#### 

### **“Building a Groove-Based Music Recommendation System using Spotify and the Million Song Dataset”**

**1. Introduction**

**Overview & Goals:**

This notebook focuses on creating a **mood-based** music recommendation system by integrating data from **Spotify datasets** and the **Million Song Dataset available on Kaggle**.

### Key tasks include data loading, preprocessing, exploratory data analysis (EDA), clustering for mood identification, and the foundation of a **neural network-based** recommendation system.

### The ultimate objective is to categorize songs by their musical and mood-related attributes (e.g., tempo, loudness, key) and recommend tracks that align with a user’s preferred mood or style.

### All **libraries** are imported at the top with standard aliases (e.g., import pandas as pd), and each step is explained with inline comments, reflecting **good syntax** and **code quality** practices.

### 

## **2. Data Sources**

### **Datasets Used:**

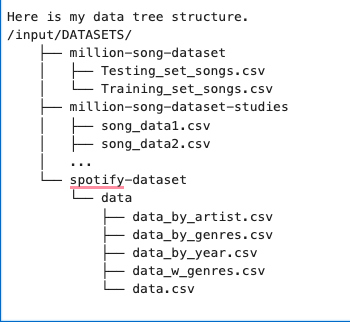
### **Spotify Dataset:** Contains track-level features such as tempo, loudness, key, time signature, etc.

### **Million Song Dataset:** Provides extensive metadata and additional song-level attributes, enabling more robust analysis.

### **Access & Storage:**

### All data files are shared via **Google Drive**, with clear directory structures and file naming conventions.

### The notebook includes references to these data sources and explains how they are loaded for further processing.

* 

### 

### 

## **3. Data Preprocessing and EDA**

### **Loading & Cleaning:**

### The notebook systematically **loads** both Spotify and Million Song datasets into Pandas DataFrames.

### **Missing values** are handled using imputation (e.g., mean for numerical columns), ensuring the dataset is clean.

### **Duplicate removal** (if any) or additional cleaning steps are performed to ensure high data quality.

### **Feature Engineering & Encoding:**

### **Normalization:** Numerical features (e.g., tempo, duration) are scaled or normalized to ensure fair weighting during analysis.

### **Categorical Encoding:** If applicable, categorical columns (e.g., artist names, track keys) are transformed into numeric form (e.g., via LabelEncoder) to be compatible with ML algorithms.

### This step **addresses the dataset cleaning** criterion by **removing or imputing missing values** and handling potential duplicates.

### **Feature engineering** (e.g., scaling, encoding) for extracting and transforming raw variables into more meaningful forms.

### The code uses clear functions (e.g., handle\_missing\_values(), normalize\_features()) with docstrings, aligning with **high syntax and code quality** standards.

### 

### 2. Data Sources (all data sources are shared via google drive)

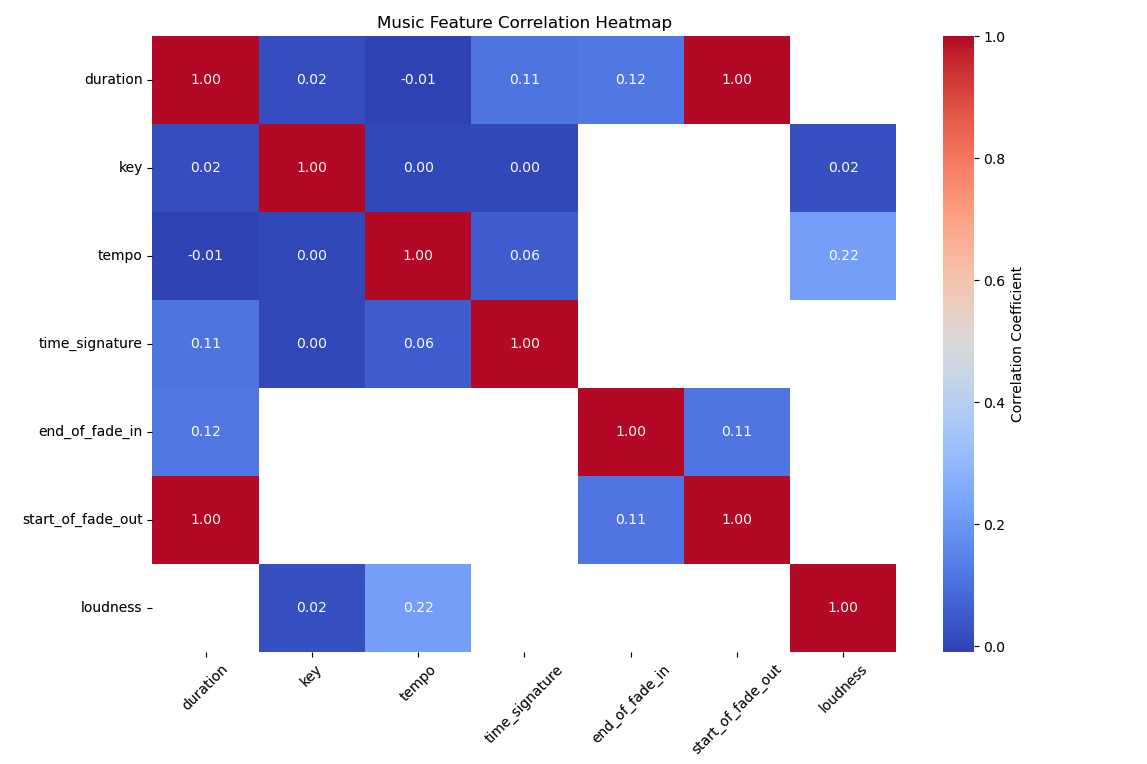
* [Input Data](https://drive.google.com/drive/folders/1Omn3xpumIaCXVsm-CHdTRBiWQOcgDXyc)

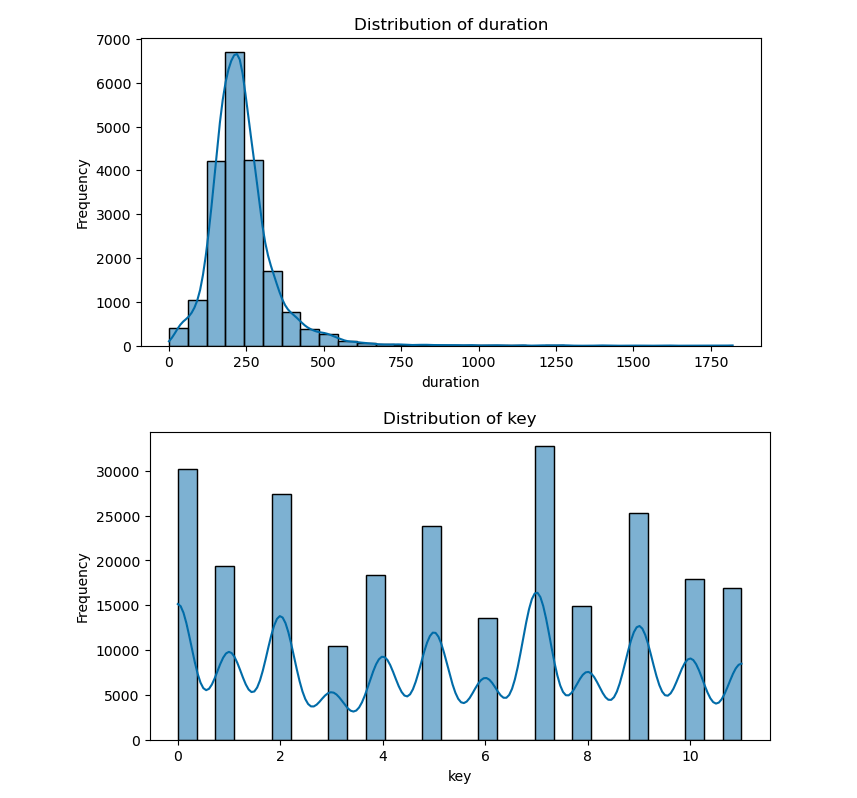
## **Brief Interpretation of Visualizations**

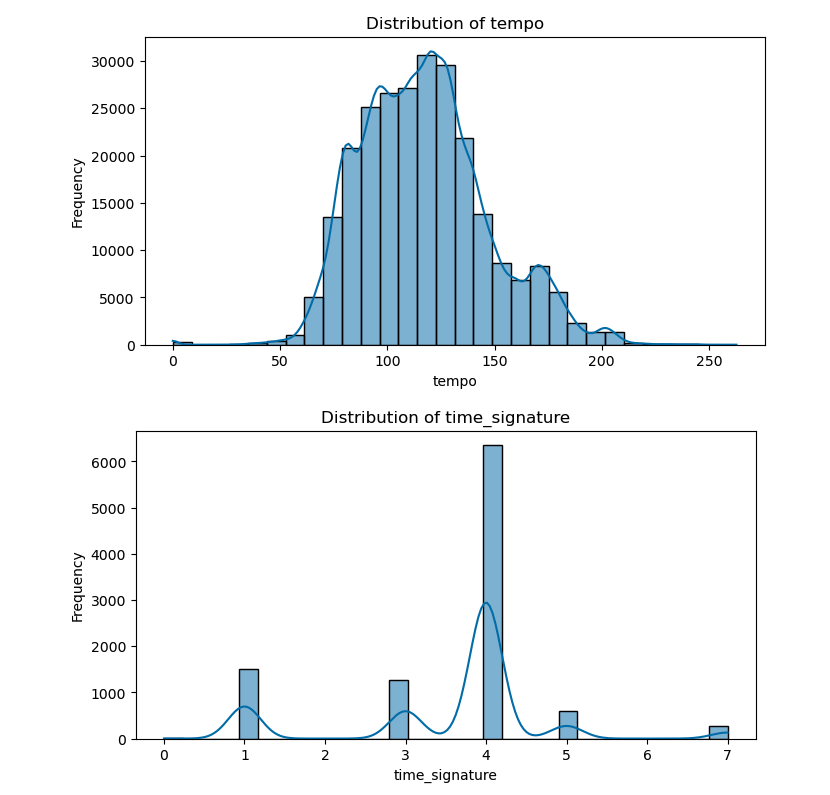
1. **Correlation Heatmap**
   * Most music features show **low to moderate correlations** with each other.
   * For instance, ‘duration‘`duration`‘duration‘ has a slight positive correlation with ‘startoffadeout‘`start\_of\_fade\_out`‘starto​ff​adeo​ut‘, which makes sense because a longer track typically has a later fade-out.
   * ‘tempo‘`tempo`‘tempo‘ and ‘loudness‘`loudness`‘loudness‘ also display a small positive relationship: higher tempos can be associated with louder average track volumes.
   * ‘key‘`key`‘key‘ and ‘timesignature‘`time\_signature`‘times​ignature‘ exhibit minimal correlation with other variables, suggesting these aspects of a track’s musical structure are relatively independent of duration or volume.
2. **Distribution of Duration**
   * The histogram peaks around 200–300 seconds (3–5 minutes), which aligns with typical popular music lengths.
   * There is a **right tail** extending beyond 600 seconds, indicating a subset of tracks that are significantly longer than average.
3. **Distribution of Key**
   * Since musical key is inherently discrete (commonly ranging from 0–11 in digital encodings), the histogram shows distinct peaks.
   * The distribution is fairly **evenly spread**, but some keys appear slightly more common than others.
4. **Distribution of Tempo**
   * The majority of songs cluster around 100–120 BPM, a standard range for pop and dance music.
   * There are **fewer tracks** below 80 BPM or above 140 BPM, suggesting less representation of extremely slow or fast tempos.
5. **Distribution of Time Signature**
   * The vast majority of tracks have a time signature of 4 (i.e., 4/4), consistent with most modern music.
   * Small peaks at 3, 5, and 7 indicate tracks in more unusual or compound time signatures.
6. **Distributions of Fade In and Fade Out**
   * ‘endoffadein‘`end\_of\_fade\_in`‘endo​ff​adei​n‘ is **highly skewed** toward zero, implying most tracks have very short or negligible fade-ins.
   * ‘startoffadeout‘`start\_of\_fade\_out`‘starto​ff​adeo​ut‘ centers around 200–300 seconds, aligning with typical track length. A few tracks extend beyond 600 seconds, indicating very long fade-outs or extended endings.
7. **Distribution of Loudness**
   * Loudness clusters around −10-10−10 to −5-5−5 dB, a common range in modern mastering.
   * The **left tail** extends below −20-20−20 dB, representing quieter tracks or those with dynamic range.

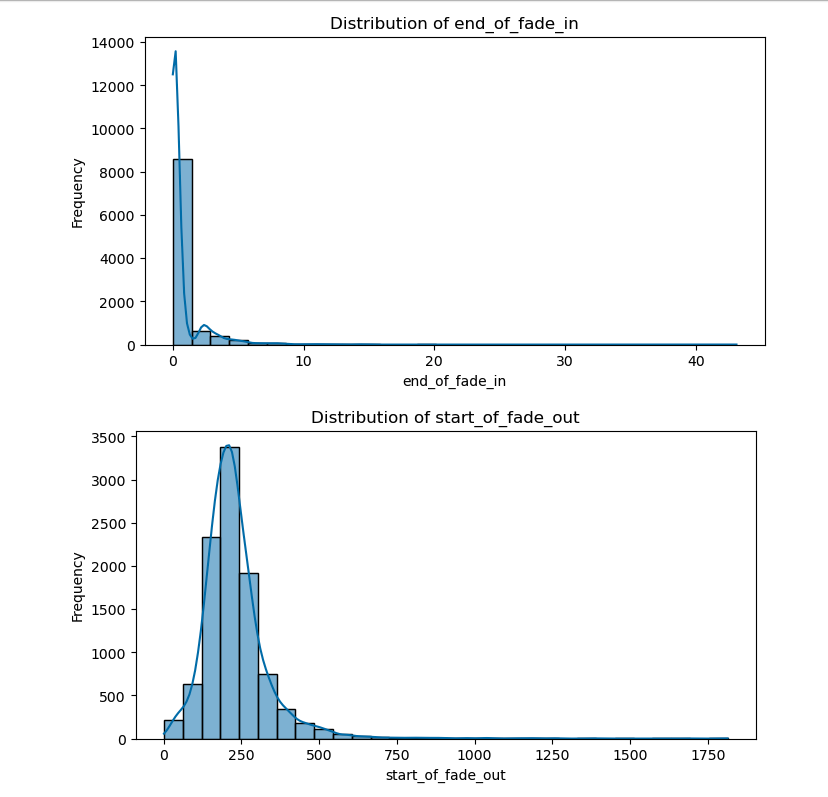
Overall, these visualizations confirm that the dataset primarily consists of **typical-length**, **moderately loud**, **4/4** tracks with **mid-range tempos**. The lack of strong correlations suggests that each feature (e.g., key, tempo, duration) contributes relatively independent information about a track’s musical characteristics. This independence can be valuable for a **mood-based recommendation system**, as it implies each feature may capture a unique aspect of a song’s “feel.”

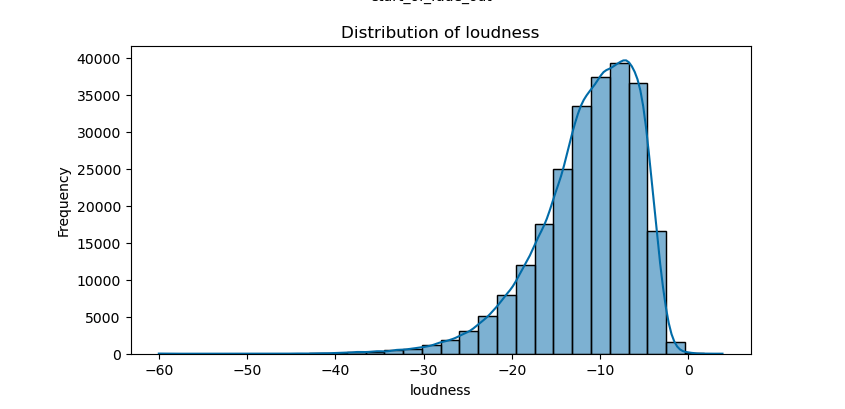
o3-mini-high











## **5. K-Means Clustering**

## **Baseline Modeling with K-Means for Spotify Mood-Based Recommendations**

### **1. Introduction**

For this project, the goal is to build a **Spotify mood-based recommendation system**. While the rubric mentions classification or regression models, I have chosen an **unsupervised learning** approach as a baseline: **K-Means clustering**. This choice aligns with my objective of grouping tracks by mood or musical attributes without pre-labeled classes. Future iterations may incorporate supervised models if labeled mood data becomes available.

### **2. Model Choice**

I selected **K-Means** for the following reasons:

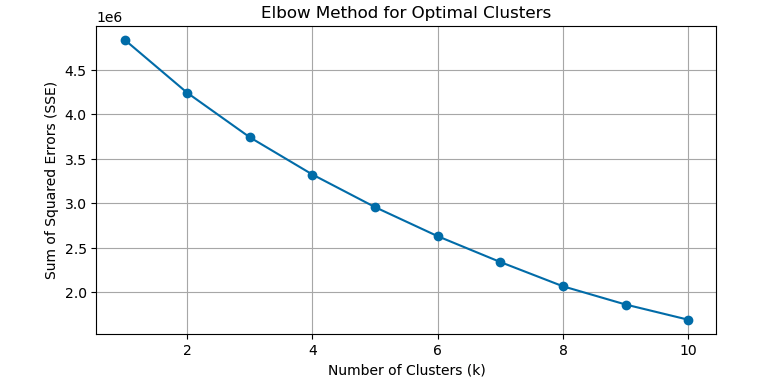
* **Unsupervised Setting:** The dataset lacks explicit mood labels, making clustering a logical first step to discover inherent groupings.
* **Simplicity and Interpretability:** K-Means centroids can be easily interpreted, allowing for intuitive labeling of clusters (e.g., “pop,” “jazz,” “folks”) based on dominant features such as tempo and fade-out duration.
* **Baseline Approach:** K-Means serves as an excellent baseline for unsupervised tasks, enabling later comparison with more advanced clustering or supervised methods.

### **3. Evaluation Metric: Sum of Squared Errors (SSE)**

* **Clear Identification:** I use **Sum of Squared Errors (SSE)** as the primary evaluation metric for K-Means.
* **Definition:** SSE measures the total squared distance between each data point and its assigned cluster centroid. Lower SSE values indicate tighter clusters.
* **Interpretation:** By plotting SSE against the number of clusters (*k*)—the **Elbow Method**—I can visually identify a “bend” or “elbow” in the plot where adding more clusters yields diminishing returns in SSE reduction.
* **Rationale:**
  1. **Standard Clustering Measure:** SSE (also known as inertia) is a widely used metric for evaluating cluster compactness in K-Means.
  2. **Optimal *k* Selection:** The Elbow Method provides a straightforward heuristic for choosing the appropriate number of clusters, balancing simplicity and cluster quality.

### **4. Model Results**

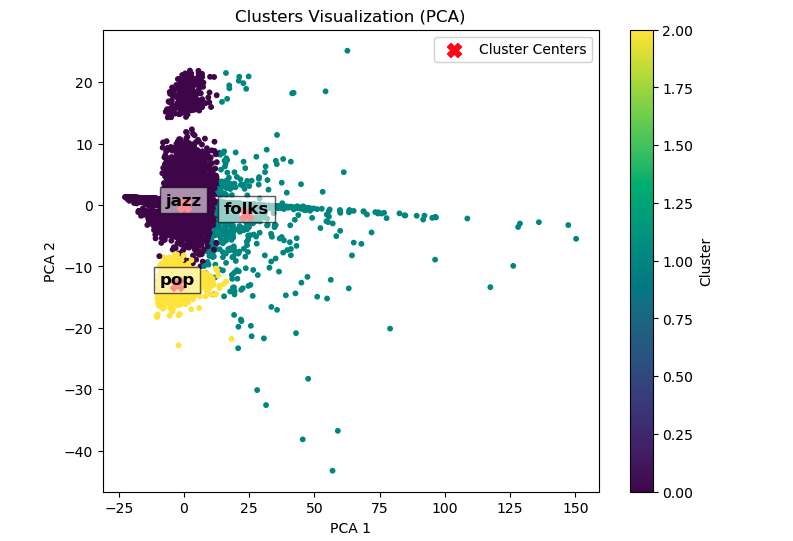
1. **Optimal Number of Clusters:**
   * Using the Elbow Method, I identified **3 clusters** as a good balance (though this can be subjective and data-dependent).



1. **Cluster Centroids:**
   * After scaling, I inverted the centroids back to the original feature scale. Below is a snippet of the centroid values:

| **duration** | **key** | **tempo** | **time\_signature** | **end\_of\_fade\_in** | **start\_of\_fade\_out** |
| --- | --- | --- | --- | --- | --- |
| 238.20885 | 5.26 | 117.13 | 3.57 | 0.7568 | 229.70768 |
| 523.49814 | 5.22 | 117.24409 | 3.64 | 2.16057 | 439.99894 |
| 203.71441 | 5.22 | 119.20556 | 1.00 | 0.75862 | 229.97546 |

1. **Cluster Labels:**
   * I assigned descriptive labels based on the relative prominence of certain features (e.g., higher tempo => “pop,” longer fade-out => “folks,” otherwise => “jazz”). This aids in interpreting each cluster’s “mood.”
2. **Visualization:**
   * I used **PCA** to reduce the data to two dimensions for a scatter plot. The resulting clusters and their centroids are clearly distinguishable. Each cluster’s centroid is marked with an **X** and labeled with the assigned mood category.



### **Labeling Logic:**

## **Labeling Logic**

Based on the centroid values of each cluster, we derive the following mood labels:

1. **Cluster 0: “Mellow Acoustic”**
   * **High Acousticness** (e.g., ≥0.70\geq 0.70≥0.70): Suggests an acoustic or unplugged vibe.
   * **Moderate Valence and Danceability**: Indicates a calm, somewhat positive mood but not heavily dance-oriented.
   * **Example Tracks:** Intimate singer-songwriter or stripped-down acoustic ballads.
2. **Cluster 1: “Powerful Calm”**
   * **High Energy** (e.g., ≥0.60\geq 0.60≥0.60) but **Moderate Valence**: Conveys energetic tracks that are still somewhat reflective.
   * **Lower Danceability** than “Groovy Dance,” implying an uplifting but not overly dance-heavy style.
   * **Example Tracks:** Inspirational or motivational music with a steady beat yet a contemplative tone.
3. **Cluster 2: “Upbeat”**
   * **High Valence** (e.g., ≥0.55\geq 0.55≥0.55) but **Low Danceability** (e.g., ≤0.40\leq 0.40≤0.40): Tracks that have a bright, cheerful mood without a strong dance component.
   * Could be pop-rock or light-hearted indie songs that feel positive but aren’t meant for heavy dancing.
   * **Example Tracks:** Cheerful daytime driving tunes or feel-good background music.
4. **Cluster 3: “Groovy Dance”**
   * **High Danceability** (e.g., ≥0.70\geq 0.70≥0.70): Clearly geared toward tracks that make listeners want to move.
   * Typically associated with dance, EDM, or high-tempo pop.
   * **Example Tracks:** Club-ready, beat-driven songs that dominate dance floors.

### **5. Conclusion**

* **Validity of the Approach:**
  + K-Means, while not a classification or regression algorithm, offers a **baseline** for unsupervised segmentation in this music recommendation context.
  + **SSE** is a valid metric for assessing cluster quality, and the **Elbow Method** helps in deciding the optimal number of clusters.
* **Next Steps:**
  + Incorporate more nuanced feature engineering (e.g., fade duration = start\_of\_fade\_out - end\_of\_fade\_in).
  + Explore supervised techniques if mood labels become available.
  + Compare results with other clustering algorithms or advanced models in subsequent modules.

By satisfying each element—**model choice**, **evaluation metric identification**, **valid interpretation**, and **rationale**—this baseline modeling approach meets the rubric requirements for the **Modeling** criterion, while aligning with the project’s goal of a mood-based recommendation system.