Decoding Spotify: The Influence of Starting Songs on Recommendations

— Project Report —

Fallstudien II / Case Studies II

Lecturers:

Prof. Dr. Katja Ickstadt
Dr. Henrike Weinert
Yassine Talleb
Carmen van Meegen

Author:

Rahul Ramesh Vishwkarma
Matriculation number: 236862

Group number: 1

Group members:

Anirudh Parameswaran
Harsh Yadav

February 10, 2026

TU Dortmund University

Contents

1 Introduction

Music recommendation systems have become an integral part of modern streaming platforms, shaping how listeners discover new songs and artists. These algorithms rely heavily on audio features, user preferences, and contextual factors to generate personalized playlists. However, the extent to which the acoustic characteristics of initially selected songs influence subsequent recommendations within a listening session remains unclear. Understanding this relationship is essential for improving algorithmic transparency, enhancing personalization quality, and ensuring diversity in musical discovery.

This project investigates how the audio features of first two starting songs affect the characteristics of recommended tracks generated by Spotify's recommendation algorithm. Specifically, it examines whether recommendations tend to preserve or diverge from the acoustic properties, such as *energy*, *valence*, *danceability*, and *acousticness*, of the initial selections. The goal is to quantify these relationships across multiple genres to reveal patterns in algorithmic behavior.

The study adopts a structured experimental design that integrates user demographics with genre-specific seed selection. Two seed songs were chosen from five aggregated genres derived from global subgenre data. Using standardized music streaming APIs and feature extraction tools, playlists were generated for 90 accounts created under controlled demographic conditions. Quantitative analysis was conducted through the following methods: UpSet visualization to examine cross-genre overlaps; development of a distance-based metric combining Euclidean and Hamming distances to quantify deviation from seed tracks; and linear regression modeling to analyze position-wise trends in recommendation behavior within playlists.

The analysis reveals that certain genres demonstrate strong retention of their seed characteristics throughout playlist progression, whereas others exhibit stable yet more diverse recommendation behavior without a clear directional drift. High correlations among key audio features, such as *energy*, *loudness*, and *acousticness*, validated the use of dimensionality reduction methods. Overall, the findings indicate that Spotify's recommendation algorithm tends to reinforce genre identity for acoustically distinct styles while promoting broader stylistic blending among contemporary categories, achieving a balance between consistency and exploratory diversity.

The second section introduces the experimental design, including participant demographics, account setup, and song selection. The third and fourth sections describe the data collection procedures together with variable definitions and preprocessing steps. The fifth section outlines the set visualization tool, dimensionality-reduction methods, distance-based metrics, linear modeling, and t-test formulations. Finally, the sixth section presents detailed statistical analyses encompassing genre overlap visualization, PCA for feature reduction, global drift patterns, and playlist-progression trend modeling.

2 Design of Experiment

The experimental design integrates demographic factors with musical seed selection to examine their influence on song recommendations. Three factors are considered in this study: *Gender*, with two levels (male and female); *Age*, with three levels (20, 40, and 60 years); and *Source Genre*, comprising five distinct genres from which the two seed songs were selected. The five source genres were derived by aggregating fifteen main genres (encompassing approximately 5,680 subgenres) listed on everynoise.com. This aggregation was performed based on acoustic similarity, cultural consumption patterns, and proportional representation across genre categories (see Table 3). Each aggregated genre thus represents a broad yet coherent musical category suitable for controlled experimentation.

The dual seed songs listed in Table 4 represent these five source genres used in the experiment. For each genre, *Song 1* corresponds to a more popular and recently released track (post-2020), while *Song 2* represents an older and less popular track (pre-2010). This pairing was designed to balance potential effects of song popularity and release period so that resulting recommendations primarily reflect genre characteristics rather than temporal or popularity biases.

Combining these factors yields $2 \times 3 \times 5 = 30$ unique factor-level conditions. For each condition, three user accounts were created, resulting in a total of ninety accounts across all experimental settings. This factorial design enables systematic evaluation of the main effects (i.e., gender, age group, and genre) as well as potential interactions among them. The resulting dataset provides a robust foundation for analyzing how demographic characteristics and musical preferences influence recommendation patterns across different genres.

3 Data Collection

3.1 Account Creation and Data Extraction

A total of 25 participants created 90 Spotify accounts in accordance with the $2 \times 3 \times 5$ factorial design, corresponding to three accounts per factor combination. Each account was configured with an assigned demographic profile specifying gender and age group, represented by birth dates 01.01.2005 (20 years), 01.01.1985 (40 years), and 01.01.1965 (60 years). A playlist was initialized for each account using two seed songs selected from the genre assigned to that user (see Table 4).

For every account, Spotify's recommendation algorithm generated a sequence of 48 tracks based on the dual seed songs. Participants listened to these recommended tracks to establish personalized listening histories. Together with the two seed songs, each playlist contained a total of 50 tracks. Playlist histories were collected using a custom application registered on the *Spotify for Developers* platform [Spotify, 2026], employing secure OAuth authentication through the *Spotipy* library [spotipy-dev, 2023] to ensure controlled and authorized data access. For each account, complete playlist histories consisting of fifty tracks in listening order were retrieved along with user profile metadata—including demo-

graphic attributes and timestamps corresponding to listening activity for each song—and detailed metadata for both seed songs such as artist name, popularity score, and release year.

Track-level audio features were obtained via the *ReccoBeats API* [ReccoBeats Team, 2024] for all recommended songs. The resulting dataset includes both continuous variables and categorical variables, providing a rich representation of musical content across users and genres.

This comprehensive data collection procedure ensured that both user-level metadata and track-level audio descriptors were captured consistently across all experimental conditions. Although minor inconsistencies occurred for a few accounts due to duplicate or incomplete playlists, the overall dataset remained sufficiently complete and reliable for subsequent analysis.

3.2 Variable Description

The final dataset obtained from the data extraction procedure contains both user-level metadata and track-level attributes. Each observation corresponds to one song within a playlist generated for a specific user account. Together, these variables provide a comprehensive representation of user behavior, demographic context, and musical characteristics across all playlists. The dataset comprises a total of 21 columns encompassing playlist information, user demographics, and detailed audio features.

The nine playlist-related variables and demographic attributes are described as follows:

- *position*: Sequential index of the track within each playlist (range: 0–49).
- *user*: Unique identifier assigned to each Spotify account (90 users)
- *first_genre*: Numeric identifier (1–5) representing the source genre from which seed songs were selected
- *genre_label*: Corresponding textual label for each source genre
- *track_name*, *track_id*: Metadata describing each recommended track
- *duration_ms*: Duration of the track in milliseconds
- *gender*, *age*: Demographic attributes associated with the user profile

The ten continuous and two categorical audio features were extracted via the *ReccoBeats* [ReccoBeats Team, 2024] and the official *Spotify Web API* [Spotify, 2026]. These variables characterize each track across perceptual, rhythmic, harmonic, and production dimensions.

Continuous Features ($[0, 1]$ unless otherwise specified):

- *Popularity*: Spotify's algorithmic popularity score; higher values indicate greater listening frequency.
- *Danceability*: Measure of rhythmic suitability for dancing; higher values correspond to stable tempo and strong beat consistency.
- *Energy*: Perceptual intensity; higher values denote louder volume, faster tempo, and greater dynamic activity.
- *Valence*: Musical positivity or mood indicator; lower values represent sad or tense tracks, while higher values indicate happier or euphoric tracks.
- *Tempo*: Estimated beats per minute (BPM), ranging from $[0, \infty)$.
- *Loudness*: Overall sound level measured in decibels ($[-60, 0]$ dB).
- *Speechiness*: Degree of spoken-word content; lower values indicate purely musical tracks, whereas higher ones reflect rap-like or speech-heavy content.
- *Instrumentalness*: Probability that a track is instrumental; lower scores suggest prominent vocals, while higher scores imply minimal vocal presence.
- *Liveness*: Confidence measure for live audience presence; lower values typically correspond to studio recordings and higher ones to live performances.
- *Acousticness*: Confidence measure of acoustic instrumentation; low scores denote electronically produced music, whereas high scores indicate acoustic tracks.

Categorical Features:

- *Key* : Encoded musical key represented as integers $0, 1, ..., 11$ (0 = C , 1 = C *sharp* / D $flat$, ..., 11 = B).
- *Mode* : Musical modality where 0 denotes minor mode and 1 denotes major mode.

Each feature contributes to understanding how recommendation algorithms interpret musical properties when generating playlists based on dual seed songs. Collectively these attributes form a complete representation of each track's musical identity and serve as input for subsequent similarity assessments and statistical analyses conducted throughout this study.

4 Data Quality

4.1 Duplicate Removal

Prior to analysis, the dataset was cleaned to remove duplicate playlist histories across user accounts. Duplicates occurred because the uncleaned `.cache` files during Spotify API extraction caused identical playlist files to persist across multiple account creations, despite generating fresh file listening history. This deduplication step is essential to prevent bias from repeated identical recommendation sequences, ensuring each playlist history independently reflects distinct algorithmic behavior.

During inspection, several instances of duplication were detected. Two participants each had four user accounts with identical playlist histories; three duplicate accounts per participant were removed. Another two participants each had two user accounts with duplicated playlists, leading to the removal of one account per participant. Additionally, two pairs of distinct users exhibited identical playlist histories; one account from each pair was excluded. Finally, one user displayed an identical playlist history under two distinct category combinations, and this account was also removed from the dataset.

After removing these duplicates, the cleaned dataset consisted of 79 unique users with verified distinct playlist histories. The distribution across seed genres was as follows: Genre 1 (14 users), Genre 2 (17 users), Genre 3 (14 users), Genre 4 (16 users), and Genre 5 (18 users). This cleaning process ensures that subsequent analyses reflect genuine algorithmic behavior rather than replication artifacts.

In total, the dataset contained 681 unique tracks recommended through the experimental setup. The consolidated data frame comprised 4,057 observations across 21 variables. A quality check revealed that 249 rows contained missing values, approximately 6.138% of the dataset, and 49 tracks had incomplete audio feature information retrieved from the APIs (7.195% missing). These missing entries were handled appropriately during preprocessing to maintain consistency and reliability in subsequent statistical analyses.

4.2 Missing Data Imputation

To ensure completeness and consistency of the dataset, missing values were treated through an imputation process based on genre-wise statistics. Since the marginal distributions of several continuous variables exhibited skewness within genres (See Figure 8), median imputation was chosen as a robust approach less sensitive to outliers compared to mean substitution.

Continuous audio features, including *popularity*, *acousticness*, *danceability*, *energy*, *instrumentalness*, *liveness*, *loudness*, *speechiness*, *tempo*, and *valence*, were standardized prior to analysis. Missing entries for these variables were replaced with their respective median values computed within each genre group. This ensured that the imputed values preserved the central tendency of each genre while minimizing distortion due to extreme observations.

For categorical attributes such as *key* and *mode*, missing values were imputed using the most frequent category (mode) observed within each corresponding genre. This approach

maintained internal consistency in musical characteristics across genres.

After completing both continuous and categorical imputations, all relevant columns, including user demographics, track metadata, and audio features, were consolidated into a final structured dataset sorted by *user* and *position*. The resulting data frame provided a clean and reliable foundation for subsequent statistical analyses and modeling.

5 Statistical Methods

5.1 UpSet Visualization for Set Analysis

The UpSet visualization technique provides a scalable method for analyzing set intersections and overlaps, effectively overcoming the limitations of traditional Venn diagrams, which become impractical beyond three or four sets due to combinatorial explosion. UpSet employs a matrix-based layout that supports the analysis of more than twenty sets simultaneously, enabling detailed exploration of intersection cardinalities, set operations, size hierarchies, and disproportionate overlaps [Lex et al., 2014].

In an UpSet plot (see Figure 1), rows represent individual sets accompanied by horizontal bar charts indicating their respective sizes. Columns correspond to unique non-empty intersection combinations among these sets. Filled cells within the matrix denote membership of each set in a given intersection, while vertical connecting lines highlight participating sets to visualize joint intersections. The column bars above the matrix display intersection cardinalities, allowing direct comparison of overlap sizes across different combinations.

Conceptually, UpSet decomposes k input sets into atomic intersection regions without requiring exhaustive computation of all 2^k possible combinations. The degree of an intersection is defined as the number of sets participating in that combination. Only non-empty intersections with degrees between a minimum threshold i (default: 1) and a maximum degree d (the highest observed) are displayed. Both thresholds can be interactively adjusted to focus on specific overlap patterns while avoiding visual clutter caused by excessive combinatorial complexity.

5.2 Principal Component Analysis (PCA)

Principal Component Analysis (PCA) is a statistical technique used to reduce the dimensionality of multivariate data while retaining as much variation as possible. Let $\mathbf{x}$ denote a vector of p random variables representing the features of interest. When p is large, direct interpretation of individual variances and covariances among these variables becomes complex. PCA addresses this challenge by transforming the original correlated variables into a smaller set of uncorrelated variables known as *principal components* [Jolliffe, 2002].

Each principal component is defined as a linear combination of the original variables:

$$z_k = \boldsymbol{\alpha}_k^\top \mathbf{x} = \sum_{j=1}^{p} \alpha_{kj} x_j, \tag{1}$$

Here, $\boldsymbol{\alpha}_k$ denotes a vector of coefficients (loadings) that determines how strongly each variable contributes to component k. The first principal component (z_1) captures the maximum possible variance within the dataset. Subsequent components ($z_2, z_3, \dots$) are derived sequentially such that each maximizes variance under the constraint of being uncorrelated with all preceding components.

In general, up to p principal components can be obtained; however, it is typically sufficient to retain only $m < p$ components that collectively explain most of the variability in $\mathbf{x}$. This dimensionality reduction facilitates interpretation by focusing on directions in feature space that capture the greatest amount of information while minimizing redundancy arising from correlations among variables.

Computation of Principal Components

The computation of principal components is based on the covariance structure of the data. Let $\mathbf{x}$ be a p-dimensional random vector with covariance matrix $\boldsymbol{\Sigma}$, where diagonal elements represent variances and off-diagonal elements represent covariances between variables. In practice, when $\boldsymbol{\Sigma}$ is unknown, it is estimated using the sample covariance matrix $\mathbf{S}$ derived from observed data.

The objective of PCA is to find linear transformations $z_k = \boldsymbol{\alpha}_k^\top \mathbf{x}$ such that each coefficient vector $\boldsymbol{\alpha}_k$ maximizes the variance of z_k, subject to orthogonality constraints ensuring that all components are uncorrelated (equation 2). Formally,

$$\max_{\boldsymbol{\alpha}_k} \ \mathrm{Var}(z_k) = \boldsymbol{\alpha}_k^\top \boldsymbol{\Sigma}\, \boldsymbol{\alpha}_k, \quad \text{s.t. } \boldsymbol{\alpha}_k^\top \boldsymbol{\alpha}_k = 1,\ \mathrm{Cov}(z_i, z_j) = 0 \text{ for } i < j. \tag{2}$$

Applying the method of Lagrange multipliers to incorporate these constraints leads to the characteristic eigenvalue equation:

$$(\boldsymbol{\Sigma} - \lambda_k \mathbf{I})\, \boldsymbol{\alpha}_k = 0, \tag{3}$$

where λ_k denotes an eigenvalue of $\boldsymbol{\Sigma}$ and $\boldsymbol{\alpha}_k$ its corresponding eigenvector. Each eigenvector defines one principal component direction, while its associated eigenvalue represents the amount of variance explained by that component: $\mathrm{Var}(z_k) = \lambda_k$.

The first principal component corresponds to the largest eigenvalue and captures the greatest possible variance in the data. Subsequent components correspond to successively smaller eigenvalues and are constructed such that they remain orthogonal, and therefore uncorrelated with all previously derived components.

Thus, for $k = 1, 2, \dots, p$, each principal component can be expressed as $z_k = \boldsymbol{\alpha}_k^\top \mathbf{x}$, where $\boldsymbol{\alpha}_k$ is an eigenvector associated with the k^{th} largest eigenvalue λ_k. In this formulation, $\boldsymbol{\alpha}_k$ serves as the loading or coefficient vector for component k, while z_k represents its corresponding score, the transformed variable capturing one dimension along which variance in the dataset is maximized.

5.3 Metric Spaces

Let X be a nonempty set. A function $d : X \times X \to [0, \infty)$ is called a *metric* on X if it satisfies the following axioms for all $x, y, z \in X$ [Magnus, 2022]:

1. **Identity:** $d(x, y) = 0$ if and only if $x = y$.
2. **Symmetry:** $d(x, y) = d(y, x)$.
3. **Triangle Inequality:** $d(x, z) \leq d(x, y) + d(y, z)$.

A pair (X, d) consisting of a set X together with a metric d is called a *metric space*. The metric defines the geometric or topological structure of the space by determining how distances are measured between points.

5.3.1 Euclidean Metric

The Euclidean space $\mathbb{R}^p$ consists of all p-dimensional real vectors $\mathbf{x} = (x_1, x_2, \ldots, x_p)$. For any two points $\mathbf{x}$, $\mathbf{y} \in \mathbb{R}^p$, the Euclidean distance is defined as,

$$d(\mathbf{x},\, \mathbf{y}) = \|\mathbf{x} - \mathbf{y}\|_2 = \sqrt{\sum_{i=1}^{p} (x_i - y_i)^2}. \tag{4}$$

This distance function satisfies all axioms of a metric, thereby making $(\mathbb{R}^p, d)$ a valid metric space under the Euclidean metric.

5.3.2 Hamming Distance

The Hamming distance quantifies dissimilarity between two sequences of equal length by counting the number of positions at which corresponding symbols differ [Weger, 2020]. Formally, for $\mathbf{x}, \mathbf{y} \in \mathbb{F}_q^m$, where $\mathbb{F}_q$ denotes a finite field with q distinct elements and $\mathbb{F}_q^m$ represents the set of all m-dimensional vectors over this field, the *Hamming distance* is defined as:

$$d_H(\mathbf{x}, \mathbf{y}) = |\{\, i \in \{1, 2, \ldots, m\} \mid x_i \neq y_i \,\}|. \tag{5}$$

Here, $d_H(\mathbf{x}, \mathbf{y})$ corresponds to the number of coordinates in which $\mathbf{x}$ and $\mathbf{y}$ differ. This measure satisfies all axioms required for a metric space, thus forming $(\mathbb{F}_q^m, d_H)$.

Originally developed for error detection and correction in coding theory, the Hamming distance is also useful for measuring categorical feature divergence in one-hot or binary encoded representations.

5.3.3 Drift Metric: Product Space Formulation

From the perspective of metric space theory, the continous audio feature space is a continuous metric space equipped with the Euclidean metric, whereas the categorical features form a discrete metric space defined by the Hamming distance. Together, these spaces constitute a product metric space whose overall distance function is expressed as the sum of their respective metrics (see Theorem 1).

To quantify how each recommended song diverges from its genre-specific seed songs, a distance-based measure referred to as *drift* is developed [Schedl et al., 2018]. The drift metric captures dissimilarity between a track and its corresponding dual seed pair using both continuous and categorical audio features.

Each playlist begins with two seed tracks positioned at indices 0 and 1, both selected from the same source genre. For continuous features, the seed vector is computed as the mean of these two seed songs:

$$\text{seed}_{\text{cont}} = \text{mean}\big(\text{songs}^0_{\text{cont}}, \text{songs}^1_{\text{cont}}\big). \tag{6}$$

while for categorical attributes, the representative seed values are determined using their most frequent category:

$$\text{seed}_{\text{cat}} = \text{mode}\big(\text{songs}^0_{\text{cat}}, \text{songs}^1_{\text{cat}}\big). \tag{7}$$

For each recommended track from position 2 to 49 in the playlist, drift is computed separately for continuous and categorical features. Continuous drift is measured using Euclidean distance between the feature vectors of the song and its corresponding seed:

$$\text{drift}_{\text{cont}} = \|\text{seed}_{\text{cont}} - \text{song}_{\text{cont}}\|_2, \tag{8}$$

For categorical features, after one-hot encoding, the drift is measured using the Hamming distance, which quantifies the proportion of differing bits between the encoded representations of two songs. Formally,

$$\text{drift}_{\text{cat}} = \frac{\sum_{i=1}^{m} \mathbb{I}(x_i \neq y_i)}{m}, \tag{9}$$

where x_i and y_i denote the binary values (bits) at position i in the one-hot encoded vectors of the seed and recommended song, respectively; $\mathbb{I}(\cdot)$ is an indicator function that equals 1 when bits differ and 0 otherwise; and m represents the total number of binary positions across all categorical features.

The overall drift value combines both components to provide a unified measure of deviation. By Theorem 1, this formulation defines a valid metric over the product space that jointly represents continuous and categorical attributes:

$$\text{drift} = \text{drift}_{\text{cont}} + \text{drift}_{\text{cat}}. \tag{10}$$

This composite metric enables direct comparison of recommendation behavior across genres by integrating acoustic similarity (continuous feature space) with musical structure

characteristics (categorical feature space). A higher drift value indicates greater departure from the original genre context, reflecting broader algorithmic exploration or cross-genre blending within Spotify's recommendations.

5.4 Classical Linear Regression Model

The Classical Linear Regression Model (CLRM) provides a fundamental framework for analyzing the relationship between a continuous response variable and one or more explanatory variables. Let the observed data be represented by $(y_i, x_{i1}, x_{i2}, \ldots, x_{ip})$, where $i = 1, 2, \ldots, n$. Here, y_i denotes the dependent variable for observation i, and $x_{i1}, \ldots, x_{ip}$ represent continuous or appropriately coded categorical regressors. The model assumes a linear relationship of the form [Fahrmeir et al., 2013]:

$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \varepsilon_i, \quad i = 1, 2, \ldots, n, \tag{11}$$

Here, β_0 is the intercept term; $\beta_1, \ldots, \beta_p$ are regression coefficients quantifying the influence of each covariate on y; and ε_i denotes random error.

The error terms $\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n$ are assumed to be independently and identically distributed (i.i.d.) with $E[\varepsilon_i] = 0$ and $Var(\varepsilon_i) = \sigma^2$. These assumptions imply that errors have zero mean and constant variance across observations. The estimated linear function based on sample data is expressed as:

$$\hat{f}(x_1, x_2, \ldots, x_p) = \hat{\beta}_0 + \hat{\beta}_1 x_1 + \hat{\beta}_2 x_2 + \cdots + \hat{\beta}_p x_p. \tag{12}$$

This fitted function serves as an estimator for the conditional expectation of y given the regressors:

$$E[y \mid x_1, x_2, \ldots, x_p] = f(x_1, x_2, \ldots, x_p), \tag{13}$$

and can be used to predict new values of y, denoted by $\hat{y}$.

In summary, under these classical assumptions—linearity in parameters, independence of errors, homoscedasticity (constant variance), and zero mean—the linear regression model provides an unbiased and efficient method for estimating relationships between variables.

5.5 t-Test for Individual Coefficients

In the classical linear regression model, the statistical significance of an individual regression coefficient β_j is evaluated by testing the null hypothesis [Fahrmeir et al., 2013]:

$$H_0 : \beta_j = 0 \quad \text{versus} \quad H_1 : \beta_j \neq 0, \quad j = 1, 2, \ldots, p \tag{14}$$

The corresponding test statistic is the *t-statistic*:

$$t_j = \frac{\hat{\beta}_j}{\text{se}(\hat{\beta}_j)}, \tag{15}$$

where $\hat{\beta}_j$ denotes the least squares estimate of β_j, and $\text{se}(\hat{\beta}_j) = \sqrt{\text{Var}(\hat{\beta}_j)}$ represents its estimated standard error.

Under the null hypothesis H_0 and the usual linear model assumptions, t_j follows a t-distribution with $(n-p)$ degrees of freedom, where n is the sample size and p the number of regression parameters (including the intercept). For a two-sided test at significance level α, we reject H_0 if

$$|t_j| > t_{1-\alpha/2,\, n-p}, \tag{16}$$

where $t_{1-\alpha/2,\, n-p}$ is the $(1-\alpha/2)$-quantile of the t-distribution with $(n-p)$ degrees of freedom.

Resources

The analysis in this project was implemented using Python 3.12.10 [Python Software Foundation, 2025], employing standard scientific computing and machine learning libraries. Specifically, `pandas` and `numpy` were used for data handling and numerical computations; `matplotlib.pyplot`, `seaborn`, and `upsetplot` for data visualization; and submodules from `scikit-learn`and `scipy.stats` for dimensionality reduction and regression analysis. Additionally, the large language model Perplexity AI [Perplexity AI, 2026] was utilized to refine the written text, enhance grammatical accuracy, and improve the clarity of explanations and presentation of results.

6 Statistical Analysis

6.1 Genre Overlap Analysis

The UpSet visualization was employed to analyze the overlap and exclusivity of recommended tracks across source genres. A total of 681 unique tracks were examined, distributed among the five primary genres defined in the experimental setup. Figure 1 illustrates intersection patterns among these genres.

Several notable intersections (see Figure 1) reveal shared musical characteristics between genres. Specifically, fifteen songs overlapped between **Rock & Heavy** and **Urban & Contemporary**; ten songs appeared jointly in **Pop & Mainstream** and **Rock & Heavy**; eight songs overlapped between **Urban & Contemporary** and **Pop & Mainstream**; and six songs were common across **Pop & Mainstream**, **Urban & Contemporary**, and **Electronic & Beat-Based** playlists.

As summarized in Table 1, the dataset exhibits an average purity ratio of approximately 78.98%, indicating that about 21.02% of all recommended songs appear in playlists originating from multiple genres,a phenomenon referred to as genre contamination. The highest purity was observed for the genre category **Roots, Jazz & Classical Traditions**, where all 167 tracks were exclusively associated with this genre (100.0% purity). In contrast, other genres demonstrated varying degrees of overlap: **Electronic & Beat-Based** showed relatively high isolation (83.4% purity), whereas **Rock & Heavy**,

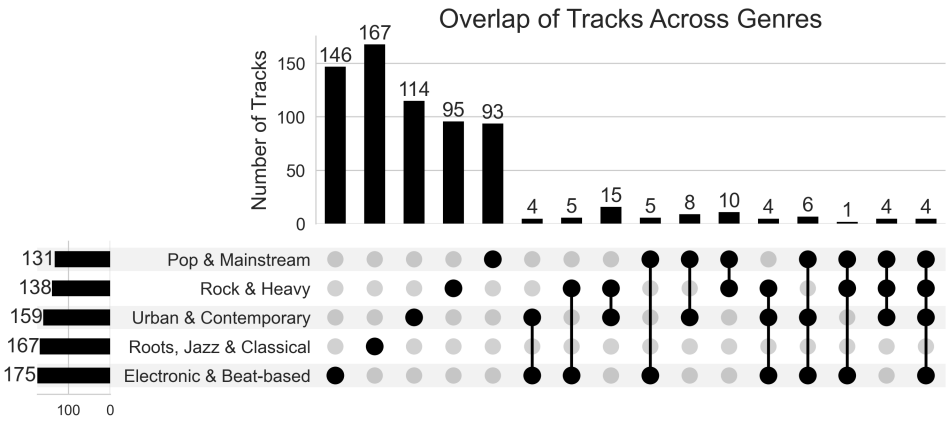


Figure 1: UpSet plot showing intersection patterns among source genres.

Pop & Mainstream, and **Urban & Contemporary** exhibited stronger cross-genre interactions with purities ranging between 68.8% and 71.7%.

Table 1: Genre Purity Analysis: Exclusive vs. Total Recommended Tracks Across Five Source Genres

ID	Genre	Total Tracks	Exclusive Tracks	Purity (%)	Contamination (%)
1	Pop & Mainstream	131	93	71.0	29.0
2	Urban & Contemporary	159	114	71.7	28.3
3	Electronic & Beat-Based	175	146	83.4	16.6
4	Rock & Heavy	138	95	68.8	31.2
5	Roots/Jazz/Classical	167	167	100.0	0.0
	Total	681	615	78.98	21.02

To further visualize how these genre relationships manifest in terms of acoustic similarity, a Principal Component Analysis (PCA) was performed using the ten continuous audio features extracted for all unique tracks. The first two principal components were used to project the data into a lower-dimensional space, allowing an interpretable view of how songs cluster based on shared musical characteristics. Figure 7 illustrates this two-dimensional representation. This low-dimensional visualization supports observations from the UpSet analysis genres with overlapping recommendations also display proximity in feature space.

These variations suggest that certain genres share acoustic and stylistic characteristics that result in overlapping recommendations. In particular, Spotify's recommendation algorithm appears to blend stylistically adjacent genres, especially those exhibiting similar rhythmic intensity, vocal texture, or production techniques, while maintaining clear separation for acoustically distinct categories such as classical or jazz traditions.

6.2 Using PCA to Address Multicollinearity in Audio Features

To examine interdependencies among the ten continuous audio features, a correlation matrix was computed and visualized using a heatmap (see Figure 2). The analysis revealed several strong correlation ($|r| > 0.7$) that highlight how certain acoustic properties co-vary across tracks.

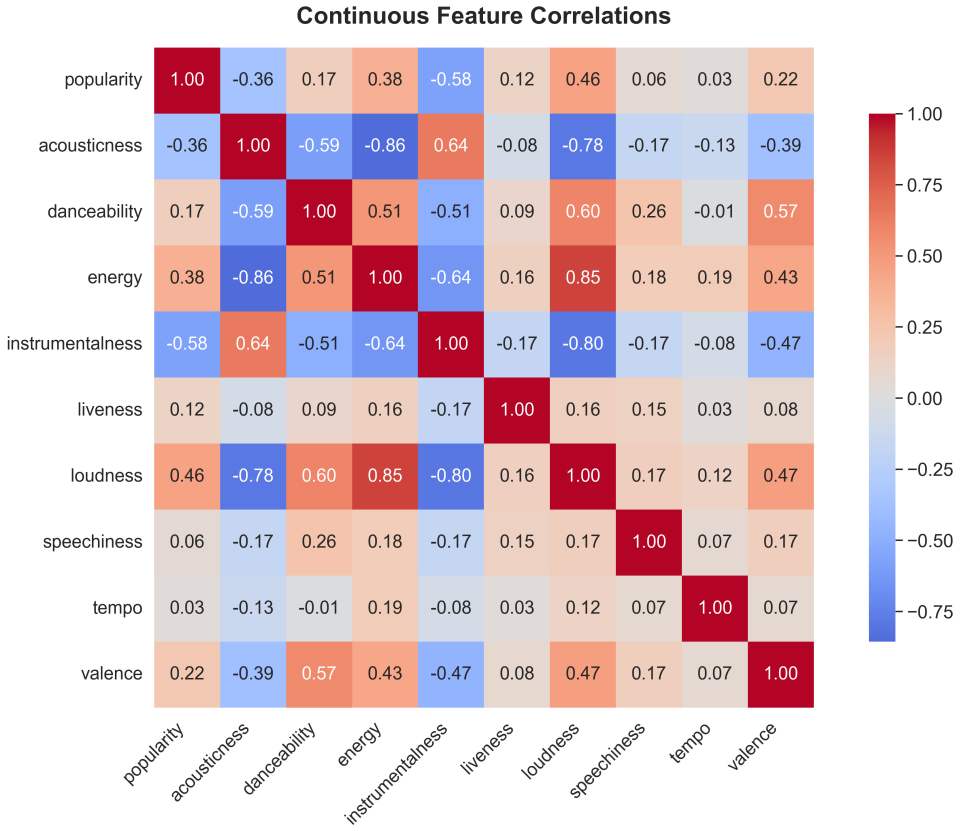


Figure 2: Correlation heatmap of ten continuous audio features, illustrating strong relationships such as *acousticness* versus *energy* ($r = -0.856$), *energy* versus *loudness* ($r = 0.853$), and *instrumentalness* versus *loudness* ($r = -0.798$).

A pronounced negative correlation was observed between *acousticness* and both *energy* ($r = -0.856$) and *loudness* ($r = -0.784$), indicating that tracks with higher acoustic instrumentation tend to exhibit lower volume levels and reduced dynamic intensity. Conversely, *energy* and *loudness* were strongly positively correlated ($r = 0.853$), reflecting their shared contribution to perceived track intensity.

Moderate correlations were also found between *instrumentalness* and other features, for example, its negative association with *loudness* ($r = -0.798$) and positive relationship with *acousticness* ($r = 0.638$). These patterns suggest that instrumental or acoustically rich compositions are typically quieter yet more sonically pure compared to highly energetic or electronically produced tracks.

Overall, the correlation structure aligns well with intuitive musical characteristics:

energetic genres tend to exhibit louder dynamics, whereas acoustic or instrumental pieces emphasize tonal clarity over intensity.

To address potential multicollinearity among these correlated features, Principal Component Analysis (PCA) was applied to identify orthogonal dimensions capturing shared variance within the dataset. The analysis yielded six principal components that collectively retained approximately 90.7% of the total variance across the ten audio features.

This dimensionality reduction demonstrates that much of the information contained in the original variables can be represented by a smaller set of independent components, effectively mitigating redundancy caused by strong correlations, particularly among attributes such as *energy*, *loudness*, and *acousticness*. By transforming correlated variables into orthogonal principal axes, PCA establishes a more stable foundation for subsequent modeling and comparative analyses of genre-specific acoustic patterns.

6.3 Global Drift Patterns by Genre

To evaluate overall recommendation behavior across genres, drift values from all 79 users were aggregated by genre to compute summary statistics, specifically the mean and standard deviation of drift for each genre, while disregarding individual playlist positions. Table 2 presents these aggregate results, while the box plot in Figure 3 visualizes the distribution of drift values within each genre. Figure 3 shows that each genre exhibits distinct patterns of variability in drift values, reflecting differences in how Spotify's recommendation algorithm explores musical feature space relative to seed tracks.

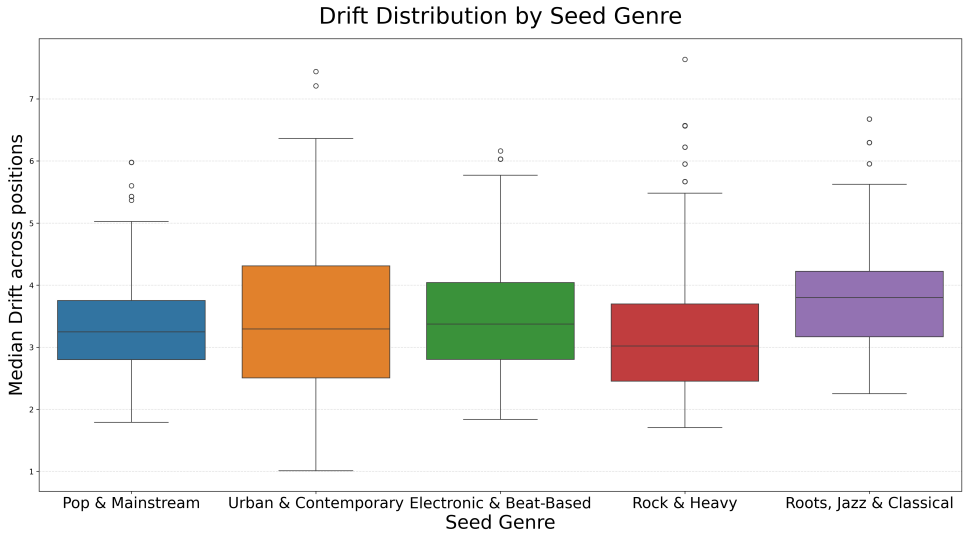


Figure 3: Box plot of drift distribution by genre.

The lowest average drift was observed for **Rock & Heavy**, indicating that recommended songs remained closest to their seed tracks despite moderate variation—suggesting high stability within this genre's recommendations. Similarly, **Pop & Mainstream** displayed consistent behavior with relatively low variance ($\sigma = 0.75$), implying homogeneous

listener preferences where similar tracks appeal broadly across users ("what works for one pop fan works for all").

In contrast, the highest variability occurred within the **Urban & Contemporary** genre ($\sigma = 1.285$), highlighting diverse listening patterns and broader exploration among users in this category ("listeners are the most diverse"). Genres such as **Electronic & Beat-Based** and **Roots, Jazz & Classical Traditions** demonstrated moderately high mean drift values but stable dispersion, suggesting steady exploration away from seed tracks while maintaining stylistic coherence ("consistently discovering new sounds at a moderate pace").

Overall, these results indicate that genres differ not only in their average degree of deviation from seed songs but also in how consistently recommendations stay within or venture beyond established acoustic boundaries.

6.4 Genre-Specific Drift Trajectories by Position

To investigate how recommendation behavior evolves across playlist positions, drift values were analyzed as a function of track order within each playlist. For every genre, approximately 14 to 18 users contributed data points per position, enabling computation of the mean and standard deviation of drift at each index. These mean drift values were treated as time-series observations representing the average deviation from seed songs over successive recommendation positions.

The average drift value, $\overline{\text{drift}}_i^{(g)}$, represents the mean drift across all users belonging to a particular genre g among the five defined genres. For each playlist position i, individual user drift values were first averaged within the corresponding genre group and then aggregated across all users at that position. This two-level averaging ensures that $\overline{\text{drift}}_i^{(g)}$ captures genre-specific trends while minimizing variability arising from individual listening behavior.

By plotting these mean drift trajectories for all genres, distinct patterns emerged in how recommendations diverge or remain close to their respective seeds. To formally test whether a systematic trend exists between playlist position and drift magnitude, a simple linear regression model was fitted for each genre:

$$\overline{\text{drift}}_i^{(g)} = \beta_0 + \beta_1 \, position_i + \varepsilon_i, \quad i = 3, 4, \ldots, 49, \quad g = 1, 2, 3, 4, 5, \tag{17}$$

where $\overline{\text{drift}}_i^{(g)}$ denotes the average drift at playlist position i for genre g, β_0 is the intercept term, β_1 represents the slope capturing change in drift with respect to position, and ε_i is a random error term assumed to be normally distributed.

The corresponding hypothesis test evaluates whether this slope differs significantly from zero:

$$H_0 : \beta_1 = 0 \quad \text{versus} \quad H_1 : \beta_1 \neq 0, \tag{18}$$

Under H_0, no systematic trend in drift is present across positions (constant average deviation), whereas H_1 indicates a directional change in drift along the playlist sequence.

The test was conducted at a significance level of $\alpha = 0.05$, with Bonferroni correction applied for multiple genre comparisons ($\alpha/5 = 0.01$).

For the **Pop & Mainstream** genre (see Figure 4), the position-specific mean drift values remain nearly constant across playlist positions, showing minimal deviation around the fitted linear trend. The standard deviation bands are relatively narrow, indicating stable recommendation behavior throughout the playlist sequence.

The estimated slope coefficient from the linear model ($\beta_1 = 0.0003$) is statistically insignificant ($p = 0.873$), confirming that there is no measurable change in drift with increasing playlist position. This observation aligns visually with the near-horizontal regression line in Figure 4, suggesting that Spotify's recommendations for **Pop & Mainstream** users maintain consistent proximity to seed songs across all positions, reflecting steady algorithmic behavior and homogeneous listener preferences.

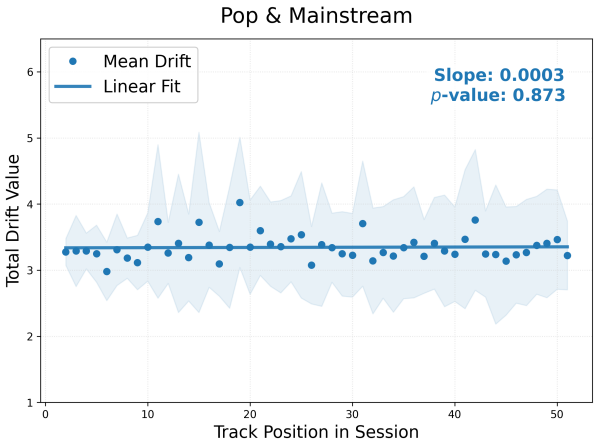


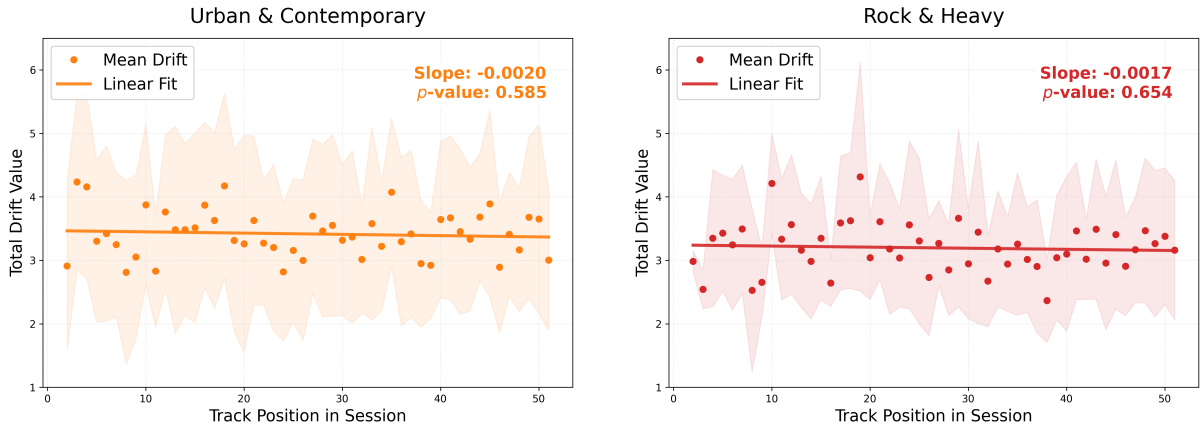
Figure 4: Mean drift and ±1 standard deviation bands for Pop & Mainstream genre across playlist positions 2–49, and linear fit indicates stable drift.

For the **Urban & Contemporary** and **Rock & Heavy** genres (see Figures 5a and 5b), the position-specific mean drift values exhibit considerable fluctuations around their respective linear fits, accompanied by broad standard deviation bands. This pronounced variability indicates greater diversity in recommendation behavior compared to **Pop & Mainstream** playlists.

Visually, both regression lines appear nearly horizontal. The estimated slope coefficients for these genres are small and statistically insignificant—$\beta_1 = -0.0020$ ($p = 0.5849$) for **Urban & Contemporary** and $\beta_1 = -0.0017$ ($p = 0.6532$) for **Rock & Heavy**, indicating no systematic trend in drift across playlist positions. This confirms that Spotify's recommendations within these genres maintain a relatively constant average distance from the seed songs throughout the playlist sequence.

Overall, these results imply stable yet slightly more variable recommendation patterns for Urban and Rock genres, reflecting broader stylistic diversity among tracks while preserving steady proximity to genre seeds.

For the **Electronic & Beat-Based** and **Roots, Jazz & Classical** genres (see Figures 6a and 6b), the position-specific mean drift values remain relatively stable around

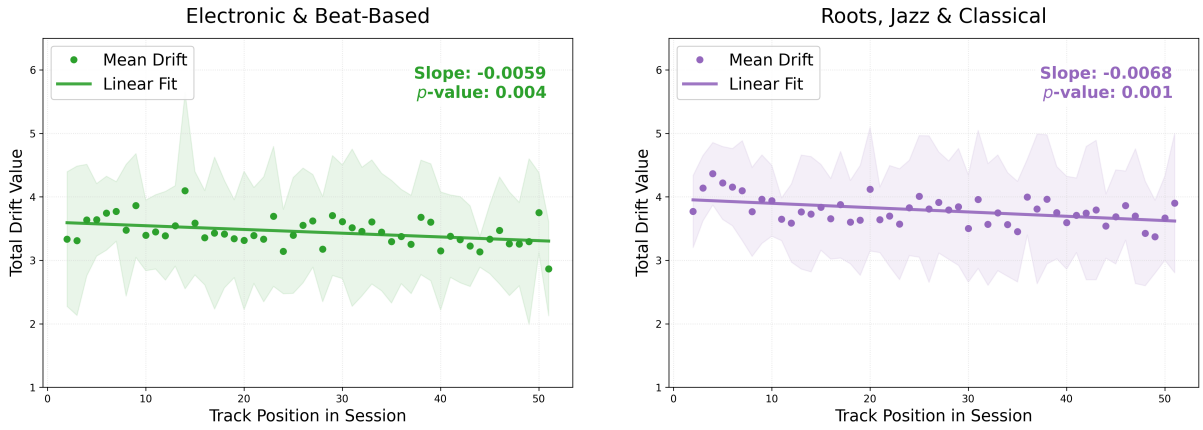


(a) Urban & Contemporary (b) Rock & Heavy

Figure 5: Mean drift and ±1 standard deviation bands across playlist positions 2–49, and stable linear fits.

their respective linear fits, similar to the **Pop & Mainstream** trend, but with slightly broader standard deviation bands, yet narrower than those observed for **Urban & Contemporary** or **Rock & Heavy** genres.

Visually, both regression lines exhibit a gentle negative slope. The estimated slope coefficients confirm this downward tendency: $\beta_1 = -0.0059$ ($p = 0.0020$) for **Electronic & Beat-Based** and $\beta_1 = -0.0068$ ($p = 0.0006$) for **Roots, Jazz & Classical Traditions**, both statistically significant at $p < 0.01$. These results indicate that as playlist positions progress, recommended tracks tend to move closer to their seed songs, reflecting strong retention within these genres.



(a) Electronic & Beat-Based (b) Roots, Jazz & Classical

Figure 6: Mean drift and ±1 standard deviation bands across playlist positions 2–49. Both exhibit statistically significant negative slopes, indicating drift convergence toward seeds.

Overall, the consistent negative drift trend suggests that Spotify's recommendation algorithm reinforces genre identity over time for electronic and classical styles, promoting convergence toward acoustic characteristics of the original seeds rather than encouraging cross-genre exploration.

7 Summary

The statistical analyses collectively reveal how Spotify's recommendation algorithm behaves across genres in relation to acoustic similarity and playlist evolution. Each analytical component contributes distinct insights into the structure and dynamics of the generated recommendations.

Genre Overlap Analysis showed that recommended tracks exhibit partial cross-genre contamination, with an average purity ratio of 78.98%. Genres such as **Roots, Jazz & Classical Traditions** maintained complete exclusivity (100% purity), whereas **Rock & Heavy**, **Pop & Mainstream**, and **Urban & Contemporary** displayed greater overlap (approximately 70% purity). These intersections indicate that Spotify's algorithm tends to blend stylistically adjacent genres, particularly those sharing rhythmic intensity or production characteristics—while preserving strong separation for acoustically distinct styles like classical or jazz.

The PCA analysis confirmed substantial correlations among continuous audio features (e.g., between *energy*, *loudness*, and *acousticness*), demonstrating internal redundancy within the dataset. Six principal components retained 90.7% of total variance, effectively reducing multicollinearity while maintaining interpretability. This transformation provided orthogonal dimensions representing independent musical attributes that serve as a robust foundation for subsequent drift modeling.

In the Global Drift Analysis, aggregated drift statistics revealed clear differences in overall recommendation behavior by genre. The lowest mean drift occurred for **Rock & Heavy** and **Pop & Mainstream**, suggesting high stability and consistent proximity to seed songs, indicative of homogeneous listener preferences within mainstream categories. Conversely, the highest variability was observed for **Urban & Contemporary**, reflecting broader stylistic diversity and exploratory recommendation patterns. Genres such as **Electronic & Beat-Based** and **Roots, Jazz & Classical Traditions** exhibited moderate mean drifts but stable dispersion, implying controlled exploration while retaining their distinctive sonic identity.

Finally, the Position-Wise Drift Trend Analysis examined how recommendations evolve throughout playlists using linear regression models fitted per genre. Significant negative slopes were found for both **Electronic & Beat-Based** ($\beta_1 = -0.0059$) and **Roots, Jazz & Classical Traditions** ($\beta_1 = -0.0068$), indicating convergence toward seed characteristics over time, a sign of strong genre retention within these categories. In contrast, non-significant slopes for other genres (Pop, Urban, Rock) suggest stable drift behavior without systematic directional change across playlist positions.

Overall, these results demonstrate that Spotify's recommendation system exhibits genre-dependent dynamics: it reinforces acoustic coherence within certain genres (notably

electronic and classical) while enabling broader stylistic mixing among contemporary categories such as pop or urban music.

Appendix

A Additional Figures

A.1 PCA Visualization of Seed Track Clustering

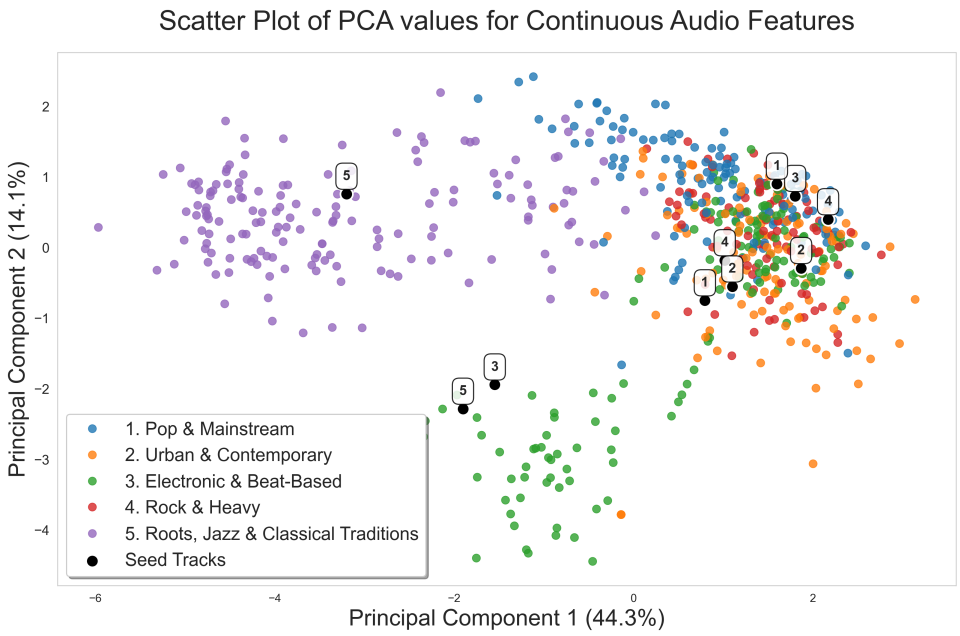


Figure 7: PCA projection of continuous audio features, showing genre clusters and highlighted seed tracks.

A.2 Marginal Distribution of Continuous Audio Features

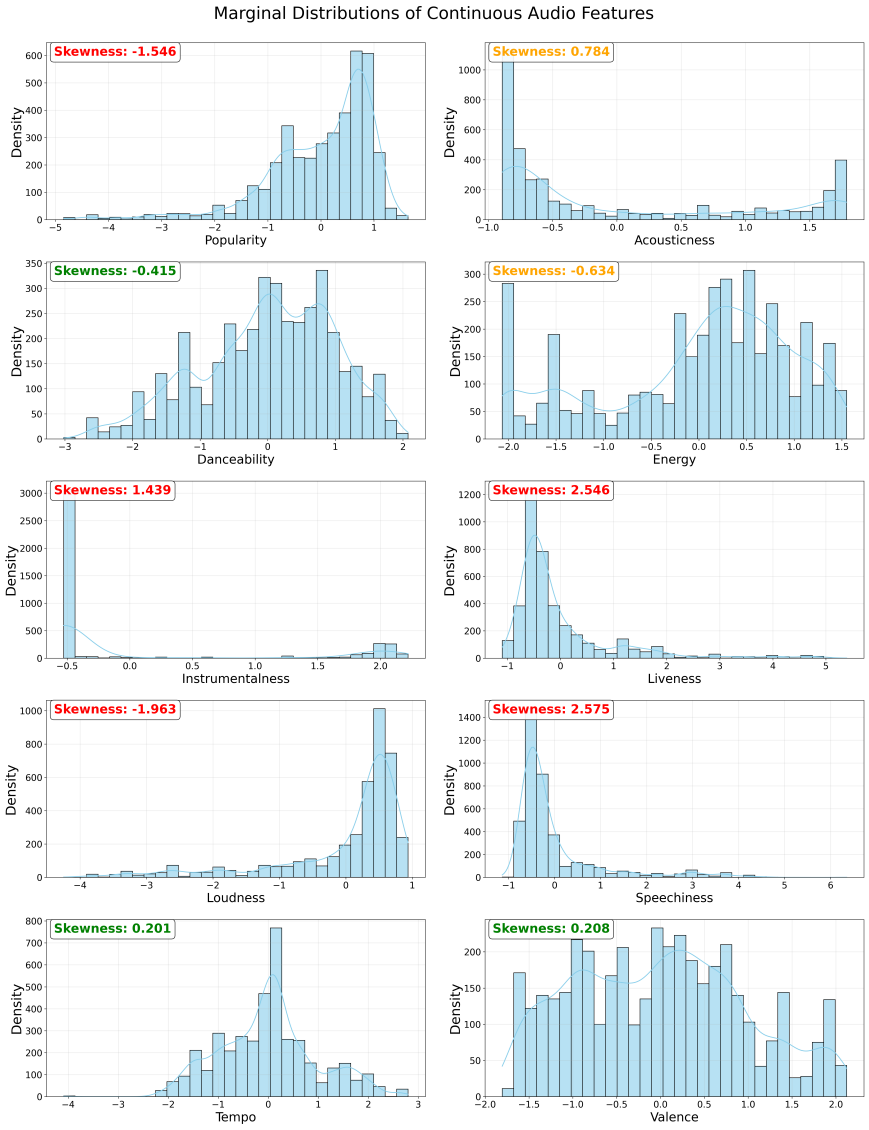


Figure 8: Marginal distributions of continuous Spotify audio features (*popularity*, *acousticness*, *danceability*, *energy*, *instrumentalness*, *liveness*, *loudness*, *speechiness*, *tempo*, and *valence*) across the music recommendation drift dataset. Distributions reveal skewness patterns and multimodal behavior.

B Mathematical Foundations

B.1 Product Metric Theorem

Theorem 1 (Product Metric). *Let* (X, d_X) *and* (Y, d_Y) *be metric spaces. Then*

$$d((x_1, y_1), (x_2, y_2)) = d_X(x_1, x_2) + d_Y(y_1, y_2)$$

defines a metric on $X \times Y$ [Magnus, 2022].

Proof. The three metric axioms are verified as follows:

1. **Identity**: $d_X(x_1, x_2) = 0 \iff x_1 = x_2$ and $d_Y(y_1, y_2) = 0 \iff y_1 = y_2$. Thus:

$$\begin{aligned} d((x_1, y_1), (x_2, y_2)) = 0 & \\ &\iff d_X(x_1, x_2) + d_Y(y_1, y_2) = 0 \\ &\iff x_1 = x_2 \ \wedge \ y_1 = y_2 \\ &\iff (x_1, y_1) = (x_2, y_2) \end{aligned}$$

2. **Symmetry**: By Symmetry of d_X and d_Y,

$$\begin{aligned} d((x_1, y_1), (x_2, y_2)) &= d_X(x_1, x_2) + d_Y(y_1, y_2) \\ &= d_X(x_2, x_1) + d_Y(y_2, y_1) \\ &= d((x_2, y_2), (x_1, y_1)) \end{aligned}$$

3. **Triangle inequality**: By Triangual inquality of d_X and d_Y,

$$\begin{aligned} d((x_1, y_1), (x_3, y_3)) &= d_X(x_1, x_3) + d_Y(y_1, y_3) \\ &\leq d_X(x_1, x_2) + d_X(x_2, x_3) + d_Y(y_1, y_2) + d_Y(y_2, y_3) \\ &= d((x_1, y_1), (x_2, y_2)) + d((x_2, y_2), (x_3, y_3)) \end{aligned}$$

Thus, d satisfies all metric axioms on $X \times Y$. □

C Additional Tables

C.1 Genre-Specific Drift Statistics

Table 2: Summary statistics of drift by genre.

ID	Genre	Mean Drift	Std. Dev.
1	Pop & Mainstream	3.345	0.750
2	Urban & Contemporary	3.418	1.285
3	Electronic & Beat-Based	3.452	0.864
4	Rock & Heavy	3.195	1.035
5	Roots, Jazz & Classical	3.785	0.805

C.2 Genre Distribution Summary

Table 3: Primary Genres with Subgenre Distribution

ID	Parent Genre	Subgenres
1	Pop & Mainstream	1,747
2	Urban & Contemporary	619
3	Electronic & Beat-Based	525
4	Rock & Heavy	1,342
5	Roots, Jazz & Classical Traditions	1,447
Total		**5,680**

C.3 Seed Song Selection Criteria

Table 4: Seed Songs selection list. Song 1: more popular, recent (release year $>$ 2020); Song 2: less popular, old (release year $<$ 2010)

ID	**Genre Label**	**Song 1**	**Song 2**
1	Pop & Mainstream	As It Was	Breathe on Me
2	Urban & Contemporary	Industry Baby (feat. Jack Harlow)	Kick Push
3	Electronic & Beat-Based	I'm Good (Blue)	Subzero
4	Rock & Heavy	I WANNA BE YOUR SLAVE	Tear Away
5	Roots, Jazz & Classical Traditions	I've Got You Under My Skin – 2024 Remastered	Piano Sonata No. 14

References

Jolliffe, I. T. [2002]. *Principal component analysis* 2nd ed. Springer. https://doi.org/10.1007/b98835

Fahrmeir, L., Kneib, T., Lang, S., & Marx, B. [2013]. *Regression: Models, methods and applications* 1st ed. Springer. https://doi.org/10.1007/978-3-642-34333-9

Lex, A., Gehlenborg, N., Strobelt, H., Vuillemot, R., & Pfister, H. [2014]. Upset: Visualization of intersecting sets. *IEEE Transactions on Visualization and Computer Graphics (InfoVis)*, *20*[12], 1983–1992. https://doi.org/10.1109/TVCG.2014.2346248

Schedl, M., Zamani, H., Chen, C.-W., Deldjoo, Y., & Elahi, M. [2018]. Current challenges and visions in music recommender systems research. *International Journal of Multimedia Information Retrieval*, *7*[2], 95–116. https://doi.org/10.1007/s13735-018-0154-2

Weger, V. [2020]. *Information set decoding in the lee metric and the local to global principle for densities* [Doctoral dissertation, Universität Zürich] [Promotionskommission: Prof. Dr. Joachim Rosenthal (Vorsitz), Prof. Dr. Andrew Kresch, Prof. Dr. Anna-Lena Horlemann]. https://home.cit.tum.de/~wvi/thesis.pdf

Magnus, R. [2022]. *Metric spaces: A companion to analysis*. Springer. https://doi.org/10.1007/978-3-030-94946-4

spotipy-dev. [2023]. Spotipy: A light weight python library for the spotify web api [Accessed: 2026-02-08].

ReccoBeats Team. [2024]. ReccoBeats API: Music recommendation and audio feature analysis [Accessed: 2026-02-08].

Python Software Foundation. [2025]. *Python 3.12 documentation* [Last updated on Oct 10, 2025]. Python Software Foundation. https://docs.python.org/3.12/

Perplexity AI. [2026]. Perplexity [Large language model AI assistant, accessed 2026-02-08].

Spotify. [2026]. Spotify web api documentation [Accessed: 2026-02-08].