Title Page

Ginyard International Co.

Creating Cohorts of Songs

Group 3

Title Page Reporters

Group Members

Group 3

Members

Alanood

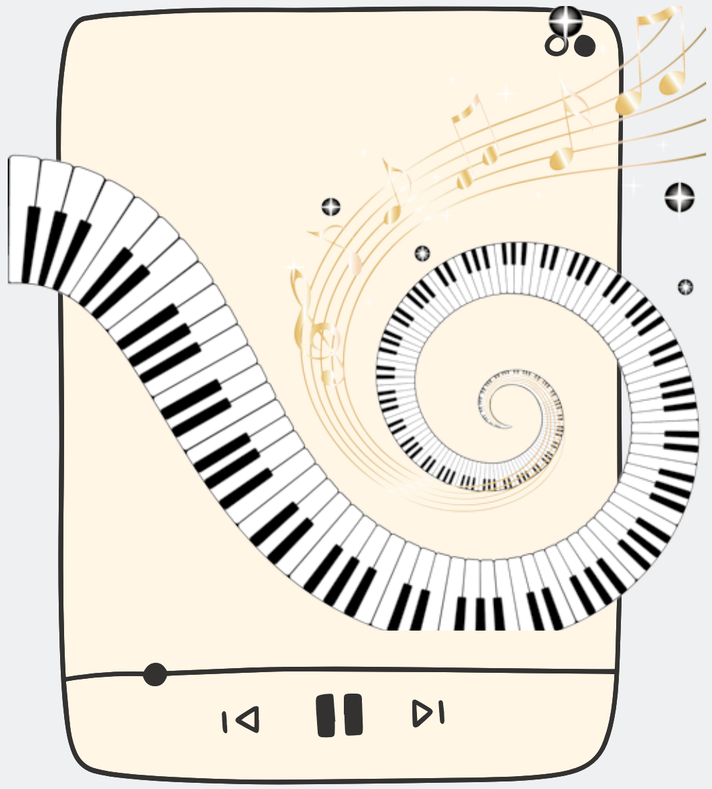
Raghad

Reem

Raghad

Introduction

With continuous growth of web applications around the globe, it is a challenge to
find the suitable information needed for the user in a limited time. most of the business revolves around the correct search of the data
Without a proper recommender system it is very difficult to get required information. Web applications use recommender systems to provide suitable data to users based on their choices and interests.



Insert your topic here

Process

solve this project

1. Initial data inspection and data cleaning
2. Perform Exploratory Data Analysis and Feature Engineering
3. Perform Cluster Analysis

Check whether the data has duplicates, missing values, irrelevant (erroneous entries) values, or outliers

In [1]:
```
import pandas as pd
import matplotlib.pyplot as plt
import seaborn as sns
import plotly.express as px
```

In [2]:
```
df = pd.read_csv('rolling_stones_spotify.csv')
df
```

Out[2]:

	Unnamed: 0	name	album	release_date	track_number	id	uri	acousticness	danc
0	0	Concert Intro Music - Live	Licked Live In NYC	2022-06-10	1	2IEkywLJ4ykbhi1yRQvmsT	spotify:track:2IEkywLJ4ykbhi1yRQvmsT	0.0824	
1	1	Street Fighting Man - Live	Licked Live In NYC	2022-06-10	2	6GVgVJBKkGJoRfarYRvGTU	spotify:track:6GVgVJBKkGJoRfarYRvGTU	0.4370	
2	2	Start Me Up - Live	Licked Live In NYC	2022-06-10	3	1Lu761pZDdBTGpzxaQoZNW	spotify:track:1Lu761pZDdBTGpzxaQoZNW	0.4160	
3	3	If You Can't Rock Me - Live	Licked Live In NYC	2022-06-10	4	1agTQzOTUnGNggyckEqiDH	spotify:track:1agTQzOTUnGNggyckEqiDH	0.5670	
4	4	Don't Stop - Live	Licked Live In NYC	2022-06-10	5	7piGJR8YndQBQWVXv6ktQw	spotify:track:7piGJR8YndQBQWVXv6ktQw	0.4000	
...	...	...	...	...	...	...	...	...	
1605	1605	Carol	The Rolling Stones	1964-04-16	8	08I7M5UpRnffGI0FyuRiQZ	spotify:track:08I7M5UpRnffGI0FyuRiQZ	0.1570	
1606	1606	Tell Me	The Rolling Stones	1964-04-16	9	3JZIIQBsTM8WWwoJdzFDLhx	spotify:track:3JZIIQBsTM8WWwoJdzFDLhx	0.0576	
1607	1607	Can I Get A Witness	The Rolling Stones	1964-04-16	10	0t2qvfSBQ3Y08lzRRoVTdb	spotify:track:0t2qvfSBQ3Y08lzRRoVTdb	0.3710	
1608	1608	You Can Make It If You Try	The Rolling Stones	1964-04-16	11	5ivIs5vwSj0RChOIvIY3On	spotify:track:5ivIs5vwSj0RChOIvIY3On	0.2170	
1609	1609	Walking The Dog	The Rolling Stones	1964-04-16	12	43SkTJJ2xleDaeiE4TIM70	spotify:track:43SkTJJ2xleDaeiE4TIM70	0.3830	

1610 rows × 18 columns

In [3]:
```
# check for duplicated
df[df.duplicated()]
```

Out[3]:

Unnamed: 0	name	album	release_date	track_number	id	uri	acousticness	danceability	energy	instrumentalness	liveness	loudness	speechines

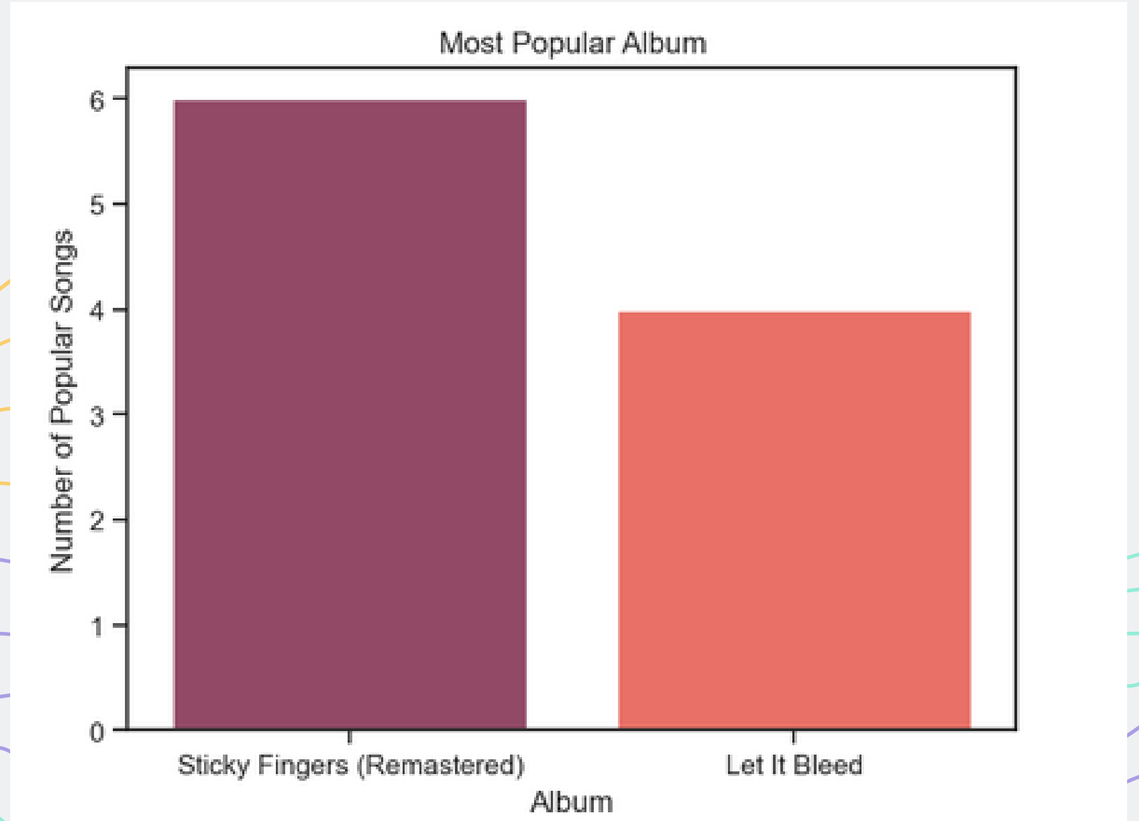
In [4]:
```
# check if there is null value
df.isnull().sum()
```

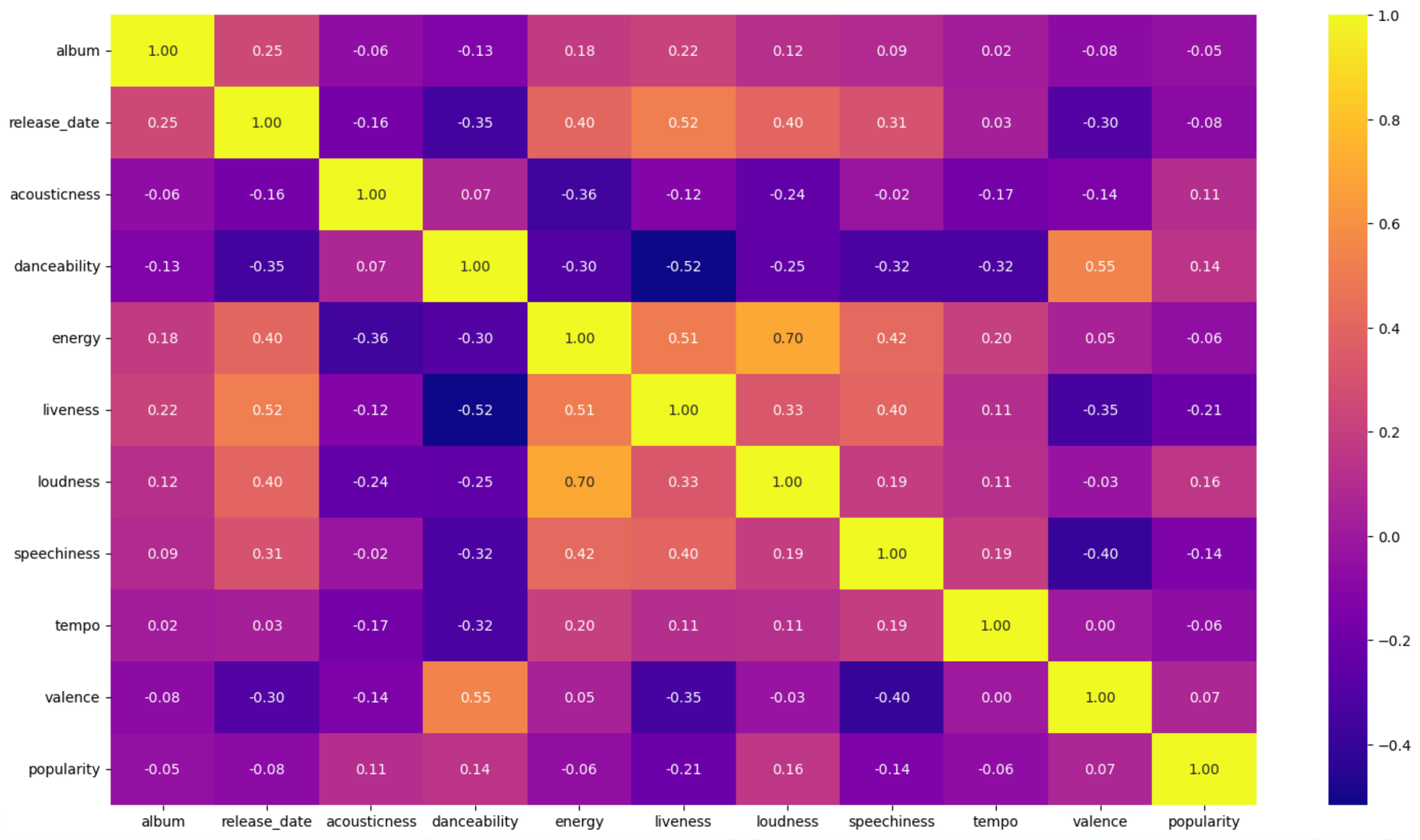
Out[4]:
```
Unnamed: 0          0
name                0
album               0
release_date        0
track_number        0
id                  0
uri                 0
acousticness        0
danceability        0
energy              0
instrumentalness    0
liveness            0
loudness            0
speechiness         0
tempo               0
valence             0
popularity          0
duration_ms         0
dtype: int64
```

In [5]:
```
df.info()
```

```
<class 'pandas.core.frame.DataFrame'>
RangeIndex: 1610 entries, 0 to 1609
Data columns (total 18 columns):
 #   Column            Non-Null Count  Dtype
---  ------            --------------  -----
 0   Unnamed: 0        1610 non-null   int64
 1   name              1610 non-null   object
 2   album             1610 non-null   object
 3   release_date      1610 non-null   object
 4   track_number      1610 non-null   int64
 5   id                1610 non-null   object
 6   uri               1610 non-null   object
 7   acousticness      1610 non-null   float64
 8   danceability      1610 non-null   float64
 9   energy            1610 non-null   float64
 10  instrumentalness  1610 non-null   float64
 11  liveness          1610 non-null   float64
 12  loudness          1610 non-null   float64
 13  speechiness       1610 non-null   float64
 14  tempo             1610 non-null   float64
 15  valence           1610 non-null   float64
 16  popularity        1610 non-null   int64
 17  duration_ms       1610 non-null   int64
dtypes: float64(9), int64(4), object(5)
memory usage: 226.5+ KB
```

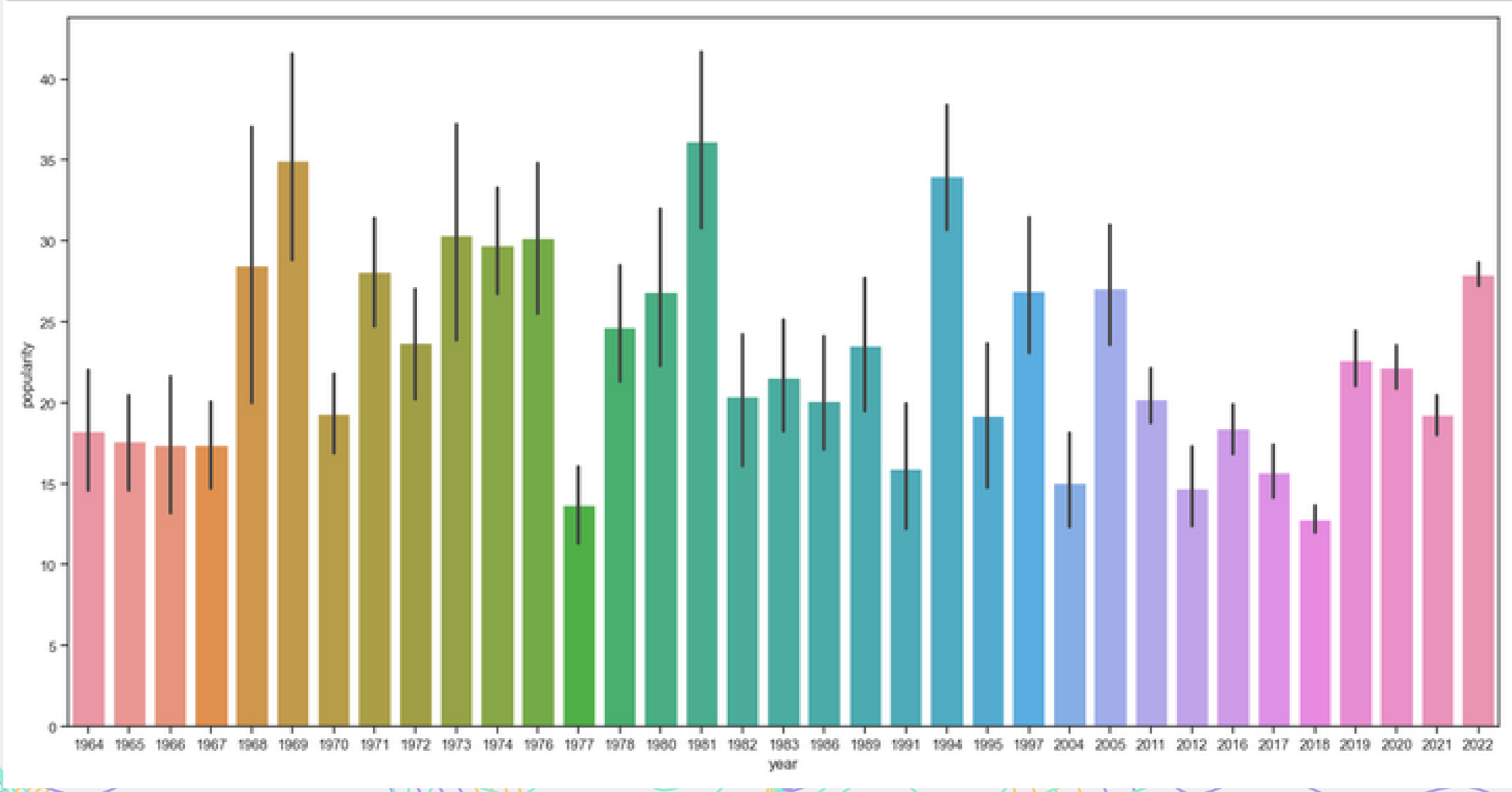
Use appropriate visualizations to find out which two albums should be recommended to anyone based on the number of popular songs in an album.




album 1.00 0.25 -0.06 -0.13 0.18 0.22 0.12 0.09 0.02 -0.08 -0.05
release_date 0.25 1.00 -0.16 -0.35 0.40 0.52 0.40 0.31 0.03 -0.30 -0.08
acousticness -0.06 -0.16 1.00 0.07 -0.36 -0.12 -0.24 -0.02 -0.17 -0.14 0.11
danceability -0.13 -0.35 0.07 1.00 -0.30 -0.52 -0.25 -0.32 -0.32 0.55 0.14
energy 0.18 0.40 -0.36 -0.30 1.00 0.51 0.70 0.42 0.20 0.05 -0.06
liveness 0.22 0.52 -0.12 -0.52 0.51 1.00 0.33 0.40 0.11 -0.35 -0.21
loudness 0.12 0.40 -0.24 -0.25 0.70 0.33 1.00 0.19 0.11 -0.03 0.16
speechiness 0.09 0.31 -0.02 -0.32 0.42 0.40 0.19 1.00 0.19 -0.40 -0.14
tempo 0.02 0.03 -0.17 -0.32 0.20 0.11 0.11 0.19 1.00 0.00 -0.06
valence -0.08 -0.30 -0.14 0.55 0.05 -0.35 -0.03 -0.40 0.00 1.00 0.07
popularity -0.05 -0.08 0.11 0.14 -0.06 -0.21 0.16 -0.14 -0.06 0.07 1.00
album release_date acousticness danceability energy liveness loudness speechiness tempo valence popularity
1.0 0.8 0.6 0.4 0.2 0.0 −0.2 −0.4

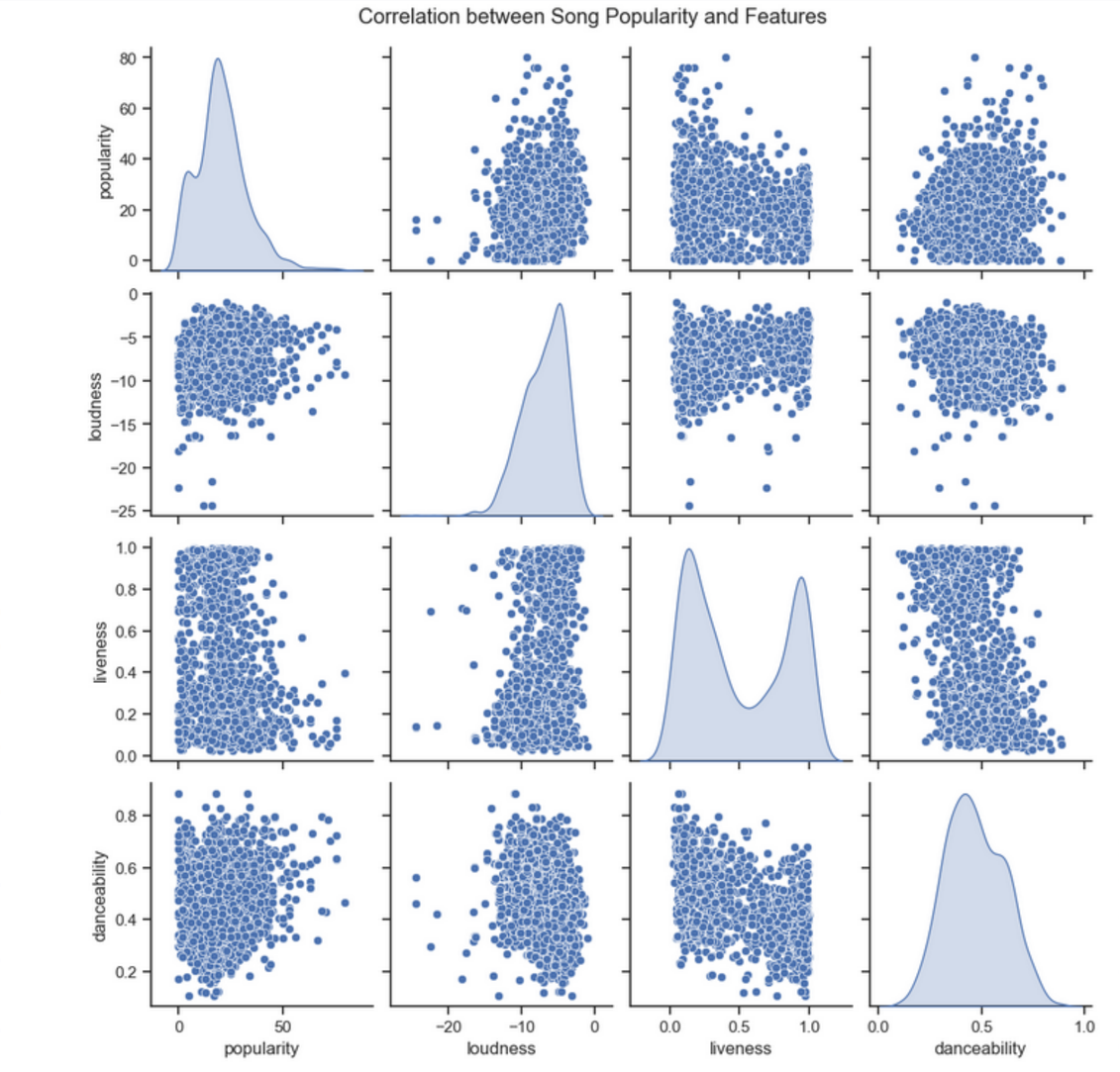
Insert your topic here

Perform exploratory data analysis to dive deeper into different features of songs and identify the pattern.



Insert your topic here

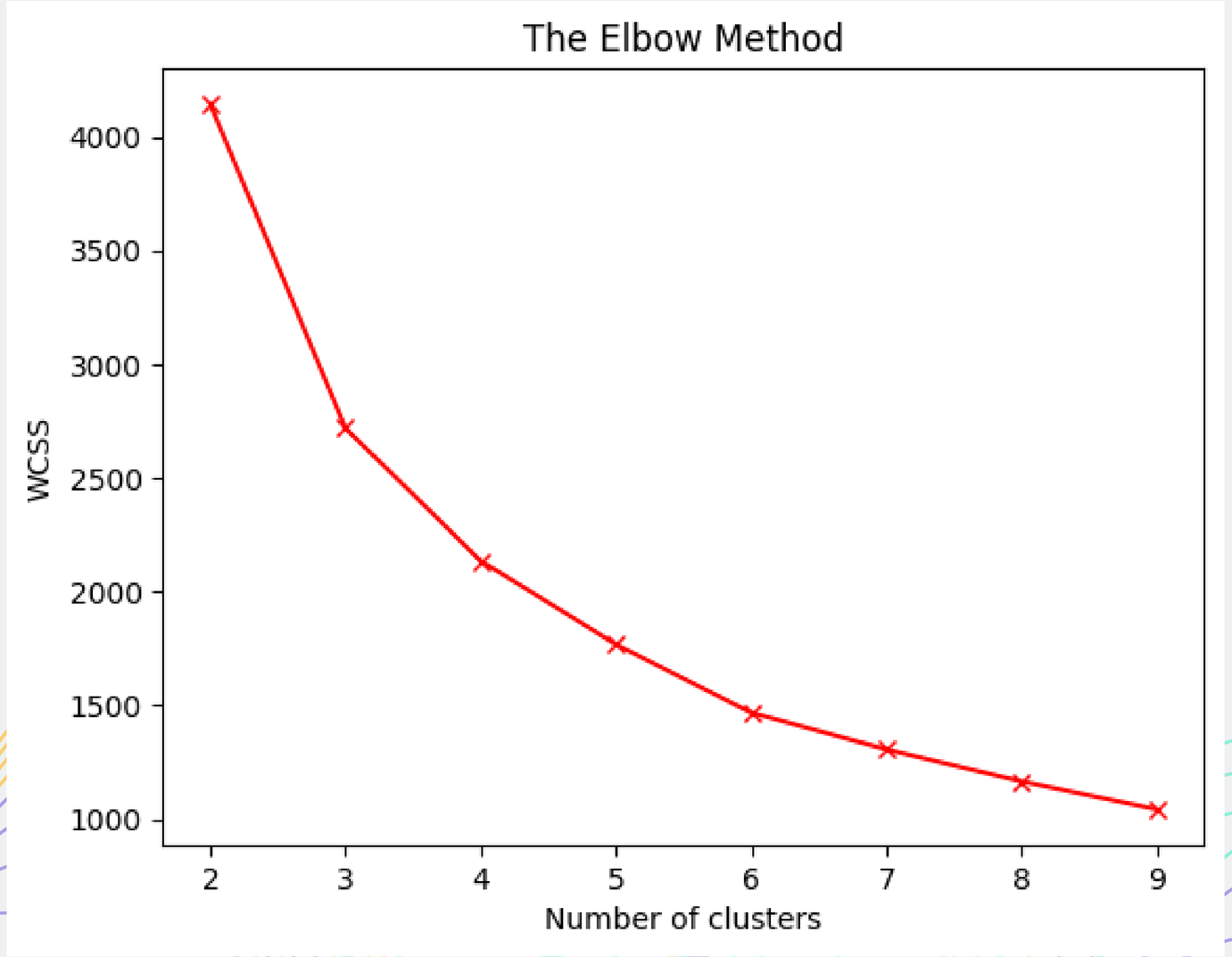
Discover how a song's popularity relates to various factors and how this has changed over time.

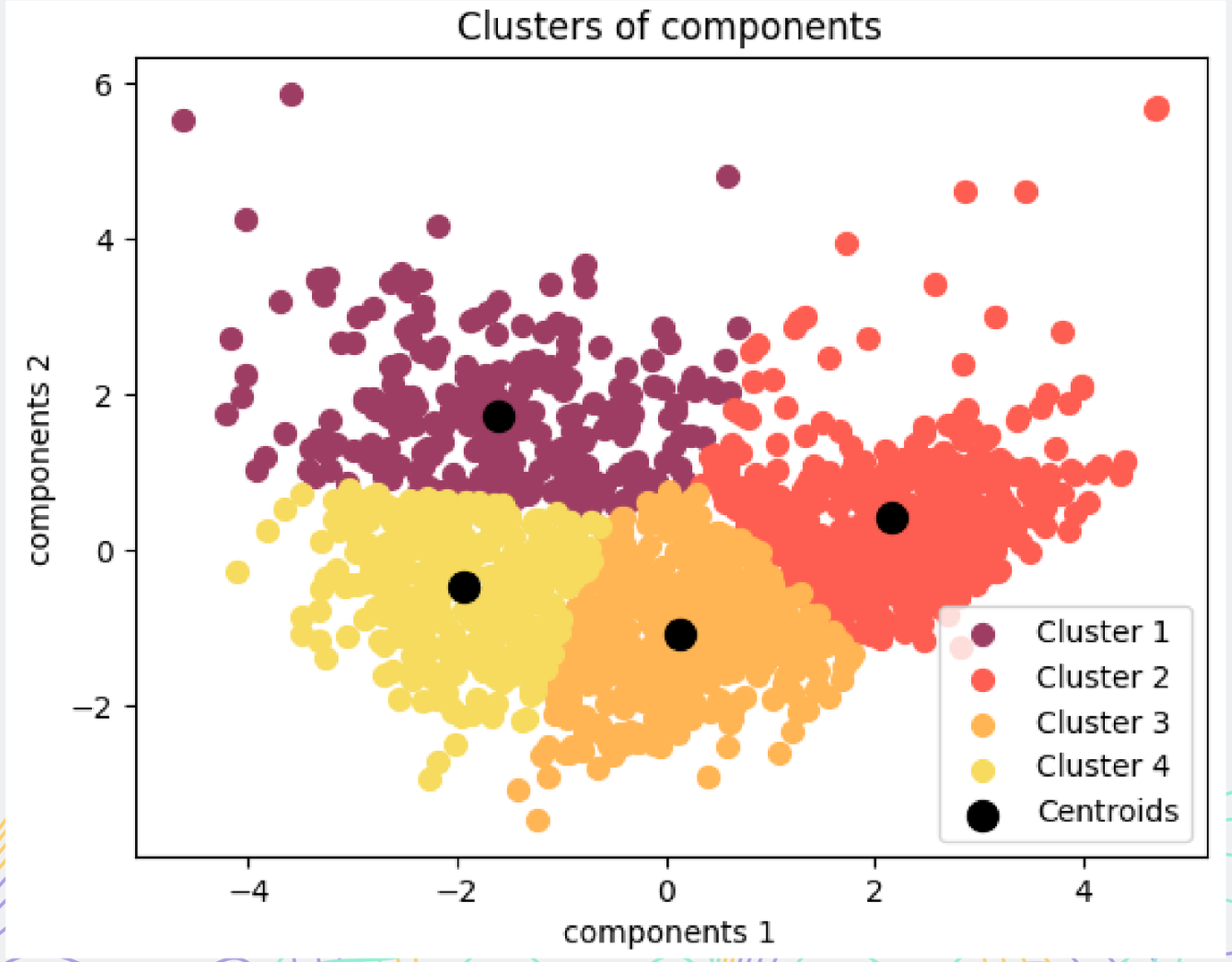


Comment on the importance of dimensionality reduction techniques, share your ideas and explain your observations.

```
[8]  pca = PCA(n_components=2)
     pca.fit(scaled_df)
     X_new = pca.transform(scaled_df)
     X_new.shape

     (1610, 2)
```


The Elbow Method
WCSS
4000
3500
3000
2500
2000
1500
1000
2
3
4
5
6
7
8
9
Number of clusters


Clusters of components
components 2
components 1
6
4
2
0
−2
−4
−2
0
2
4
Cluster 1
Cluster 2
Cluster 3
Cluster 4
Centroids

Title Page | Reporters | Introduction | Overview

Insert your topic here

Questions !

Title Page | Reporters | Introduction | Overview

Insert your topic here

Questions !

Give me 3 examples of a recommendation?

Insert your topic here

Questions !

diffrent between supervised learning and unsopervised?

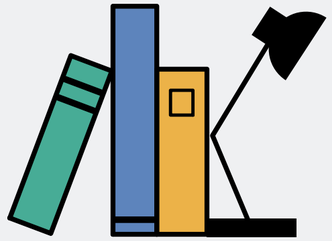
Insert your topic here

Questions !

how many type of machine learning?

Insert your topic here

The End ! Thank You For Listening :)



Sources

01 **Insert a website or book title for references**