Music Mood Mapping using PCA and Gaussian Mixture Models

Scientific Computing Project

Yerassyl Kudaibergenov

Nazarbayev University

Fall 2025

Common Problem

Project Motivation

- Music recommendation systems rarely understand user **mood** in real time.
- Existing platforms rely on collaborative filtering or past history.
- Goal: Build a system that maps every song into a **continuous emotional space**.
- Use PCA and GMM to uncover latent mood structure from Spotify audio features.
- Output: Recommendations filtered by mood, genre, popularity, and randomness.

Dataset

- Dataset: Spotify audio features (Kaggle, 1921–2020).
- 114,000 tracks with 21 audio features.
- Key features:
  - `danceability`, `energy`, `valence`, `tempo`
  - `acousticness`, `instrumentalness`, `loudness`
  - `speechiness`, `liveness`, `popularity`
- We use 9 numeric features as input to PCA.

Principal Component Analysis (PCA): Idea and Math

Goal: compress high-dimensional audio features into a low-dimensional mood space.

Data:

- Let $X \in \mathbb{R}^{n \times d}$ be the matrix of standardized features (e.g. danceability, energy, valence, tempo, ...).
- Each row $x_i \in \mathbb{R}^d$ is one song.

Covariance matrix:

$$\Sigma = \frac{1}{n-1} X^\top X \in \mathbb{R}^{d \times d}.$$

Eigen-decomposition:

$$\Sigma v_k = \lambda_k v_k,$$

where

- v_k are eigenvectors (principal directions),
- λ_k are eigenvalues (variance along v_k),
- sorted as $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_d$.

so each song becomes $z_i \in \mathbb{R}^2$ (PC1, PC2).

PCA for Mood Embedding

Projection to 2D:

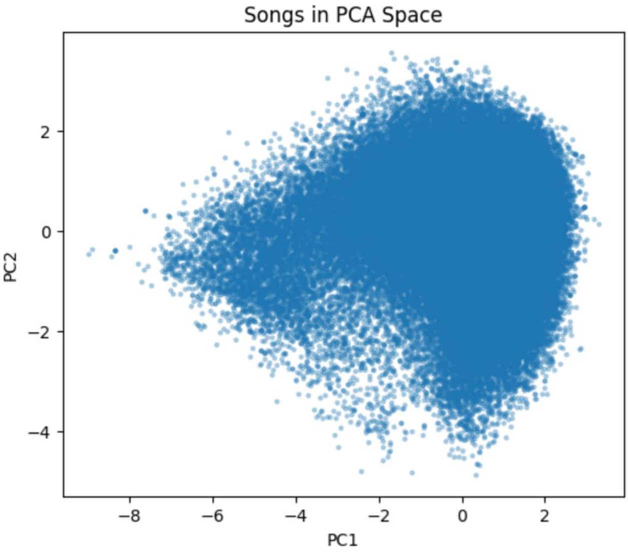
$$Z = XV_2, \quad V_2 = [v_1 \ \ v_2] \in \mathbb{R}^{d \times 2},$$

- Input: Standardized audio feature matrix $X \in \mathbb{R}^{n \times d}$.
- PCA finds eigenvectors of covariance matrix:

$$\Sigma = \frac{1}{n-1} X^\top X$$

- First two principal components describe high-level musical characteristics.
- PC directions are aligned with:
  - PC1 $\rightarrow$ "valence-like" variation
  - PC2 $\rightarrow$ "energy-like" variation

PCA Visualization



Alignment to Mood Axes

- Correlate PCs with true Spotify `valence` and `energy`.
- Determine which PC matches which axis.
- Flip sign of PCs if needed:

$$\mathrm{mood\_x} = \tanh(\mathrm{PC}_{\text{valence-aligned}})$$

$$\mathrm{mood\_y} = \tanh(\mathrm{PC}_{\text{energy-aligned}})$$

- Result: A 2D emotional space:
  - x: Happy $\leftrightarrow$ Sad
  - y: Energetic $\leftrightarrow$ Calm

```
corr(pc1, energy)  = 0.8582112828914792
corr(pc2, energy)  = -0.3458035898279236
corr(pc1, valence) = 0.5089142281547303
corr(pc2, valence) = 0.6302361219718385
Energy-like axis: pc1
Valence-like axis: pc2
```

	pc1	pc2	mood_x	mood_y
0	0.687034	1.049035	0.705316	0.383449
1	-3.189227	1.035571	0.699609	-0.954120
2	-1.315996	-0.216925	-0.179549	-0.649271
3	-3.189109	-0.499216	-0.395016	-0.954113
4	-0.886531	0.271094	0.223031	-0.478808

Goal: discover $K = 8$ soft mood clusters in the 2D mood space.

Model: each song's mood coordinate $x \in \mathbb{R}^2$ is drawn from a mixture of Gaussians:

$$p(x) = \sum_{k=1}^{K} \pi_k \, \mathcal{N}(x \mid \mu_k, \Sigma_k),$$

where

- π_k are mixture weights, $\sum_k \pi_k = 1$,
- $\mu_k \in \mathbb{R}^2$ are cluster centers (mood prototypes),
- $\Sigma_k \in \mathbb{R}^{2\times 2}$ are covariance matrices.

Expectation Step (Probability sample i-th belongs to gaussian k-th):

$$\gamma_{ik} = P(\text{cluster} = k \mid x_i) = \frac{\pi_k \, \mathcal{N}(x_i \mid \mu_k, \Sigma_k)}{\sum_{j=1}^{K} \pi_j \, \mathcal{N}(x_i \mid \mu_j, \Sigma_j)}.$$

Interpretation:

- Each song can belong to several moods with different probabilities.

Gaussian Mixture Model: Maximization (M) Step

$$\hat{\phi}_k = \frac{1}{N}\sum_{i=1}^{N} \hat{\gamma}_{ik}$$

The new kth **mixture weight** $\hat{\phi}_k$ becomes the average probability that a data point belongs to component k.

$$\hat{\mu}_k = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik} x_i}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$$

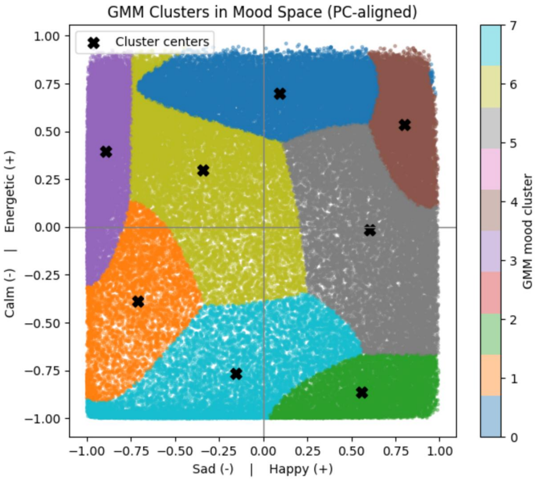
The new kth **mean** $\hat{\mu}_k$ is the **responsibility-weighted average** of all data points.

$$\hat{\sigma}_k^2 = \frac{\sum_{i=1}^{N} \hat{\gamma}_{ik}(x_i - \hat{\mu}_k)^2}{\sum_{i=1}^{N} \hat{\gamma}_{ik}}$$

The new kth **variance** $\hat{\sigma}_k^2$ is the **responsibility-weighted variance** of all data points.

- Here $\hat{\gamma}_{ik}$ are the responsibilities from the E-step: $\hat{\gamma}_{ik} = P(z_k = 1 \mid x_i)$.
- N is the number of data points, $k = 1, \dots, K$ indexes mixture components.

Mood Space Visualization



- Each dot = a song projected into (mood_x, mood_y) space.
- Clear structure emerges along the 2 axes.

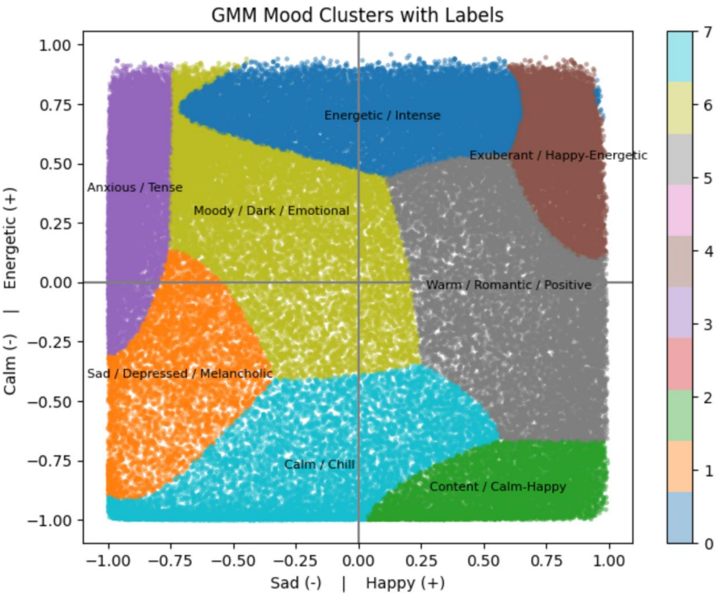
Gaussian Mixture Model

- Fit an 8-component GMM on the mood space:

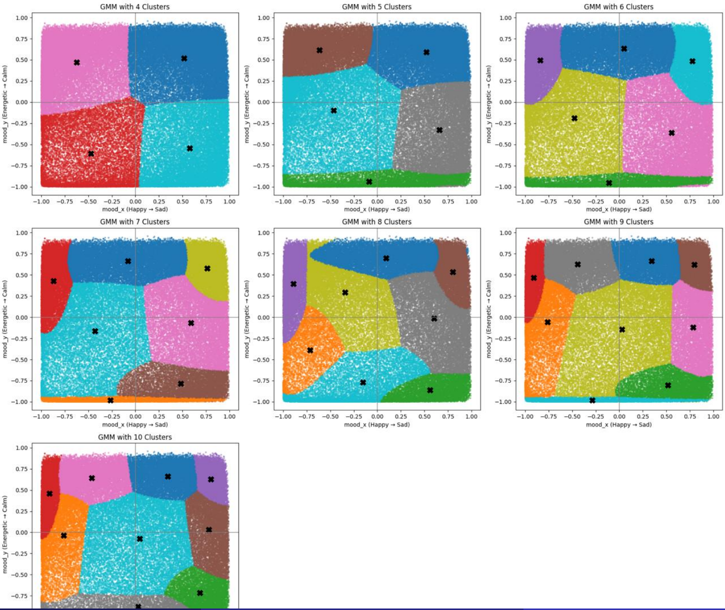
$$p(x) = \sum_{k=1}^{8} \pi_k \mathcal{N}(x \mid \mu_k, \Sigma_k)$$

- Soft clustering: each song has probabilities for each mood cluster.
- Estimated cluster centers reflect distinct emotional regions.
- GMM chosen over k-means because moods are elliptical and overlapping.

8 Mood Clusters



Different Cluster Numners



Recommendation Method

- User selects:
  - Desired mood point (x, y) OR one of 8 clusters.
  - Preferred genres.
  - Popularity range.
- Compute distance:

$$d = \sqrt{(mood\_x - x)^2 + (mood\_y - y)^2}$$

- Take N closest songs $\rightarrow$ randomly sample 10.
- Ensures variety and prevents repetition.

Recommendation Example

```python
target = (0.8, 0.6)  # very happy + energetic

recs = recommend_songs_mood_space(
    df,
    target_mood=target,
    genres=["pop", "dance pop"],
    min_popularity=80,
    max_popularity=100,
    neighborhood_size=50,
    n_random=10
)

recs
```

✓ 0.0s

	track_name	artists	track_genre	popularity	mood_x	mood_y	mood_distance
0	Blank Space	Taylor Swift	pop	85	0.693350	0.538896	0.122914
1	Sorry	Justin Bieber	pop	82	-0.026516	0.573384	0.826945
2	Happier Than Ever	Billie Eilish	pop	88	0.317551	-0.848757	1.526975
3	Until I Found You	Stephen Sanchez	pop	90	0.239013	-0.477542	1.214827
4	Payphone	Maroon 5;Wiz Khalifa	pop	84	0.338811	0.677447	0.467646
5	Say You Won't Let Go	James Arthur	pop	85	0.345705	-0.510700	1.200016
6	Shape of You	Ed Sheeran	pop	86	0.966057	0.553854	0.172349
7	Baby	Justin Bieber;Ludacris	pop	82	0.621856	0.613577	0.178660
8	Night Changes	One Direction	pop	88	0.707068	-0.407057	1.011335
9	Watermelon Sugar	Harry Styles	pop	89	-0.078075	0.574145	0.878456

- For the mood "Happy-Energetic", pop and dance pop genre, popularity 80–100.
- System retrieves 10 closest tracks.

Evaluation

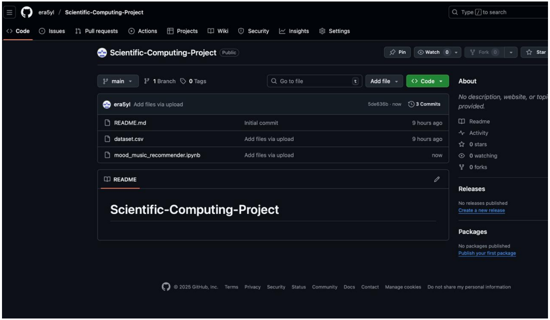
- PCA explained variance: usually 35-45% in first 2 PCs.
- GMM clusters visually separable.
- Recommendations are:
    - Consistent across runs.
    - Smoothly transitioning across moods.
    - Sensitive to user-selected filters.

Conclusion

- Built a full pipeline from raw audio features to mood-aware song recommendations.
- PCA creates interpretable 2D emotional map.
- GMM with 8 components discovers natural mood clusters.
- Recommendation engine supports:
    - Mood selection
    - Genre filtering
    - Popularity filtering
    - Randomized sampling
- Extensible to personalization, neural models, or more mood dimensions.

Version Control and Tools

- The version control I used is GitHub.comThe link: https://github.com/era5yl/Scientific-Computing-Project



- The programming language: Python
- Libraries: NumPy, Pandas, Matplotlib, Scikit-Learn

Questions?