INFO-I590 Data Visualization
Fall 2024

Blended Visual Beats - Visualizing Playlist Comparisons and Personalized Music Recommendations

Group 7– Gandhar Ravindra Pansare, Rutuja Jangle, Swarn Gaba

GitHub Link - https://github.iu.edu/sgaba/Blended_Visual_Beats

NOTE – This repository contains all the code used to build the dashboard, Python notebooks for visualizations and EDA, datasets, and the presentation.

ABSTRACT

Spotify, the world's leading audio streaming subscription service, boasts over 574 million users (226 million subscribers) across more than 180 countries [1]. With its vast collection of musical data, Spotify offers a rich opportunity for analysis. This project leverages Spotify data to explore and compare two distinct playlists, analyzing their similarities and differences, identifying shared trends, and providing personalized music recommendations tailored to listeners of each playlist. Through simple yet intuitive visualizations, this project combines static and dynamic techniques to present insights. The result is an interactive dashboard that allows users to compare playlists, understand their music preferences, and discover new recommendations based on common or similar musical interests. This approach bridges data analysis and music personalization, creating an engaging experience for users.

1. INTRODUCTION

1.1 MOTIVATION

This project is inspired by Spotify's *Blend* feature [2], a unique recommendation system that offers multi-user personalization. *Blend* enables multiple users to collaborate on a shared playlist, generating recommendations based on their overlapping interests. However, this feature lacks a visual component, offering only a similarity percentage and a shared song between playlists. By incorporating visual elements into this shared recommendation approach, users can gain deeper insights into their music preferences and discover new tracks in a more engaging way. The potential for enhancing music discovery through visualizing relationships between artists, genres, and preferences is supported by research in music recommendation systems. Studies have shown that visualizing user preferences can significantly enhance the music discovery process [3]. This project aims to address this gap by supplementing shared recommendations with intuitive visualizations, enabling users to explore and understand their musical connections more effectively.

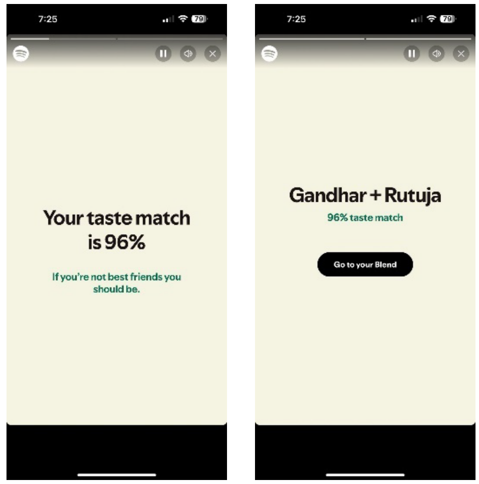


Figure 1: Current Spotify Blend Visualization (Images from own profile)

Another key motivation for this project is to develop a flexible tool for comparative analysis of two playlists. Such a tool could serve various purposes, including comparing personal musical tastes between friends, analyzing top tracks from different years (e.g., 2022 vs. 2024), or even examining listening trends across countries (e.g., top tracks in the USA vs. India). By comparing attributes such as top artists, genres, and song features like energy or acousticness, the tool aims to uncover trends and patterns in music preferences, making it versatile for multiple use cases.

Additionally, this project is motivated by the opportunity to explore musical data, a unique and rich data domain. It seeks to analyze this data through a variety of visualizations, such as scatterplots and bar charts, to assess their effectiveness in representing musical features. Furthermore, the project aims to identify whether certain visualization techniques are particularly well-suited for analyzing music-related data, contributing to the broader understanding of how to effectively present such insights.

1.2 EXISTING WORK

1.2.1 Stats for Spotify

One of the most widely used platforms for analyzing Spotify trends is *Stats for Spotify* [4]. This website provides a straightforward way to view top tracks, top artists, and top genres over various time periods. While it is simple and easy to understand, the platform offers limited insights beyond basic information.

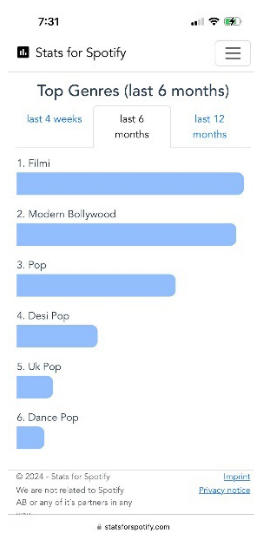


Figure 2: Bar plot showing top genres in 'Stats for Spotify'

The bar plot used to display top genres, for example, lacks a scale, making it unclear what the bar lengths represent. This ambiguity can lead to confusion. Furthermore, the platform focuses solely on individual user profiles, offering no functionality for comparing data across multiple users or playlists. Additionally, it does not provide personalized music recommendations, limiting its utility for users seeking deeper insights or new discoveries.

1.2.2 Spotify Pie

Spotify Pie [5] is another platform that provides a basic visualization of a user's music preferences. It generates a pie chart displaying the top genres, along with a list of top artists shown below the chart. Users can also download the generated chart as an image file.

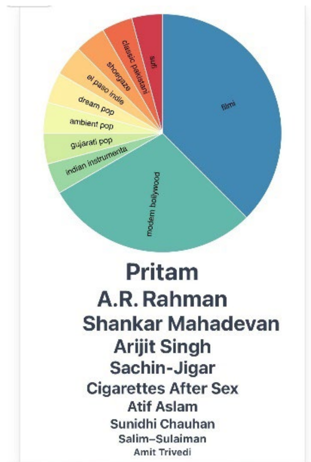


Figure 3: 'The Spotify Pie' interface

However, the tool has significant limitations. The list of artists lacks any meaningful context. This creates a confusing and misleading visualization. Similar to *Stats for Spotify*, this tool only works for individual user profiles and does not offer comparison features or personalized music recommendations. As a result, it provides little value for users looking for deeper insights or shared trends across playlists.

1.2.3 Obscurify

Obscurify Music [6] is an interactive dashboard offering several unique and engaging features. Its primary objective is to quantify and compare how obscure a user's musical taste is relative to general trends in a selected country. The platform provides insights into top tracks, artists, and genres, accompanied by various interesting visualizations.

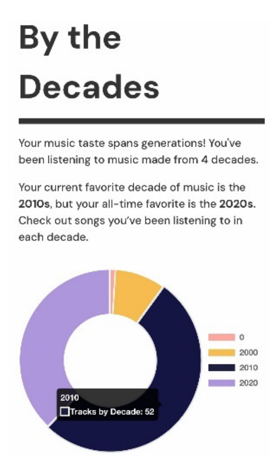


Figure 4: Donut plot showing number of songs for each decade

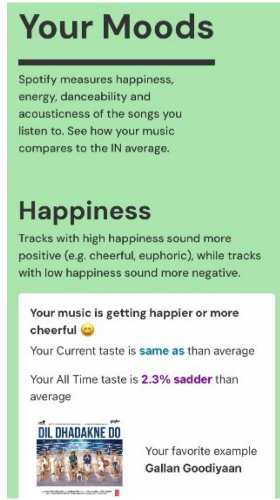


Figure 5: Comparing user moods

While *Obscurify* allows users to share results and compare with others, the comparisons are not visually represented, and no combined trends are extracted. Additionally, it generates recommendations that can be saved as a playlist; however, these recommendations are based solely on an individual user's preferences, and the attributes of the recommended songs are not visually explained. The platform leans heavily on text-based insights, with fewer visual explanations of the data.

Despite these limitations, *Obscurify* offers a comprehensive overview of individual music preferences and excels in interactivity and trend reporting. However, it falls short for users seeking visual comparisons or shared insights across multiple playlists.

1.2.4 Musictaste.space

Musictaste.space [7] is a highly interactive platform that provides insights into individual listening habits and allows comparisons with other Spotify users. The platform also highlights general trends, mood-related trends, and integrates obscurity scores from *Obscurify* as an added feature. One of its standout functions is the ability to compare two playlists.

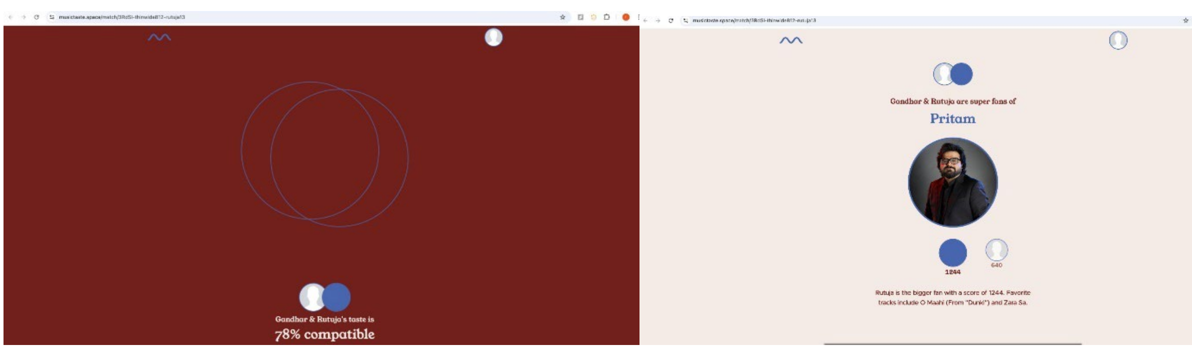


Figure 6: 'Musictaste' data comparison

While it generates playlists based on shared interests, it lacks visual representations of the resulting data. The platform relies heavily on text-based insights and offers minimal visualizations, making it less engaging for users seeking graphical interpretations. Although it includes interactivity, data comparison, and shared recommendations, it does not visually explain the underlying numbers or the logic behind the results.

Additionally, *Musictaste.space* lacks flexibility in playlist selection. Users cannot choose which playlists to compare, limiting the tool's applicability to scenarios beyond personal analysis. While it comes closest to aligning with the goals of our project, it falls short in offering diverse and meaningful visualizations, restricting its potential for broader comparative analyses.

1.3 PROJECT OBJECTIVES AND VALUE PROPOSITION

The existing works discussed above do not contain all the features we are looking for, namely:

- Compare any two playlists and visualize the similarities/differences in their musical attributes.
- Gather common traits and present them as visuals.
- Generate recommendations based on the visualized common traits.
- Present the newly formed playlist through visualizations.
- Visualize the musical features of the new playlist.

We want to give users the flexibility to decide the context of the comparative analysis, and not just rigidly compare two personal profiles to study personal tastes. If users want to study the change in musical preferences from five years ago to now, they can do so by comparing Spotify's 'Top Songs' playlists for specific years. Similarly, the tool can compare musical tastes between countries using Spotify-generated playlists for the most popular songs in each country. Another use case is comparing musical features for different years or decades to explain the general vibes of songs prevalent during those times, using the recommender to find new songs with similar vibes. It could also generate music that blends the styles of two artists. These are just a few examples of the use cases our project can handle.

The existing works also do not explore the artist network formed by collaborations between artists from the two playlists. We analyze this aspect to discover unexpected collaborations and uncover lesser-known tracks, which can lead to interesting insights for many use cases.

Lastly, the existing methods either use visualizations that lack clarity or rely heavily on textual reporting of trends. Our project focuses on creating meaningful, easily interpretable, and aesthetically appealing visualizations. We aim to ensure interactivity enhances the information provided by the plots, adding value rather than being included just for the sake of engagement.

2. DATA

Data gathering was carried out using the Spotify API, which provided a straightforward and efficient process. A pipeline was established to:

- Retrieve playlist hyperlinks.
- Extract playlist data from Spotify.
- Perform comparative analysis.
- Generate personalized recommendations.

The collected data includes the following columns:

- **Track_Name**: Song name
- **Track_ID**: Spotify ID for the song
- **URL**: Web URL for the son
- **Artist(s)**: All artists associated with the song
- **Genres**: Genres associated with the artists
- **Album**: Album that the song is part of
- **Song Popularity**: song popularity score (0-100), based on no. of streams
- **Artist Popularity**: artist popularity score (0-100), based on no. of streams
- **Duration**: Song runtime
- **Tempo**: Approx. beats per minute
- **Loudness**: How loud a song is (dB)
- **Key**: 0-11, one of the 12 notes on a musical scale
- **Mode**: Major or minor
- **Danceability**: A probabilistic measure of how danceable the song is
- **Energy**: A probabilistic measure of how energetic the song is
- **Speechiness**: A probabilistic measure of the presence of spoken words in the track
- **Acousticness**: A probabilistic measure of how acoustic a song is
- **Instrumentalness**: A probabilistic measure of the vocals in the song
- **Liveliness**: A probabilistic measure of the presence of a live audience
- **Valence**: A probabilistic measure of how positive the song is
- **Time_Signature**: The time signature (beats in a measure) of the song

3. VISUALIZATION METHODS

Before finalizing the visualizations, extensive trial and error was conducted to refine the presentation and design. However, some core design principles guided the development of the visuals:

a. Comparison of Single Continuous Variable

To compare and analyze individual continuous variables, density plots or overlaid histograms were considered ideal. These methods effectively highlight differences and overlaps between playlists. An alternative approach explored was using CCDFs or ECDFs, which provide cumulative insights into the data distribution. Each method offers unique perspectives, aiding in contrasting playlist characteristics.

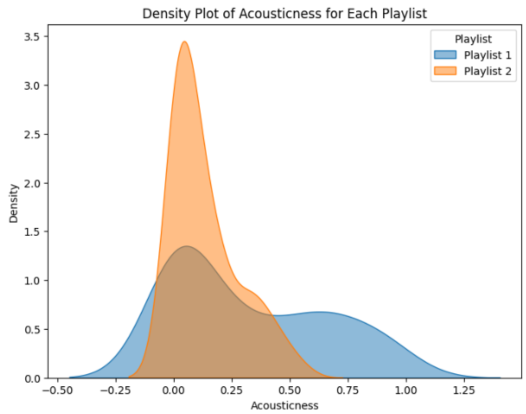


Figure 7: Test plot to compare a single continuous variable

b. Comparison of Multiple Continuous Variables

When analyzing multiple continuous variables simultaneously, scatterplots are an ideal choice. The X and Y axes represent two primary features being compared, while the size of the points can be adjusted based on a third variable. Color is used to either encode a fourth variable or differentiate between the playlists being compared. This approach is effective for contrasting two playlists or exploring the attributes of a newly generated playlist from recommendations. It provides a multidimensional perspective, allowing users to identify patterns, clusters, or trends across various musical attributes.

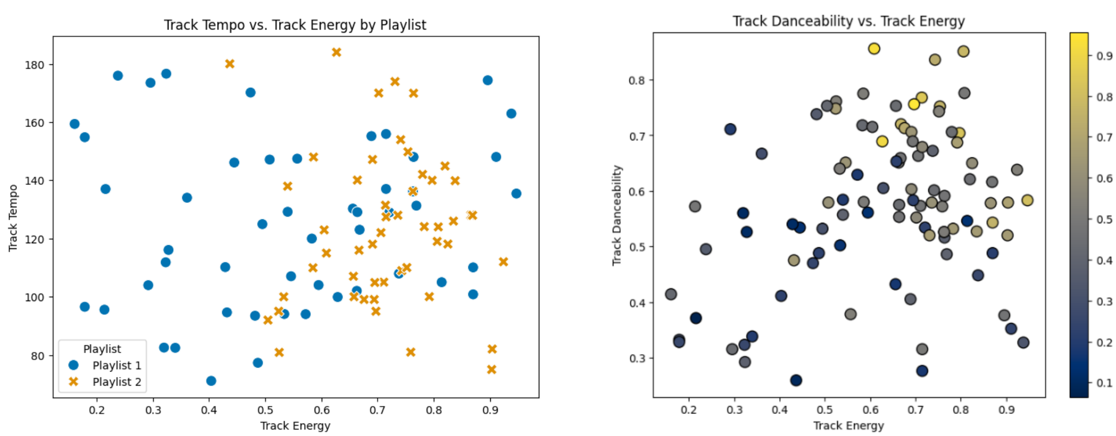


Figure 8: Test plots to compare multiple continuous variables together

c. Aggregated View of Continuous Variable

Another approach to analyze all significant continuous variables together is to use a spider plot or radar plot. This visualization aggregates the values of continuous variables and makes it easier to compare datasets based on the selected variables. The radial axes allow for a holistic view, highlighting differences and overlaps in multiple dimensions simultaneously. This method is particularly useful for comparing playlists based on multiple continuous attributes.

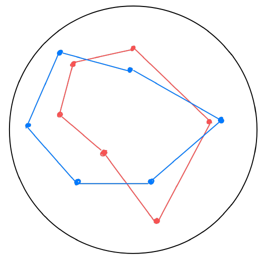


Figure 9: Rough sketch for proposed spider plot

d. Top 5 Genres/ Artists

When analyzing common artists or genres, both bar plots and bubble plots are suitable options. Bar plots provide simplicity and clarity, making it easy to compare and sort values by order. On the other hand, bubble plots allow for the incorporation of multiple visual encodings, such as size and color, to represent additional dimensions like popularity or total counts. Side-by-side bar plots can also be used to display individual top artists or genres from different playlists, offering a clear comparison of preferences.

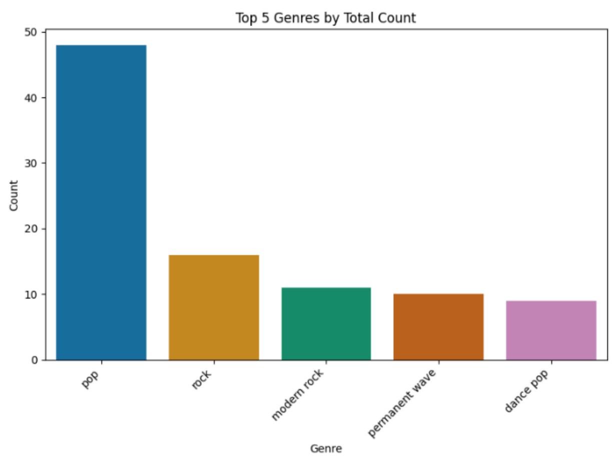


Figure 10: Test plot for top genres

e. Collaboration Network

A network map is ideal for visualizing collaborations between artists. Edges represent collaborative links, and their weights can encode additional metrics like collaboration frequency or popularity. Nodes incorporate multiple layers of information, such as size to represent artist popularity and color to indicate the playlist they belong to. Artists without collaborations are shown as isolated nodes, emphasizing their lack of connections.

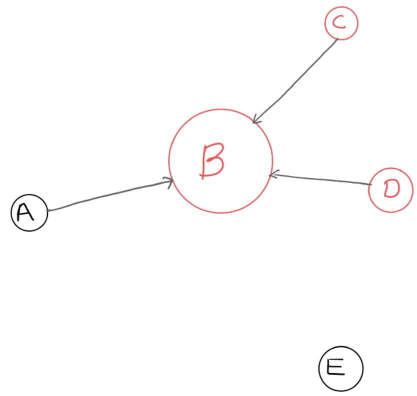


Figure 11: Sample sketch of the collaboration network

f. Recommendations Visualization

To display recommendations based on common artists or genres, a treemap is an effective choice. The hierarchical levels are structured as: Artist/Genre → Song
The size of each rectangle represents the popularity of the song, while the color encodes the artist or genre it belongs to.

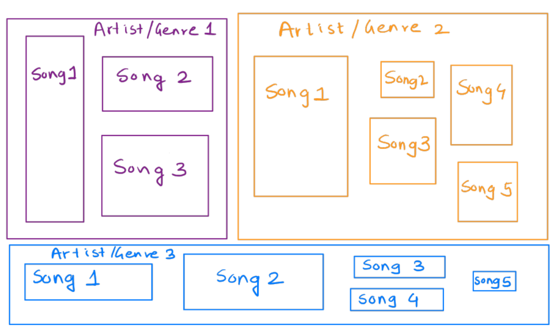


Figure 12: Sample sketch for treemap

The initial phase of plotting was done to make static graphs. The following results were obtained:

1. Top Artists by Song Frequency:

This static plot compares the top 5 most frequent artists from both playlists. The bar chart on the left displays the distribution of artist frequencies in Playlist 1, while the chart on the right does the same for Playlist 2. This visualization provides a quick insight into the dominant artists across the two playlists, highlighting overlaps or differences in preferences. (Experimental) (Alternative – Bubble Chart)

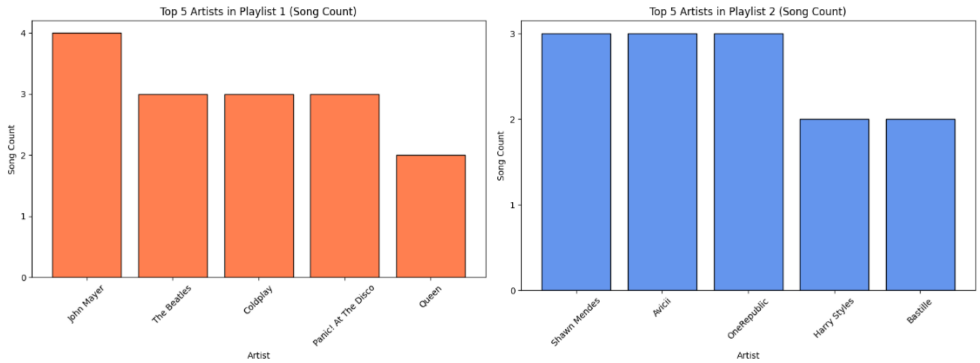


Figure 13: Top artists in playlists by song frequency

2. Top Genres by Frequency

This plot compares the top 5 music genres and their frequencies across Playlist 1 and Playlist 2. Each bar represents a genre, with its height indicating frequency. The left subplot shows genre distribution in Playlist 1, while the right focuses on Playlist 2. (Experimental) (Alternative – Bubble Chart)

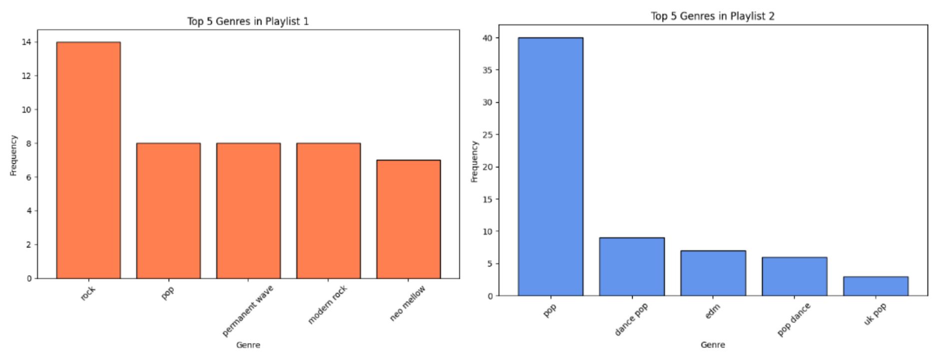


Figure 14: Top genres by frequency

3. Popularity Distribution Comparison

The plot displays the popularity distribution of songs in Playlist 1 and Playlist 2 using histograms and KDE curves. Playlist 1 (left subplot) shows a balanced spread of popularity, while Playlist 2 (right subplot) emphasizes higher popularity levels.

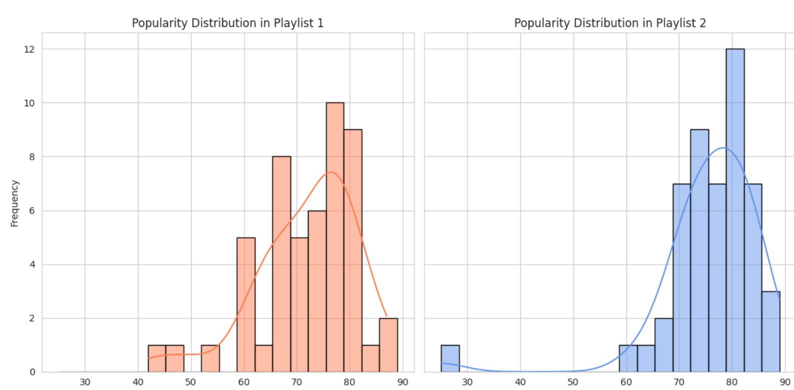


Figure 15: KDE with histogram for popularity distribution

4. Popularity Vs Energy Scatterplot

This scatterplot compares song popularity and energy levels for Playlist 1 and Playlist 2. Songs from Playlist 1 show a wider spread across the energy axis, while Playlist 2 focuses more on high-energy and popular tracks. The plot visually highlights the distinct preferences in energy levels and popularity between the two playlists.

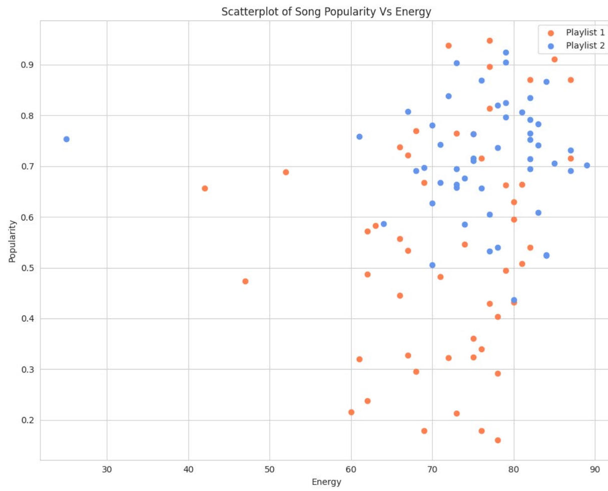


Figure 16: Scatterplot of multiple variables for the two playlists

5. Song Duration Distribution

This histogram highlights the distribution of song durations across Playlist 1 and Playlist 2. The x-axis denotes the duration in seconds, while the y-axis indicates the frequency of songs within each duration range. This comparison illustrates differing preferences for song lengths in the two playlists.

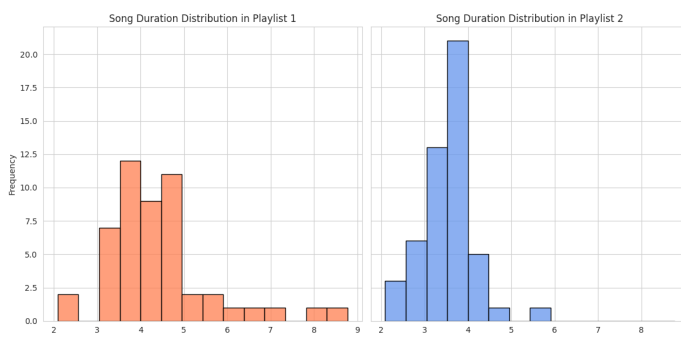


Figure 17: Histogram distribution to show song durations in playlists

6. Parallel Coordinate Plots

These plots visualize the relationship between various song attributes like Danceability, Energy, Loudness, Speechiness, and Acousticness for Playlist 1 and Playlist 2. The lines represent individual songs, and their color intensity reflects popularity allowing for a visual assessment of feature patterns in relation to popularity within each playlist. (Experimental) (Alternative - Spider plot)

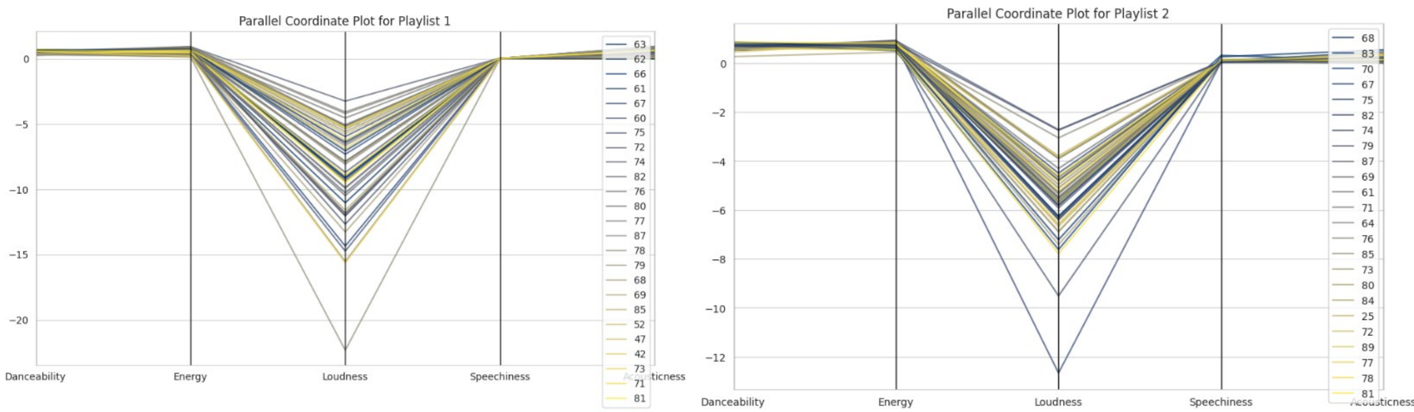
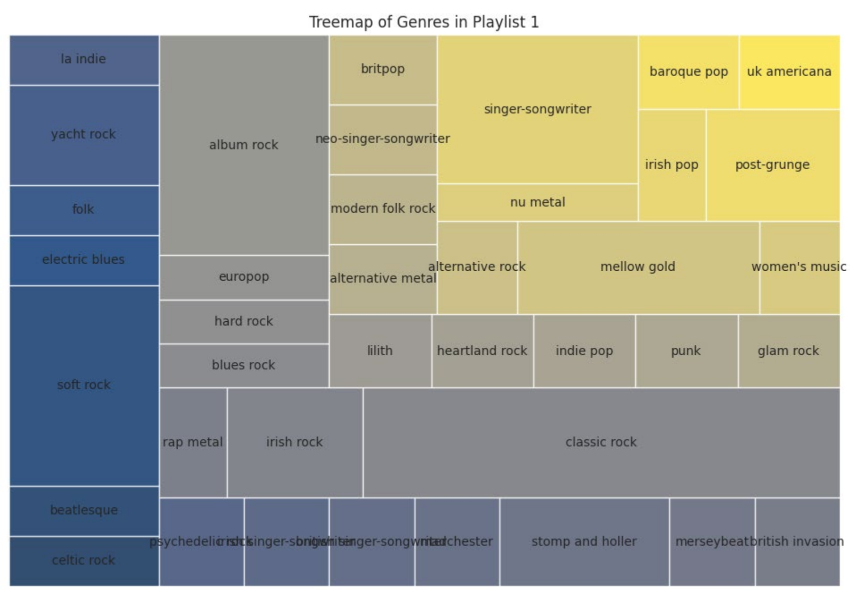


Figure 18: Parallel coordinate plots for multiple variables

7. Treemaps

The treemaps visualize the genre diversity of two playlists, with each rectangle representing a genre and its size proportional to the number of artists associated with that genre. These plots effectively highlight the distinct genre distributions and focus areas in the playlists, aiding in understanding their musical inclinations.



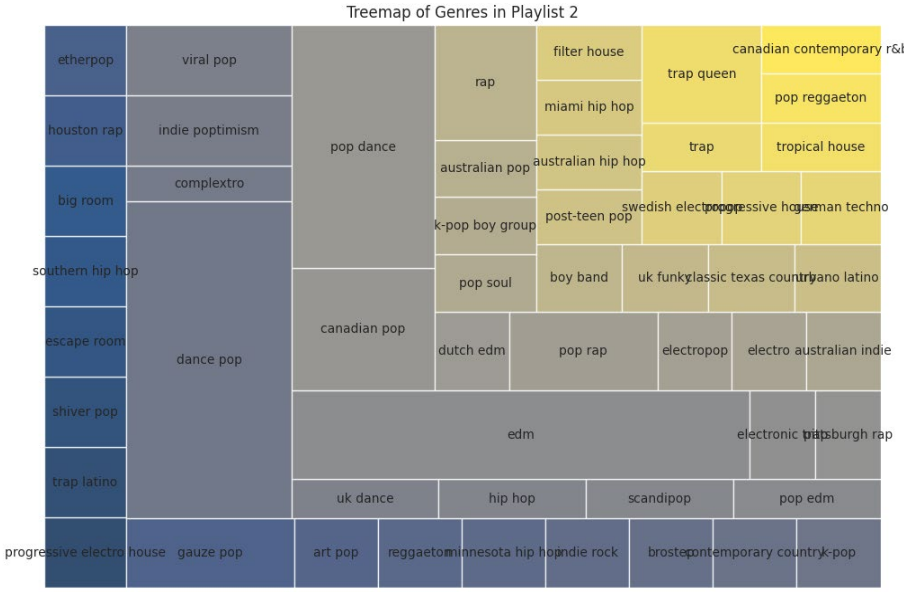


Figure 19: Treemaps for genre

8. Wordclouds

The word clouds represent the most common music genres in Playlist 1 and Playlist 2, with word size corresponding to genre frequency. Playlist 1 prominently features genres like "classic rock" and "singer-songwriter," while Playlist 2 highlights "pop," "dance," and "edm" as dominant genres. This visual approach quickly identifies the genre focus and differences in musical inclinations between the two playlists. (Experimental)

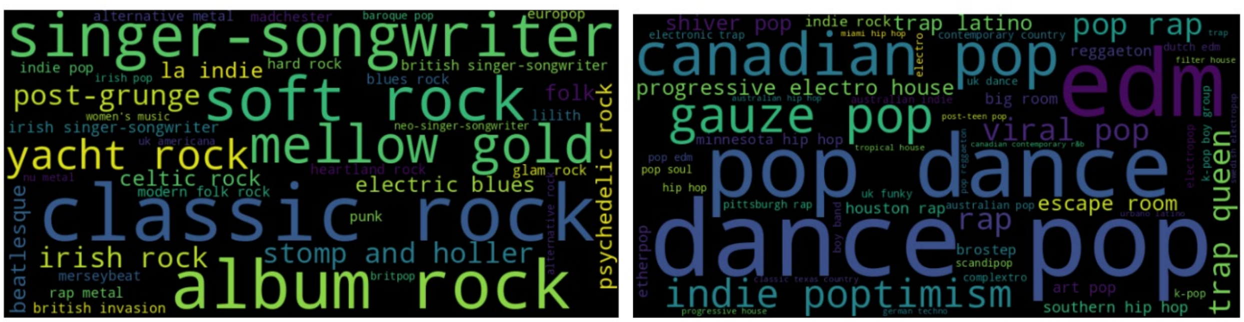


Figure 20: Word clouds for genres

9. Radar Chart

The radar chart illustrates the mean values of musical attributes between Playlist 1 and Playlist 2. Playlist 2 exhibits higher Energy, Danceability, and Valence, indicating a more lively and engaging vibe, while Playlist 1 scores higher in Acousticness, reflecting a calmer tone. Instrumentalness and Speechiness are nearly identical, suggesting similarity in these aspects across both playlists.

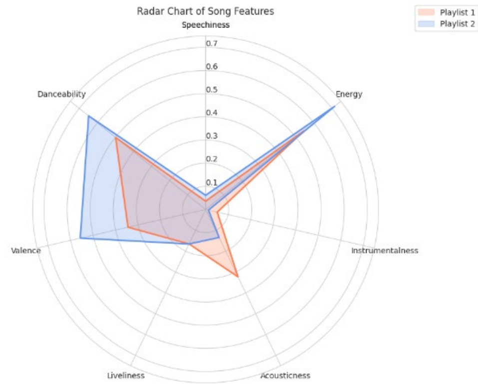


Figure 21: Radar chart for various musical attributes

The final visualizations and the reasons for choosing the visualizations is explained in the results.

4. RESULTS AND DISCUSSION

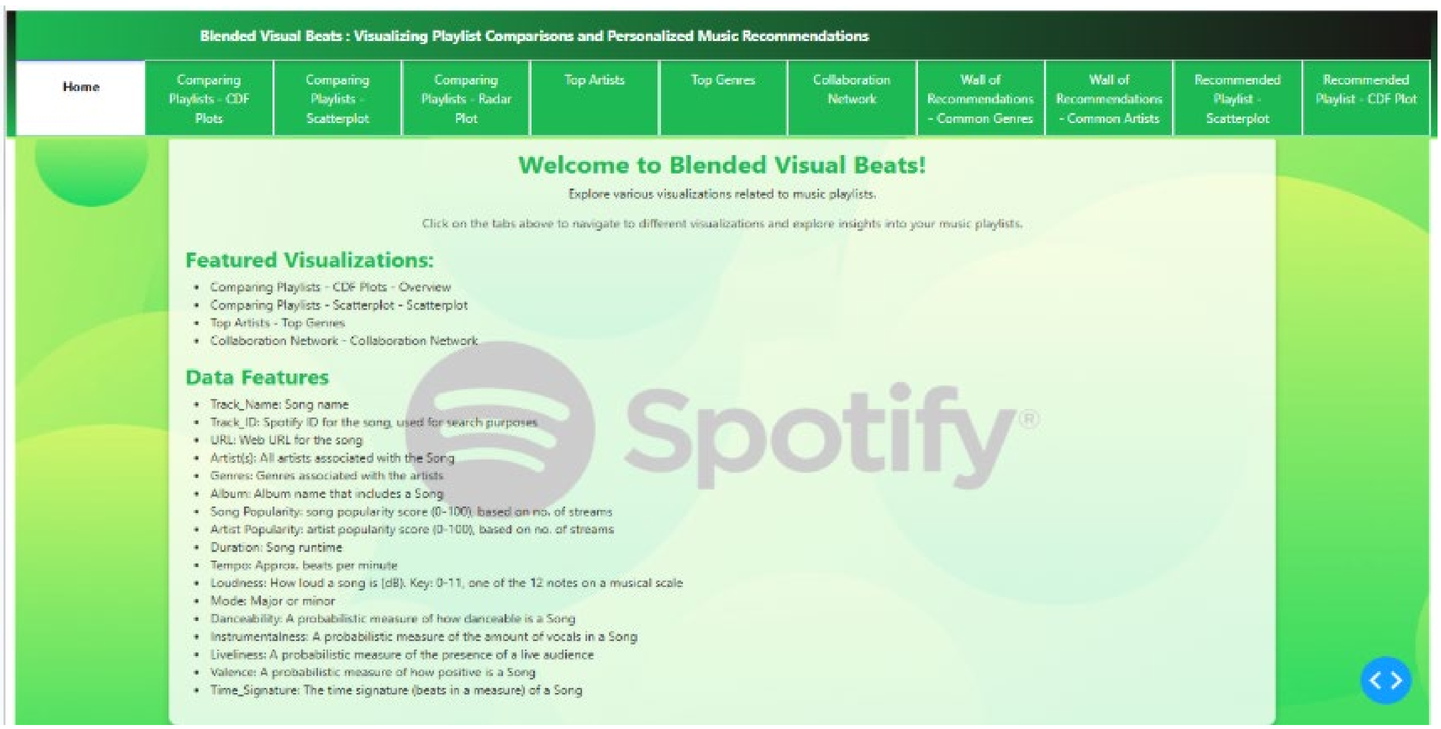


Figure 22: Home Page

1. We employed a cumulative distribution plot to compare various continuous numerical variables in the data. The CDF creates a stepwise function that intuitively displays the cumulative distribution by visually connecting each data point through a continuous, monotonically increasing line. This connectivity and continuity, along with our natural tendency for perceptual closure (filling in gaps to perceive a complete shape), enhance the plot's readability according to Gestalt principles.

However, the uniform stepping can introduce visual noise, potentially disrupting figure-ground separation and obscuring exact distribution shapes. Despite this limitation, the cumulative distribution plot achieves a balance between visual coherence and distribution resolution, making it an effective tool for our comparative analysis.

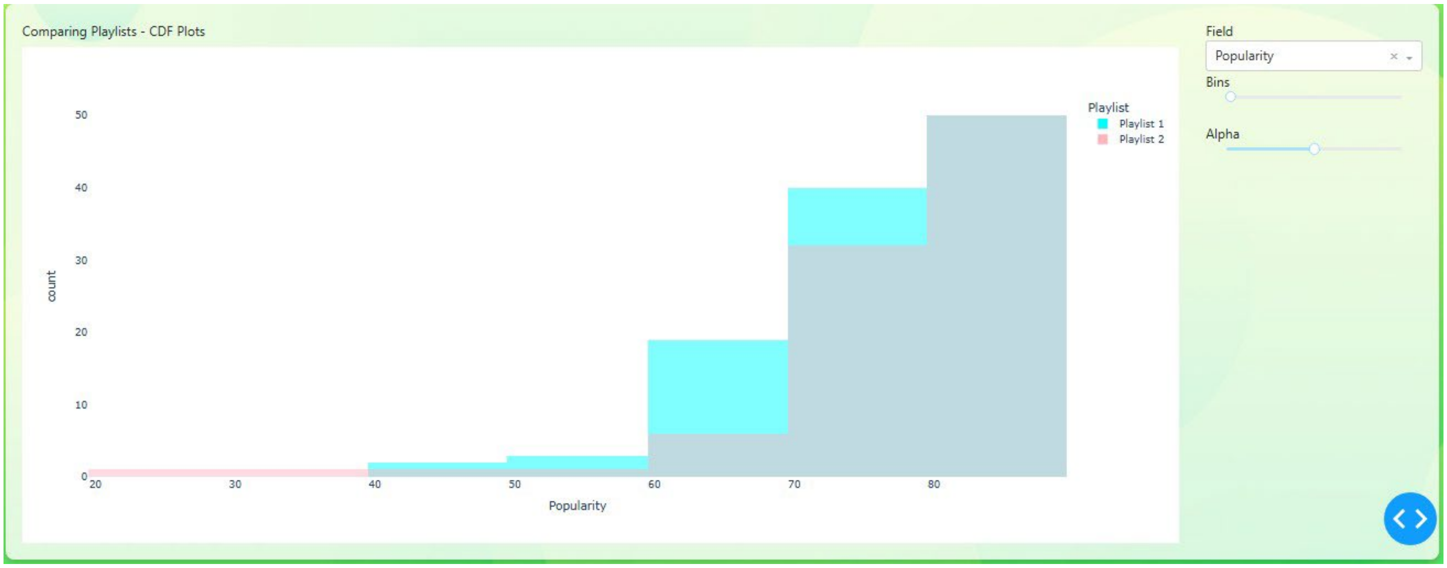


Figure 23: Plot 1 – Comparing Playlists using CDF

Insights: Playlist 2 has a higher density of popular tracks concentrated in the 70-80 range, while Playlist 1 shows a more gradual increase, suggesting a broader distribution of popularity values across its tracks.

2. A scatterplot was used to compare songs from the two playlists, enabling users to dynamically choose variables for the X-axis, Y-axis, and bubble size. By incorporating multiple visual encodings such as position, size, and color, the scatterplot intuitively conveys multidimensional data patterns. Gestalt principles play a significant role in this visualization. Similarity groups elements by shared properties like color or size, while proximity binds related elements spatially, indicating correlations. Continuity fosters trends across multiple dimensions through consistent encodings, allowing patterns to emerge naturally from the data.

Despite these strengths, overloading the visual field can disrupt figure-ground separation and grouping continuity, potentially leading to cognitive overload. To prevent this, we carefully layered visual variables upon a foundational base encoding, ensuring clarity and coherence. By striking a balance between unification and complexity, the scatterplot effectively unlocks multidimensional insights while maintaining readability and interpretability.

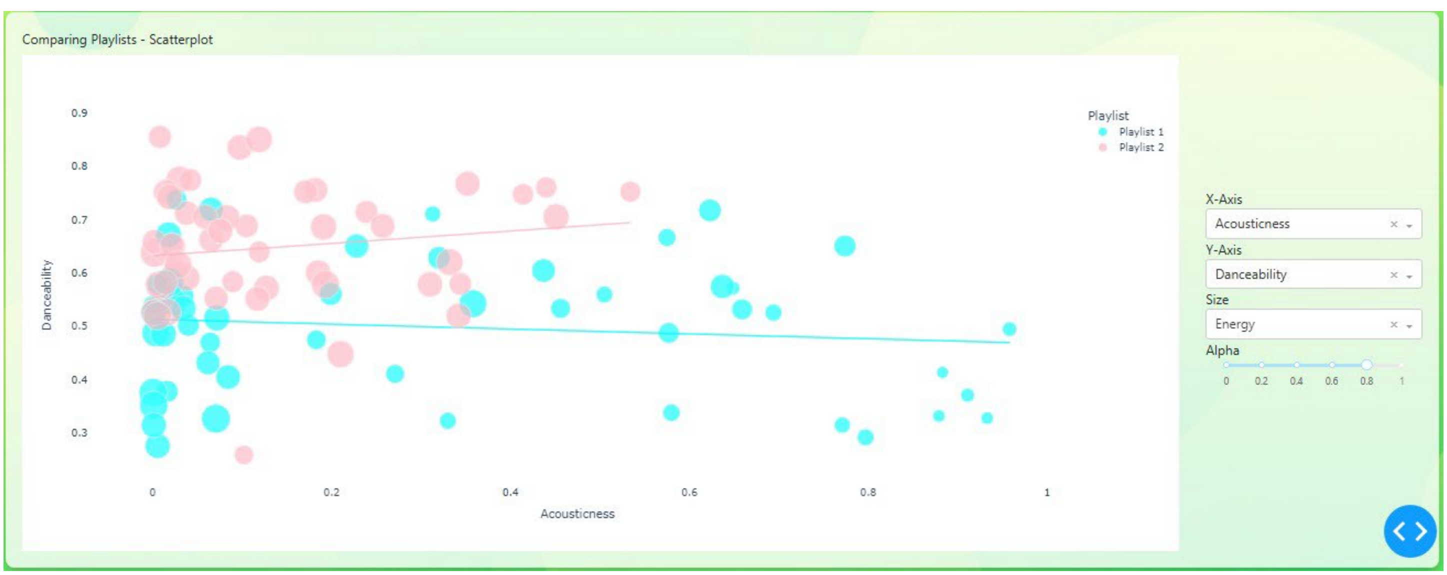


Figure 24: Plot 2 – Comparing Playlists using Scatterplot

Insights: Playlist 1 includes tracks with a wide range of acousticness but shows a tendency toward lower danceability values overall. Whereas, playlist 2, in contrast, has a tighter cluster of tracks with moderate to high danceability, and the tracks cluster around low acousticness values.

3. A radar plot was used to compare the average values of various musical attributes between the two playlists. This visualization highlights multivariate patterns by leveraging the similarity and continuity of visual elements. The proximity of adjacent and enclosed axes naturally groups factors, while the closed polygons take advantage of perceptual closure, making the data appear as cohesive shapes. Smooth contours enhance visual continuity, making patterns more intuitive compared to jagged or sparse lines. Additionally, segment color coding improves category distinction, and the alpha control allows users to adjust the transparency of the plot, further clarifying overlapping areas and enabling better visual differentiation.

However, radar plots can become visually overwhelming when interleaving or crossing lines disrupt continuity, potentially obscuring trends. To address this, we focused on maintaining simplicity through careful selection of axes, smoothing techniques, and interactive controls like the alpha slider. By reducing visual clutter and emphasizing essential data dimensions, the radar plot effectively maps multidimensional data into an intuitive format, balancing insight and readability.

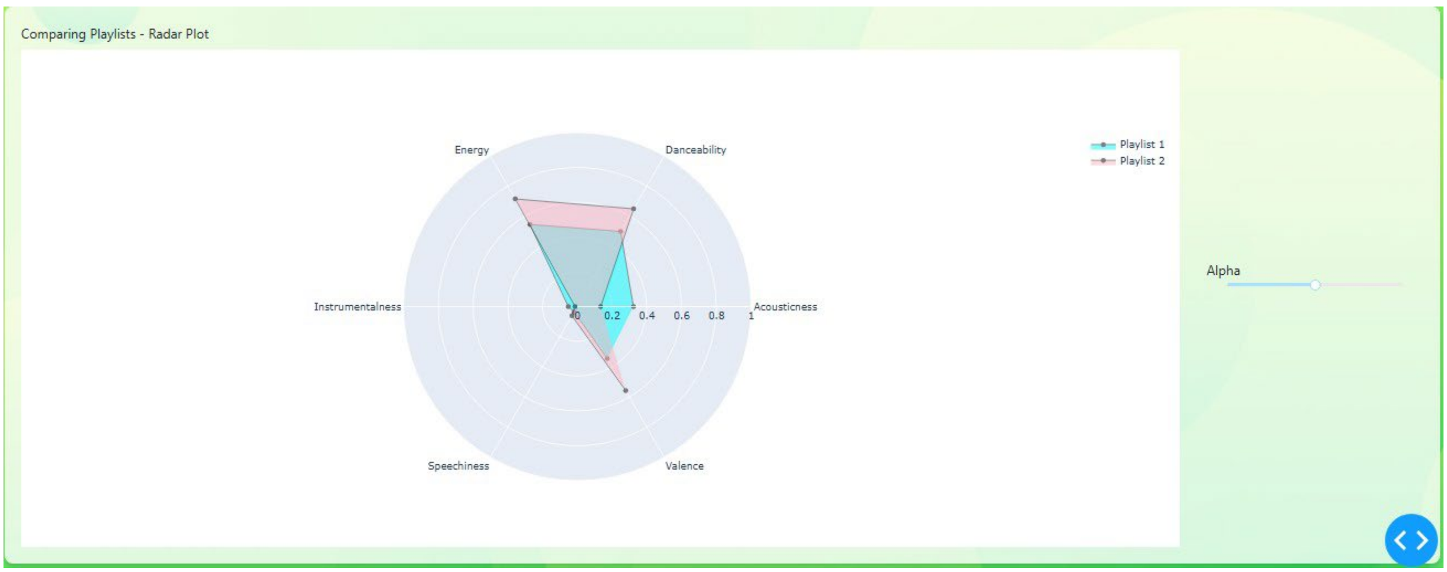


Figure 25: Plot 3 – Comparing Playlists using Radarplot

Insights: Playlist 2 has higher values for energy, danceability, and valence, indicating a preference for upbeat, rhythmic, and emotionally positive tracks. Playlist 1 shows a significantly higher acousticness, highlighting a preference for softer and more acoustic sounds. Instrumentalness is nearly the same for both playlists, suggesting similar levels of instrumental components in the tracks.

4. A bubble chart was created to visualize the top artists across both playlists. The chart includes a slider to control the number of artists displayed at once, allowing users to explore the data interactively. Bubble colors indicate the playlist each artist belongs to, while bubble size represents the total number of songs the artist has in both playlists combined. Users can switch between viewing top artists based on popularity or Spotify follower count, offering flexibility in the analysis.

This visualization uses intuitive visual cues to represent multidimensional data. Similar bubble sizes group related elements, while spatial proximity links entities, making patterns emerge naturally. However, excessive variation in bubble scaling can distort trends and disrupt readability. To address this, we maintained simplicity through consistent sizing and spacing, which ensured that the underlying data distributions remain clear and uncluttered. The result is a visually engaging and insightful representation of artist prominence across the playlists.

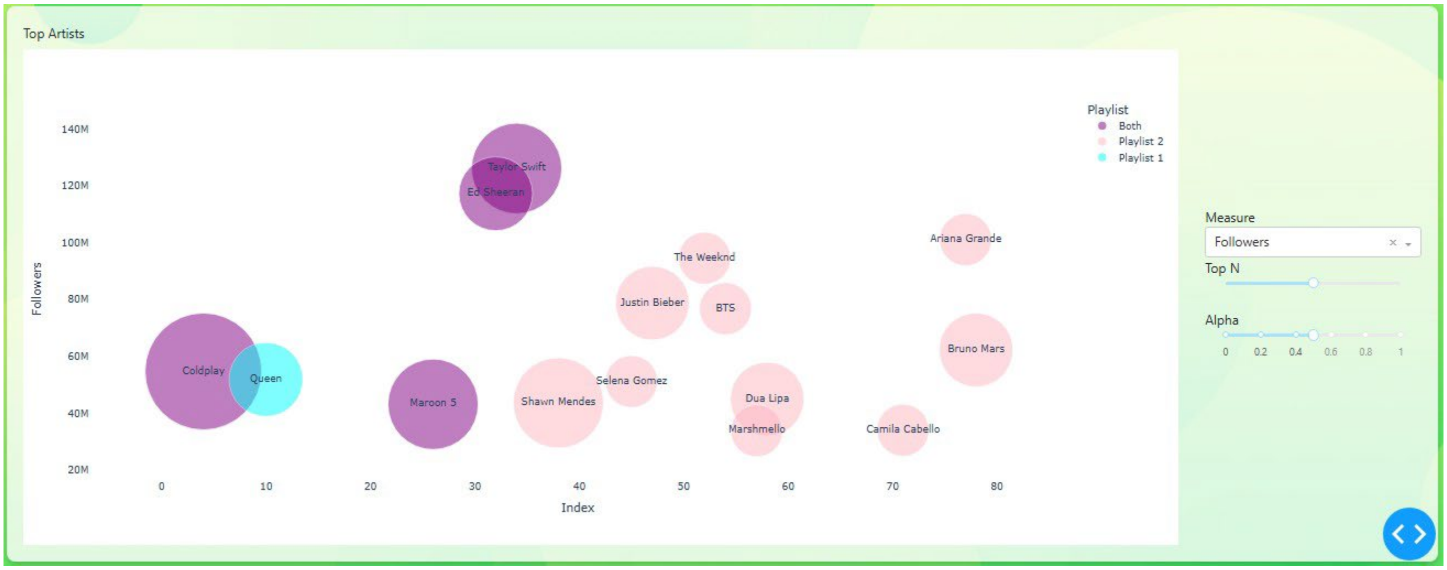


Figure 26: Plot 4 – Top Artists

Insights: Coldplay has the largest bubble, indicating it is the most followed artist across both playlists. It dominates with a strong presence in both Playlist 1 and Playlist 2. Taylor Swift and Ed Sheeran follow as the next most popular shared artists across both playlists. Queen stands out in Playlist 1, showcasing its popularity among tracks exclusive to that playlist. In Playlist 2, artists like Shawn Mendes, Justin Bieber, and Bruno Mars show significant follower counts, highlighting their importance in this playlist.

5. A bubble chart was used to visualize the top genres by count across both playlists. The chart includes a slider to adjust the number of genres displayed, providing users with flexibility to explore the data interactively. The bubble colors indicate which playlist the genre is associated with, while the bubble size represents the total number of songs of that genre across both playlists combined.

The choice of a bubble chart here aligns with its ability to intuitively convey multidimensional data, similar to its application in the top artists plot. By using size and color as visual encodings, the chart effectively highlights both the prominence and playlist association of each genre. This approach ensures clarity and ease of interpretation, allowing users to uncover trends and make comparisons effortlessly. The addition of the alpha slider enables users to adjust transparency, further enhancing the readability of overlapping elements and improving visual clarity.

Figure 27: Plot 5 – Top Genres

Insights: Pop dominates as the most represented genre, with the largest bubble indicating its strong presence across both playlists. Rock also has a significant presence, in both the playlists. Playlist 1 emphasizes softer genres like soft rock and classic rock, while Playlist 2 includes upbeat genres like pop dance, and EDM. Shared genres like modern rock and permanent wave show overlaps in taste, reflecting mutual preferences in music styles.

6. A network map was created to visualize the relationships and collaborations between artists in the two playlists. The nodes represent artists, and their colors indicate the playlist they belong to. Node size is determined by degree centrality, reflecting the number of connections (collaborations) an artist has. Edges between nodes represent collaborations, while unconnected nodes are displayed with a default size of 1 to indicate their presence in the playlists.

Network maps are highly effective in showing connections and relationships, leveraging Gestalt principles like continuity, closure, and connectedness. These principles help users perceive clusters and communities of artists naturally, making patterns of collaboration easily noticeable. Although edge weighting based on collaboration popularity was considered, it was excluded as the results did not meet expectations. Here, edges remain unweighted, focusing solely on the existence of connections.

Unconnected nodes were retained to ensure all artists in the combined playlists are represented. This approach ensures a comprehensive view of artist relationships, highlighting both collaborative networks and isolated artists, while maintaining clarity and interpretability through interactive features like the alpha slider for node transparency.

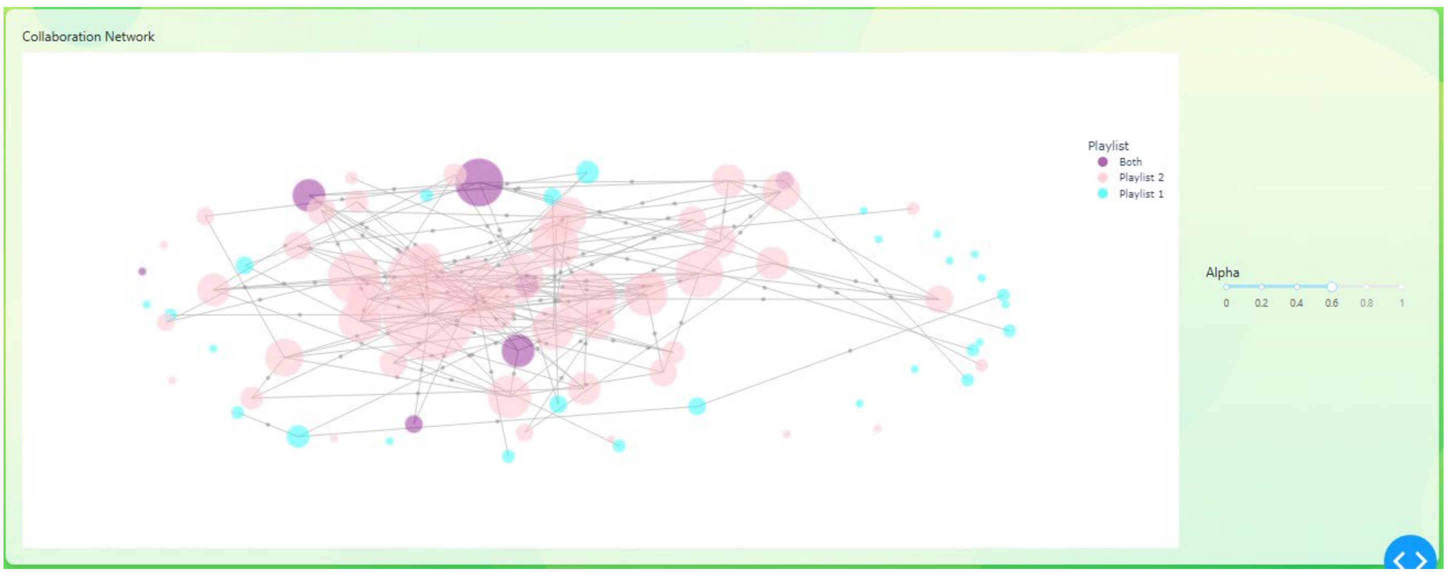


Figure 28: Plot 6 – Collaboration Network

Insights: The collaboration network reveals that Playlist 2 features artists with the most collaborations, forming denser clusters in the network. In contrast, Playlist 1 includes artists with fewer collaborations, resulting in more sparse and isolated nodes. Shared artists bridge the two playlists, emphasizing their mutual popularity and collaborative influence.

Building on the common traits and insights identified through the comparison of the two playlists, we moved forward to extract **personalized song recommendations**. These recommendations were carefully curated by leveraging the shared characteristics and unique features of both playlists.

7. A treemap was used to display the top new recommended songs based on common genres between the two playlists. Users can adjust the number of songs shown using a slider and select the measure (e.g., popularity, energy) to control the size and color of the treemap boxes. The visualization dynamically updates to reflect the chosen parameters, providing an interactive and personalized experience.

Treemaps utilize nested rectangles to represent hierarchical data effectively. The Gestalt principles of similarity and closure make it easy to group related items by color and identify enclosed subgroups intuitively. However, discontinuities caused by uneven rectangles or imbalanced hierarchy can sometimes obscure understanding. To address this, we focused on maintaining well-shaped containment, which guides the viewer's eye across levels and enhances pattern comprehension.

This treemap offers a concise and visually engaging way to explore recommendations, helping users discover new songs that align with their preferences while balancing visual complexity and interpretability.

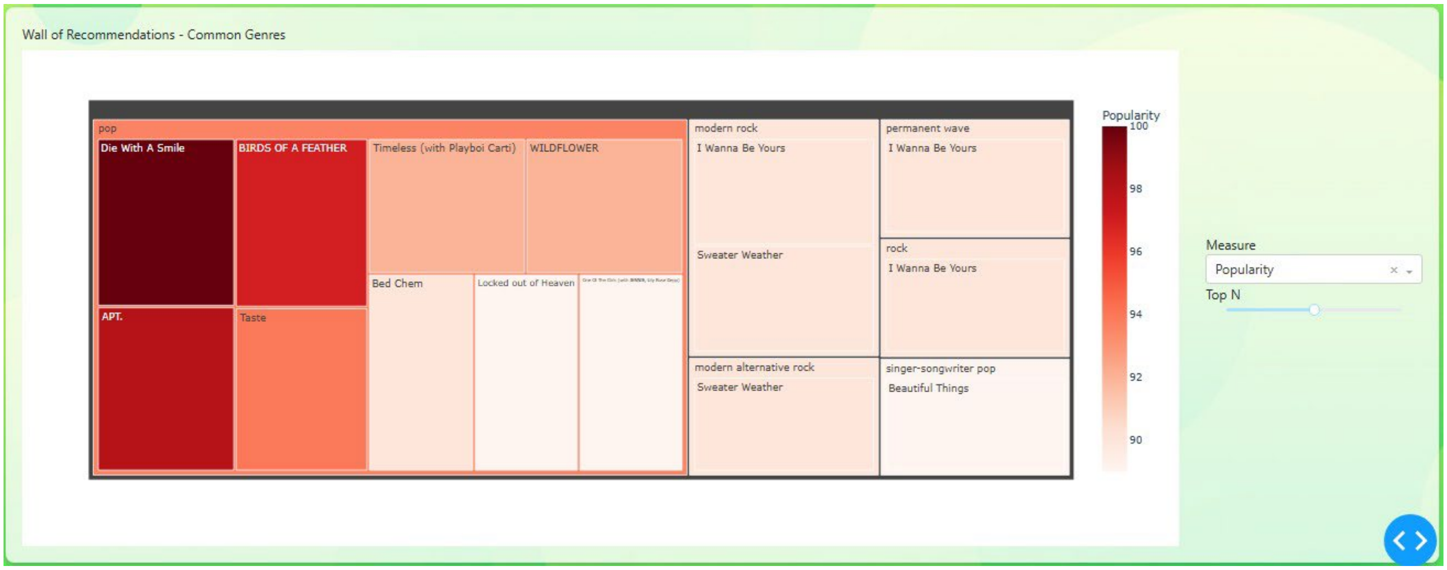


Figure 29: Plot 7 – Wall of Recommendations for Common Genres

Insights: The treemap highlights top recommended songs grouped by common genres, with pop dominating the recommendations as the most popular genre. Genres like modern rock and permanent wave also contribute significantly to the recommendations. Larger and darker rectangles indicate songs with higher popularity scores, while smaller rectangles show less prominent tracks. This visualization emphasizes genre-based song discovery, catering to shared listener preferences.

8. This treemap visualizes the top new recommended songs from common artists across both playlists. Users can adjust the number of songs displayed using a slider and select a measure (e.g., popularity, energy) to control the size and color of the rectangles. The rationale for using a treemap remains the same as the previous plot: it effectively organizes hierarchical data, leveraging similarity and closure for intuitive groupings. This treemap provides a visually appealing and interactive way to explore song recommendations based on shared artists.

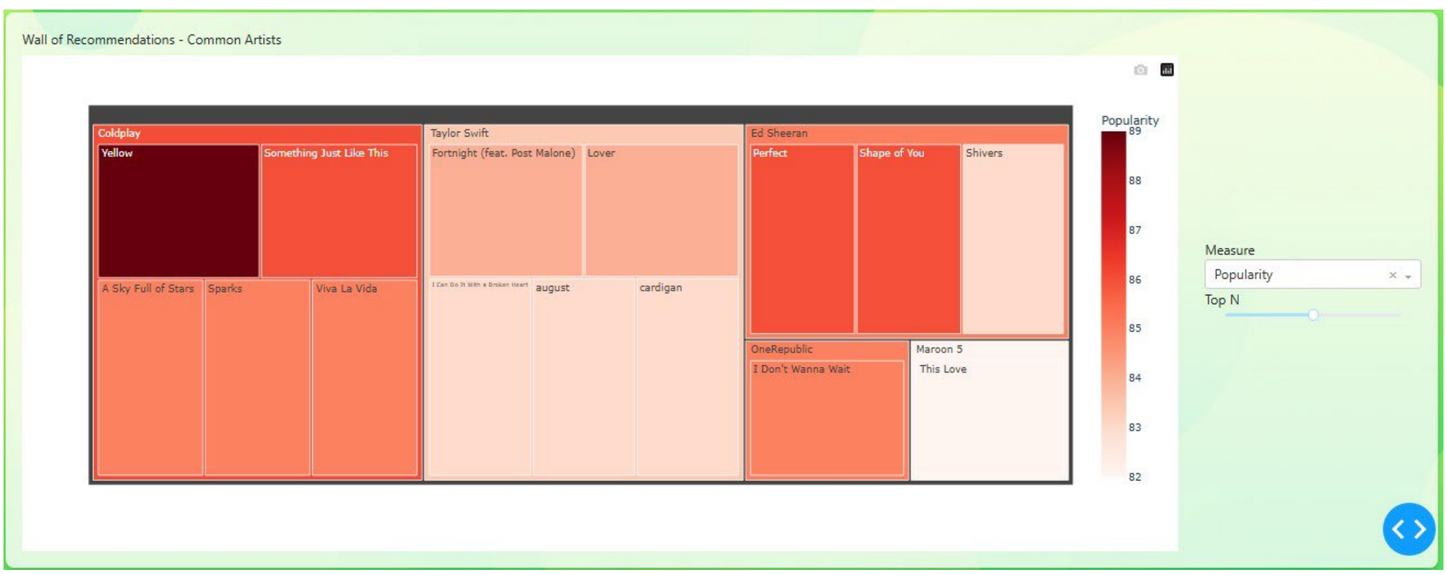


Figure 30: Plot 8 – Wall of Recommendations for Common Artists

Insights: This treemap displays top recommended songs grouped by common artists. Coldplay, Taylor Swift, and Ed Sheeran dominate the recommendations with their most popular tracks, such as Yellow, Lover, and Shape of You. Larger and darker rectangles indicate higher popularity, while smaller rectangles represent less prominent recommendations.

9. A scatterplot that visualizes the songs from the new playlist, allowing users to dynamically select variables for the X-axis, Y-axis, bubble size, and bubble color. This flexibility helps users explore relationships between different musical attributes, such as popularity, tempo, danceability, and energy.

The scatterplot was chosen for its ability to efficiently represent multidimensional data through multiple visual encodings in a single plot. It also provides a way to estimate trends and patterns, making it a powerful tool for analyzing the new playlist. Interactive controls like the alpha slider further enhance clarity by adjusting the transparency of overlapping points.

Figure 31: Plot 9 - Scatterplot for Recommended Playlist

Insights: In this plot, we observe that songs with higher tempo values tend to cluster in the mid-range of popularity. Additionally, the bubble size, representing danceability, and the color gradient, showing energy, highlight tracks that are both upbeat and popular, suggesting potential top recommendations for active or high-energy playlists. These insights allow users to identify standout tracks and trends within the recommended playlist.

10. A cumulative distribution plot (CDF) is used to study the distribution of various musical attributes within the new playlist. Users can adjust the number of bins and opacity using interactive sliders, ensuring flexibility and clarity in visualization. This plot highlights the cumulative distribution of data points, allowing for an intuitive understanding of how specific attributes, such as popularity or energy, are distributed across the playlist.

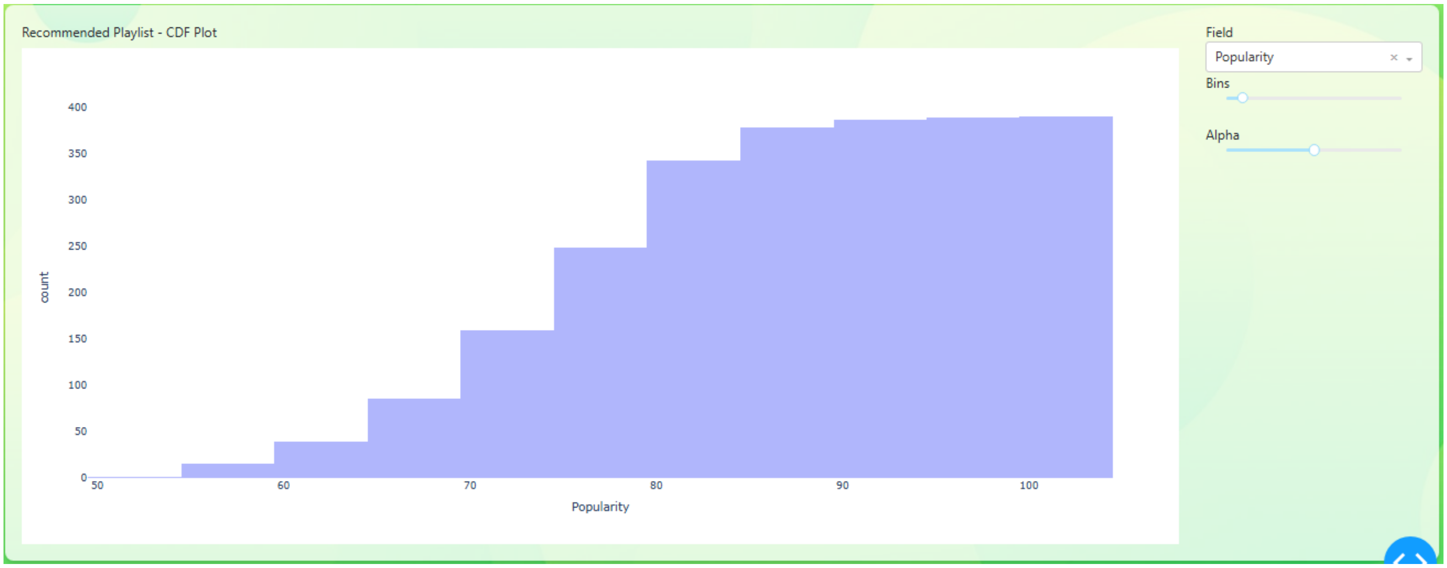


Figure 32: Plot 10 – CDF Plot of Recommended Playlist

Insights: The CDF plot enables users to identify trends, such as the proportion of songs within a specific popularity range. For example, in this demonstration, most songs in the new playlist have popularity values above 70, reflecting a strong preference for widely appreciated tracks.

5. CONCLUSION

In this project, we aimed to create visualizations for Spotify data that effectively allow the comparison of two playlists while also providing an intuitive way to understand how shared recommendations are generated. Starting with static plots we transitioned to a dynamic dashboard.

The final visualizations can either be studied together to tell a cohesive story or used individually for specific analytical purposes. Plots like scatterplots are versatile and can serve multiple objectives, such as identifying songs based on mood or comparing trends. The collaboration network and dynamic scatterplots proved particularly insightful for modeling musical data.

To ensure accessibility, the visualizations are colorblind-friendly, making them inclusive for a diverse audience. While some redundancy exists in the plots, this was a deliberate choice to provide users with flexible ways to explore their data.

Overall, the project meets its motivation by offering an easy-to-interpret, interactive view of Spotify playlist data.

6. FUTURE WORK

While the project fulfills its objectives, there are several areas for improvement that can be considered for future work:

- Streamlining Scripts: Integrating the scripts into a single workflow will simplify the process, allowing users to input playlist URLs directly.
- Enhanced Visualizations: Exploring more complex and advanced visualizations using sophisticated tools could elevate the analytical depth of the project.
- Multiple Playlist Comparisons: Adding functionality to compare more than two playlists and enabling user selection among them would increase flexibility.
- Seamless Integration: Tighter integration with Spotify and its API will allow playlists to be fetched directly and facilitate effortless collaborations with friends.
- Expanded Use Cases: Incorporating features for analyzing playlists beyond individual users, such as trends across different regions or time periods, would broaden the tool's applicability.

By addressing these enhancements, the project can evolve into a powerful tool ready for public release.

7. REFERENCES

[1] "Spotify — About Spotify," Spotify, Oct. 24, 2023.
https://newsroom.spotify.com/company-info/
[2] Stacy.Goldrick@groupsjr.com, "How Spotify's Newest Personalized Experience, Blend, Creates a Playlist for You and Your Bestie — Spotify," Spotify, Aug. 29, 2023.
https://newsroom.spotify.com/2021-08-31/how-spotifys-newest-personalized-experience-blend-creates-a-playlist-for-you-and-your-bestie/
[3] Herlocker, Jonathan L., et al. "Evaluating collaborative filtering recommender systems." ACM Transactions on Information Systems (TOIS) 22.1 (2004): 5-53.
https://grouplens.org/site-content/uploads/evaluating-TOIS-20041.pdf
[4] "Stats for Spotify."
https://www.statsforspotify.com/
[5] "My Spotify Pie - Top artists - Top genres," My Spotify Pie - Top Artists - Top Genres.
https://huangdarren1106.github.io/spotify-pie
[6] "Obscurify Music."
https://www.obscurifymusic.com/home
[7] "Compare your Spotify music taste! - Musictaste.space."
https://musictaste.space/