The University of Sheffield

School of Information, Journalism and Communication

MSc Data Science

Audio Characteristic Analysis of Popular Songs (2000-2023): Clustering Billboard Songs and Examining the Relationship Between Song Duration and Energy

Course Code: IJC437
Course: Introduction to Data Science
Student Registration Number: 250208702
Word Count: 3236

1. Introduction, Aim and Research Questions

1.1 Introduction

Popular music can be viewed as both a cultural object and an analytics domain driven by creativity and technology. The availability of large datasets for digital music allows for the measurement of music attributes in a way that was only possible in a qualitative fashion in the past. The development of music retrieval and streaming allows for features such as energy, danceability, and acousticness to be extracted from popular music in order to analyse it through data science techniques.

There has been a predictable pattern in the features of audio files over time, due to the influence of production and listener preferences (Mauch et al., 2015). Research into the audio features of the popular streaming service, Spotify, indicates that there is not a single formula for success, but many audio profiles that could also perform well commercially, and therefore further research into the relationship between audio features and chart success could be explored (Perez-Verdejo et al., 2021).

The current research utilises cluster analysis and PCA to analyse high-dimensional musical data to discover hidden trends without using unreliable musical genres (Langensiepen et al., 2018; Cai et al., 2021). Cluster analysis can reveal musical trends, but their relation to success is uncertain.

Attention is also paid to the relationship between structural factors, such as duration and perceptual features such as energy and intensity. Some research implies that listening behaviour affects song structure, while others contend that energy is a product of production factors rather than duration (Thiesen et al., 2020). Researching the relationship between energy and duration could enhance the role of structural constraints on musical intensity.

With a Billboard Hot 100 dataset paired with Spotify audio features, this research seeks to answer these questions using exploratory data science. Through the integration of dimensionality reduction techniques, clustering analysis, and correlation analysis, this research seeks to find connections between audio features of popular songs.

1.2 Aim of the Study

The overall goal of this research is to uncover insights on Billboard Hot 100 songs' audio feature distributions via unsupervised learning techniques, as well as to find out connections between audio features, performance, and essential structural attributes of songs.

The objectives of this research are to:

- identify patterns of popular songs on Spotify based on audio features using principal component analysis (PCA) and k-means clustering, and

- examine whether the duration of songs is linked to the energy in the songs for which the full feature data is available.

The study uses an exploratory research design, which focuses more on looking for patterns in the data.

1.3 Research Questions

To achieve the above-stated objective, the research seeks to answer the following research questions:

RQ1: Can the Billboard Hot 100 songs be clustered based on Spotify audio features, and are the resulting clusters different regarding their performance on the charts?

This question assesses whether the unsupervised clustering method can produce distinct and interpretable results, categorising songs based on their audio characteristics, and whether these groupings align with variations in Billboard rankings.

RQ2: Is there a relationship between song duration and energy for complete Billboard Hot 100 songs with audio features on Spotify?

The question being asked here is whether there is any correlation between the actual length of the song structure and the energy level, which is measured by the energy feature provided by the music service, Spotify.

1.4 Scope and Contribution

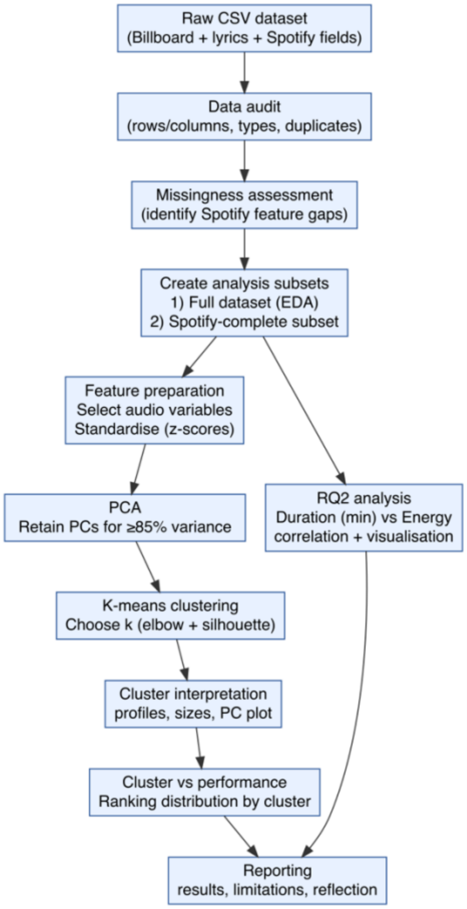
The work presented herein contributes to existing literature on music analytics by applying existing data science methods on a popular and well-studied popular music dataset. The work emphasises data completeness and limitations of methods by being very transparent and reproducible in its findings, which help explain the richness of audio profiles that exist within popular music and limitations of single features in explaining musical success.

2. Methodology

2.1 Data Source and Initial Processing

The data for this study consists of 3,397 observations of Billboard Hot 100 data from 2000 to 2023. Every observation corresponds to a song that charted on Billboard Hot 100, with accompanying data such as chart position, year, artist data, and lyrics. Moreover, some of these songs are connected to audio attributes on Spotify, such as danceability, energy, loudness, acousticness, valence, tempo, and duration.

All data processing and analysis were done using the R programming environment. After the data was loaded into R, the names of the variables were standardised, as well as the data types, to ensure the correct interpretation of the numerical audio features. Preliminary data exploration was carried out to validate the data ranges for the year variables as well as the ranking values.



2.2 Handling Missing Data and Analytical Scope

Results from the analysis of missing data revealed that audio features on the Spotify platform were only present for a small percentage of data on this particular dataset. In particular, full audio features on the Spotify platform were collected for 486 songs, which comprised about 14% of the full data set. Most of these data were from the beginning of the 2000s (2000-2005), with very few from later years. (See Appendix A.2)

To prevent the imposition of artificial similarity as well as possible biases in the analysis, the missing features in the audio files were not imputed. This means that the analysis of the features was done only in the Spotify-complete dataset. This is because imputing missing values in distance-based methods such as clustering analysis can introduce biases in the results (Wongoutong, 2024).

Accordingly, the analyses with audio characteristics, such as PCA, clustering, and the relationship between duration and energy, have been carried out only on the Spotify-complete subset. Descriptive analyses, which do not require audio characteristics, have been conducted on the full dataset whenever appropriate.

2.3 Feature Selection and Scaling

The audio features selected for evaluation include the common set of features considered in music computing, such as rhythm, intensity, timbre, and mood (danceability, energy, loudness, acousticness, valence, tempo, and duration), and have been shown in previous works to represent meaningful musical style and perceptual variations.

Before the use of the multivariate technique, all the audio feature variables were subjected to z-score normalisation. Feature scaling is a critical step in k-means clustering because k-means relies on the distance between the centres of the clusters. Variables with different scales may affect the results differently because of the distance calculation (Lala et al., 2023).

2.4 Principal Component Analysis

The scaled audio features were subjected to Principal Component Analysis (PCA), a technique that reduces the dimensionality of the features, as well as the multicollinearity that exists among the variables. PCA helps in the transformation of the correlated variables into orthogonal components that explain the highest possible variance in the variables. This technique is commonly applied in the analysis of music to extract hidden musical variables, as well as to ensure that the process of clustering is stable (Langensiepen et al., 2018; Cai et al., 2021).

The number of principal components considered for retention was based on a cumulative variance criterion. A component was retained until at least 85% of the total variance was accounted for, resulting in the selection of ten principal components. Eighty-five per cent is a threshold that is generally considered in exploratory multivariate data analyses. (See Appendix A.3)

2.5 K-means Clustering

K-Means clustering was employed on the retained principal components to cluster the songs based on similar audio features. The simplicity, efficiency, and popularity of the algorithm in clustering songs made it an ideal choice (Jondya & Iswanto, 2017; Marlia et al., 2024). To reduce the bias caused by initialisations, the algorithm was initialised several times.

The value of k for which the optimal clustering is obtained was found using the elbow method as well as the silhouette method. The elbow method compared the reduction in the sum of squares for the within-cluster values for different values of k, while the silhouette method compared the separation between the clusters for different values of k. Even as the silhouette method revealed the highest separation for k = 2, a value of k = 4 was used to improve interpretability and capture stylistic variability.

After clustering, I calculated the means of audio features for each cluster. The Billboard rankings of songs were also studied to investigate how audio-based clusters relate to rankings.

2.6 Analysis of Song Duration and Energy

To answer the second research question, the duration of the song in minutes (after converting from milliseconds) was analysed in correlation with the Spotify energy feature. The scatter diagram was used for the graphical representation, and the correlation coefficient was calculated using Pearson’s correlation formula.

This analysis was conducted only within the Spotify-complete subset to ensure a consistent level of available features. Because the subset only covers a short time span, the analysis was focused on finding a general association rather than a long-term trend. This helps avoid the risks of trying to read too much from a small dataset, while still being able to gain some information from it regarding the association between time and musical intensity. (See Appendix A.4)

2.7 Reproducibility and Output Management

To ensure the reproducibility of the results, the random number seed was set to the same number for all the clustering algorithms. The intermediate results, the summary tables, and the graphical representations were saved in the appropriate output directories. For the graphical representations, the images were saved in the format of PNG files with high resolutions, which could then be included in the final report. (See Appendix A.5)

3. Results and Discussion

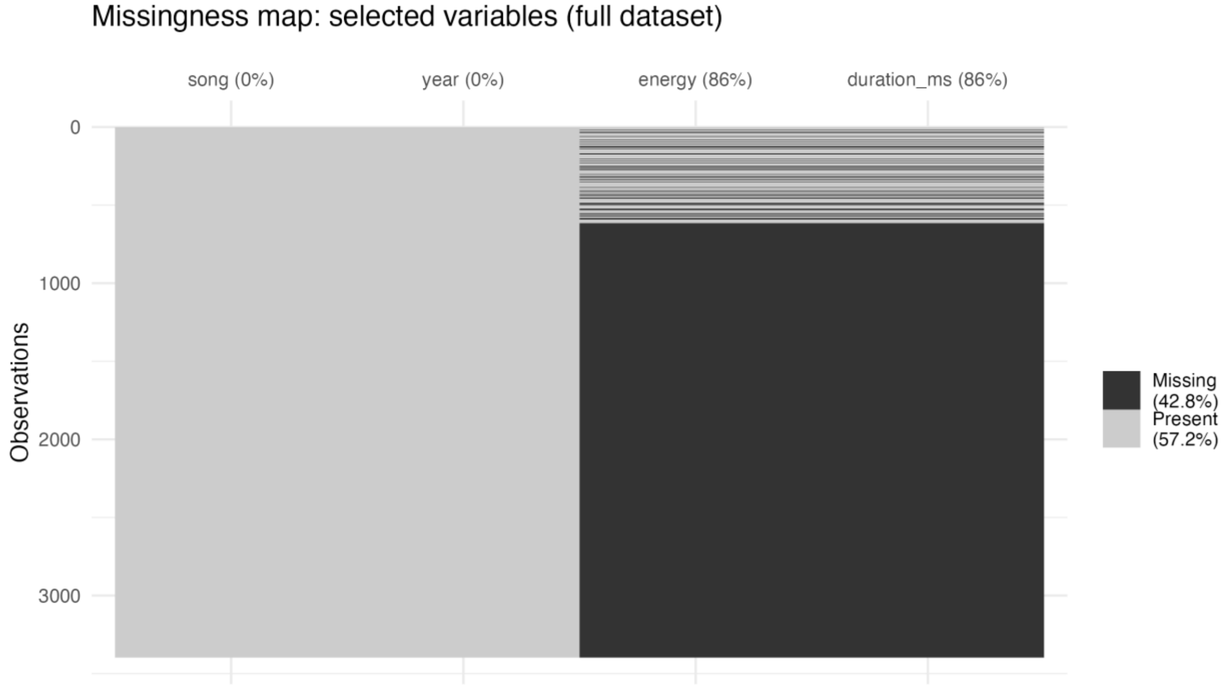
3.1 Overview of Analytical Scope

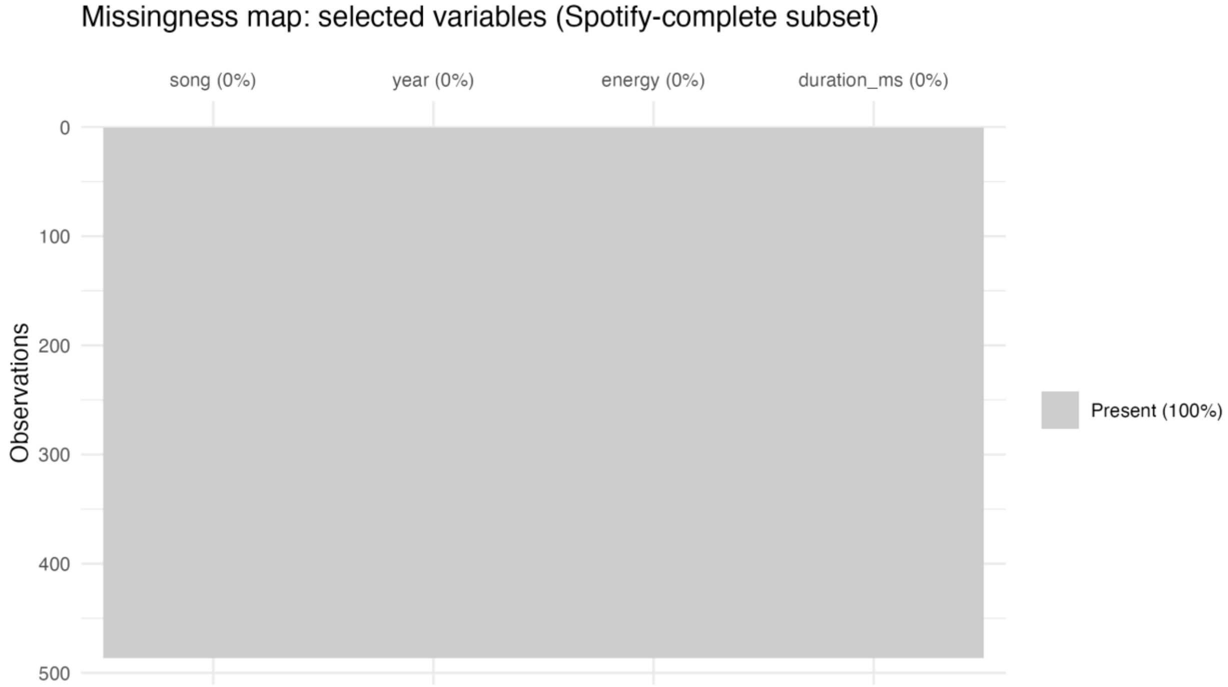
The research was conducted on a dataset of 3,397 Billboard Hot 100 songs that existed between the years 2000 and 2023. In cases where analyses involved audio features on Spotify, the research was limited to the Spotify complete subset of 486 songs because these are the ones that had full information on the audio features that had to be considered in the analysis (Pérez-Verdejo et al., 2021; Marlia et al., 2024).

Results from the assessment for temporal coverage showed that the vast preponderance of Spotify-complete data exists from the beginning of the 2000s (years 2000-2005), with very few songs from the latter years. Although this reduces the ability to generalise from the results over time, the sample set is sufficient for exploratory analysis, especially for audio features in popular music (Mauch et al., 2015).

Two research questions were answered in this study

(1) whether songs from Billboard can be grouped based on audio features, and (2) whether there is a relationship between song duration and energy.


Missingness map: selected variables (full dataset)

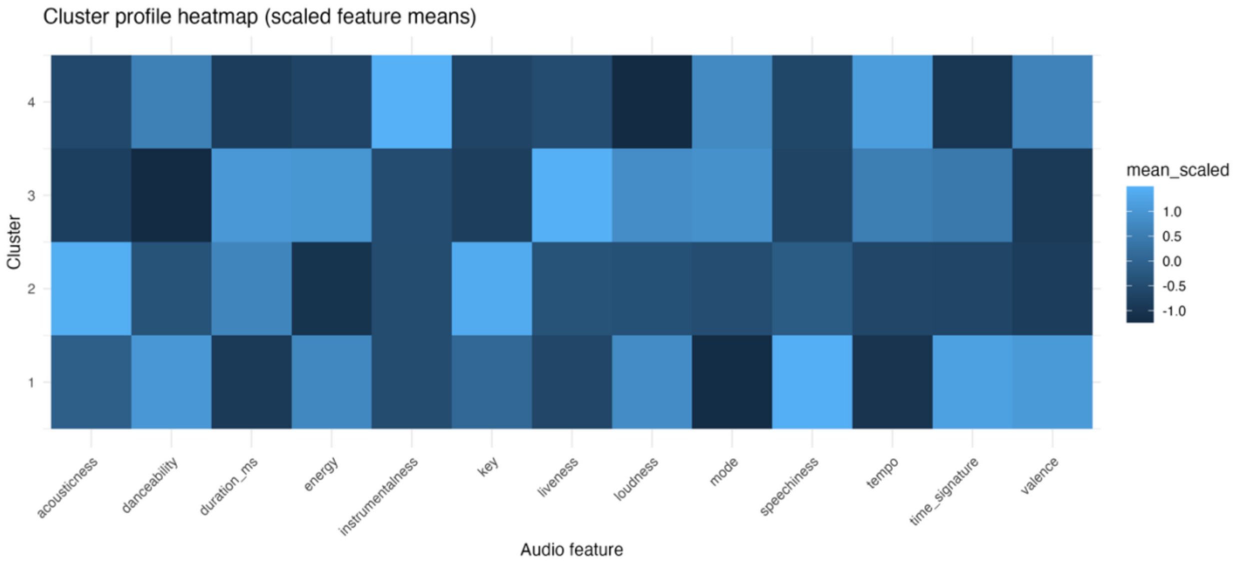

Missingness map: selected variables (Spotify-complete subset)

3.2 RQ1: Clustering Songs by Audio Characteristics

3.2.1 Principal Component Analysis

The standardised Spotify audio features were then subjected to Principal Component Analysis (PCA) to perform dimensionality reduction and address the problem of multicollinearity. The application of PCA in music analysis for the extraction of hidden musical variables and the facilitation of the clustering process has been extensive (Langensiepen et al., 2018; Cai et al., 2021).

The scree plot revealed that the components explain the variances gradually without any dominant component. Ten principal components were retained that accounted for about 85% of the total variance. This reveals that audio features of popular music are represented in various dimensions and are not controlled by dominant features. This has also been found in genre and popularity-based music classification studies in which music features are found to be multidimensional in nature (Long et al., 2021).



3.2.2 K-means Clustering and Selection of *k*

The K-means clustering algorithm was applied to the retained principal component scores. The elbow technique revealed a continually decreasing pattern in within-cluster sum of squares past the fourth cluster, whereas silhouette values revealed well-separated clusters for k = 2, and acceptable values for k = 4. Rather than maximising one criterion, a compromise was made between model adequacy and interpretability, and a value of k = 4 was chosen.

This is because interpretability and relevance are also taken into account in music clustering research, besides diagnostic quality (Jondya & Iswanto, 2017; Lala et al., 2023). It also supports recent literature that stresses the need for feature scaling and dimensionality reduction before clustering is performed (Wongoutong, 2024).

3.2.3 Cluster Structure in PCA Space

Analysis of clustering in the PC1-PC2 subspace shows some level of separation between clusters, despite some overlap. One cluster lies in the area of higher values of PC1, which might correspond to higher energy or loudness, while another lies in the area of lower values of PC2, which might correspond to lower valence or energy. Another cluster lies in the central area, which might correspond to more balanced audio characteristics, while a small fourth cluster corresponds to more extreme values.

This overlap can be expected because musical characteristics are found on continua and not on discrete points. Nevertheless, these results of the clustering algorithm prove that Billboard songs are not musically identical but are rather found in different parts of the audio feature space. This result confirms that popular music consists of various coexisting styles rather than being characterised by one dominant style (Mauch et al., 2015).

3.2.4 Cluster Profiles and Billboard Ranking

Analysis of the audio features at the level of clusters indicates the existence of distinct profiles. Some clusters tend to be more energy and loudness-oriented, as opposed to others, which tend to be more acousticness and lower energy-oriented. These findings echo previous observations, which suggested that the success of popular music can be achieved through a variety of audio features (Pérez-Verdejo et al., 2021).

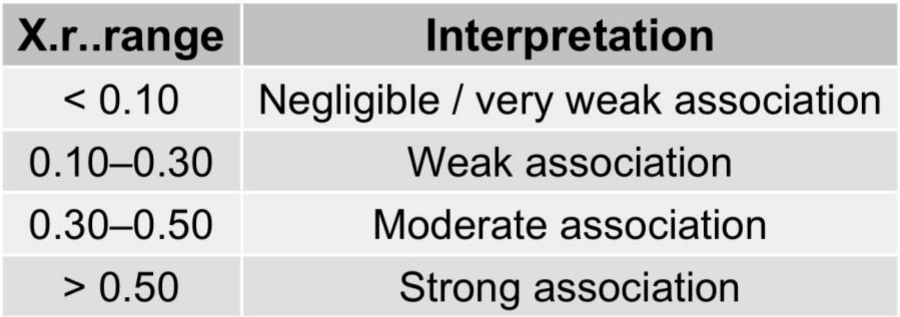
Comparison of Billboard rankings for each cluster reveals differences in median and mean rankings, with one cluster reaching a lower median ranking (indicating greater success on the Billboard charts). Nonetheless, there is some degree of overlap among all clusters. This shows that while audio-related factors are important for stylistic divergence, they are not the sole determinant of success on the Billboard charts. This is supported by other studies on the evolution of popular music (Nickerson, 2019)

3.3 RQ2: Relationship Between Song Duration and Energy

3.3.1 Overall Association

The correlation between song duration and energy was tested within the Spotify-complete dataset. The scatter plot of song duration in minutes versus energy shows a wide scatter of points, with a very weak tendency to decrease. The Pearson correlation coefficient is very weak and negative (r = −0.099).

What this means is that longer songs will be slightly less energetic on average, but the difference is small enough that it wouldn’t be a significant factor in a real-world context. Song length, then, appears to be a poor indicator of energy level when it comes to popular music.

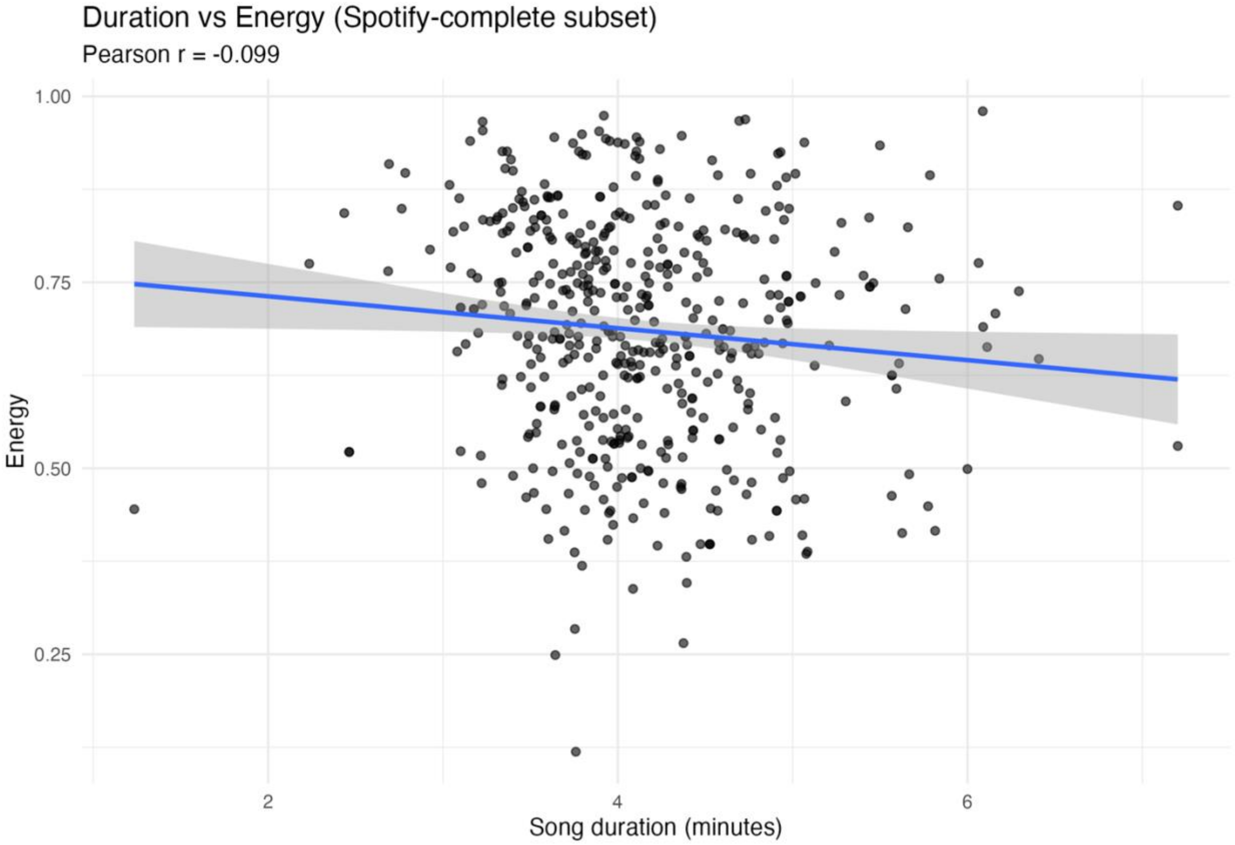


X.r..range	Interpretation
< 0.10	Negligible / very weak association
0.10–0.30	Weak association
0.30–0.50	Moderate association
> 0.50	Strong association

3.3.2 Interpretation and Context

The fact that there is a weak correlation between duration and energy indicates that these variables function relatively independently in popular music production. High-energy tracks appear to occupy all possible duration slots, while longer tracks do not necessarily appear to be more mellow or less dynamic. This is consistent with findings suggesting that energy perception is more closely related to production elements such as arrangement, rhythm, and production choices, as opposed to duration (Thiesen et al., 2020).

Industry-wise, this result defies simplistic beliefs that shorter songs are more likely to be energetic or successful. Energetic measures seem more a reflection of composition choices than time limitations, which is consistent with research on the longitudinal trends of popular music (Tan & Ko, 2025).



3.3.3 Scope and Limitations

Duration-Energy analysis is largely representative of Billboard music from the early 2000s, and so findings cannot be considered indicative of industry-wide practices across longer periods of time. However, it can be shown, via existing data, that duration and energy have a very weak association.

3.4 Integrated Discussion

On the whole, it is clear from the results that Billboard Hot 100 songs do form meaningful clusters based on audio features, though these clusters do not completely account for variations in popularity. Successful songs tend to reside in various regions of audio feature space, thus supporting the view that stylistic diversity, as opposed to convergence, is a hallmark of popular music.

The results of the clustering analysis demonstrate the strength of the combination of PCA and the k-means algorithm in revealing hidden musical structure, as in other music analysis tasks (Cai et al., 2021; Langensiepen et al., 2018). On the other hand, the analysis of duration and energy shows the weakness of the sole dependence on individual features in the analysis of musical success, as in the criticism of reductionist approaches in the analysis of musical data (Mauch et al., 2015).

3.5 Summary of Key Findings

PCA reveals that music traits vary along several dimensions instead of being concentrated in terms of a single trait.

K-means clustering (k = 4) identifies clear song clusters based on audio features.

Billboard rankings have considerable variation within the clusters, although they have considerable overlap, which indicates that a particular audio profile is not a determinant of Billboard success.

Song duration and energy have a very weak negative association. This shows that these factors are nearly independent of each other.

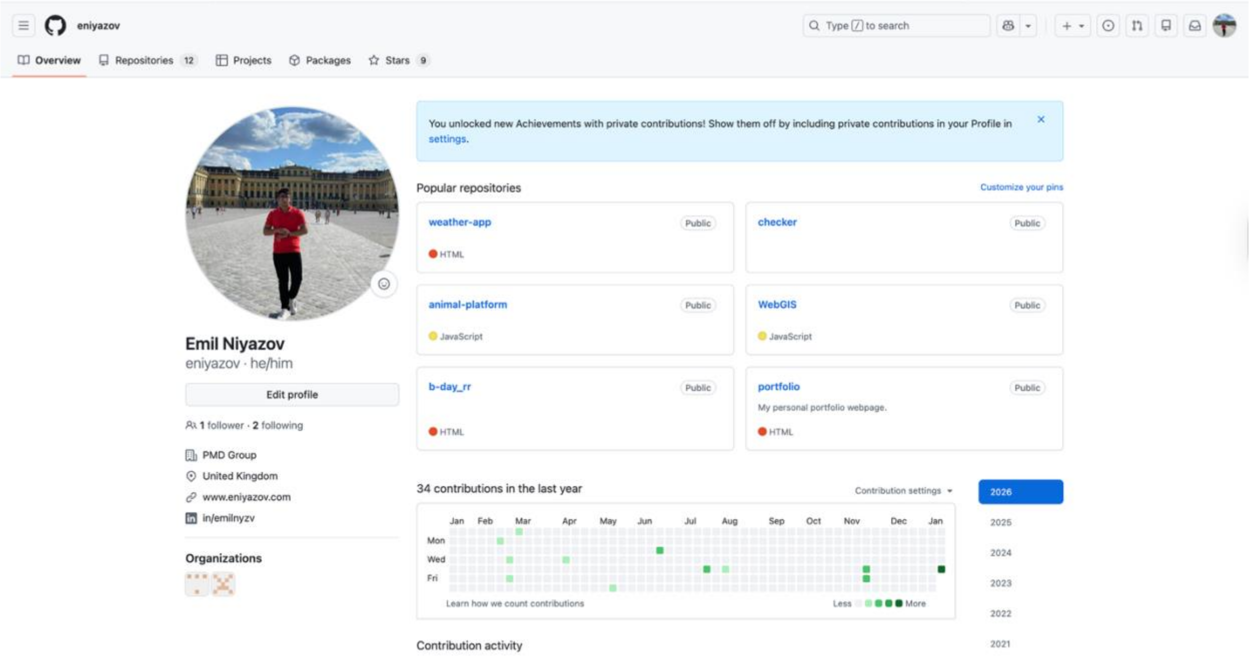
The findings point clearly to the power and weaknesses of feature analysis in the analysis of popular music.

4. R Code&GitHub

Links:

- Profile: https://github.com/eniyazov
- Repository: https://github.com/eniyazov/IJC437-Intro_to_Data_Science
- Code: https://github.com/eniyazov/IJC437-Intro_to_Data_Science/tree/main/code
- Instructions: https://github.com/eniyazov/IJC437-Intro_to_Data_Science/blob/main/README.md

Overall information about the project, instructions on how to run the code are given in README.md file. Below are the screenshots of the GitHub profile, repository and structure.



You unlocked new Achievements with private contributions! Show them off by including private contributions in your Profile in settings.

Emil Niyazov
eniyazov · he/him

1 follower · 2 following

PMD Group
United Kingdom
www.eniyazov.com
in/emilnyzv

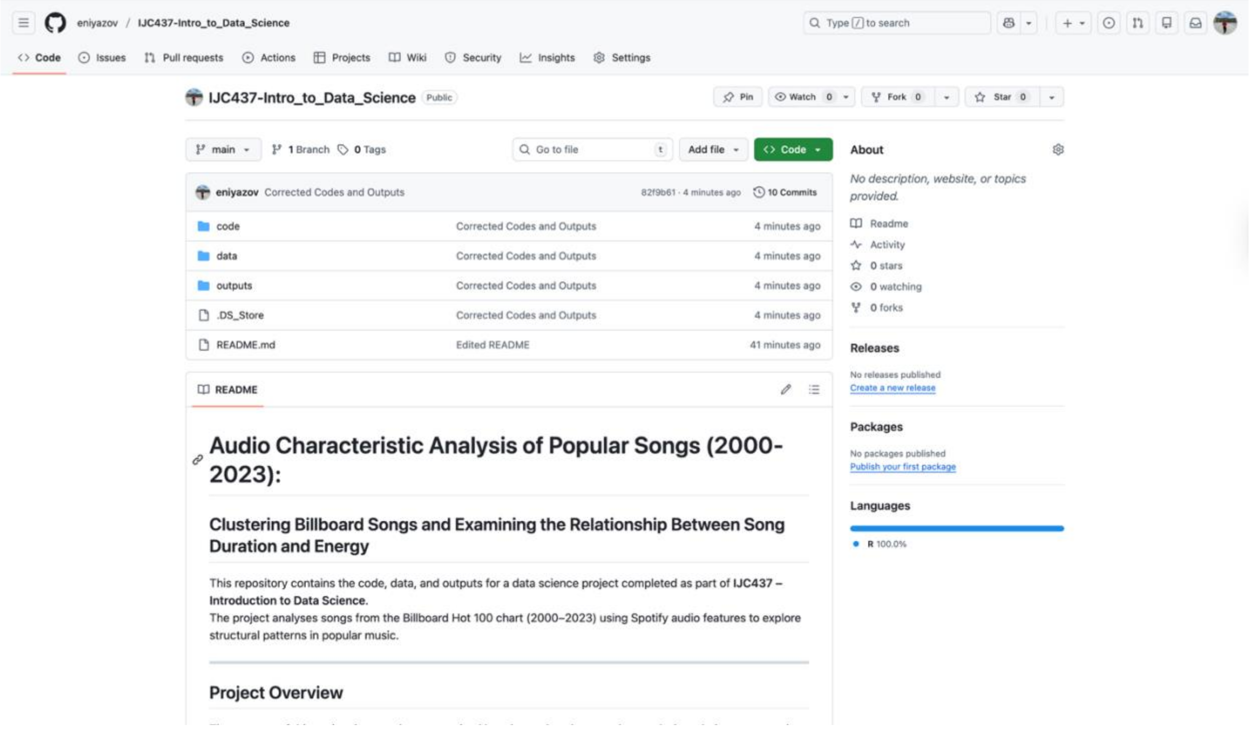
Organizations

Popular repositories

weather-app Public HTML	checker Public
animal-platform Public JavaScript	WebGIS Public JavaScript
b-day_rr Public HTML	portfolio Public My personal portfolio webpage. HTML

34 contributions in the last year

Contribution activity



IJC437-Intro_to_Data_Science Public

eniyazov Corrected Codes and Outputs 82f9b61 · 4 minutes ago 10 Commits

code	Corrected Codes and Outputs	4 minutes ago
data	Corrected Codes and Outputs	4 minutes ago
outputs	Corrected Codes and Outputs	4 minutes ago
.DS_Store	Corrected Codes and Outputs	4 minutes ago
README.md	Edited README	41 minutes ago

About

No description, website, or topics provided.

Readme
Activity
0 stars
0 watching
0 forks

Releases

No releases published
Create a new release

Packages

No packages published
Publish your first package

Languages

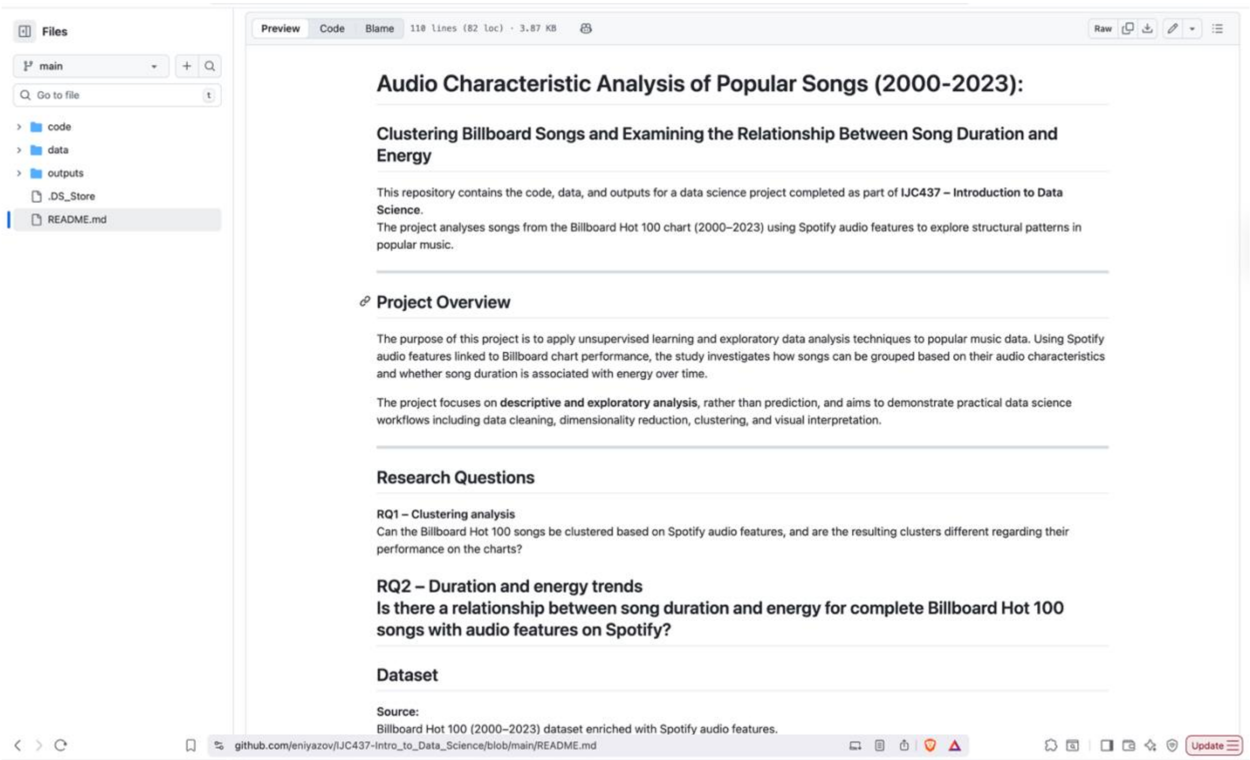
R 100.0%

Audio Characteristic Analysis of Popular Songs (2000-2023):

Clustering Billboard Songs and Examining the Relationship Between Song Duration and Energy

This repository contains the code, data, and outputs for a data science project completed as part of **IJC437 – Introduction to Data Science**.
The project analyses songs from the Billboard Hot 100 chart (2000–2023) using Spotify audio features to explore structural patterns in popular music.

Project Overview



Audio Characteristic Analysis of Popular Songs (2000-2023):

Clustering Billboard Songs and Examining the Relationship Between Song Duration and Energy

This repository contains the code, data, and outputs for a data science project completed as part of **IJC437 – Introduction to Data Science**.
The project analyses songs from the Billboard Hot 100 chart (2000–2023) using Spotify audio features to explore structural patterns in popular music.

Project Overview

The purpose of this project is to apply unsupervised learning and exploratory data analysis techniques to popular music data. Using Spotify audio features linked to Billboard chart performance, the study investigates how songs can be grouped based on their audio characteristics and whether song duration is associated with energy over time.

The project focuses on **descriptive and exploratory analysis**, rather than prediction, and aims to demonstrate practical data science workflows including data cleaning, dimensionality reduction, clustering, and visual interpretation.

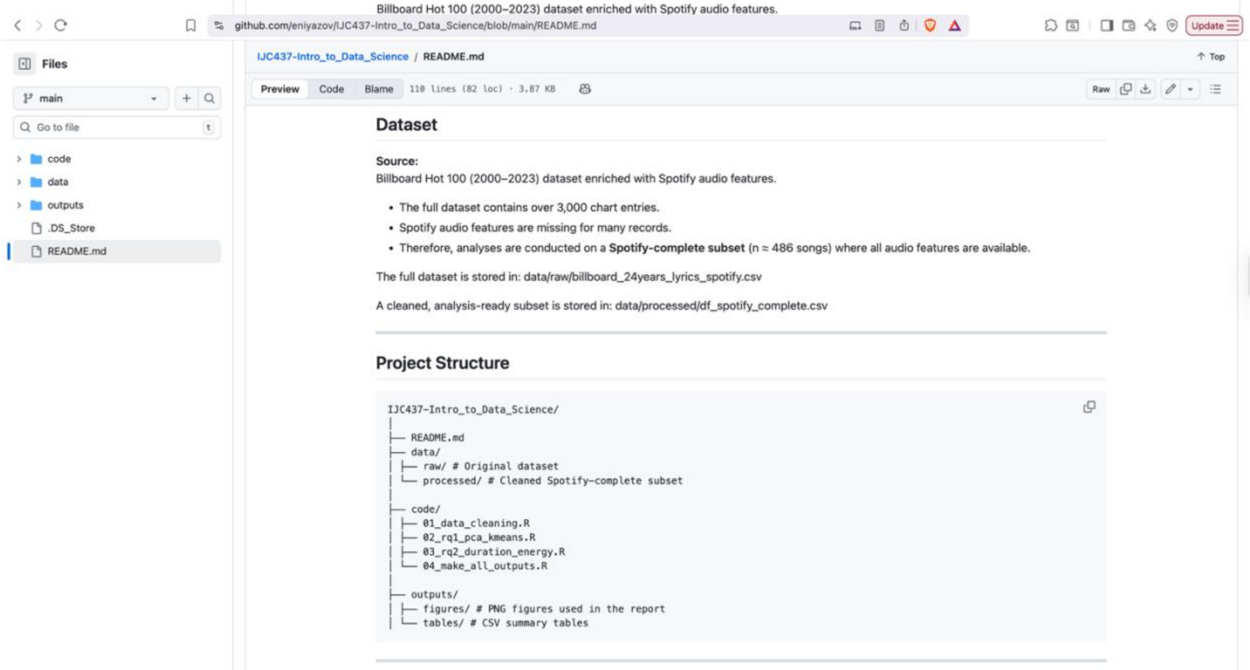
Research Questions

RQ1 – Clustering analysis
Can the Billboard Hot 100 songs be clustered based on Spotify audio features, and are the resulting clusters different regarding their performance on the charts?

RQ2 – Duration and energy trends
Is there a relationship between song duration and energy for complete Billboard Hot 100 songs with audio features on Spotify?

Dataset

Source:
Billboard Hot 100 (2000–2023) dataset enriched with Spotify audio features.


github.com/eniyazov/IJC437-Intro_to_Data_Science/blob/main/README.md
Update
IJC437-Intro_to_Data_Science / README.md
Top
Files
main
Go to file
code
data
outputs
.DS_Store
README.md
Preview
Code
Blame
110 lines (82 loc) · 3.87 KB
Raw

Dataset

Source:
Billboard Hot 100 (2000–2023) dataset enriched with Spotify audio features.

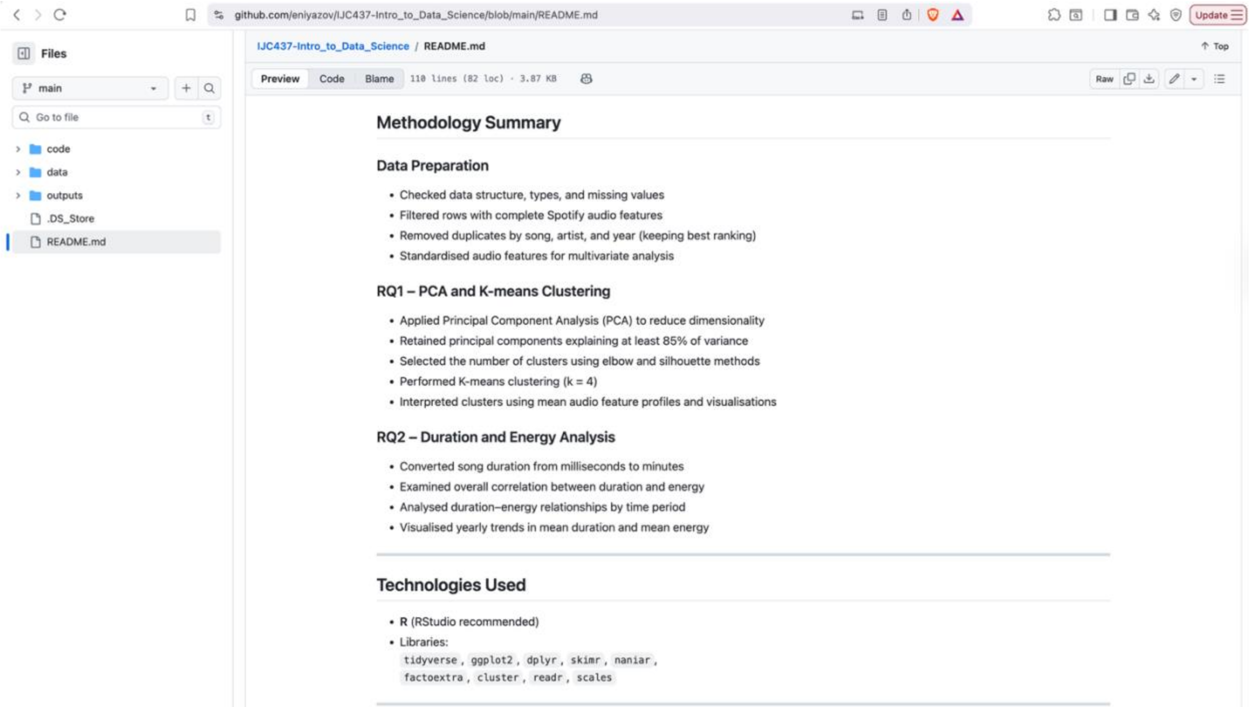
- The full dataset contains over 3,000 chart entries.
- Spotify audio features are missing for many records.
- Therefore, analyses are conducted on a **Spotify-complete subset** (n ≈ 486 songs) where all audio features are available.

The full dataset is stored in: data/raw/billboard_24years_lyrics_spotify.csv

A cleaned, analysis-ready subset is stored in: data/processed/df_spotify_complete.csv

Project Structure

```
IJC437-Intro_to_Data_Science/
│
├── README.md
├── data/
│   ├── raw/ # Original dataset
│   └── processed/ # Cleaned Spotify-complete subset
│
├── code/
│   ├── 01_data_cleaning.R
│   ├── 02_rq1_pca_kmeans.R
│   ├── 03_rq2_duration_energy.R
│   └── 04_make_all_outputs.R
│
├── outputs/
│   ├── figures/ # PNG figures used in the report
│   └── tables/ # CSV summary tables
```


github.com/eniyazov/IJC437-Intro_to_Data_Science/blob/main/README.md
Update
Files
main
Go to file
code
data
outputs
.DS_Store
README.md
IJC437-Intro_to_Data_Science / README.md
Top
Preview
Code
Blame
110 lines (82 loc) · 3.87 KB
Raw

Methodology Summary

Data Preparation

- Checked data structure, types, and missing values
- Filtered rows with complete Spotify audio features
- Removed duplicates by song, artist, and year (keeping best ranking)
- Standardised audio features for multivariate analysis

RQ1 – PCA and K-means Clustering

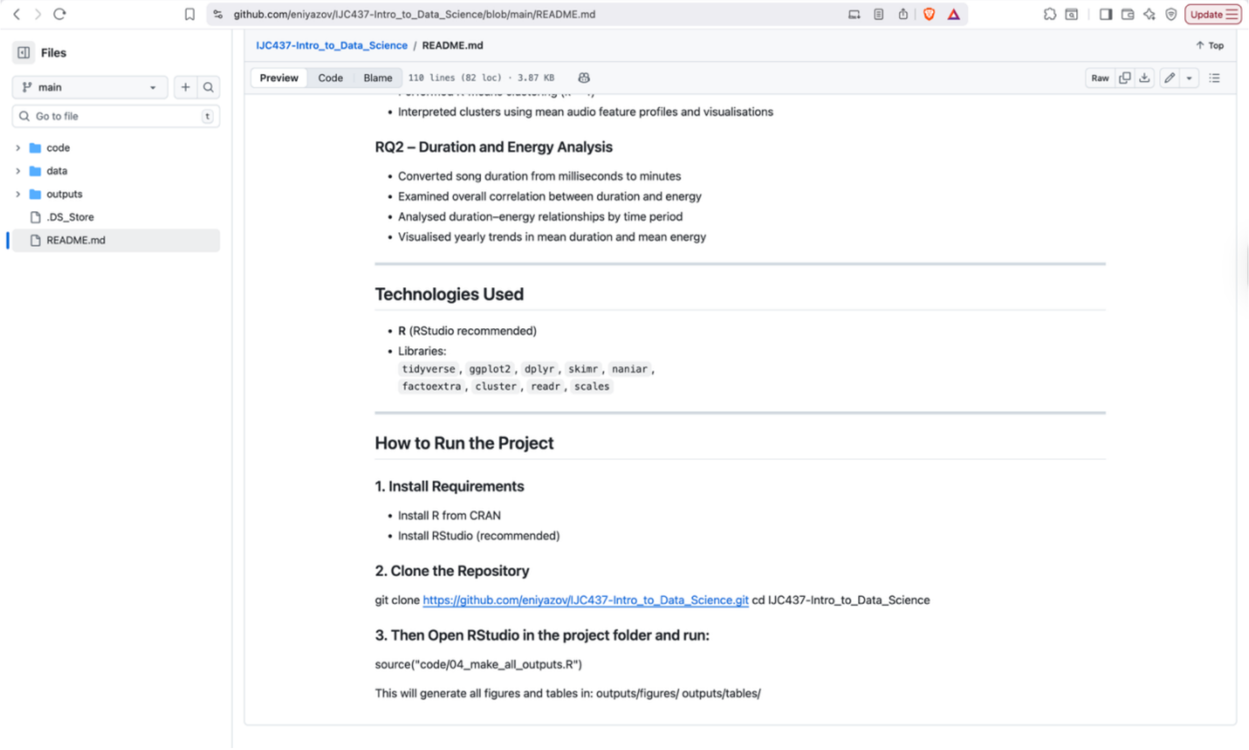
- Applied Principal Component Analysis (PCA) to reduce dimensionality
- Retained principal components explaining at least 85% of variance
- Selected the number of clusters using elbow and silhouette methods
- Performed K-means clustering (k = 4)
- Interpreted clusters using mean audio feature profiles and visualisations

RQ2 – Duration and Energy Analysis

- Converted song duration from milliseconds to minutes
- Examined overall correlation between duration and energy
- Analysed duration–energy relationships by time period
- Visualised yearly trends in mean duration and mean energy

Technologies Used

- **R** (RStudio recommended)
- Libraries:
  `tidyverse`, `ggplot2`, `dplyr`, `skimr`, `naniar`,
  `factoextra`, `cluster`, `readr`, `scales`


github.com/eniyazov/IJC437-Intro_to_Data_Science/blob/main/README.md
Update
Files
main
Go to file
code
data
outputs
.DS_Store
README.md
IJC437-Intro_to_Data_Science / README.md
Top
Preview
Code
Blame
110 lines (82 loc) · 3.87 KB
Raw

- Interpreted clusters using mean audio feature profiles and visualisations

RQ2 – Duration and Energy Analysis

- Converted song duration from milliseconds to minutes
- Examined overall correlation between duration and energy
- Analysed duration–energy relationships by time period
- Visualised yearly trends in mean duration and mean energy

Technologies Used

- **R** (RStudio recommended)
- Libraries:
  `tidyverse`, `ggplot2`, `dplyr`, `skimr`, `naniar`,
  `factoextra`, `cluster`, `readr`, `scales`

How to Run the Project

1. Install Requirements

- Install R from CRAN
- Install RStudio (recommended)

2. Clone the Repository

git clone https://github.com/eniyazov/IJC437-Intro_to_Data_Science.git cd IJC437-Intro_to_Data_Science

3. Then Open RStudio in the project folder and run:

source("code/04_make_all_outputs.R")

This will generate all figures and tables in: outputs/figures/ outputs/tables/

5. Conclusion, Reflections and Engagement

5.1 Conclusion and Reflections

In this study, the Spotify-complete dataset of the Billboard Year-End songs chart (from 2000 to 2023) on the Spotify platform has been examined to determine the extent to which popular songs tend to cluster in their audio features, such as duration versus energy features. PCA extracted ten components that cumulatively explained at least 85% of the variance, and the k-means algorithm yielded four interpretable clusters.

There was a slight difference in rankings among the clusters, with one of them tending to perform relatively better (lower) rankings, but this indicates that the audio attributes alone are not the complete picture, with marketing and popularity of artists also playing a role.

Limitations: (i) only 486 songs contained complete audio data from Spotify, which is biased towards earlier songs; (ii) missing data could be systematic, so results may not generalise to all hit songs on Billboard.

However, it is achieved by designing a clear and reproducible process for music analysis and interpreting results through statistics.

5.2 Reflection on Engagement and Professional Context

Engagement activities (industry and alumni talks) were informative about how stakeholder goals shape data requests, ways in which biases can be embedded in platform metrics, and the importance of methodological transparency. These have shaped key project choices, such as the decision to limit analysis to a Spotify-complete dataset, reporting limitations clearly, and avoiding claims of causality based on correlation patterns.

Final Summary

With a Spotify-complete set of Billboard Year-End Top 100 songs (from 2000 to 2023), this research utilised standardised music features to investigate trends on popular songs. Principal component analysis yielded 10 principal components (accounting for a total variance of at least 85%), and k-means clustering grouped songs into four categories, suggesting that top songs have multiple music features, not just one formulaic approach to music creation. Rank distributions are not identical across clusters, yet there is some overlap, suggesting that non-musical factors are also at play in determining a song's success. There is a negative correlation (r = -0.10) between length and energy, suggesting that longer songs are only slightly less energetic on average.

Refererences

1. Cai, Z., Fu, L., & Li, W. (2021). Research and analysis of music development based on k-means and PCA algorithm. *Journal of Physics. Conference Series*, *2083*(3), 32044. https://doi.org/10.1088/1742-6596/2083/3/032044
2. Enhanced Music Recommendation Systems: A Comparative Study of Content-Based Filtering and K-Means Clustering Approaches. (2024). *Revue d Intelligence Artificielle*. https://doi.org/10.18280/ria.380138
3. Evolution of Popular Music: Analyzing Two Decades of Shifts in the Billboard Top 10 (2003–23). (n.d.). *International Journal of Humanities and Social Science*. https://doi.org/10.14445/23942703/ijhss-v11i6p101
4. Haghbayan, H., Coomes, E. A., & Curran, D. (2020). Temporal Trends in the Loudness of Popular Music over Six Decades. *Journal of General Internal Medicine : JGIM*, *35*(1), 394–395. https://doi.org/10.1007/s11606-019-05210-4
5. Jondya, A. G., & Iswanto, B. H. (2017). Indonesian's Traditional Music Clustering Based on Audio Features. *Procedia Computer Science*, *116*, 174–181. https://doi.org/10.1016/j.procs.2017.10.019

6. Juhász, Z. (2024). Revealing Footprints of Ancient Sources in Recent Eurasian and American Folk Music Cultures Using PCA of the Culture-Dependent Moment Vectors of Shared Melody Types. *Music & Science*, *7*. https://doi.org/10.1177/20592043241228982
7. Langensiepen, C., Cripps, A., & Cant, R. (2018). Using PCA and K-Means to Predict Likeable Songs from Playlist Information. *2018 UKSim AMSS 20th International Conference on Computer Modelling and Simulation (UKSim)*, 26–31. https://doi.org/10.1109/UKSim.2018.00017
8. Lala, R., Patankar, G., Patil, A., & Deshpaande, H. (2023). Enhancing Music Data Clustering: An Empirical Analysis for Music Clustering and Scaling Optimization. *2023 6th International Conference on Advances in Science and Technology (ICAST)*, 313–318. https://doi.org/10.1109/ICAST59062.2023.10454979
9. Leung, C. K. S., & Zhang, Y. (2019). An HSV-Based Visual Analytic System for Data Science on Music and Beyond. *International Journal of Art, Culture and Design Technologies*, *8*(1), 68–83. https://doi.org/10.4018/IJACDT.2019010105
10. Long, M., Hu, L., & Jin, F. (2021). Analysis of Main Characteristics of Music Genre Based on PCA Algorithm. *2021 IEEE 2nd International Conference on Big Data, Artificial Intelligence and Internet of Things Engineering (ICBAIE)*, 101–105. https://doi.org/10.1109/ICBAIE52039.2021.9389878
11. Marlia, S., Setiawan, K., & Juliane, C. (2024). Analysis of Music Features and Song Popularity Trends on Spotify Using K-Means and CRISP-DM. *Sistemasi : Jurnal Sistem Informasi (Online)*, *13*(2), 595–607. https://doi.org/10.32520/stmsi.v13i2.3757
12. Mauch, M., MacCallum, R. M., Levy, M., & Leroi, A. M. (2015). The evolution of popular music: USA 1960–2010. *Royal Society Open Science*, *2*(5), 150081–150081. https://doi.org/10.1098/rsos.150081
13. Nickerson, B. J. (2019). The Underlying Underwriter: An Analysis of the Spotify Direct Listing. *The University of Chicago Law Review*, *86*(4), 985–1025.
14. Pérez-Verdejo, J. M., Piña-García, C. A., Ojeda, M. M., Rivera-Lara, A., & Méndez-Morales, L. (2021). The rhythm of Mexico: an exploratory data analysis of Spotify's top 50. *Journal of Computational*

Social Science, *4*(1), 147–161. https://doi.org/10.1007/s42001-020-00070-z

15. Tan, E. E.-L., & Ko, A. M.-S. (2025). Dynamics of Popular Music Over a Decade: A Longitudinal Analysis of Spotify's Top Tracks and the Impact of the COVID-19 Pandemic. *Journal of Student Research (Houston, Tex.)*, *14*(1). https://doi.org/10.47611/jsrhs.v14i1.8744

16. Thiesen, F. C., Kopiez, R., Müllensiefen, D., Reuter, C., & Czedik-Eysenberg, I. (2020). Duration, song section, entropy: Suggestions for a model of rapid music recognition processes. *Journal of New Music Research*, *49*(4), 334–348. https://doi.org/10.1080/09298215.2020.1784955

17. Wongoutong, C. (2024). The impact of neglecting feature scaling in k-means clustering. *PloS One*, *19*(12), e0310839. https://doi.org/10.1371/journal.pone.0310839

Appendix A: Reproducible Analysis Code

Appendix A.1: Package Setup Script

```r
# ==========================================
# 00_setup_packages.R
# Sets CRAN mirror + installs/loads packages
# ==========================================

options(repos = c(CRAN = "https://cloud.r-project.org"))

required_packages <- c(
  "tidyverse", "skimr", "naniar", "here",
  "factoextra", "cluster", "scales"
)

to_install <- required_packages[!required_packages %in% installed.packages()[, "Package"]]

if (length(to_install) > 0) {
  install.packages(to_install)
}

invisible(lapply(required_packages, library, character.only = TRUE))
```

Appendix A.2: Data Cleaning and Preparation

```
# ============================================
# 01_data_cleaning.R
# Data loading, audit, missingness, subset creation
# ============================================

# # ---- Packages ----
# pkgs <- c("tidyverse", "skimr", "naniar", "here")
# to_install <- pkgs[!pkgs %in% installed.packages()[, "Package"]]
# if (length(to_install) > 0) install.packages(to_install)

library(tidyverse)
library(skimr)
library(naniar)
library(here)

# ---- Paths ----
dir.create(here("data/processed"), recursive = TRUE, showWarnings = FALSE)
dir.create(here("outputs/figures"), recursive = TRUE, showWarnings = FALSE)
dir.create(here("outputs/tables"),  recursive = TRUE, showWarnings = FALSE)

raw_path <- here("data/raw/billboard_24years_lyrics_spotify.csv")

# ---- Load ----
raw <- readr::read_csv(raw_path, show_col_types = FALSE)

# ---- Quick audit ----
cat("Raw rows:", nrow(raw), "\n")
cat("Raw cols:", ncol(raw), "\n")
cat("Year range:", min(raw$year, na.rm = TRUE), "-", max(raw$year, na.rm = TRUE), "\n")

# Save audit tables (helpful for appendices)
```

```r
raw %>% slice(1:20) %>%
  write_csv(here("outputs/tables/data_example_first20_raw.csv"))

# Missingness by variable (full dataset)
naniar::miss_var_summary(raw) %>%
  write_csv(here("outputs/tables/missingness_by_variable_raw.csv"))

# Save a light "df_all_basic" table for repo use (optional but useful)
df_all <- raw %>%
  filter(!is.na(year), year >= 2000, year <= 2023) %>%
  filter(!is.na(ranking), ranking >= 1) %>%
  mutate(
    year = as.integer(year),
    ranking = as.integer(ranking)
  )

write_csv(df_all, here("outputs/tables/df_all_basic.csv"))

# ---- Define Spotify audio variables ----
audio_vars <- c(
  "danceability","energy","key","loudness","mode","speechiness",
  "acousticness","instrumentalness","liveness","valence","tempo",
  "duration_ms","time_signature"
)

# ---- Create Spotify-complete subset ----
# Keep only rows with complete audio variables (no NA in any of these)
df_spotify <- df_all %>%
  filter(if_all(all_of(audio_vars), ~ !is.na(.)))

cat("Spotify-complete rows:", nrow(df_spotify), "\n")

# Deduplicate: keep best ranking per (song, artist, year)
df_spotify <- df_spotify %>%
  arrange(song, band_singer, year, ranking) %>%
  group_by(song, band_singer, year) %>%
  slice(1) %>%
  ungroup()

cat("Spotify-complete rows after dedup:", nrow(df_spotify), "\n")

# Save Spotify subset for reproducible analysis
write_csv(df_spotify, here("data/processed/df_spotify_complete.csv"))

# Missingness check on Spotify subset (should be 0 for audio vars)
naniar::miss_var_summary(df_spotify %>% select(all_of(audio_vars))) %>%
  write_csv(here("outputs/tables/missingness_df_spotify.csv"))

# ---- Visual: missingness map for RQ2-selected columns ----
p_miss_selected <- df_all %>%
  select(song, year, energy, duration_ms) %>%
  naniar::vis_miss() +
  theme_minimal() +
  labs(title = "Missingness map: selected variables (full dataset)")

ggsave(
  filename = here("outputs/figures/missingness_full_selected_vars.png"),
  plot = p_miss_selected,
  width = 8, height = 4.5, dpi = 300
)

# ---- Visual: songs per year (full dataset) ----
p_songs_year <- df_all %>%
  count(year) %>%
  ggplot(aes(x = year, y = n)) +
  geom_col() +
  theme_minimal() +
  labs(title = "Number of songs per year (full dataset)", x = "Year", y = "Count")

ggsave(
  filename = here("outputs/figures/eda_songs_per_year_full.png"),
  plot = p_songs_year,
  width = 8, height = 4.5, dpi = 300
)
```

```
cat("\n01_data_cleaning.R complete.\n")
```

Appendix A.3: PCA and K-Means Clustering (RQ1)

```
# ==========================================
# 02_rq1_pca_kmeans.R
# PCA + K-means clustering + cluster interpretation outputs
# ==========================================

# # ---- Packages ----
# pkgs <- c("tidyverse", "here", "factoextra", "cluster", "scales")
# to_install <- pkgs[!pkgs %in% installed.packages()[, "Package"]]
# if (length(to_install) > 0) install.packages(to_install)

library(tidyverse)
library(here)
library(factoextra)
library(cluster)
library(scales)

# ---- Paths ----
dir.create(here("outputs/figures"), recursive = TRUE, showWarnings = FALSE)
dir.create(here("outputs/tables"),  recursive = TRUE, showWarnings = FALSE)

# ---- Load processed Spotify-complete dataset ----
df_spotify <- readr::read_csv(here("data/processed/df_spotify_complete.csv"), show_col_types = FALSE)

audio_vars <- c(
  "danceability","energy","key","loudness","mode","speechiness",
  "acousticness","instrumentalness","liveness","valence","tempo",
  "duration_ms","time_signature"
)

# ---- Scaling + PCA ----
X <- df_spotify %>% select(all_of(audio_vars)) %>% as.data.frame()
```

```r
X_scaled <- scale(X)

pca <- prcomp(X_scaled, center = TRUE, scale. = FALSE)

# Variance table
pca_var <- (pca$sdev^2) / sum(pca$sdev^2)
pca_var_tbl <- tibble(
  pc = paste0("PC", seq_along(pca_var)),
  variance = pca_var,
  cumulative_variance = cumsum(pca_var)
)

write_csv(pca_var_tbl, here("outputs/tables/rq1_pca_variance_table.csv"))

# Scree plot
p_scree <- ggplot(pca_var_tbl, aes(x = pc, y = variance)) +
  geom_col() +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(title = "PCA Scree Plot (Spotify-complete subset)", x = "Principal Components", y = "Proportion of variance")

ggsave(here("outputs/figures/rq1_pca_scree.png"), p_scree, width = 8, height = 5, dpi = 300)

# Choose number of PCs for >= 85% variance
k_pcs <- pca_var_tbl %>%
  filter(cumulative_variance >= 0.85) %>%
  slice(1) %>%
  mutate(k = as.integer(str_remove(pc, "PC"))) %>%
  pull(k)

cat("Number of PCs retained for >=85% variance:", k_pcs, "\n")

pc_scores <- as.data.frame(pca$x[, 1:k_pcs, drop = FALSE])

# ---- Choose k: elbow + silhouette ----
p_elbow <- factoextra::fviz_nbclust(pc_scores, kmeans, method = "wss") +
  theme_minimal() +
  labs(title = "Elbow method for choosing k (Spotify-complete subset)")
ggsave(here("outputs/figures/rq1_kmeans_elbow.png"), p_elbow, width = 8, height = 5, dpi = 300)

p_sil <- factoextra::fviz_nbclust(pc_scores, kmeans, method = "silhouette") +
  theme_minimal() +
  labs(title = "Silhouette method for choosing k (Spotify-complete subset)")
ggsave(here("outputs/figures/rq1_kmeans_silhouette.png"), p_sil, width = 8, height = 5, dpi = 300)

# ---- Fit final K-means ----
set.seed(123)
k_final <- 4

km <- kmeans(pc_scores, centers = k_final, nstart = 50)
df_spotify <- df_spotify %>% mutate(cluster = factor(km$cluster))

write_csv(df_spotify, here("outputs/tables/rq1_clustered_df_spotify.csv"))

# ---- Cluster plot PC1 vs PC2 ----
pc12 <- as.data.frame(pca$x[, 1:2]) %>%
  mutate(cluster = df_spotify$cluster)

p_clusters <- ggplot(pc12, aes(PC1, PC2, color = cluster)) +
  geom_point(alpha = 0.7) +
  theme_minimal() +
  labs(title = "Song clusters in PCA space (Spotify-complete subset)", x = "PC1", y = "PC2")

ggsave(here("outputs/figures/rq1_clusters_pc1_pc2.png"), p_clusters, width = 8, height = 5.5, dpi = 300)

# ---- Cluster sizes ----
p_cluster_sizes <- df_spotify %>%
  count(cluster) %>%
  ggplot(aes(cluster, n)) +
  geom_col() +
  theme_minimal() +
  labs(title = "Cluster sizes (Spotify-complete subset)", x = "Cluster", y = "Number of songs")

ggsave(here("outputs/figures/rq1_cluster_sizes.png"), p_cluster_sizes, width = 7, height = 5, dpi = 300)
```

```
# ---- Cluster profiles (mean audio features) ----
cluster_profile <- df_spotify %>%
  group_by(cluster) %>%
  summarise(across(all_of(audio_vars), \(x) mean(x, na.rm = TRUE)),
            n = n(), .groups = "drop") %>%
  arrange(cluster)

write_csv(cluster_profile, here("outputs/tables/rq1_cluster_profiles_means.csv"))

# ---- Cluster profile heatmap (scaled means) ----
profiles_long <- cluster_profile %>%
  select(cluster, all_of(audio_vars)) %>%
  pivot_longer(-cluster, names_to = "feature", values_to = "mean_value") %>%
  group_by(feature) %>%
  mutate(mean_scaled = as.numeric(scale(mean_value))) %>%
  ungroup()

p_heat <- ggplot(profiles_long, aes(feature, cluster, fill = mean_scaled)) +
  geom_tile() +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1)) +
  labs(
    title = "Cluster profile heatmap (scaled feature means)",
    x = "Audio feature", y = "Cluster"
  )

ggsave(here("outputs/figures/rq1_cluster_profile_heatmap.png"), p_heat, width = 11, height = 5, dpi = 300)

# ---- Ranking by cluster ----
cluster_ranking <- df_spotify %>%
  group_by(cluster) %>%
  summarise(
    n = n(),
    mean_ranking = mean(ranking, na.rm = TRUE),
    median_ranking = median(ranking, na.rm = TRUE),
    .groups = "drop"
  ) %>%
  arrange(cluster)

write_csv(cluster_ranking, here("outputs/tables/rq1_cluster_ranking_summary.csv"))

p_rank_box <- ggplot(df_spotify, aes(cluster, ranking)) +
  geom_boxplot() +
  theme_minimal() +
  labs(
    title = "Ranking distribution by cluster (Spotify-complete subset)",
    x = "Cluster",
    y = "Billboard ranking (lower is better)"
  )

ggsave(here("outputs/figures/rq1_ranking_by_cluster_boxplot.png"), p_rank_box, width = 7, height = 5, dpi = 300)

# ---- Save summary info ----
summary_rq1 <- tibble(
  n_rows_spotify_complete = nrow(df_spotify),
  k_pcs_85pct = k_pcs,
  kmeans_k_final = k_final
)
write_csv(summary_rq1, here("outputs/tables/rq1_summary.csv"))

cat("\n02_rq1_pca_kmeans.R complete.\n")
```

Appendix A.4: Duration and Energy Analysis (RQ2)

```
# ==========================================
# 03_rq2_duration_energy.R
# RQ2: Duration vs Energy analysis
# ==========================================

# # ---- Packages ----
# pkgs <- c("tidyverse", "here")
# to_install <- pkgs[!pkgs %in% installed.packages()[, "Package"]]
```

```r
# if (length(to_install) > 0) install.packages(to_install)

library(tidyverse)
library(here)

dir.create(here("outputs/figures"), recursive = TRUE, showWarnings = FALSE)
dir.create(here("outputs/tables"),  recursive = TRUE, showWarnings = FALSE)

# ---- Load clustered Spotify dataset (includes audio vars) ----
df_spotify <- readr::read_csv(here("outputs/tables/rq1_clustered_df_spotify.csv"), show_col_types = FALSE)

# Duration in minutes
df_spotify <- df_spotify %>%
  mutate(duration_min = duration_ms / 60000)

# ---- Overall correlation ----
cor_overall <- cor(df_spotify$duration_min, df_spotify$energy, use = "complete.obs")
cat("Overall correlation (duration_min vs energy):", cor_overall, "\n")

write_csv(tibble(correlation = cor_overall),
          here("outputs/tables/rq2_duration_energy_correlation_overall.csv"))

# ---- Scatter plot (overall) ----
p_rq2_all <- ggplot(df_spotify, aes(duration_min, energy)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = TRUE) +
  theme_minimal() +
  labs(
    title = "Duration vs Energy (Spotify-complete subset)",
    subtitle = paste0("Pearson r = ", round(cor_overall, 3)),
    x = "Song duration (minutes)",
    y = "Energy"
  )

ggsave(here("outputs/figures/rq2_duration_energy_scatter_all.png"),
       p_rq2_all, width = 8, height = 5.5, dpi = 300)
```

```r
# ---- Period analysis (will show the data sparsity clearly) ----
df_spotify <- df_spotify %>%
  mutate(
    period = case_when(
      year >= 2000 & year <= 2005 ~ "2000–2005",
      year >= 2006 & year <= 2010 ~ "2006–2010",
      year >= 2011 & year <= 2015 ~ "2011–2015",
      year >= 2016 & year <= 2020 ~ "2016–2020",
      year >= 2021 & year <= 2023 ~ "2021–2023",
      TRUE ~ "Other"
    ),
    period = factor(period, levels = c("2000–2005","2006–2010","2011–2015","2016–2020","2021–2023"))
  )

cor_by_period <- df_spotify %>%
  group_by(period) %>%
  summarise(
    n = n(),
    correlation = ifelse(sd(duration_min, na.rm = TRUE) == 0 | sd(energy, na.rm = TRUE) == 0,
                         NA_real_,
                         cor(duration_min, energy, use = "complete.obs")),
    .groups = "drop"
  )

write_csv(cor_by_period,
          here("outputs/tables/rq2_duration_energy_correlation_by_period.csv"))

# Faceted scatter by period
p_rq2_facet <- ggplot(df_spotify, aes(duration_min, energy)) +
  geom_point(alpha = 0.6) +
  geom_smooth(method = "lm", se = FALSE) +
  facet_wrap(~ period) +
  theme_minimal() +
  labs(
    title = "Duration vs Energy by period (Spotify-complete subset)",
    x = "Song duration (minutes)",
    y = "Energy"
  )

ggsave(here("outputs/figures/rq2_duration_energy_scatter_by_period.png"),
       p_rq2_facet, width = 10, height = 6, dpi = 300)

# ---- Yearly trends (means) ----
yearly_trends <- df_spotify %>%
  group_by(year) %>%
  summarise(
    n = n(),
    mean_duration_min = mean(duration_min, na.rm = TRUE),
    mean_energy = mean(energy, na.rm = TRUE),
    .groups = "drop"
  ) %>%
  arrange(year)

write_csv(yearly_trends,
          here("outputs/tables/rq2_yearly_trends_duration_energy.csv"))

p_trends <- ggplot(yearly_trends, aes(x = year)) +
  geom_line(aes(y = mean_duration_min, group = 1)) +
  geom_point(aes(y = mean_duration_min)) +
  theme_minimal() +
  labs(
    title = "Yearly mean song duration (minutes) in Spotify-complete subset",
    x = "Year", y = "Mean duration (minutes)"
  )

ggsave(here("outputs/figures/rq2_yearly_trends_duration_energy.png"),
       p_trends, width = 8, height = 5, dpi = 300)

cat("\n03_rq2_duration_energy.R complete.\n")
```

Appendix A.5: Master Execution Script

```
# ============================================
```

```r
# 04_make_all_outputs.R

# Master script: runs the whole pipeline

# ==========================================


source("code/00_setup_packages.R")

source("code/01_data_cleaning.R")

source("code/02_rq1_pca_kmeans.R")

source("code/03_rq2_duration_energy.R")


cat("\nDONE. All outputs generated in outputs/figures and outputs/tables.\n")
```