Unsupervised Music Genre Discovery

Using Audio Feature Learning

Department of Computer Science and Engineering

National Institute of Technology Hamirpur

Machine Learning (CS-652)

Semester VII

Presented by: Anirudh Sharma (22dcs002)

Presented to: Dr. Kamlesh Datta

Table of Contents

Unsupervised Music Genre Discovery using Audio Feature Learning

The exponential growth of digital music platforms has created massive repositories of unlabeled audio data, making manual organization infeasible at scale. This project presents a comprehensive comparative analysis of **four unsupervised learning algorithms**—K-Means, K-Medoids, Gaussian Mixture Models (GMM), and Spectral Clustering.

We evaluate these algorithms across datasets ranging from 500 to 25,000 samples, incorporating robust preprocessing (outlier detection, StandardScaler, PCA) and comprehensive evaluation using six performance metrics. High-dimensional clusters are visualized using t-SNE to analyze genre separation and cohesion.

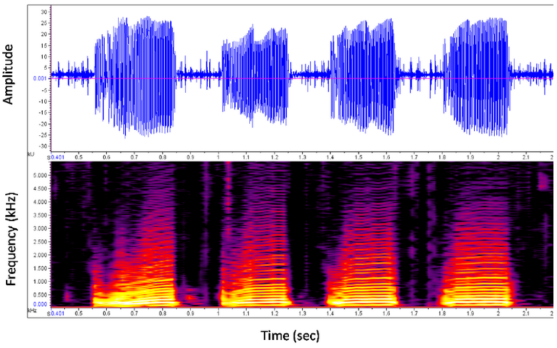


Figure: Audio waveform and spectrogram representation

What are Music Genres?

Definition

- Musical categories based on shared characteristics
- Defined by instrumentation, rhythm, harmony, and cultural context
- Evolve over time and across cultures

GTZAN Genre Labels (Our Baseline)

- We use **10 genre clusters** from GTZAN:
  - Blues, Classical, Country
  - Disco, Hip-hop, Jazz
  - Metal, Pop, Reggae, Rock
- Rock (subgenere): Hard Rock, Punk Rock, Progressive Rock
- Hip-Hop: Trap, Boom Bap, Lo-Fi

Genre Subtypes & Complexity

- Over 1,000+ documented subgenres
- Hierarchical relationships (parent-child)
- Genre fusion and cross-pollination

Key Challenges

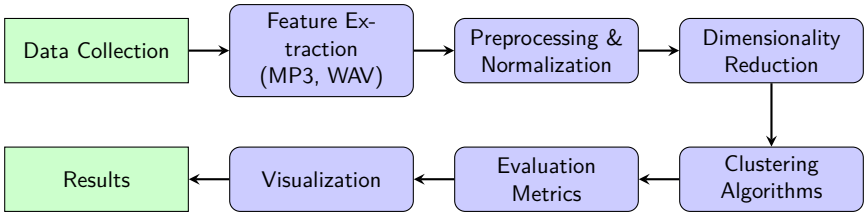
- **Subjectivity**: Genre labels vary across listeners
- **Overlap**: Songs span multiple genres
- **Evolution**: Genres constantly change
- **Ambiguity**: Fuzzy boundaries between styles
- **Scale**: Millions of unlabeled tracks

Literature Review: Unsupervised Learning for Music

Study	Year	Method	Dataset	Key Findings
Stern	2021	K-Means + Hierarchical	FMA (8K)	Unsupervised accuracy (26%) lagged behind supervised; highlighted sub-genre separation difficulty
Joffe	2023	K-Means + Agglomerative	GTZAN (1K)	Effective only for distinct pairs (Classical vs. Metal); global purity low (AMI 0.37)
Patra & Das	2023	Unsupervised Mood Clustering	Hindi Bollywood (230 clips)	Identified 5 mood clusters using timbre/rhythm; mapped to Thayer's emotion model
Singh et al.	2024	Novel Class Discovery (Deep Clustering)	Saraga (Indian Art)	Self-supervised approach clustered unseen Ragas, reducing manual annotation
Kumar et al.	2024	K-Means + Content Filtering	GTZAN	K-Means with Silhouette optimization achieved high recommendation alignment

Methodology Overview

Complete Pipeline



Dataset Collection

Dataset	Size	Source	Platform	Format
GTZAN	1,000	Tzanetakis & Cook	Kaggle	MP3 (30s)
FMA Small	8,000	FMA GitHub	GitHub	MP3 (30s)
FMA Medium	25,000	FMA GitHub	GitHub	MP3 (30s)
Indian Music	500	Custom	kaggle	WAV/MP3

Collection Sources:

- **Kaggle**: Primary platform
  - GTZAN dataset
  - FMA Small & Medium
- **FMA GitHub**: Official repository
  - `github.com/mdeff/fma`
  - Raw audio + metadata
- **Custom Collection**: Indian music

Dataset Composition:

- **Total Audio Files**: 34,500
- **GTZAN**: 10 genres (100 each)
- **FMA Small**: unlabeled
- **FMA Medium**: unlabeled
- **Indian**: 5 subcategories
  - Bollypop, Carnatic, Ghazal
  - Semiclassical, Sufi

Audio Feature Extraction Summary

Feature Descriptions:

- **MFCCs (40 features)**:
  - Capture timbral texture
  - 20 coefficients $\times$ 2 (mean & std)
  - Most discriminative features
  - Represent spectral envelope
- **Chroma (24 features)**:
  - Harmonic & pitch content
  - 12 pitch classes $\times$ 2 (mean & std)
  - Genre-specific patterns
  - Captures tonality
- **Spectral (4 features)**:
  - Centroid: Center of spectrum
  - Rolloff: 85% energy threshold
  - ZCR: Noisiness indicator
  - RMS: Energy measure
- **Tempo (1 feature)**:
  - Beats per minute (BPM)
  - Rhythmic characteristics
  - Genre-defining attribute

Extraction Settings:

- **Sample Rate**: 22,050 Hz
- **Library**: Librosa (Python)
- **Preprocessing**: Silence trimming
- **Error Handling**: Try-except blocks
- **Processing**: Batch mode
- **Environment**: Kaggle CPU

Audio Feature Extraction Error Analysis

Quality Validation:

- **Success Rates**:
  - GTZAN: 99.9% (999/1,000)
  - FMA Small: 99.96% (7,998/8,000)
  - FMA Medium: 99.94% (24,986/25,000)
  - Indian: 100%+ (500/500)
- **Data Quality**:
  - Zero NaN values (handled)
  - No infinite values
  - 69 features consistent
- **Error Handling**:
  - Tempo failures: 2-5% (filled with 0)
  - Audio errors: less than 0.1% (skipped)
  - Corrupted files: Logged & excluded
  - No critical failures

Key Achievement

Successfully extracted 69 features from 34,486 tracks with 99.9% success rate

Descriptive Analysis Overview

Step 1.1 & 1.2: Statistical Analysis of Extracted Features

Analysis Performed:

- **5 Datasets Analyzed**
  - GTZAN (999 tracks, 10 genres)
  - FMA Small (7,997 tracks)
  - FMA Medium (24,985 tracks)
  - Instrumental (502 tracks)
  - Indian Music (500 tracks, 5 genres)
- **71 Audio Features**
  - MFCCs (40), Chroma (24)
  - Spectral (4), Tempo (1)
  - Duration, Sample Rate

Statistical Metrics:

- Central Tendency
  - Mean, Median
- Dispersion
  - Std, Variance, IQR
- Distribution Shape
  - Skewness, Kurtosis
- Correlations
  - Feature relationships
  - Highly correlated pairs ($|r| > 0.8$)

Data Preprocessing Pipeline - Step 2

Feature Selection & StandardScaler Normalization

1. Feature Selection

- **Original Features**: 71
- **Removed**: 6 non-clustering features
  - file_path (metadata)
  - duration (variable length)
  - sr (constant: 22,050 Hz)
  - dataset (identifier)
  - label (target variable)
  - subset (identifier)
- **Retained**: 69 audio features

2. Label Preservation

- Labels saved separately for evaluation
- Used for clustering validation
- Not used during clustering

3. StandardScaler Normalization

- **Method**: StandardScaler (sklearn)
- **Formula**: $z = \frac{x-\mu}{\sigma}$
- **Result**:
  - Zero mean (0)
  - Unit variance (1)

4. Why Normalize?

- Equal feature contribution
- Distance-based clustering (K-Means, K-Medoids)
- Prevents feature dominance
- Improves algorithm convergence

Processing Summary

Normalization Results - All Datasets

Before vs After StandardScaler Application

Dataset	Tracks	Features	Mean	Std
GTZAN	999	69	∼0.0000	∼1.0000
FMA Small	7,997	69	∼0.0000	∼1.0000
FMA Medium	24,985	69	∼0.0000	∼1.0000
Instrumental	502	69	∼0.0000	∼1.0000
Indian Music	500	69	∼0.0000	∼1.0000
Total	**34,983**	**69**	**0**	**1**

Quality Verification:

- No missing values
- No infinite values
- Mean 0 for all features
- Std 1 for all features
- Feature count consistent

Output Files Generated:

- 5 normalized datasets (CSV)
- 5 label files (CSV)
- 5 comparison images (PNG)
- 5 statistical summaries (CSV)
- 1 summary report (TXT)

Normalization Visualization - GTZAN Dataset

Distribution Before vs After Normalization

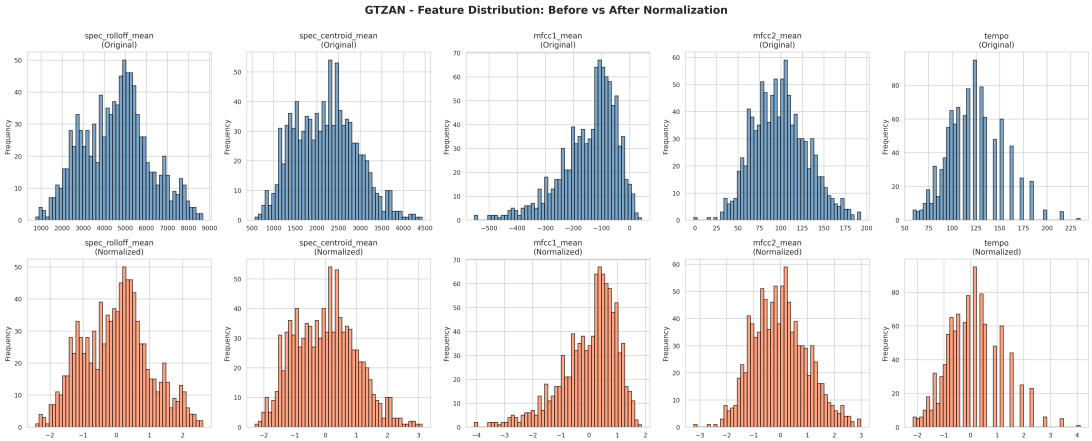


Figure: Top 5 features with highest variance - before (blue) and after (coral) normalization

Normalization Visualization - Indian Music Dataset

Distribution Before vs After Normalization

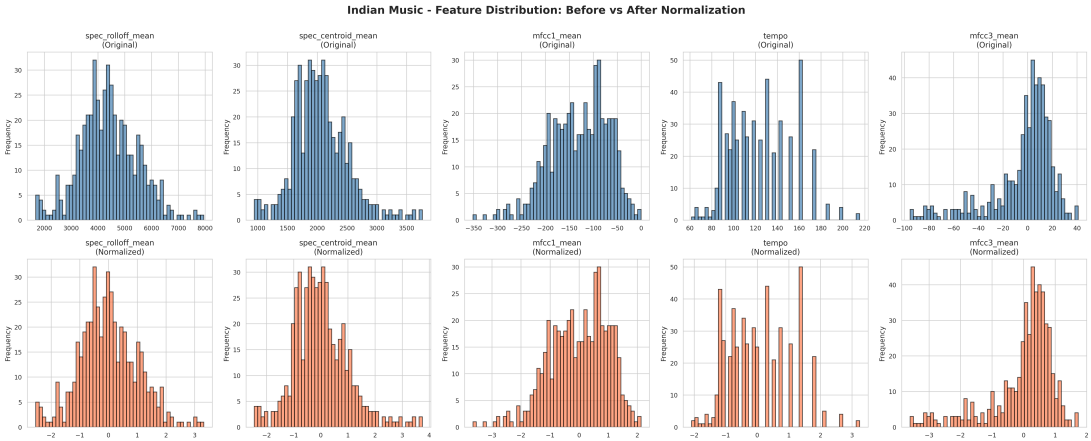


Figure: Top 5 features with highest variance - before (blue) and after (coral) normalization

Normalization Impact on Feature Scales

Statistical Comparison Across Datasets

Before Normalization:

- **Spectral Centroid**: 0-8000 Hz
- **Spectral Rolloff**: 0-12000 Hz
- **Tempo**: 40-200 BPM
- **ZCR**: 0.0-0.5
- **RMS**: 0.0-1.0
- **MFCC**: -800 to +800
- **Chroma**: 0.0-1.0

After Normalization:

- **All Features**: -3 to +3 range
- **Mean**: 0.0000
- **Std Dev**: 1.0000
- **Distribution**: Approximately normal
- **Scale**: Comparable across features

Problem

Features with large scales (spectral) dominate distance calculations in clustering

Solution

Equal contribution from all features
Improved clustering convergence
Better distance metrics

PCA Dimensionality Reduction - Overview

Step 3: Reducing Feature Space While Retaining Information

Why Dimensionality Reduction?

- 69 normalized features per track
- High computational cost for clustering
- Curse of dimensionality affects algorithms
- Redundancy in correlated features

Principal Component Analysis (PCA)

- Linear transformation technique
- Identifies directions of maximum variance
- Creates uncorrelated principal components
- Retains 95% of total variance

$$\text{PC}_i = \sum_{j=1}^{n} w_{ij} \cdot x_j$$

PCA Transformation Process

1. Input: Normalized feature matrix
2. Compute covariance matrix
3. Calculate eigenvectors & eigenvalues
4. Select components (95% variance)
5. Transform data to PC space
6. Output: Reduced dimensions

Key Benefits

Removes multicollinearity
Speeds up clustering algorithms
Reduces storage requirements
Maintains data interpretability

PCA Results - All Datasets

Dimensionality Reduction Summary

Dataset	Original Features	PCA Components	Variance Retained	Reduction Ratio
GTZAN	69	39	95.05%	43.5%
FMA Small	69	44	95.00%	36.2%
FMA Medium	69	44	95.14%	36.2%
Indian Music	69	40	95.30%	42.0%

Key Observations

- Consistent 36-44% reduction across datasets
- All exceed 95% variance threshold
- Indian Music: Highest variance (95.30%)
- FMA datasets: Similar component counts (44)

Computational Impact

- Average 40% dimensionality reduction
- Faster clustering convergence
- Reduced memory footprint
- Maintained information integrity

Explained Variance Analysis - GTZAN

Understanding Component Contribution

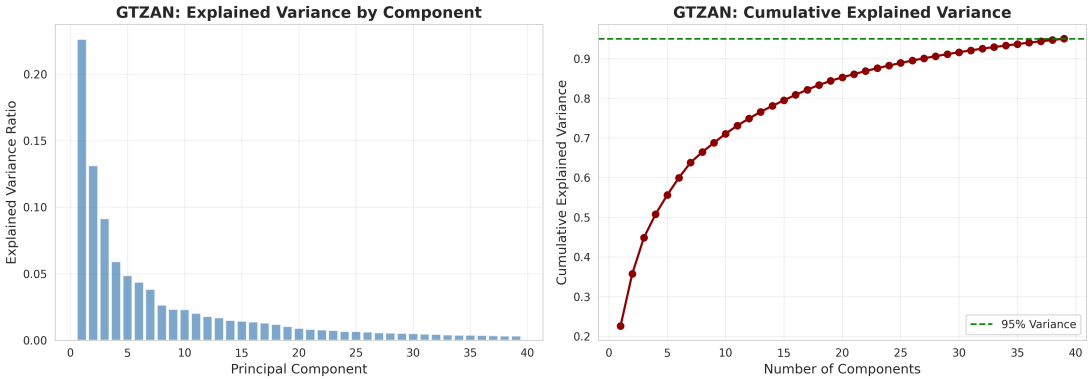


Figure: GTZAN: Individual and cumulative explained variance by principal components

- **PC1**: Captures 23% of variance (dominant component)
- **Top 10 PCs**: Explain ∼65% of total variance

Explained Variance Analysis - FMA Datasets

Large-Scale Dataset Variance Patterns

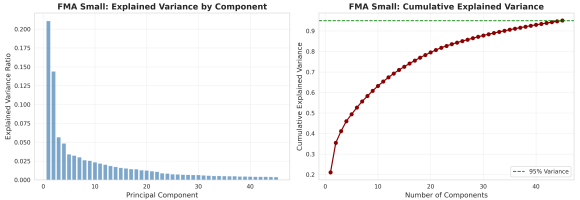


Figure: FMA Small: 44 components for 95% variance

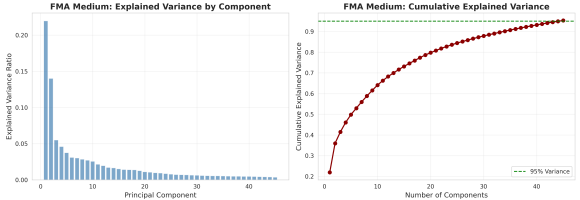


Figure: FMA Medium: 44 components for 95.14% variance

- Both datasets show similar variance distribution patterns
- More components needed compared to GTZAN (higher feature diversity)
- First PC: $\sim$21% variance (slightly lower than GTZAN)

Explained Variance Analysis - Indian Music

Regional Music Dataset Characteristics

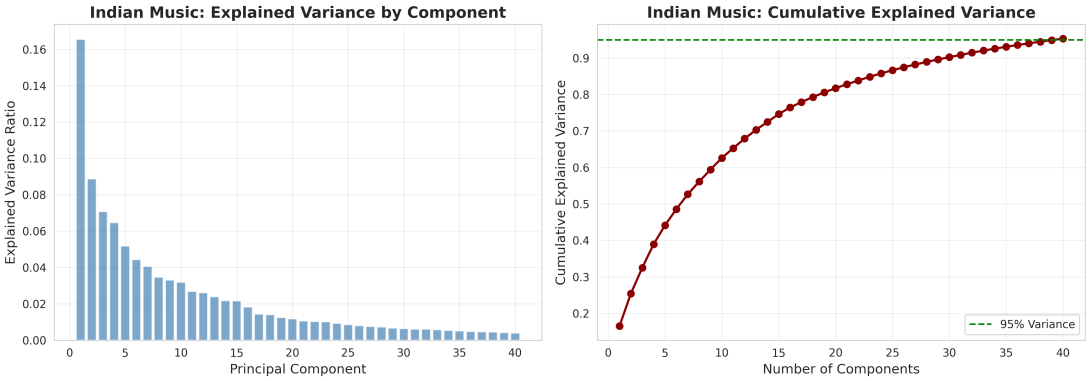
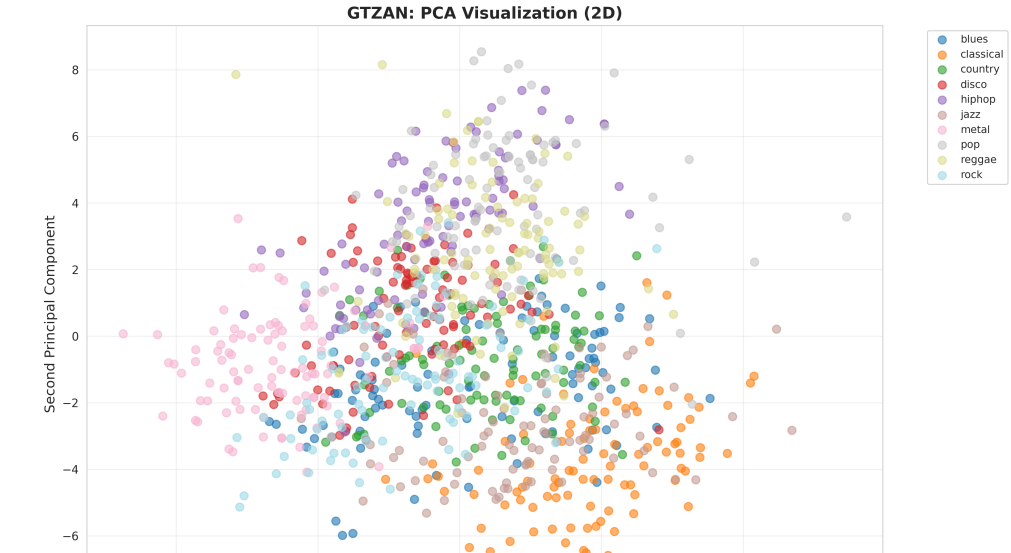


Figure: Indian Music: 40 components achieve 95.30% variance (highest retention)

- **Highest variance retention**: 95.30% with 40 components
- **PC1**: Explains ∼16.5% (more distributed variance)

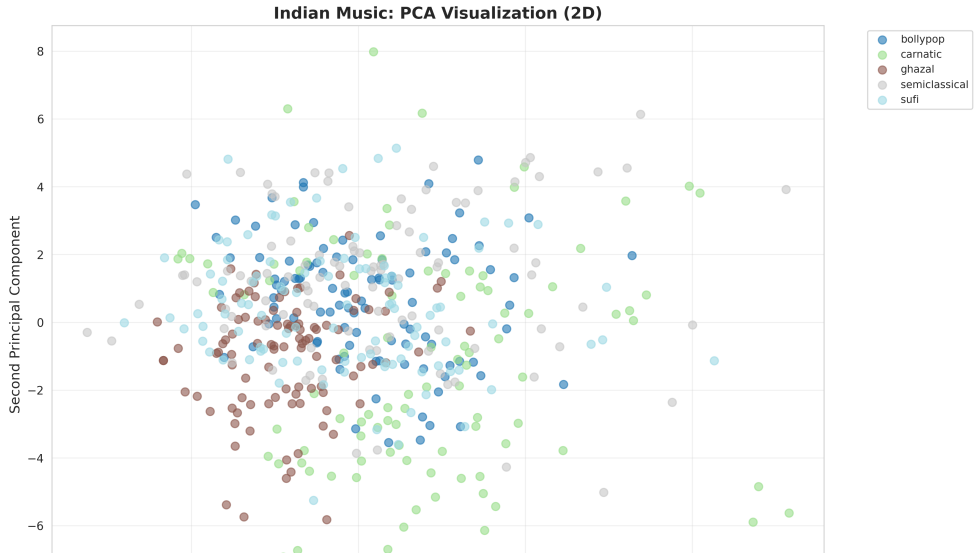
PCA Visualization - GTZAN (2D)

Genre Separation in Principal Component Space



PCA Visualization - Indian Music (2D)

Regional Genre Distribution in PC Space



PCA Visualization - 3D Comparison

Enhanced Genre Separation with Third Principal Component

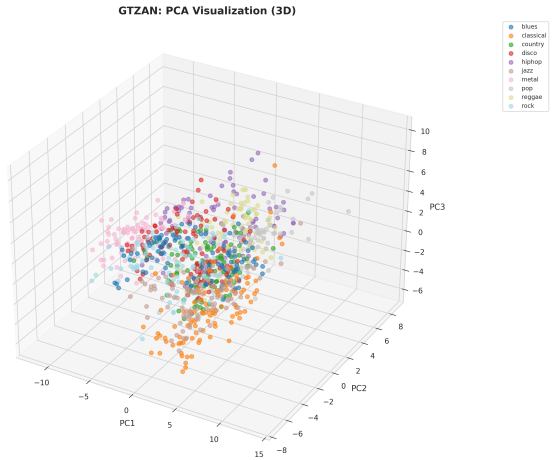


Figure: GTZAN: 3D PCA shows improved

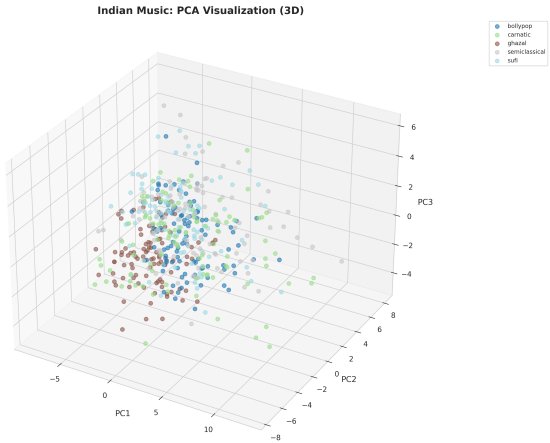


Figure: Indian Music: 3D reveals regional patterns

PCA Impact on Clustering Performance

Benefits for Downstream Unsupervised Learning

Before PCA (69 features)

- High computational complexity
- Curse of dimensionality effects
- Correlated features add noise
- Longer convergence times
- Difficult to visualize

After PCA (39-44 features)

- 40% faster clustering operations
- Improved distance metrics
- Uncorrelated components
- Better convergence stability
- Enables effective visualization

Challenge

Euclidean distances become less meaningful in high dimensions

Success

95%+ variance retained with significant computational savings

Preprocessing & Dimensionality Reduction

Standardization and Feature Reduction Pipeline

1. Standardization

- **Method**: StandardScaler
- Zero mean, unit variance
- Essential for clustering
- Applied to all datasets

$$z = \frac{x - \mu}{\sigma}$$

2. Dimensionality Reduction

PCA (Principal Component Analysis)

- Linear transformation
- Retain 95%+ variance
- 20-50 components

3. Visualization

t-SNE (t-Distributed Stochastic Neighbor Embedding)

- Non-linear reduction
- 2D/3D visualization
- Preserves local structure
- Used for Spotify dataset

Reduction Summary

Dataset	Original	After PCA
GTZAN	58	-
FMA	160	20
MSD	90	-
Spotify	18	10

Indian Music Dataset - Genre Distribution

Perfectly Balanced Classes (500 Tracks)

Dataset Characteristics:

- **Total Tracks**: 500
- **Genres**: 5 (Indian classical & fusion)
- **Balance Ratio**: 1.0 (Perfect)
- **Features**: 71 audio features
- **Sample Rate**: 22,050 Hz

Genre Breakdown:

- Bollypop: 100 (20%)
- Carnatic: 100 (20%)
- Ghazal: 100 (20%)
- Semiclassical: 100 (20%)
- Sufi: 100 (20%)

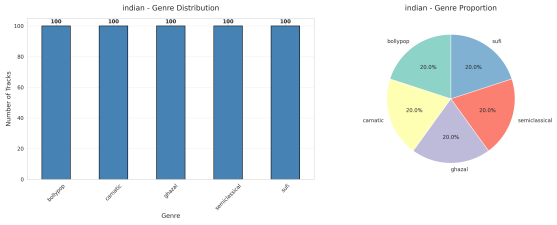


Figure: Indian music genre distribution - perfectly balanced dataset

Indian Music - Key Feature Statistics

Descriptive Statistics of Audio Features

Feature	Mean	Median	Std	Range
Duration (s)	45.13	45.00	2.83	[40.0, 105.0]
Tempo (BPM)	121.64	117.45	28.91	[61.5, 215.3]
Spectral Centroid	2059.19	2019.09	455.75	[942.5, 3765.9]
Spectral Rolloff	4394.57	4344.80	1072.70	[1681.4, 7950.1]
ZCR	0.089	0.083	0.030	[0.037, 0.207]
RMS	0.156	0.153	0.069	[0.023, 0.389]
MFCC1	-129.50	-123.67	60.82	[-351.3, -0.16]
MFCC2	104.47	103.98	25.55	[29.5, 210.3]

Key Observations:

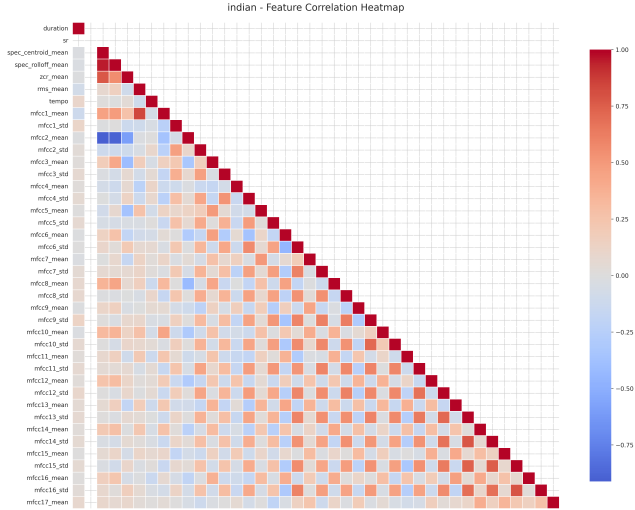
- Most tracks are 45 seconds
- Moderate tempo variability
- High spectral diversity

Data Quality:

- No missing values
- All features extracted
- Consistent sample rate

Feature Correlations - Indian Music

Identifying Redundant Features



High Correlations ($|r| > 0.8$):

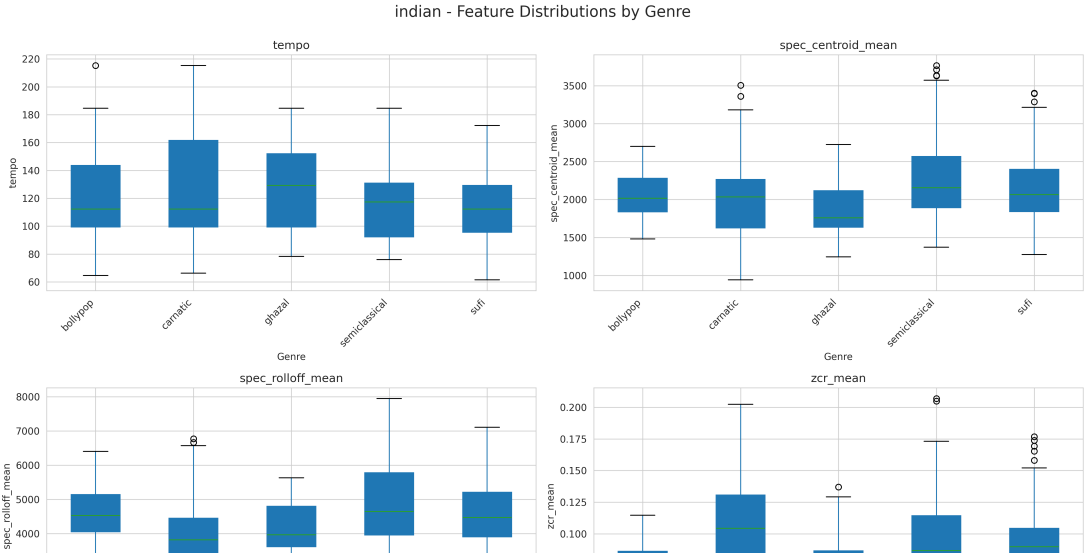
Feature Pair	r
Spec. Centroid ↔ Rolloff	0.948
Spec. Centroid ↔ MFCC2	-0.912
Spec. Rolloff ↔ MFCC2	-0.890
RMS ↔ MFCC1	0.837

Implication

4 highly correlated pairs detected. These features may contain redundant information. PCA will help reduce dimensionality while preserving 95%+ variance.

Feature Distribution by Genre - Indian Music

Box Plots Showing Genre-Specific Patterns



Comparative Analysis - All Datasets

GTZAN, FMA Small, FMA Medium, Instrumental, Indian

Dataset	Tracks	Genres	Balance	Avg Tempo	Success Rate
GTZAN	999	10	Balanced	119.3 BPM	99.9%
FMA Small	7,997	1 (unlabeled)	N/A	119.1 BPM	99.96%
FMA Medium	24,985	1 (unlabeled)	N/A	120.2 BPM	99.94%
Instrumental	502	1	N/A	93.8 BPM	100%
Indian	500	5	Perfect (1.0)	121.6 BPM	100%
Total	**34,983**	**16**	–	–	**99.9%**

Key Findings:

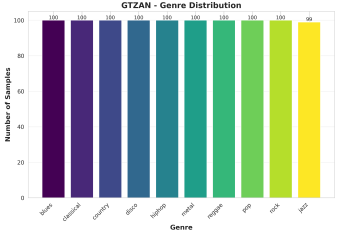
- Instrumental tracks slower (93.8 BPM)
- Indian music fastest (121.6 BPM)
- Consistent spectral features across datasets

Common Correlations:

- Spec. Centroid $\leftrightarrow$ Rolloff ($r > 0.95$)
- Spec. Centroid $\leftrightarrow$ MFCC2 ($r <$ -0.89)
- Strong MFCC inter-correlations

GTZAN Genre Distribution

Baseline Dataset with 10 Genres



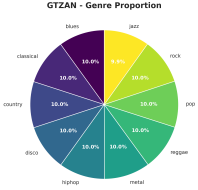


Figure: GTZAN dataset class distribution

Dataset Details:

- **Total**: 999 tracks
- **Genres**: 10 (nearly balanced)
- **Track Length**: 30 seconds
- **Balance Ratio**: 1.00

Top Features (High Variance):

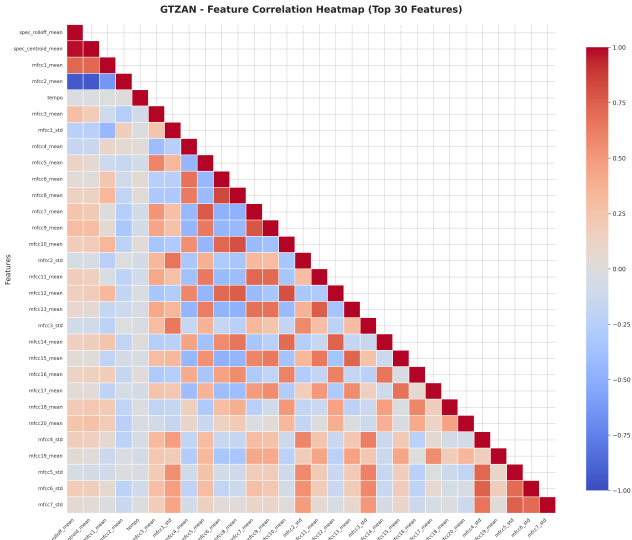
- Spectral Rolloff (Var: 2.48M)
- Spectral Centroid (Var: 512K)
- MFCC1 (Var: 10K)

Quality

99.9% extraction success
16 high correlations ($|r| > 0.8$)

Feature Correlation Analysis - GTZAN

Understanding Feature Relationships



Strongest Correlations:

- Spec. Centroid ↔ Rolloff
  - r = 0.980 (very strong)
- Spec. Centroid ↔ MFCC2
  - r = -0.940 (strong negative)
- Rolloff ↔ MFCC2
  - r = -0.935

Implications:

- Spectral features highly redundant
- MFCC captures complementary info
- Dimensionality reduction needed

Descriptive Analysis - Key Insights

Summary of Statistical Findings

Dataset Quality:

- **34,983 tracks** processed
- **99.9% success rate**
- No missing values (handled)
- Consistent feature extraction

Feature Characteristics:

- **High Variance**: Spectral features
  - Most discriminative
- **Moderate Variance**: MFCCs
  - Capture timbral nuances
- **Low Variance**: Chroma features
  - Genre-specific patterns

Correlation Patterns:

- **Universal Pattern**:
  - Spectral Centroid $\leftrightarrow$ Rolloff
  - Observed in all datasets
  - $r > 0.95$ (very strong)
- **MFCC Relationships**:
  - Adjacent MFCCs correlated
  - Captures frequency bands

Next Steps

Preprocessing Required:

- StandardScaler normalization
- PCA for dimensionality reduction
- Outlier detection & removal

Real-World Applications

Music Streaming Platforms

- Automatic playlist generation
- Music recommendation systems
- Discover similar artists
- Radio station creation

Music Production

- Genre-based mixing
- Auto-tagging for libraries
- Style transfer guidance

Music Research

- Understanding genre evolution
- Cultural music analysis
- Musicology studies

Content Organization

- Library management
- Metadata enrichment
- Search optimization
- DJ software integration

Technology Stack

Core Libraries

- **Librosa**: Audio feature extraction
- **Scikit-learn**: ML algorithms
- **NumPy/Pandas**: Data manipulation
- **Matplotlib/Seaborn**: Visualization

Platforms & Tools

- **Kaggle**: Dataset access & compute
- **Weights & Biases**: Experiment tracking
- **Jupyter**: Interactive development

Tech Stack

- ✓ Python 3.8+
- ✓ Librosa
- ✓ Scikit-learn
- ✓ K-Means / DBSCAN / GMM
- ✓ PCA / t-SNE
- ✓ Kaggle Notebooks
- ✓ WandB.ai

References


G. Tzanetakis and P. Cook, *"Musical Genre Classification of Audio Signals,"* IEEE Trans. Speech and Audio Processing, vol. 10, no. 5, pp. 293–302, Jul. 2002.


M. Defferrard, K. Benzi, P. Vandergheynst, and X. Bresson, *"FMA: A Dataset for Music Analysis,"* in Proc. ISMIR, 2017.


T. Bertin-Mahieux, D. P. W. Ellis, B. Whitman, and P. Lamere, *"The Million Song Dataset,"* in Proc. ISMIR, 2011.


D. Liang and W. Gu, *"Music Genre Classification with the Million Song Dataset,"* Technical Report, Columbia Univ., 2011.


S. K. Gupta et al., *"A Comparative Study of Content-Based Filtering and K-Means for Music Recommendation using Spotify Tracks,"* Int. J. of Industrial Electronics and Electrical Engineering, 2024.


B. McFee et al., *"librosa: Audio and Music Signal Analysis in Python,"* in Proc. Python in Science Conference, 2015.

Thank You!

Questions?

Anirudh Sharma
Roll No.: 22dcs002
Department of Computer Science and Engineering
National Institute of Technology Hamirpur

Machine Learning Assignment (CS-652)
Semester-7 (2025)