522 -0.60344343 0.00373762 -0.0218234 0.00262524 0.00830382 0.00789986 -0.16448019]]

	acousticness	danceability	duration_ms	energy	instrumentalness	key	liveness	loudness	mode	speechiness	tempo	time_signature	track_popularity	valence
acousticness	1.000207	-0.381318	-0.125131	-0.751178	0.511804	-0.021501	-0.136997	-0.647398	0.052126	-0.119197	-0.173051	-0.199106	-0.233087	-0.351718
danceability	-0.381318	1.000207	-0.140745	0.387217	-0.349568	0.024572	0.000984	0.557932	-0.122744	0.256309	0.019802	0.199493	0.128472	0.513714
duration_ms	-0.125131	-0.140745	1.000207	0.125348	-0.141090	-0.000418	-0.002964	0.070681	0.030252	-0.096676	0.032071	-0.018877	0.021198	-0.036013
energy	-0.751178	0.387217	0.125348	1.000207	-0.564894	0.040002	0.192652	0.798993	-0.079151	0.133931	0.197645	0.196918	0.195023	0.491818
instrumentalness	0.511804	-0.349568	-0.141090	-0.564894	1.000207	-0.024980	-0.119160	-0.641841	0.025429	-0.209171	-0.124888	-0.139113	-0.263188	-0.427741
key	-0.021501	0.024572	-0.000418	0.040002	-0.024980	1.000207	0.007333	0.045839	-0.149986	0.016735	0.013778	-0.003709	0.028709	0.033481
liveness	-0.136997	0.000984	-0.002964	0.192652	-0.119160	0.007333	1.000207	0.154254	-0.014234	0.097289	0.047025	0.029172	0.022283	0.067179
loudness	-0.647398	0.557932	0.070681	0.798993	-0.641841	0.045839	0.154254	1.000207	-0.097102	0.178735	0.161902	0.217944	0.217470	0.471390
mode	0.052126	-0.122744	0.030252	-0.079151	0.025429	-0.149986	-0.014234	-0.097102	1.000207	-0.087344	0.007654	-0.003079	0.003386	-0.062318
speechiness	-0.119197	0.256309	-0.096676	0.133931	-0.209171	0.016735	0.097289	0.178735	-0.087344	1.000207	0.064001	0.108641	0.019055	0.161473
tempo	-0.173051	0.019802	0.032071	0.197645	-0.124888	0.013778	0.047025	0.161902	0.007654	0.064001	1.000207	-0.009585	0.060059	0.088135
time_signature	-0.199106	0.199493	-0.018877	0.196918	-0.139113	-0.003709	0.029172	0.217944	-0.003079	0.108641	-0.009585	1.000207	0.003295	0.140105
track_popularity	-0.233087	0.128472	0.021198	0.195023	-0.263188	0.028709	0.022283	0.217470	0.003386	0.019055	0.060059	0.003295	1.000207	0.096797
valence	-0.351718	0.513714	-0.036013	0.491818	-0.427741	0.033481	0.067179	0.471390	-0.062318	0.161473	0.088135	0.140105	0.096797	1.000207