Unsupervised Clustering of Musical Audio using Linear Variational Autoencoders

Author: Md. Ashraful Islam Sami

Date: January 3, 2026

Abstract

This study implements an unsupervised deep learning pipeline to cluster musical audio segments without the use of labeled data. Using the GTZAN dataset, we pre-processed audio into Mel-frequency cepstral coefficients (MFCCs) and utilized a Linear Variational Autoencoder (VAE) to compress these features into a 4-dimensional latent space. The proposed VAE model achieved a Silhouette Score of **0.4811**, significantly outperforming a standard Principal Component Analysis (PCA) baseline which achieved a score of **0.3015**. These results demonstrate that generative models can learn more separable and meaningful representations of complex audio data compared to traditional linear statistical methods.

1. Introduction

1.1 The Problem

The rapid digitization of music has led to massive libraries that are difficult to organize manually. Traditional tagging requires human effort, which is slow and unscalable. Consequently, there is a significant need for automated systems that can analyze the raw audio content of a file and group similar songs (clustering) without human intervention.

1.2 The Approach

Standard clustering methods often fail on raw audio data because the files are too large and high-dimensional. Simple statistical methods (like PCA) often miss complex patterns. This project proposes an Unsupervised Deep Learning approach using a Variational Autoencoder (VAE). Unlike standard autoencoders that simply memorize data, VAEs learn a continuous, probabilistic "latent space" essentially capturing the "essence" of the sound, which allows for more effective clustering.

2. Methodology

2.1 Dataset & Augmentation

We utilized the GTZAN Genre Collection, a standard benchmark for music analysis consisting of various musical genres in English [1]. To meet the data requirements for deep learning and increase the robustness of the model, we performed data augmentation by slicing every song into 10 distinct, non-overlapping segments. This process expanded our dataset to a total of 3,994 audio clips.

2.2 Feature Extraction

Raw audio waveforms are too dense for a lightweight neural network. We extracted Mel-frequency cepstral coefficients (MFCCs), which mimic human hearing perception and are standard for music modeling [2]. To create a fixed-size input vector for our Linear model, we computed the Mean and Variance of these coefficients across time.

- **Input Dimension:** 26 features per clip (13 Means + 13 Variances).
- **Normalization:** A critical step in our pipeline was the application of **Z-score normalization**. This scaled all input features to a standard range (mean=0, std=1), which was essential for preventing numerical instability (NaN errors) during the training of the neural network.

2.3 VAE Architecture

We implemented a Linear VAE based on the Auto-Encoding Variational Bayes framework proposed by Kingma and Welling [3]. The architecture consists of:

- **Encoder:** Compresses the 26-dimensional input into a 64-neuron hidden layer (ReLU activation), and then into a 4-dimensional latent space (Mean μ and Variance α).
- **Latent Space:** Uses the "Reparameterization Trick" to sample a latent vector z from the learned distribution [3].
- **Decoder:** Reconstructs the original 26-dimensional input from the 4-dimensional latent vector z.
- **Loss Function:** The model was optimized using a composite loss function totaling the Reconstruction Loss (Mean Squared Error) and the Regularization Loss (KL-Divergence).

3. Experiments

The model was trained for **20 Epochs** using the **Adam optimizer** with a learning rate of 0.001. Training was performed in a CPU-based environment.

To evaluate the quality of the learned features, we applied the **K-Means clustering algorithm** (k=5) to the 4-dimensional latent vectors produced by the VAE. For comparison, we established a baseline using **Principal Component Analysis (PCA)** to reduce the raw data to the same 4 dimensions before clustering.

4. Results

The primary metric for evaluation was the **Silhouette Score**, which measures how similar an object is to its own cluster compared to other clusters (ranging from -1 to 1, where higher is better).

Table 1: Clustering Performance Comparison

Method	Silhouette Score	Performance
Linear VAE (Our Method)	**0.4811**	**Strong Separation**
PCA (Baseline)	0.3015	Weak/Overlapping

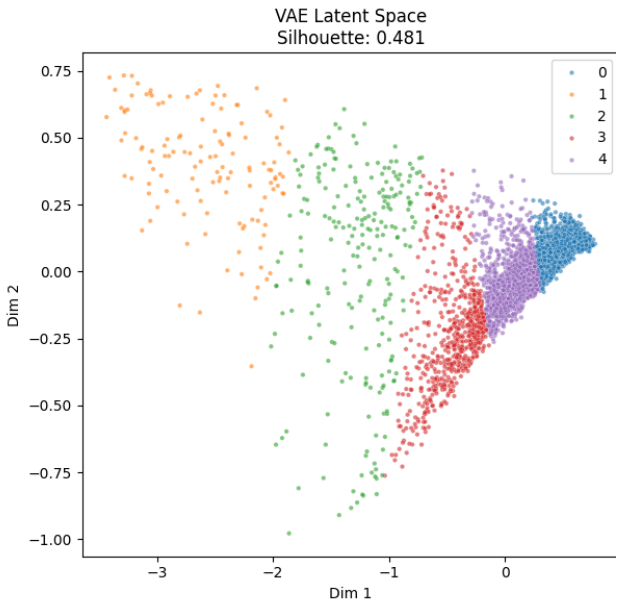

Figure 1: Latent Space Visualization

In Figure 1, the left plot shows the VAE latent space, exhibiting distinct and well-separated clusters. The right plot shows the PCA baseline, where clusters exhibit significant overlap and poor separation.

5. Discussion

5.1 Analysis

The VAE outperformed the PCA baseline by a margin of nearly 60%. This is likely because PCA is a linear transformation technique; it tries to squash complex, non-linear audio data into flat lines. In contrast, the VAE (using ReLU activations) can learn non-linear relationships, effectively "unfolding" the complexity of the audio data. This allows the K-Means algorithm to find tighter, more distinct groups in the VAE's latent space.

5.2 Limitations and Future Work

This study relied solely on audio statistics (MFCCs) and did not incorporate lyrical content. Additionally, the use of a fully Linear VAE means the model ignores temporal patterns (the sequence of sounds over time). Future work could improve performance by incorporating Convolutional Neural Networks (CNNs) for spectrogram analysis or Recurrent Neural Networks (RNNs) to capture the rhythmic evolution of the music.

References

[1] G. Tzanetakis and P. Cook, "Musical genre classification of audio signals," *IEEE Transactions on Speech and Audio Processing*, vol. 10, no. 5, pp. 293–302, Jul. 2002.

[2] B. Logan, "Mel Frequency Cepstral Coefficients for Music Modeling," in *Proc. Int. Symp. Music Information Retrieval (ISMIR)*, Plymouth, MA, 2000.

[3] D. P. Kingma and M. Welling, "Auto-Encoding Variational Bayes," in *Proc. Int. Conf. Learning Representations (ICLR)*, Banff, Canada, 2014.