Song Recommendation System

Machine Learning for Global EDM Playlists

Clustering and Content-Based Song Recommendation

Lora Chuaner Ding

Date: 04.07.2024

Project Brief

1. Data Preparation – Spotify API
2. Feature Engineering
3. Machine Learning Model – Clustering
4. Model Evaluation and Visualization
5. Model Optimization – PCA&ISOMAP
6. Limitations and Learnings

Spotify Recommendation Methods

	Collaborative Filtering	Content-Based Recommendation
Principle	• Find users with similar music tastes to you • Find songs similar to the ones you like.	Recommend songs that are similar in features
Data	Private user-Item Interaction Data **Data Security** 	Song Features Data **Public Playlists** 

Spotify API

Step1 **Authenticate** with Spotify API → Spotify Developer → **Client Key & Secret**

Step2 **Fetch Playlists** → Search and fetch to Country **public playlists**

```python
# 3.1 functions search playlist
def search_playlists(query, country, limit=50):
```

```python
# 3.2 Country playlists
def get_playlists_for_countries(query, countries,
```

Step3 **Extract Tracks** → Extract **tracks and audio features** from playlists

```python
# 4.2 Get track features
def get_audio_features_with_info(track_ids):
```

```python
# 4.3 Get all features in playlists
def get_playlist_audio_features(playlist_id):
```

```
Getting audio features for playlist: EDM Music 2024 (Top 100)
Getting audio features for playlist: mint
Getting audio features for playlist: EDM House Mix
Getting audio features for playlist: EDM BANGERS ❤️🩷🩵
Getting audio features for playlist: Chill Tracks
Getting audio features for playlist: EDM Hard Bass Mix
Getting audio features for playlist: EDM Classics (Top 100)
Getting audio features for playlist: Workout EDM Mix
Getting audio features for playlist: Summer EDM Mix
Getting audio features for playlist: Happy Beats
Getting audio features for playlist: hot girl EDM workout mix 🎧
Getting audio features for playlist: Pop Hits 2000s – 2024
Getting audio features for playlist: Club EDM Mix
Getting audio features for playlist: EDM BANGERS 🔥🎧
Getting audio features for playlist: EDM 2024
Getting audio features for playlist: EDM MIX 2024 🔥 EDM HITS 🔥TOMORROWLAND 2024
Getting audio features for playlist: Ultra Gaming
Getting audio features for playlist: Best EDM Songs of All Time - Most Popular EDM Music Playlist
Getting audio features for playlist: BRAZILIAN PHONK 🇧🇷
Getting audio features for playlist: Best EDM Playlist 2024
Getting audio features for playlist: Deep House Relax
Getting audio features for playlist: 2013-2015 EDM Hits
Getting audio features for playlist: Dance Hits
Getting audio features for playlist: Hype EDM Mix
Getting audio features for playlist: Dance Party
Getting audio features for playlist: EDM Remixes of Popular Songs
Getting audio features for playlist: Running EDM Mix
Getting audio features for playlist: EDM Workout 2024 | Training Hits
Getting audio features for playlist: frat songs that could resurrect me
Getting audio features for playlist: EDM Hits
Getting audio features for playlist: Hype
Getting audio features for playlist: EDM Pop Mix
Getting audio features for playlist: Car Music
Getting audio features for playlist: as i run (to EDM hits)
```

Step4 **Store Data** → **Loop step 3** for all playlists and save the data

**** Playlists are up-to-date***

Data Preparation

Dataset **Top 50** EDM Playlists from Germany, Netherlands, Portugal, Spain, US and the UK

Rows Total tracks: 12183 → **9819** after dropped duplicates and missing values

Columns

- **Track info:** track_id, track_name, artist_name, album_name, release_date, duration_ms, popularity
- **Audio features:** danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo → **all numerical**

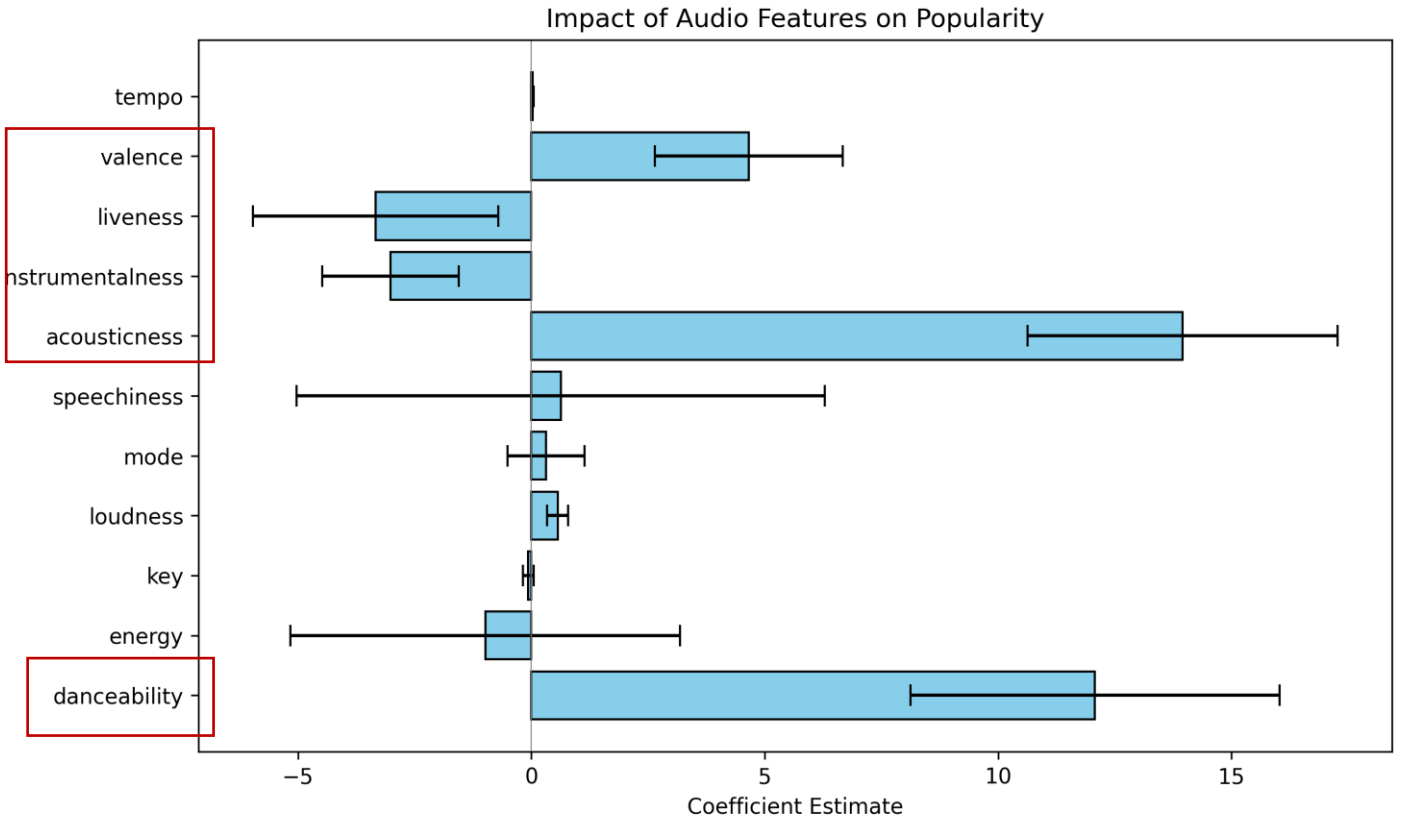
Feature Engineering

*In the context of Spotify's recommendation system, **popularity** plays a crucial role. Spotify's algorithms use a combination of user listening data, playlists, and other behavioral signals to generate personalized recommendations.*

H0 There is no significant difference in the effect of all audio features on popularity.

H1 There is a significant difference in the effect of all audio features on popularity.

REJECTED



ANOVA results:

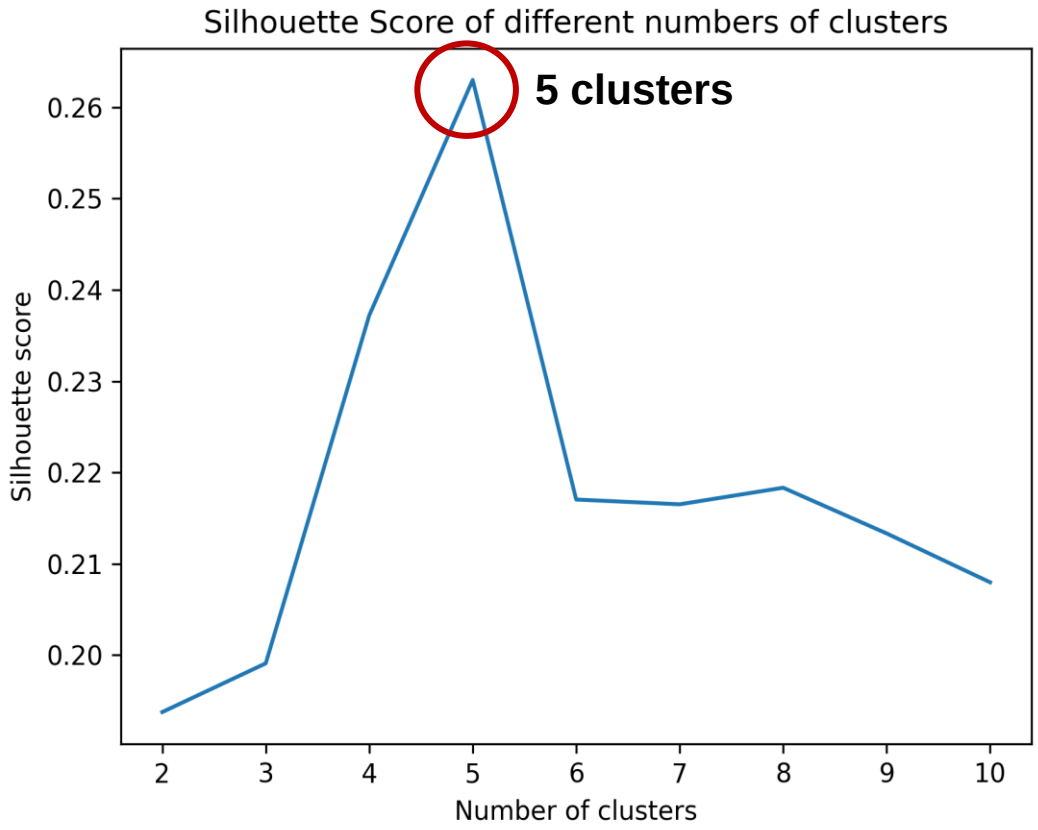
P-value = 0.0

ML Model – Clustering - Kmeans

- *Unlabeled and High-Dimensional Data*
- *Unsupervised Model*

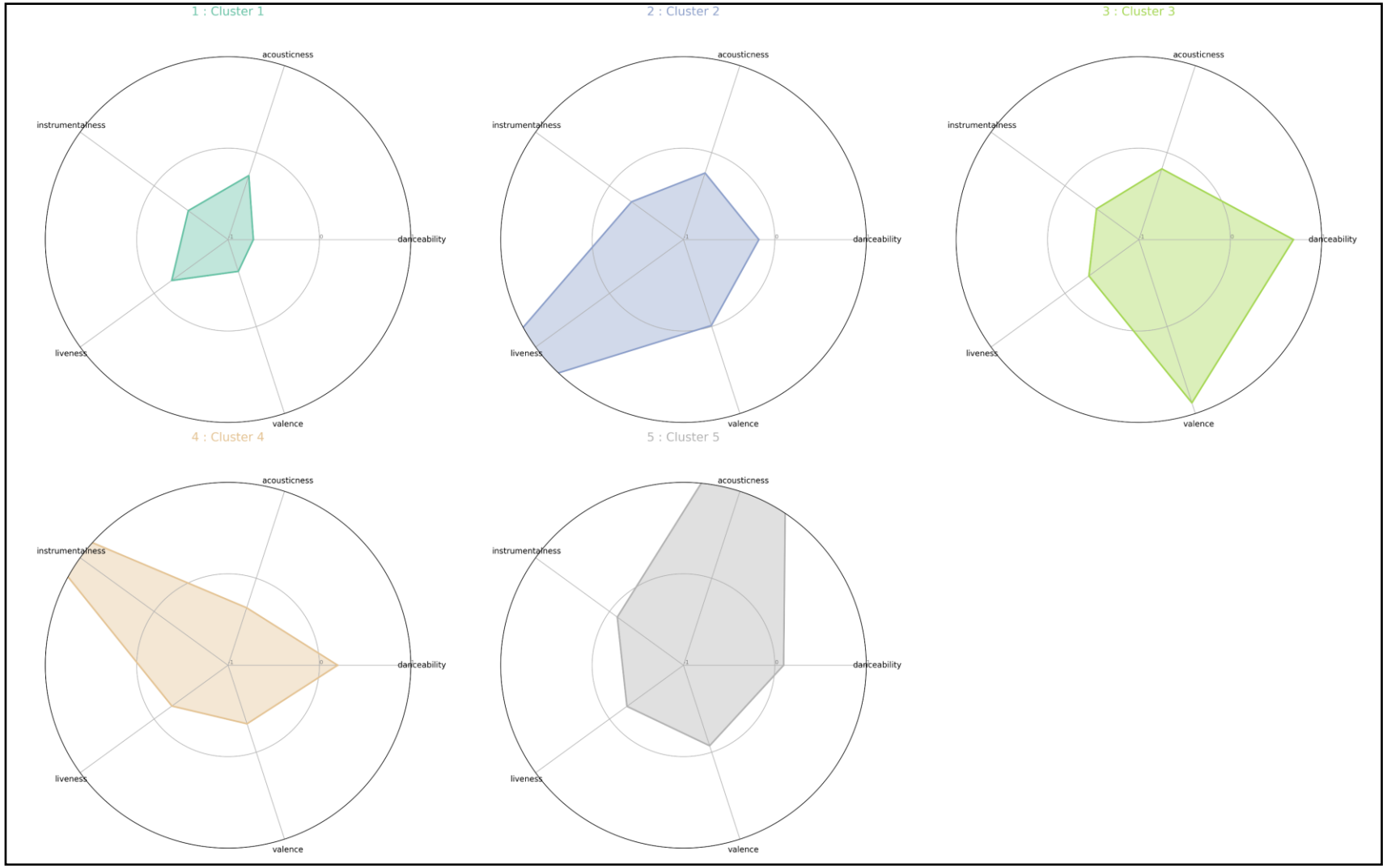
Kmeans (K = n_clusters)

Number of clusters	Silhouette Score
2	0.1937560426971352 8
3	0.1990934507497559 8
4	0.2372024614332109 7
5	**0.2629938078904257**
6	0.2170358680483024 7
7	0.2165133091914758 5
8	0.2133245452053587 6
9	0.2079695369367600 4

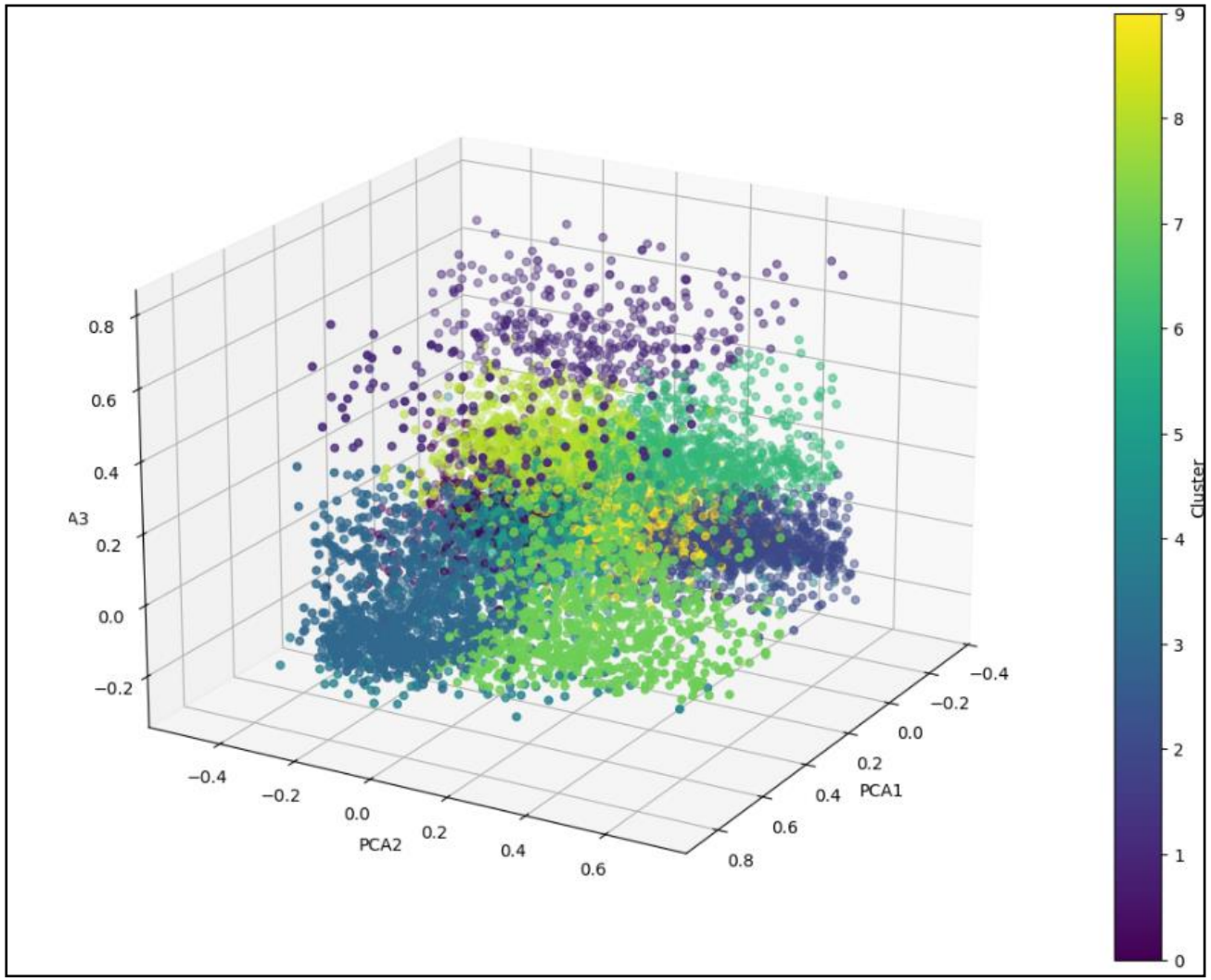


Clusters Visualization

Radar Charts of 5 Clusters



Model Optimization – PCA



PCA (Principal Component Analysis)

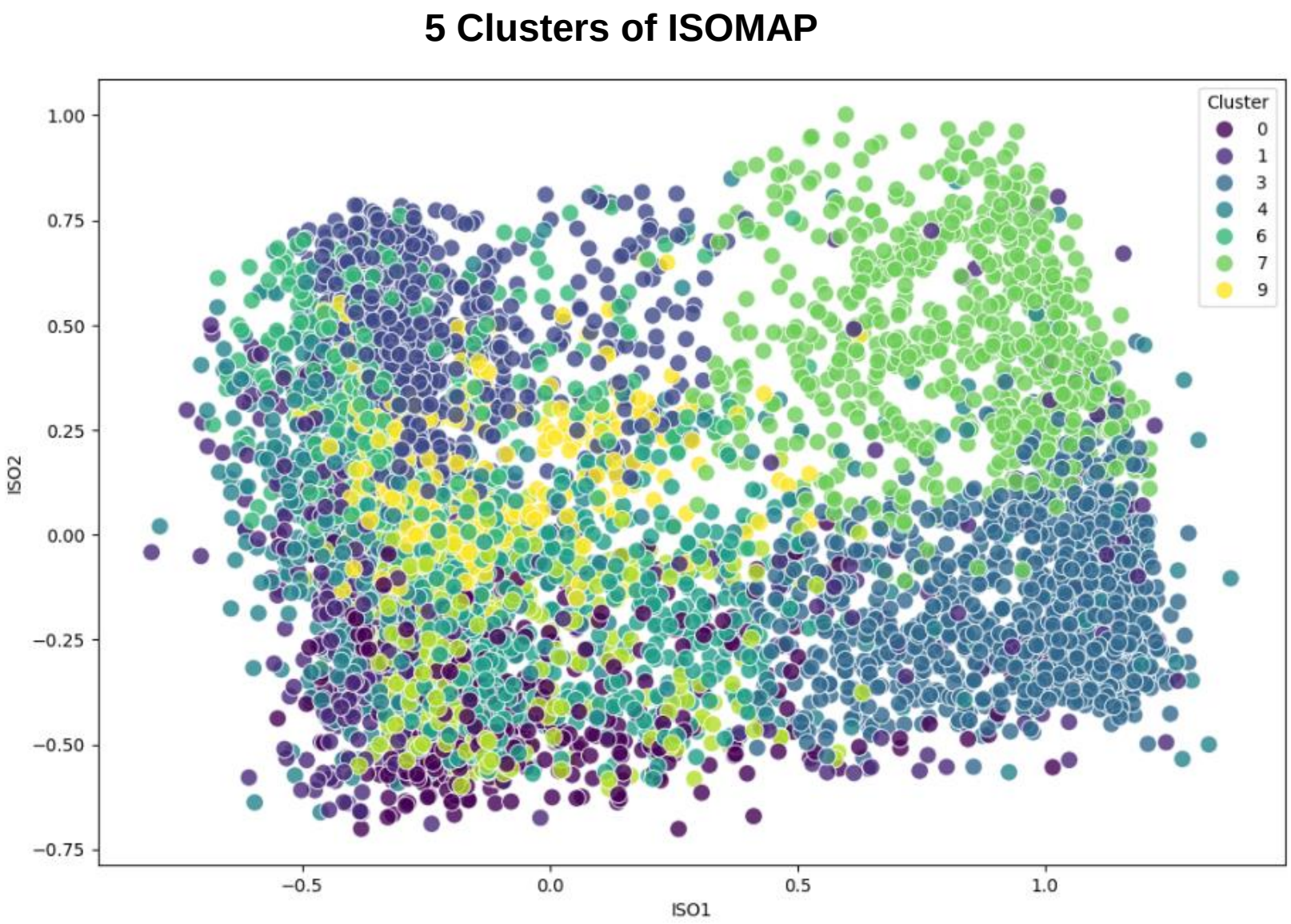
- Transform data from a high-dimensional space to a lower-dimensional space
- Noise reduction, reducing computational complexity

	PCA1	PCA2	PCA3	cluster
0	-0.133918	0.110011	0.200144	1
1	-0.150705	-0.014893	-0.110876	2
2	-0.206502	0.096055	0.030194	2
3	-0.220934	0.094950	0.137786	1
4	-0.237217	0.238084	-0.016491	2
...	...	...	...	...
12177	0.690894	0.155542	0.011461	3
12179	-0.043342	-0.200245	0.216556	4
12180	0.768750	-0.135039	0.087999	3
12181	-0.174781	0.007284	0.132586	2
12182	-0.163114	-0.178501	0.032709	4

9818 rows × 4 columns

* ***PCA (Principal Component Analysis) and ISOMAP (Isometric Feature Mapping)*** *are two commonly used dimensionality reduction algorithms for different types of data processing needs.*

Model Optimization – ISOMAP



ISOMAP (Isometric Mapping)

- A non-linear dimensionality reduction method
- Preserve the manifold structure of the data in the high-dimensional space while mapping it to a lower-dimensional space.

Recommendation Demo

Demo in Jupyter Notebook

[39]:
```
# test by name and artist
track_name = "Blinding Lights"
artist_name = "The Weeknd"
recommended_songs, error = recommend_songs(track_name, artist_name)
if error:
    print(error)
else:
    display(recommended_songs[['track_name', 'artist_name', 'album_name', 'release_date']])
```

```
Song not found in the dataset.
```

[42]:
```
# test by name and artist
track_name = "Wake Me Up"
artist_name = "Avicii"
recommended_songs, error = recommend_songs(track_name, artist_name)
if error:
    print(error)
else:
    display(recommended_songs[['track_name', 'artist_name', 'album_name', 'release_date']])
```

	track_name	artist_name	album_name	release_date
7058	Follow Me (feat. Jason Derulo) - Bingo Players...	Hardwell, Jason Derulo, Bingo Players	United We Are (Remixed)	2015-12-04
4934	Devil (feat. Busta Rhymes, B.o.B & Neon Hitch)	Cash Cash, Busta Rhymes, B.o.B, Neon Hitch	Devil (feat. Busta Rhymes, B.o.B & Neon Hitch)	2015-08-07
3503	Under The Water	Camden Cox	Under The Water	2021-02-19
10434	Lost In Sound	ROY KNOX	Lost In Sound	2019-03-23
430	Better	Sonny Fodera	Better	2022-06-10

Limitations and Learnings

Limitations

- Silhouette Score < 0.5

→ Low score

- Clusters have too many overlapping areas
- Clustering not very distinct

Learnings

- All tracks from “EDM”
- The differences in audio features of subgenres are not particularly significant

Next Step

- Using API to fetch more diverse playlists
- Streamlit – App interface
- Deep Learning Models to generative a song

THANK YOU!

Congrats on graduation, geeks! Enjoy your summer!