

```
# Import necessary libraries
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
%matplotlib inline

# Load dataset
spotify_data = pd.read_csv('/content/spotify_dataset.csv')
spotify_data.head()
# Check for missing values
spotify_data.isnull().sum()

# Remove duplicates
spotify_data.drop_duplicates(inplace=True)
# Distribution of target variable
sns.countplot(x='playlist_genre', data=spotify_data)

# Distribution of numerical features
spotify_data.hist(figsize=(20, 12))
plt.show()

# Distribution of categorical features
fig, axs = plt.subplots(ncols=2, nrows=2, figsize=(20, 12))
sns.countplot(x='key', data=spotify_data, ax=axs[0][0])
sns.countplot(x='mode', data=spotify_data, ax=axs[0][1])
sns.countplot(x='duration_ms', data=spotify_data, ax=axs[1][0])
plt.show()

# Relationship between numerical features and target variable
fig, axs = plt.subplots(ncols=3, nrows=3, figsize=(20, 12))
sns.boxplot(x='playlist_genre', y='danceability', data=spotify_data, ax=axs[0][0])
sns.boxplot(x='playlist_genre', y='energy', data=spotify_data, ax=axs[0][1])
sns.boxplot(x='playlist_genre', y='loudness', data=spotify_data, ax=axs[0][2])
sns.boxplot(x='playlist_genre', y='speechiness', data=spotify_data, ax=axs[1][0])
sns.boxplot(x='playlist_genre', y='acousticness', data=spotify_data, ax=axs[1][1])
sns.boxplot(x='playlist_genre', y='instrumentalness', data=spotify_data, ax=axs[1][2])
sns.boxplot(x='playlist_genre', y='liveness', data=spotify_data, ax=axs[2][0])
sns.boxplot(x='playlist_genre', y='valence', data=spotify_data, ax=axs[2][1])
sns.boxplot(x='playlist_genre', y='tempo', data=spotify_data, ax=axs[2][2])
plt.show()

sns.catplot(x='key', y='danceability', kind='box', col='playlist_genre', data=sp
plt.show()

sns.catplot(x='mode', y='danceability', kind='box', col='playlist_genre', data=s
```

```
plt.show()

sns.catplot(x='duration_ms', y='danceability', kind='box', col='playlist_genre',
plt.show()

corr_matrix = spotify_data.corr()
sns.heatmap(corr_matrix, annot=True)
plt.show()

# Select relevant features for clustering
X = spotify_data[['danceability', 'energy', 'key', 'loudness', 'mode', 'speechin

# Standardize the data
from sklearn.preprocessing import StandardScaler
scaler = StandardScaler()
X_scaled = scaler.fit_transform(X)

# Apply KMeans clustering algorithm
from sklearn.cluster import KMeans
kmeans = KMeans(n_clusters=10, random_state=42)
kmeans.fit(X_scaled)

# Add cluster labels to dataframe
spotify_data['cluster'] = kmeans.labels_

# Scatter plot of energy vs danceability with clusters
sns.scatterplot(x='energy', y='danceability', hue='cluster', data=spotify_data)
plt.show()

# Scatter plot of loudness vs tempo with clusters
sns.scatterplot(x='loudness', y='tempo', hue='cluster', data=spotify_data)
plt.show()

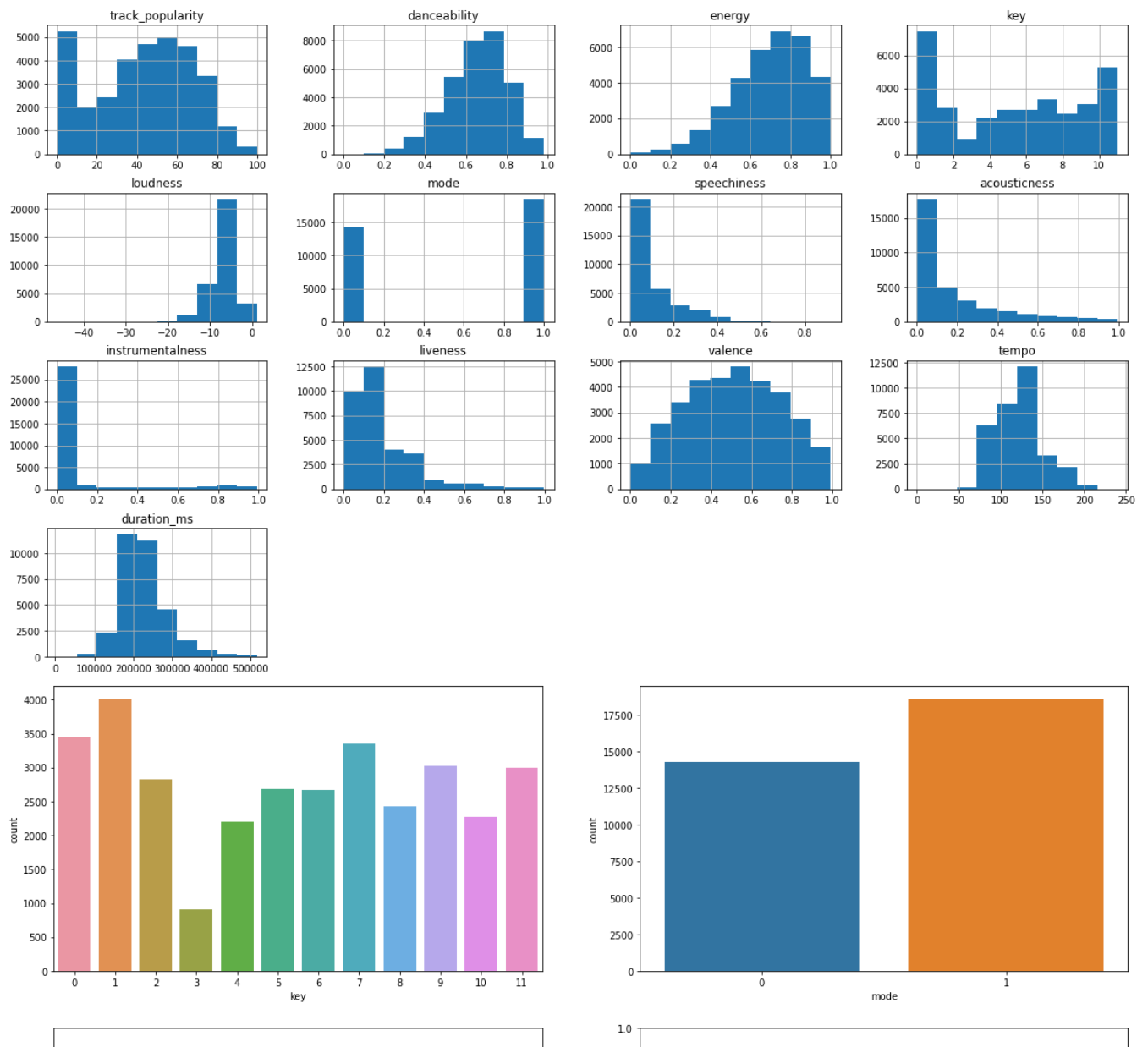
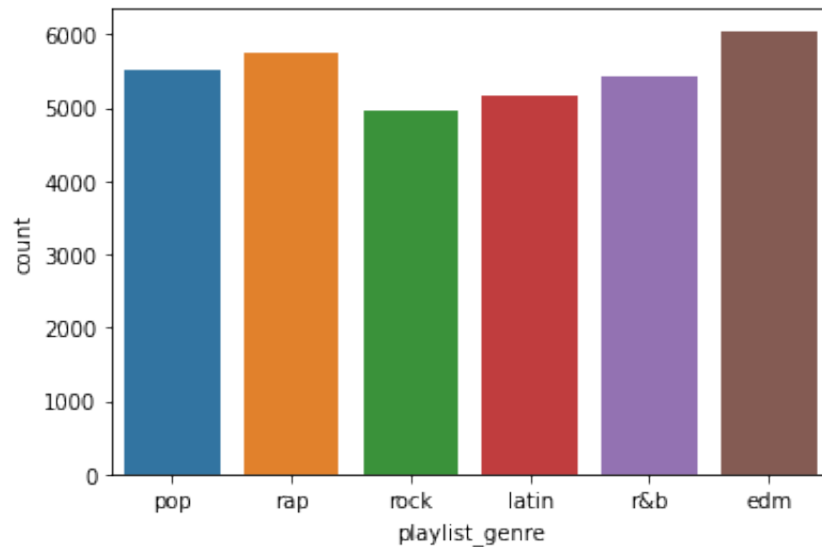
# Scatter plot of instrumentalness vs acousticness with clusters
sns.scatterplot(x='instrumentalness', y='acousticness', hue='cluster', data=spot
plt.show()

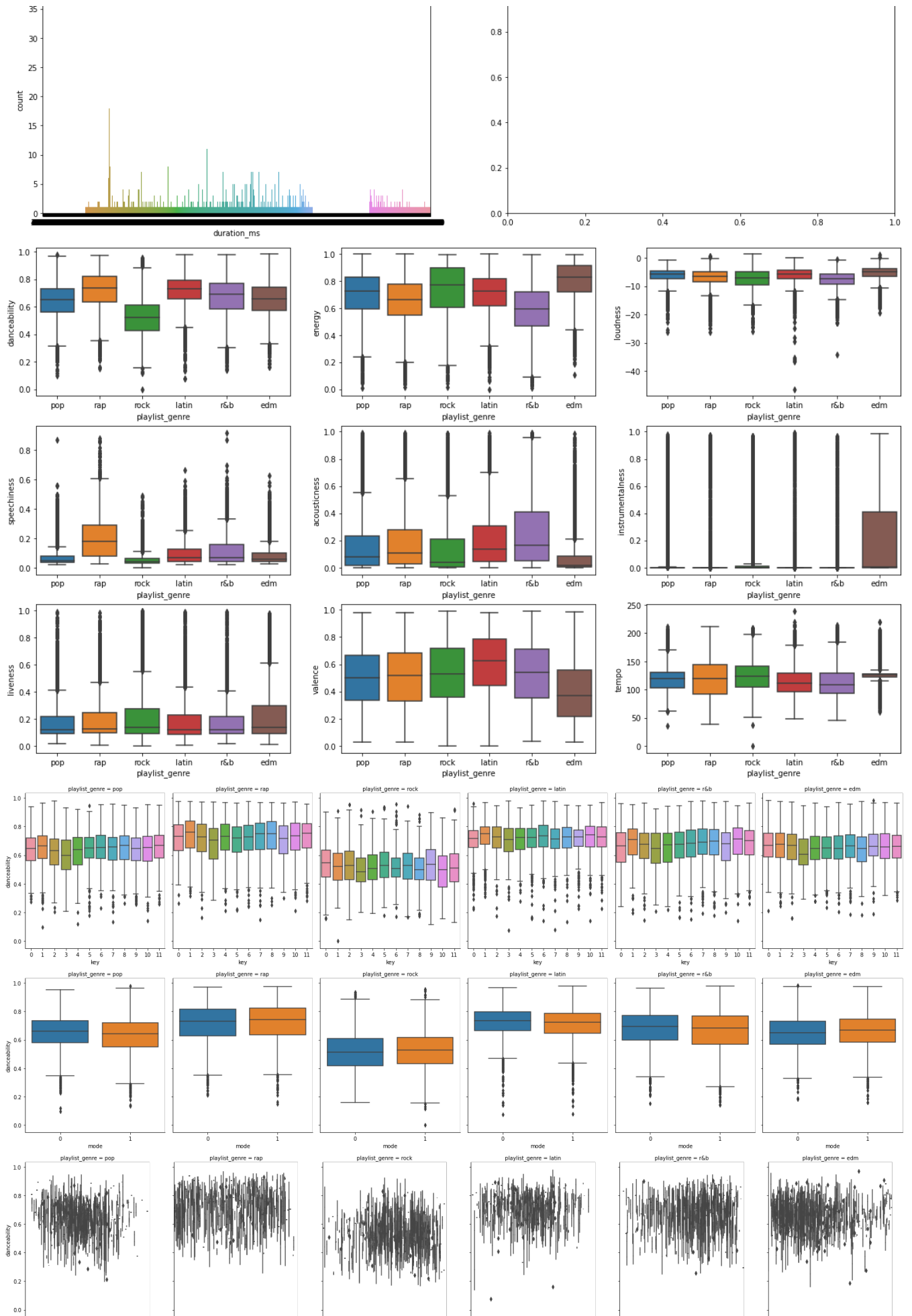
# Get list of genres in each cluster
cluster_genres = spotify_data.groupby('cluster')['playlist_genre'].apply(list)

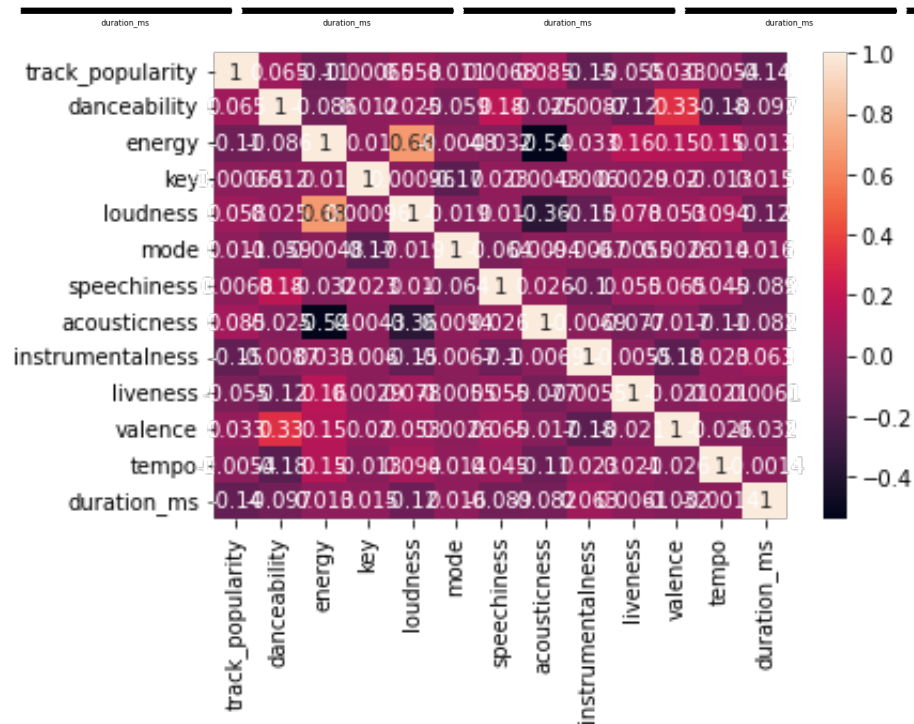
# Define function to recommend songs based on genre
def recommend_songs(playlist_genre):
    cluster = spotify_data.loc[spotify_data['playlist_genre'] == playlist_genre,
    recommended_genres = cluster_genres[cluster]
    recommended_songs = spotify_data.loc[spotify_data['playlist_genre'].isin(rec
    return recommended_songs

# Example recommendation for a user who likes rock music
```

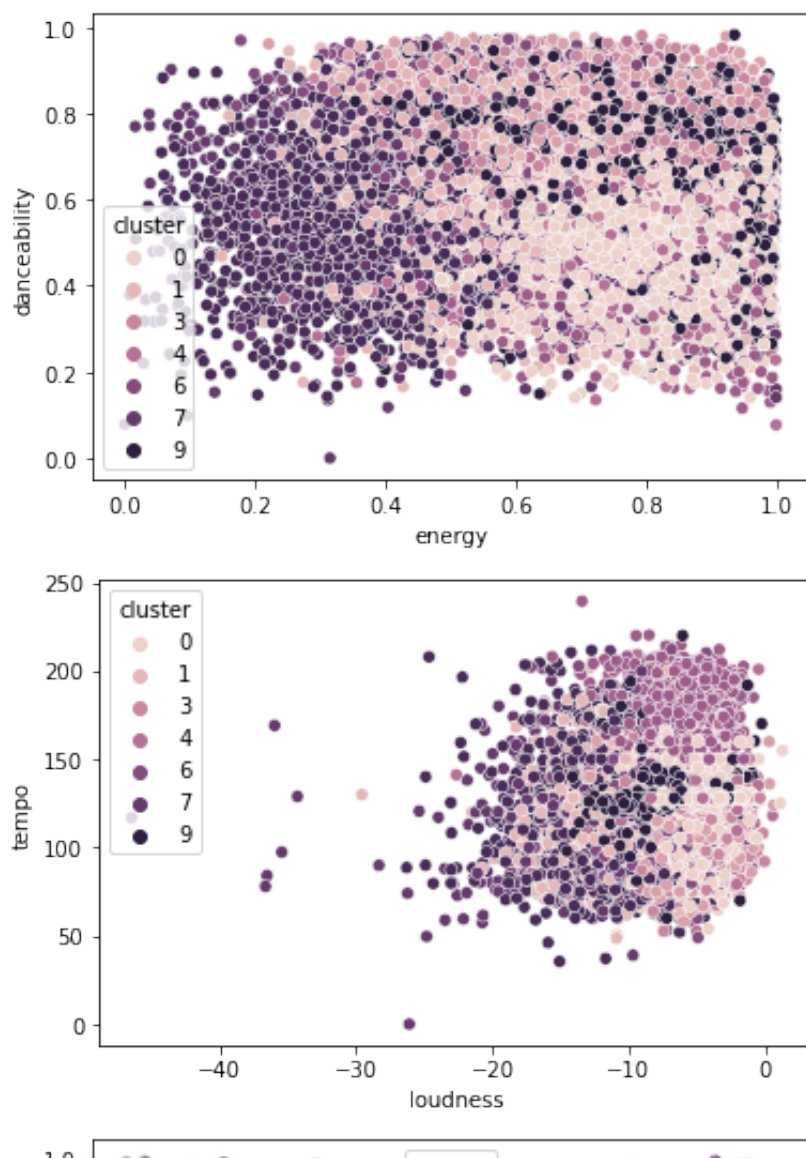
```
typee=(input("Enter any"))  
recommend_songs(typee)
```

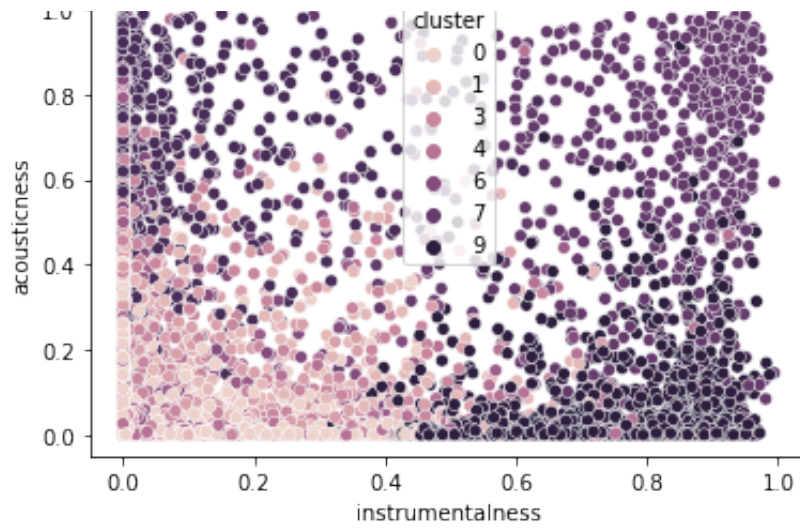






```
/usr/local/lib/python3.9/dist-packages/sklearn/cluster/_kmeans.py:870: FutureWarnings.warn(
```





Enter anyrock

```
array(["I'll Do 4 U (Re-Recorded / Remastered)", 'Kemba Walker',  
      'Break the Hold', "She's A Beauty - Remastered 1991", 'This City',  
      'Stay Late', 'Surviving (feat. Joe Cleere)', 'BOP',  
      'Revolution - Remastered 2009', 'i loved you first'], dtype=object)
```



```
import pandas as pd
import numpy as np
import matplotlib.pyplot as plt
import seaborn as sns
spotify_data = pd.read_csv('/content/spotify dataset.csv')
spotify_data.head()
```

	track_id	track_name	track_artist	track_popularity	
0	6f807x0ima9a1j3VPbc7VN	I Don't Care (with Justin Bieber) - Loud Luxur...	Ed Sheeran	66	2oCs0I
1	0r7CVbZTWZgbTCYdfa2P31	Memories - Dillon Francis Remix	Maroon 5	67	63rPSC
2	1z1Hg7Vb0AhHDiEmnDE79I	All the Time - Don Diablo Remix	Zara Larsson	70	1Ho\$
3	75FpbthrwQmzHIBJLuGdC7	Call You Mine - Keanu Silva Remix	The Chainsmokers	60	1nqYs
4	1e8PAfcKUYoKkxPhrHqw4x	Someone You Loved - Future Humans Remix	Lewis Capaldi	69	7m7

5 rows x 23 columns

[Colab paid products](#) - [Cancel contracts here](#)

