**Dataset link https://www.kaggle.com/datasets/leonardopena/top50spotify2019**

**1. Description of the project:**

The project aims to demonstrate how to perform clustering analysis on a dataset containing information about the top 50 songs of all time in Spotify. The dataset includes attributes such as the song name, artist, genre, release year, and popularity score.

The project utilizes Python and several popular libraries, including pandas, scikit-learn, matplotlib, and seaborn, to explore the dataset, preprocess the data, perform clustering using the K-means algorithm, and visualize the results.

**2. List of outputs with detailed descriptions:**

The code in the project produces several outputs, each serving a specific purpose. Here is a list of the outputs with their descriptions:

a) Data Exploration:

- Summary statistics: Provides descriptive statistics of the dataset, including count, mean, standard deviation, minimum, maximum, and quartiles for numerical columns.

- Data visualization: Generates various plots and charts to visualize different aspects of the data, such as the distribution of song popularity, genre distribution, and release year distribution.

b) Data Preprocessing:

- Feature extraction: Creates a new feature called "Duration\_min" by converting the song duration from milliseconds to minutes.

- Data cleaning: Handles missing values and removes unnecessary columns that are not relevant for clustering.

c) Clustering Analysis:

- Elbow method plot: Generates a plot to determine the optimal number of clusters based on the within-cluster sum of squares (WCSS) criterion.

- K-means clustering: Performs the K-means algorithm on the preprocessed dataset with the chosen number of clusters.

- Cluster visualization: Produces scatter plots to visualize the clusters formed by the K-means algorithm.

**3. Main output:**

The main output of the project is the scatter plot visualization of the songs, where each point represents a song and is color-coded according to the cluster it belongs to. This visualization helps in understanding the similarities and dissimilarities between songs based on their features.

**4. Detailed instructions for beginners on how to run the code:**

To run the code in the project, follow these steps:

Step 1: Set up the environment

- Install Python: If you don't have Python installed, download and install it from the official Python website (https://www.python.org).

- Install required libraries: Open a terminal or command prompt and run the following command to install the necessary libraries:

```

pip install pandas scikit-learn matplotlib seaborn

```

Step 2: Download the dataset using provided link above

Step 3: Open the code notebook

- Launch Jupyter Notebook or JupyterLab: Open a terminal or command prompt, navigate to the directory where you saved the downloaded code notebook, and run the following command:

```

jupyter notebook

```

- Open the code notebook: In the Jupyter interface, navigate to the directory where you saved the downloaded code notebook and click on it to open it.

Step 4: Run the code

- Execute the code cells: In the code notebook, you'll find multiple code cells. Execute each code cell one by one by selecting it and either clicking the "Run" button or using the Shift+Enter keyboard shortcut.

- Observe the outputs: As you run the code cells, you'll see the outputs being generated. Pay attention to any error messages that may appear, as they might indicate missing dependencies or issues with the dataset.

Step 5: Explore the results

- Analyze the generated outputs: After running all the code cells, examine the generated outputs, such as summary statistics, visualizations, and the main scatter plot visualization of song clusters.

- Interpret the results: Use the outputs to gain insights into the dataset, such as identifying trends, understanding the clusters formed, and exploring relationships between song features.

By following these instructions, beginners should be able to run the code and explore the results of the "Top 50 Music Data Clustering" project.