Northeastern University

Spotify Recommendation System

INFO7390 Advanced Data Science and Architecture

Dr. Ramkumar Hariharan

Sai Raghavendra Viravalli 001568587

Sai Saketh Ram Vootla 001568984

Introduction

- In this project we evaluate ML models such Kmeans clustering and then use content based recommendation algorithm to recommend songs based on the user’s playlist and the dataset.
- We use logistic regression to then fit the sets we have resampled after splitting our data into the pipeline.

Objectives and Constraints

Northeastern University

- **Objectives**
  - **The main goal is to cluster the data we have and then create a pipeline accordingly.**
  - **The clustering is crucial as that forms as the base for our recommendation algorithm based on the probability**
- **Constraints**
  - **Performing multiple clustering models such as agglomerative is tough as the dataset is huge and the ram crashes due to that.**
  - **The goal is to make the closest recommendation possible to the users playlist so the accuracy will be varying depending on users playlist as we will be fetching direct from the API**

Dataset Description

Northeastern University

- We have taken the dataset from the kaggle source https://www.kaggle.com/code/vatsalmavani/music-recommendation-system-using-spotify-dataset/data. The dataset splits into 170653 rows and 19 columns. Those 19 data columns are divided into 13 numerical and 6 categorical columns.

- Numerical Columns:
  - **acousticness**: The relative metric of the track being acoustic, (Ranges from 0 to 1)
  - **danceability**: The relative measurement of the track being danceable, (Ranges from 0 to 1)
  - **energy**: The energy of the track, (Ranges from 0 to 1)
  - **duration_ms**: The length of the track in milliseconds (ms), (Integer typically ranging from 5k to 300k)
  - **instrumentalness**: The relative ratio of the track being instrumental, (Ranges from 0 to 1)
  - **valence**: The positiveness of the track, (Ranges from 0 to 1)
  - **popularity**: The popularity of the song lately, default country = US, (Ranges from 0 to 100)
  - **tempo**: The tempo of the track in Beat Per Minute (BPM), (Float typically ranging from 0 to 250)
  - **liveness**: The relative duration of the track sounding as a live performance, (Ranges from 0 to 1)
  - **loudness**: Relative loudness of the track in decibel (dB), (Float typically ranging from -60 to 4.0)
  - **speechiness**: The relative length of the track containing any kind of human voice, (Ranges from 0 to 1)
  - **year**: The release year of track, (Ranges from 1921 to 2020)
  - **id**: The primary identifier for the track, generated by Spotify

- Categorical Columns:
  - **key**: The primary key of the track encoded as integers in between 0 and 11 (starting on C as 0, C# as 1 and so on...)
  - **artists**: The list of artists credited for production of the track
  - **release_date**: Date of release mostly in yyyy-mm-dd format, however precision of date may vary
  - **name**: The title of the track
  - **mode**: The binary value representing whether the track starts with a major (1) chord progression or a minor (0)
  - **explicit**: The binary value whether the track contains explicit content or not, (0 = No explicit content, 1 = Explicit content)

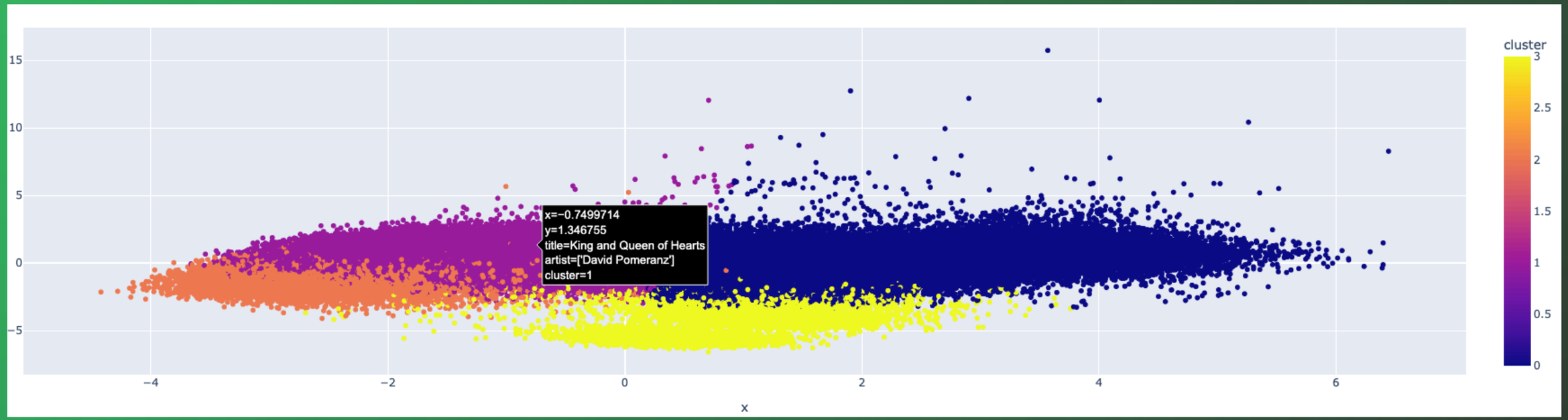
Approach

Northeastern University

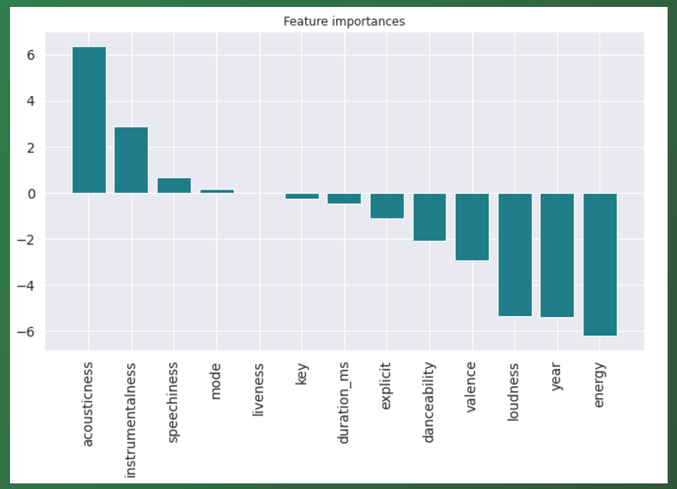
- Once we read the data we first perform some data cleaning and preprocessing. Since there are no null values we skip the part.
- Then we perform EDA to have a clear picture of it and understand the features within the dataset with visualisations.
- We then do our modeling part, where we apply k means clustering to evaluate their performance and get the results and the dataset fit into the pipeline using logistic regression.
- After the data is prepared we authenticate Spotify API credentials using their developer service. Once that’s done we retrieve the songs of that particular user.
- Once the songs/ playlist are retrieved we then compared with the clustered data then using content based recommendation algorithm we recommend songs from our dataset to the user

Plots

- **Clustered data**



- **Feature importance**



Output

- **Scores**

```
Training Metrics
Precision Score: 0.9844573610280081
Recall Score: 0.9851310124597269
Accuracy Score: 0.9847888689305163
F1 Score: 0.9847940715405444
None


Testing Metrics
Precision Score: 0.9828047775947282
Recall Score: 0.9830072090628218
Accuracy Score: 0.9844365272829552
F1 Score: 0.982905982905983
None
```

- **Recommended songs**

```
[{'artists': "['Dexter Freebish']", 'name': 'Leaving Town', 'year': 2000},
 {'artists': "['Godsmack']", 'name': 'No Rest For The Wicked', 'year': 2006},
 {'artists': "['Rascal Flatts']",
  'name': 'Life is a Highway - From "Cars"',
  'year': 2009},
 {'artists': "['Bad English']", 'name': 'Best of What I Got', 'year': 1989},
 {'artists': "['The Wallflowers']", 'name': 'Bleeders', 'year': 1996},
 {'artists': "['Social Distortion']", 'name': '99 To Life', 'year': 1992},
 {'artists': "['John Hiatt']", 'name': 'Perfectly Good Guitar', 'year': 1993},
 {'artists': "['Los Fugitivos']", 'name': 'Corazón Mágico', 'year': 1995},
 {'artists': "['Metallica']", 'name': 'Of Wolf And Man', 'year': 1991},
 {'artists': "['Danger Danger']", 'name': "Don't Walk Away", 'year': 1989}]
```

Conclusion

- We have successfully built a model using K-means clustering and obtained a 98.7% accuracy and clustered data accordingly.
- We have also recommended songs while using the API and if heard physically the recommended songs match the vibe of the similar songs we have listened to.