mayo, 2024

Análisis visual de datos de Spotify
Analítica Visual

Alberto García Martín

Máster en Sistemas Inteligentes
Universidad de Salamanca

Índice

1. Introducción

Este informe tiene como objetivo recoger el trabajo realizado para la asignatura de Analítica Visual del Máster en Sistemas Inteligentes de la Universidad de Salamanca. En este trabajo se ha realizado un análisis visual de un conjunto de datos de Spotify, con la finalidad de obtener información relevante sobre las canciones y artistas más populares en la plataforma.

El dominio de la música es un campo con un gran potencial para la visualización de datos, ya que permite representar de forma gráfica aspectos como la popularidad de las canciones, la evolución de los géneros musicales o las preferencias de los usuarios. Por ello, se ha considerado interesante realizar un análisis visual sobre este dominio a partir del conjunto de datos MusicOSet [5], que contiene información sobre miles de canciones y artistas, extraída de la plataforma Spotify.

Para realizar este análisis se ha utilizado principalmente D3 [1], una biblioteca de JavaScript que permite la creación de visualizaciones interactivas en la web. Aunque también se han explorado otras alternativas como Vega-Lite [4] y Observable Plot.

En los siguientes apartados se describen los datos utilizados, las visualizaciones realizadas, incluyendo una descripción de las decisiones de diseño tomadas, y las conclusiones obtenidas a partir del análisis visual de los datos.

2. Descripción de los datos

Para la realización de este trabajo, se ha buscado un conjunto de datos relacionado con el dominio de la música que contuviera información amplia y relevante para posteriormente poder realizar un adecuado análisis visual. Con este objetivo, el conjunto de datos seleccionado tenía que cumplir con una serie de requisitos, como que fuera de un tamaño moderado, ni demasiado pequeño ni demasiado grande, que contuviera información variada y que fuera accesible y fácil de trabajar. Además esta información debía ir más allá de una simple lista de canciones o artistas, incluyendo datos como género musical, popularidad, características musicales, etc.

Tras una búsqueda de las fuentes de datos disponibles, se ha seleccionado el conjunto de datos MusicOSet de Mariana et al. [5], al ser un conjunto de datos de acceso abierto que contiene información de artistas, canciones y álbumes, basado en su popularidad. Los datos de este conjunto fueron extraídos en 2019 de la plataforma Spotify, y contiene información sobre más de 20.000 canciones y 11.000 artistas.

Este conjunto de datos se divide en tres categorías principales: popularidad de canciones y artistas, información de metadatos, y características musicales. Cada una de estas categorías contiene diferentes tablas relacionales con la información correspondiente. En la figura 1 se muestra el esquema de las tablas del conjunto de datos MusicOSet y la relación entre ellas. En la tabla 1 se muestra un resumen del número de registros de cada uno de los elementos principales del conjunto de datos.

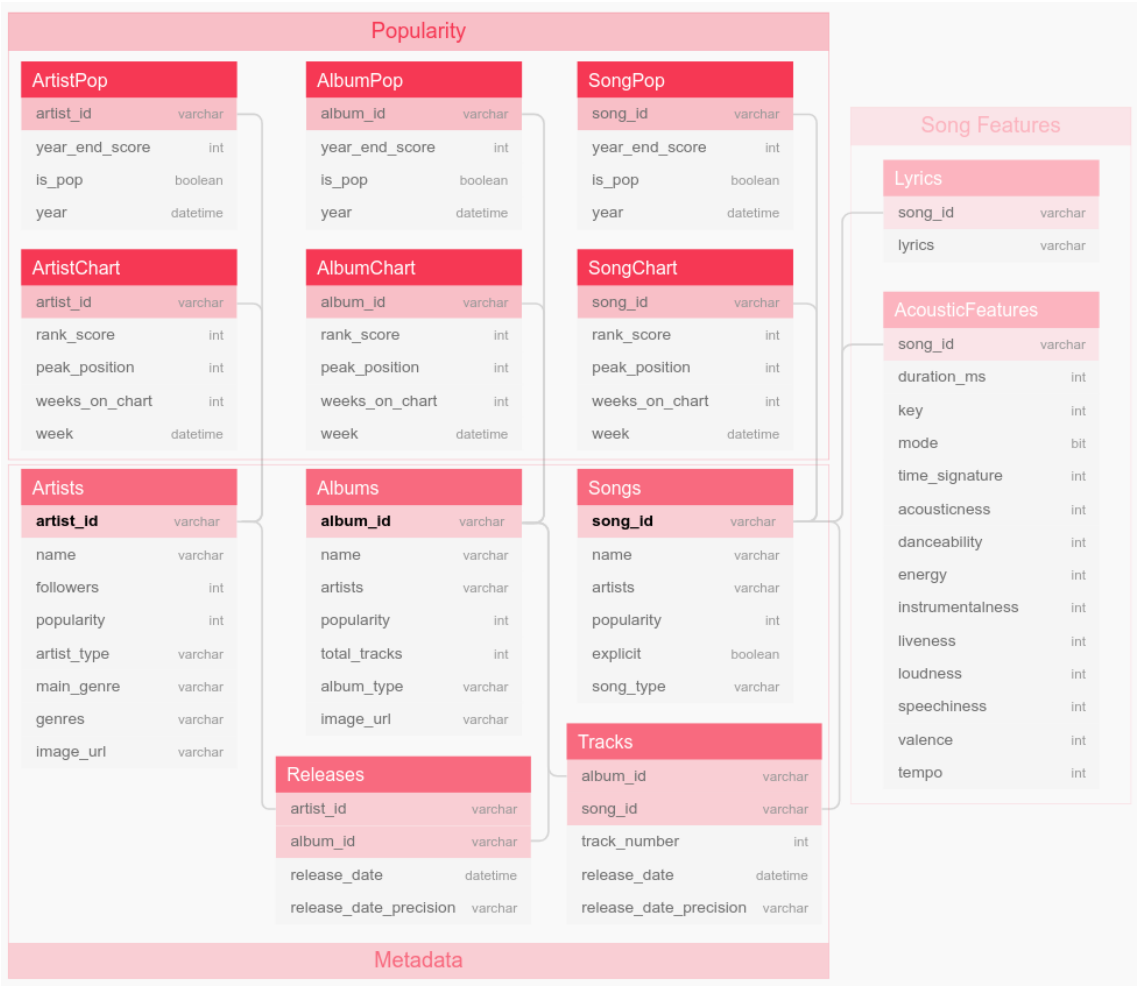

Figura 1: Esquema del conjunto de datos MusicOSet.

Categoría	Registros
Canciones	20.405
Artistas	11.518
Álbumes	26.522
Letras	19.664
Géneros	1.561

Tabla 1: Resumen del número de registros conjunto de datos MusicOSet.

Además de distribuirse en archivos ZIP con las tablas en formato CSV, el conjunto de datos MusicOSet también está disponible como una base de datos SQL, lo que facilita su acceso y consulta, previo a la realización de las visualizaciones.

Adicionalmente, también se ha utilizado el historial de reproducciones de Spotify de un usuario particular, con el objetivo de explorar la posibilidad de realizar visualizaciones personalizadas para un usuario concreto. Estos datos pueden ser descargados por el usuario desde su cuenta de Spotify. El historial usado en este trabajo se encuentra adjunto a este informe en el archivo `StreamingHistory.json`.

3. Visualizaciones realizadas

El desarrollo de las visualizaciones se ha realizado bajo el marco de trabajo Observable Framework[1], que permite la creación aplicaciones web centradas en la visualización de datos, integrando herramientas de procesamiento de datos en cualquier lenguaje de programación, con bibliotecas de visualización de gráficos en una única plataforma.

El flujo de desarrollo de las visualizaciones comienza con la conceptualización de las ideas y la selección de los datos a visualizar. Una vez se tiene claro el formato de los datos y de la presentación de los mismos, se procede a la implementación de un script en Python que se encarga de procesar los datos de forma adecuada para su posterior visualización. La transformación de los datos se realiza con consultas SQL mediante la biblioteca DuckDB [3], que permite realizar consultas SQL sobre archivos CSV de forma eficiente, y con la biblioteca Pandas [2].

Una vez se ha escrito el script de transformación de los datos, se procede a la implementación de la visualización que consume los datos transformados. Para ello, se ha comparado el uso de las bibliotecas D3, Vega-Lite y Observable Plot en una visualización inicial, con el objetivo de evaluar las ventajas y desventajas de cada una de ellas. Se ha optado por utilizar D3 para la implementación de las visualizaciones posteriores, debido a su flexibilidad y capacidad para crear visualizaciones animadas y altamente interactivas.

Todas las visualizaciones mostradas en este informe están disponibles en el siguiente enlace: `https://garcialnk.observablehq.cloud/visual-analytics`.

3.1. Visualización de la popularidad de artistas

`https://garcialnk.observablehq.cloud/visual-analytics/top-artists`

En esta primera visualización el objetivo era simple, descubrir cuáles son los artistas más populares en el conjunto de datos a trabajar. Para ello, el primer gráfico, que se muestra en la figura 2, representa a los 50 artistas más populares en un gráfico de barras vertical, donde la altura de las barras representa la popularidad de cada artista (dato relativo al número de reproducciones en el momento de creación del dataset) y su color la cantidad de seguidores que tiene. Este gráfico se ha realizado con la biblioteca Vega-Lite, una gramática de alto nivel para la creación de visualizaciones.

Aunque la creación de este gráfico es relativamente sencilla con Vega-Lite, esta biblioteca contiene algunas limitaciones intrínsecas en cuanto a personalización e interactividad. Al ser una abstracción de más alto nivel, no permite personalizar todos los aspectos de la visualización. Además, renderiza la visualización en una imagen estática, en vez de emplear algún otro método como SVG basado en vectores, lo que limita las posibilidades de interacción.

[1] `https://observablehq.com/framework/`

Adicionalmente, esta visualización demuestra algunas de las limitaciones de los gráficos de barras verticales, al tener muchos elementos, las etiquetas del eje X tienen que ser rotadas para que no se solapen, lo que dificulta la lectura de las mismas. En pantallas más estrechas, la visualización se vuelve aún más difícil de leer e interpretar. También la codificación de la popularidad en la altura de las barras no es la más adecuada, ya que hay muchos artistas con una popularidad similar y es difícil distinguirlos.

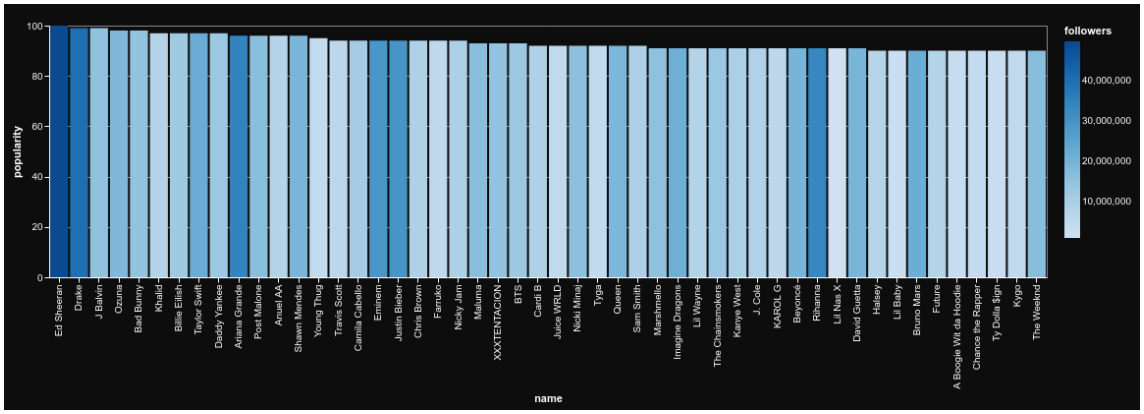

Figura 2: Gráfico de barras en Vega-Lite con los artistas más populares.

Para mejorar la visualización, se ha optado por iterar sobre el gráfico utilizando Observable Plot, una biblioteca de gráficos de alto nivel similar a Vega-Lite pero basada en D3. En la figura 3 se muestra el resultado de esta iteración, donde se ha utilizado un gráfico de barras horizontales en lugar de verticales, lo que facilita la lectura de las etiquetas. Además, los artistas se han agrupado por género musical, lo que permite una mejor organización de los datos y una visualización más clara. El ancho de las barras representa el número de seguidores de cada artista, lo que facilita la comparación entre ellos, mientras que el color de las barras representa la popularidad de cada artista.

El usuario puede cambiar el género musical que se muestra en el gráfico mediante un menú desplegable, lo que permite explorar la popularidad de los artistas en diferentes géneros. Además, al pasar el ratón sobre las barras, se muestra el número concreto de seguidores y la popularidad de cada artista, lo que facilita la interpretación de los datos, como se ve en la figura 4.

En este caso, Plot renderiza la visualización en SVG, lo que inmediatamente resulta en un gráfico de mayor resolución y calidad visual comparado con el de Vega-Lite. Además, permite una mayor personalización sin entrar en la complejidad de D3, aunque al ser una abstracción de esta, algunas funcionalidades más avanzadas relacionadas con la interacción y animación no están disponibles y es necesario recurrir a D3 directamente.

Teniendo en cuenta las limitaciones las bibliotecas anteriores, se utilizará D3 para la implementación de las visualizaciones posteriores, debido a su flexibilidad. Aunque para facilitar el desarrollo, se partirá de plantillas disponibles en la galería de D3[2].

[2] `https://observablehq.com/@d3/gallery`

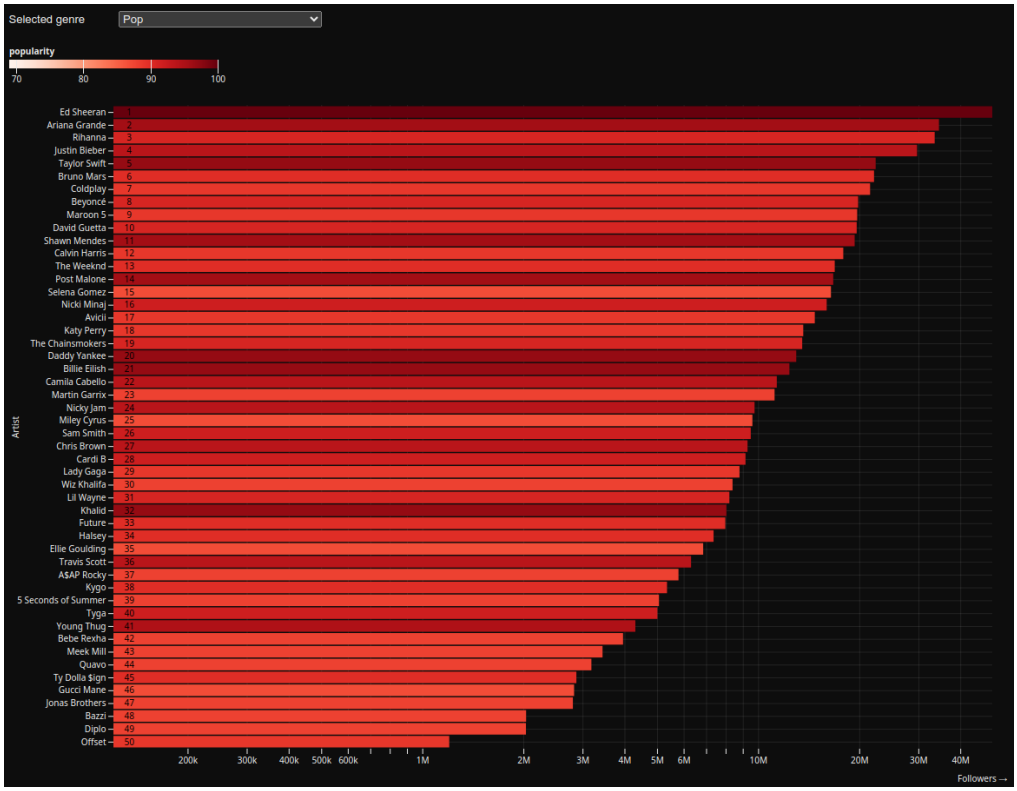

Figura 3: Gráfico en Observable Plot con los artistas de Pop más populares.

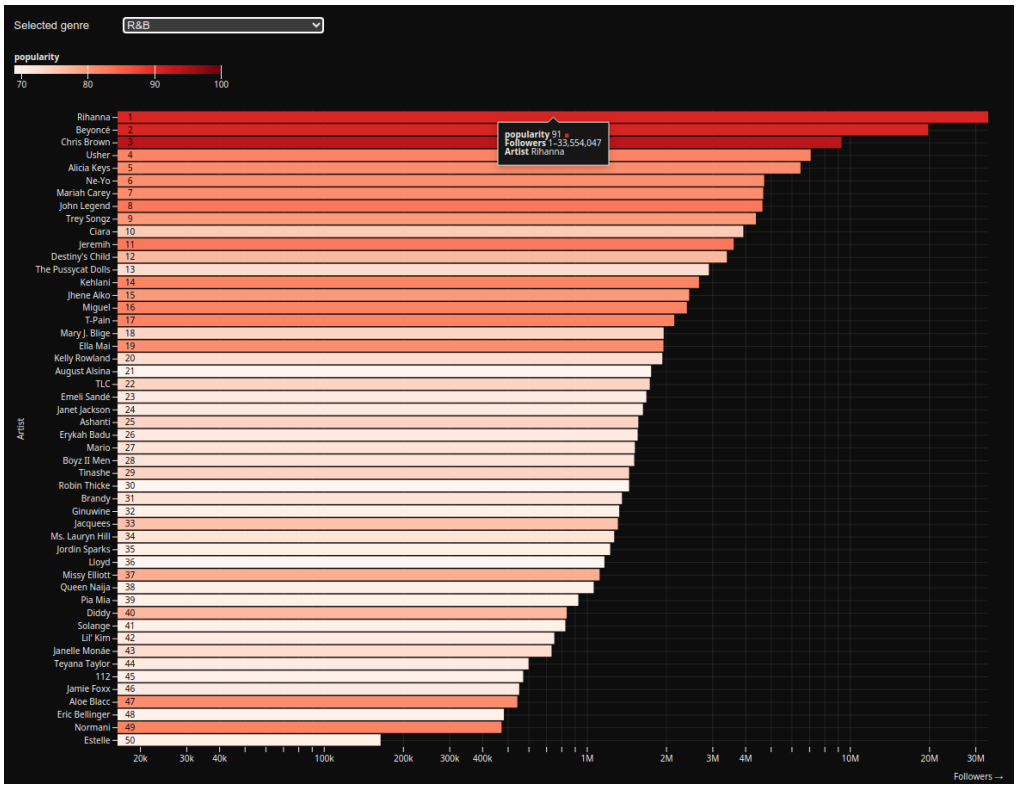

Figura 4: Gráfico en Observable Plot con los artistas de R&B más populares.

Para realizar una visualización de los artistas y su popularidad que fuera eficaz e interactiva, se ha optado por un treemap con zoom en D3, que permite representar jerarquías de datos en forma de rectángulos anidados. En la figura 5 se muestra el resultado de esta visualización, donde los rectángulos representan los géneros musicales de los distintos artistas, y el tamaño de los rectángulos es equivalente a la suma de los seguidores de los artistas que pertenecen a ese género. Al hacer clic en un rectángulo, se muestra un treemap con los artistas de ese género, como se ve en la figura 6 para el caso del rap, el color de los rectángulos representa la popularidad de los artistas.

Si se hace clic una vez más en un artista, se muestra un treemap con las canciones más populares de ese artista como se ve en la figura 7, donde el tamaño de los rectángulos representa la popularidad de las canciones. Al hacer clic en una canción, se abre una nueva pestaña con la página de la canción en Spotify.

Empleando la barra deslizadora se pueden filtrar los artistas que aparecen (y por ende los géneros) por su número de seguidores, lo que permite explorar los artistas más populares en función de su popularidad. Esta barra está en una escala logarítmica al haber un rango muy amplio de seguidores entre los distintos artistas.

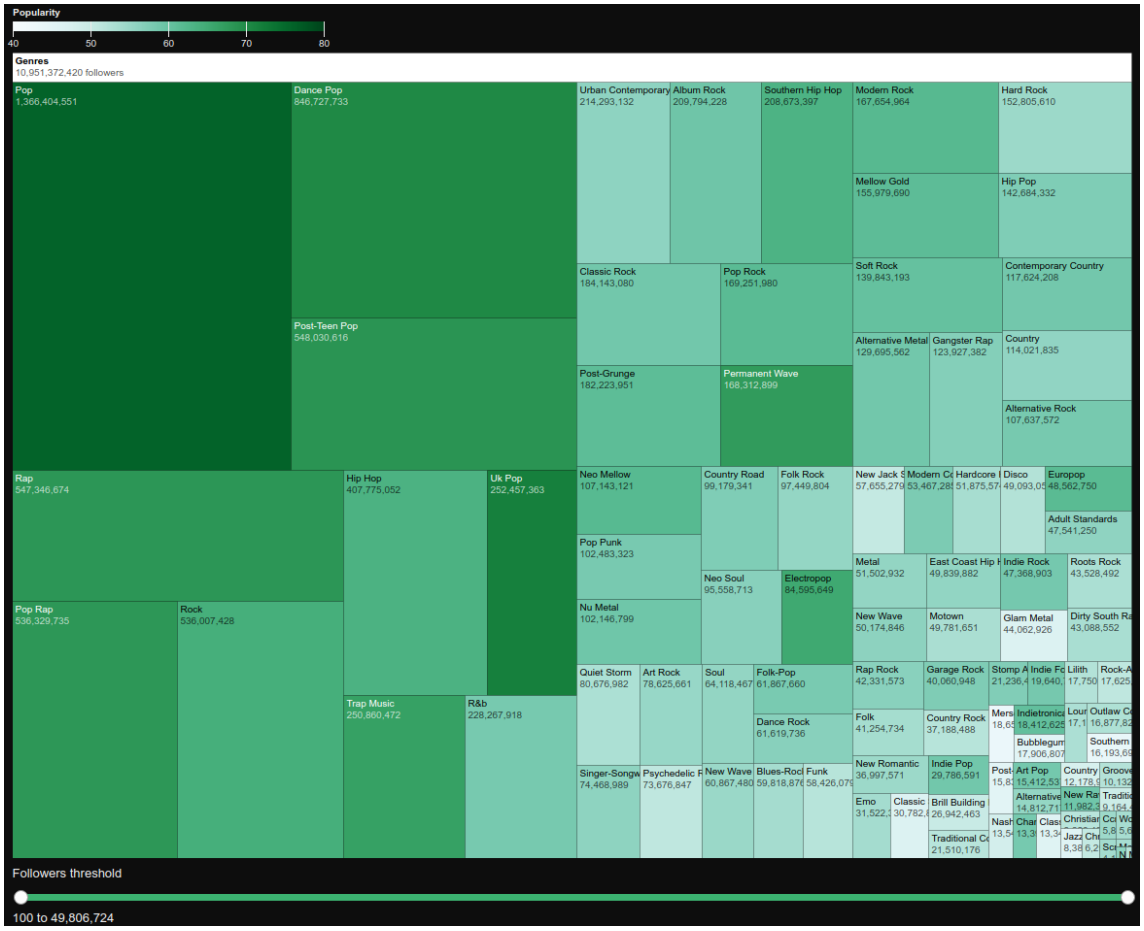

Figura 5: Treemap con zoom en D3 mostrando los distintos géneros.

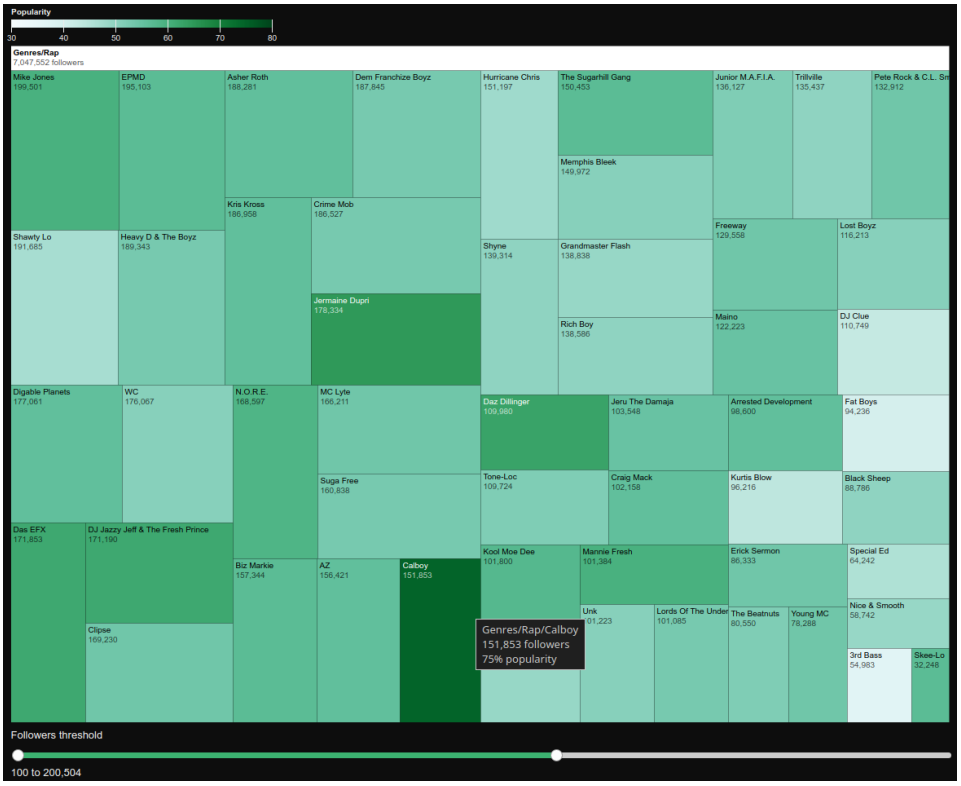

Figura 6: Treemap con zoom mostrando los artistas de rap.

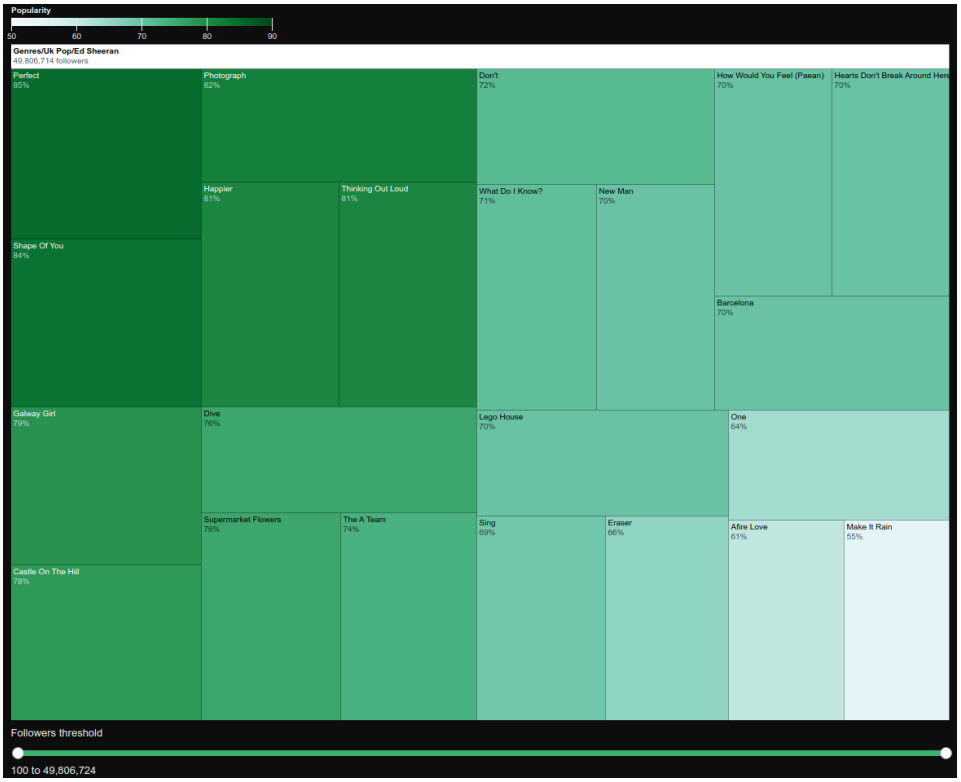

Figura 7: Treemap con zoom mostrando las canciones más populares de un artista.

3.2. Visualización de las tendencias de los artistas

`https://garcialnk.observablehq.cloud/visual-analytics/artist-trends`

En esta visualización se ha representado la evolución de la popularidad de los artistas a lo largo del último año (desde la fecha de creación del dataset) en un gráfico de líneas ampliable en D3. En la figura 8 se muestra el resultado de esta visualización. Cada línea representa la popularidad de un artista en función del tiempo, y al hacer pasar el ratón sobre una línea esta se resalta y se muestra el nombre del artista y su posición en el ranking de Spotify en ese momento.

Los datos del conjunto fueron preprocesados para no incluir artistas que no contuvieran información de cada una de las semanas de la serie temporal, para evitar huecos en las líneas del gráfico. Además, se ha establecido un límite máximo en la escala Y de 1000, para evitar que las líneas de los artistas menos populares "aplasten" a las de los más populares y se vean condensadas a un espacio muy pequeño.

Mediante la barra de búsqueda se pueden filtrar los artistas que aparecen en el gráfico, según su nombre, como se ve en la figura 9 para el caso de Rihanna. Al efectuar la búsqueda la escala del eje Y se ajusta automáticamente al valor máximo de popularidad de los artistas que aparecen en el gráfico.

También se puede hacer zoom sobre un periodo de tiempo concreto, arrastrando el ratón sobre el gráfico, y al hacer clic en él se vuelve a la vista original. En la figura 10 se muestra el resultado de hacer zoom en el gráfico sobre los últimos meses del año.

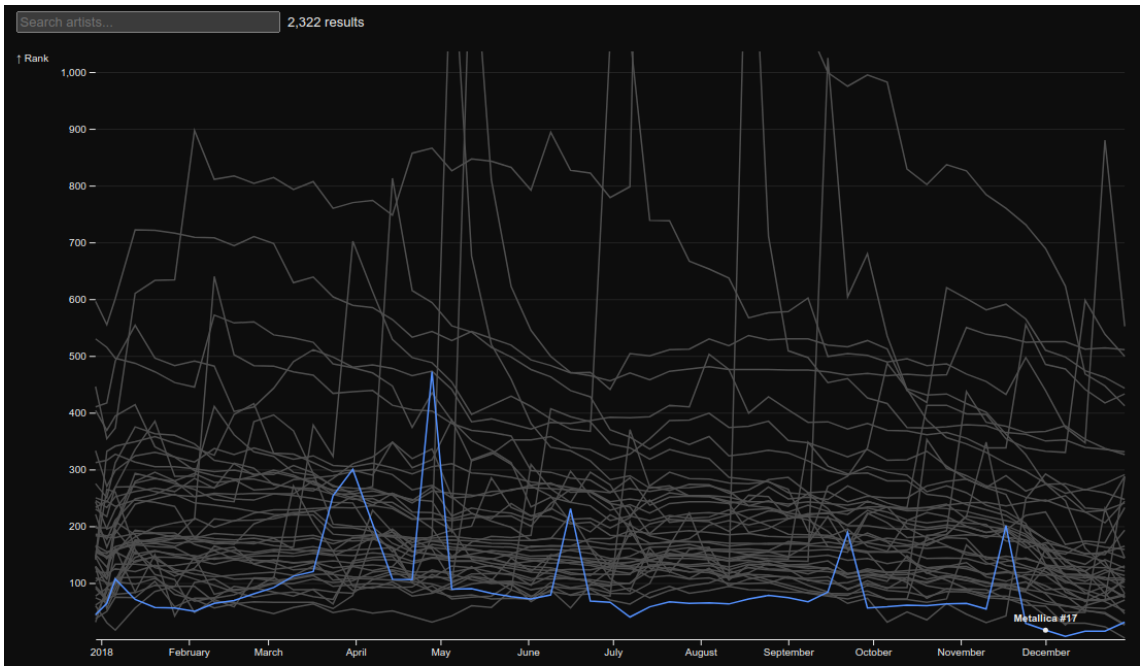

Figura 8: Gráfico de líneas en D3 con las tendencias de popularidad de los artistas.

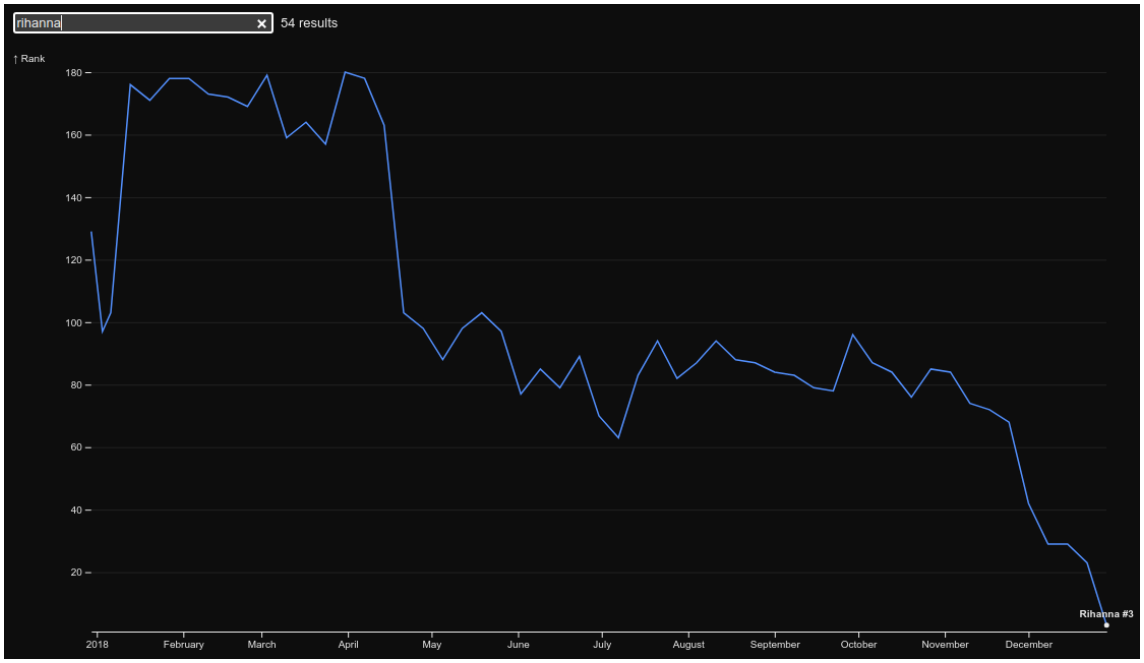

Figura 9: Gráfico de líneas con las tendencias de popularidad de Rihanna.

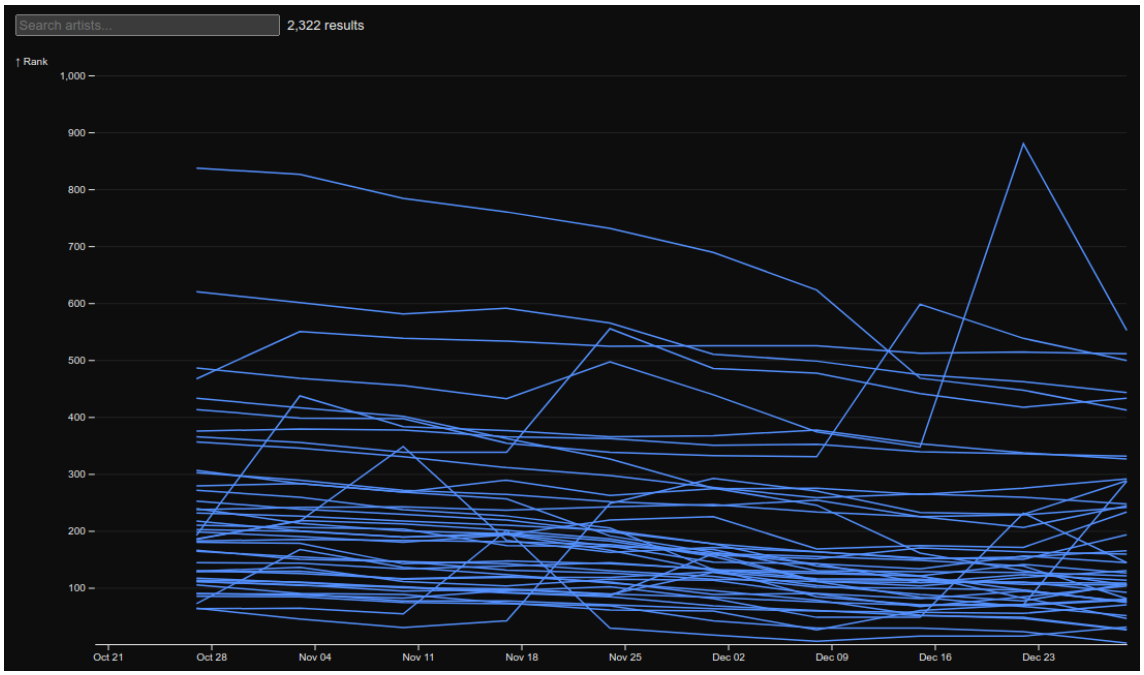

Figura 10: Gráfico de líneas con las tendencias en los últimos meses del año.

3.3. Visualización de las características musicales

`https://garcialnk.observablehq.cloud/visual-analytics/popularity-features`

Con el objetivo de explorar la relación entre las distintas características musicales de las canciones y su popularidad, se ha realizado una matriz de gráficos de dispersión en D3, donde cada punto representa una canción y su posición en cada celda está determinada por dos características musicales. El color de los puntos representa la popularidad de las canciones. En la figura 11 se muestra el resultado de esta visualización.

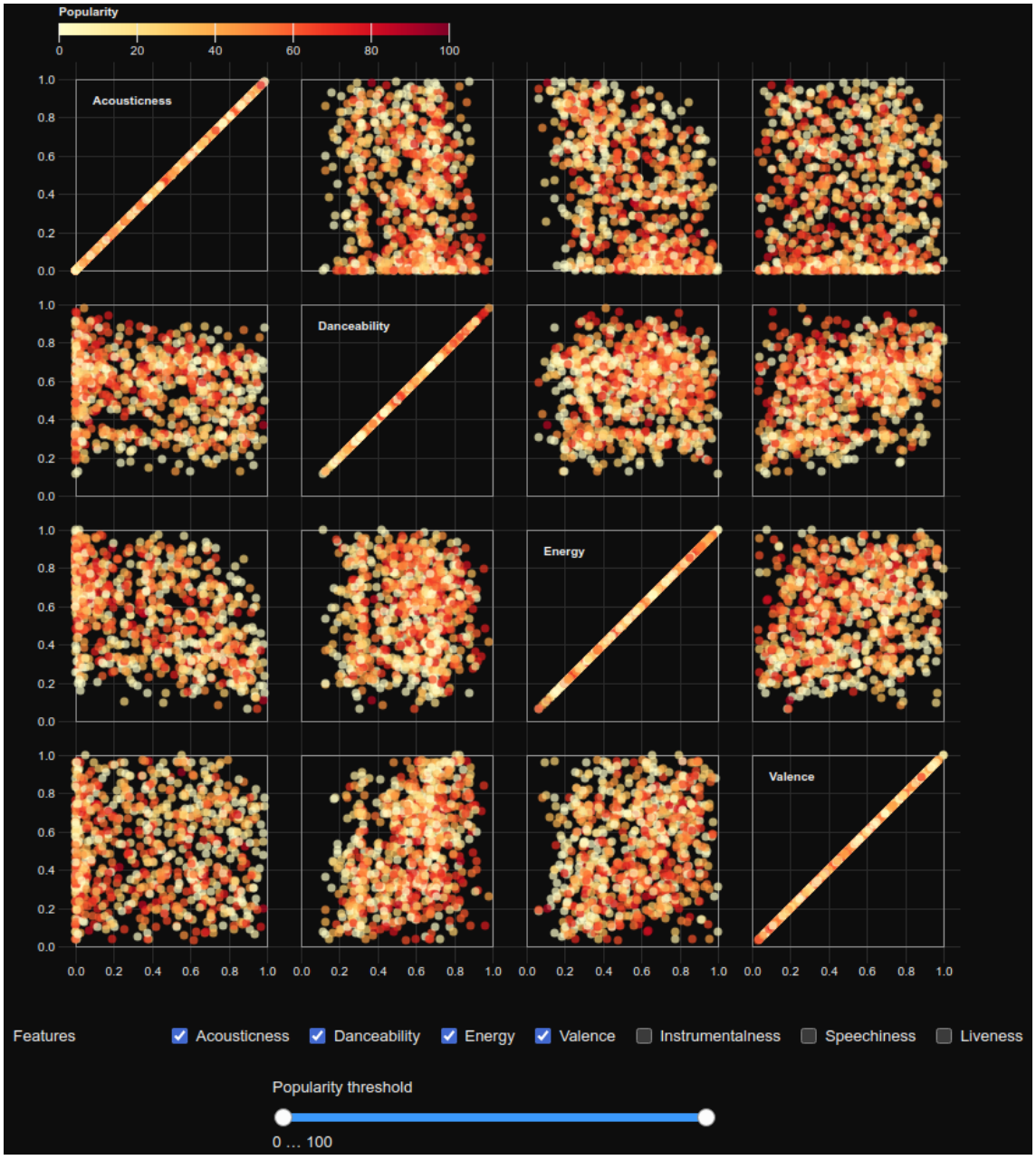

Figura 11: Matriz de gráficos de dispersión en D3 con las características musicales.

Arrastrando el ratón sobre una celda se resaltan los puntos correspondientes a la selección en todo el gráfico, lo que permite identificar patrones y relaciones entre múltiples características musicales. También se pueden seleccionar las características que se muestran en la matriz mediante checkboxes, para eliminar las que no se consideren relevantes. Además se pueden filtrar las canciones que aparecen representadas por su popularidad, mediante una barra deslizadora.

En la figura 12 se muestra el resultado de aplicar filtros a la matriz de gráficos de dispersión, donde se han seleccionado las características "Valence", "Danceability" y "Energy", se han filtrado las canciones con una popularidad menor de 20 y se han resaltado un conjunto concreto de puntos.

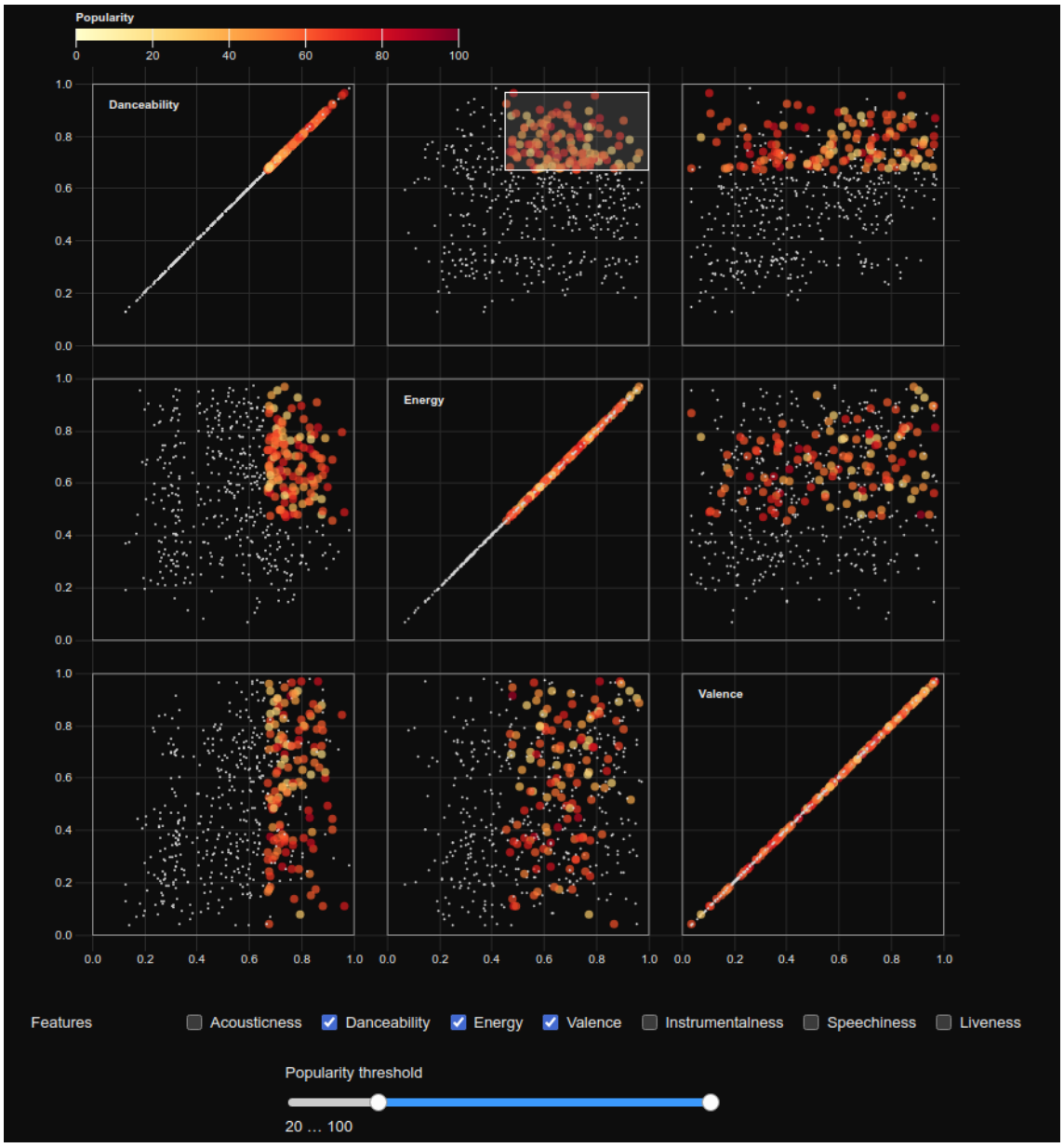

Figura 12: Matriz de gráficos de dispersión, filtrada y resaltada.

3.4. Visualización del historial de reproducción de Spotify

`https://garcialnk.observablehq.cloud/visual-analytics/streaming-history`

Esta última visualización difiere en la fuente de datos utilizada, ya que en este caso depende de un historial de reproducciones que proporcione el usuario tras descargarlo de su cuenta de Spotify. Estos datos se procesan localmente en el navegador y posteriormente se utilizan para mostrar una visualización de una carrera de barras en D3. Cada barra representa un artista y su anchura está determinada por el número de minutos que se ha escuchado al artista hasta el momento indicado. En la figura 13 se muestra el resultado de esta visualización sobre un historial de reproducciones de Spotify que se adjunta a este informe en el archivo `StreamingHistory.json`.

Según avanza el tiempo en la visualización, las barras van intercambiando posiciones según el tiempo de escucha de cada artista, lo que permite visualizar de forma dinámica la evolución del historial de reproducciones. Mediante un calendario se puede seleccionar una fecha concreta ye iniciar la visualización desde ese día, permitiendo explorar el historial de reproducciones en diferentes momentos. Además se puede pausar y reanudar la visualización, así como reiniciarla desde el principio.

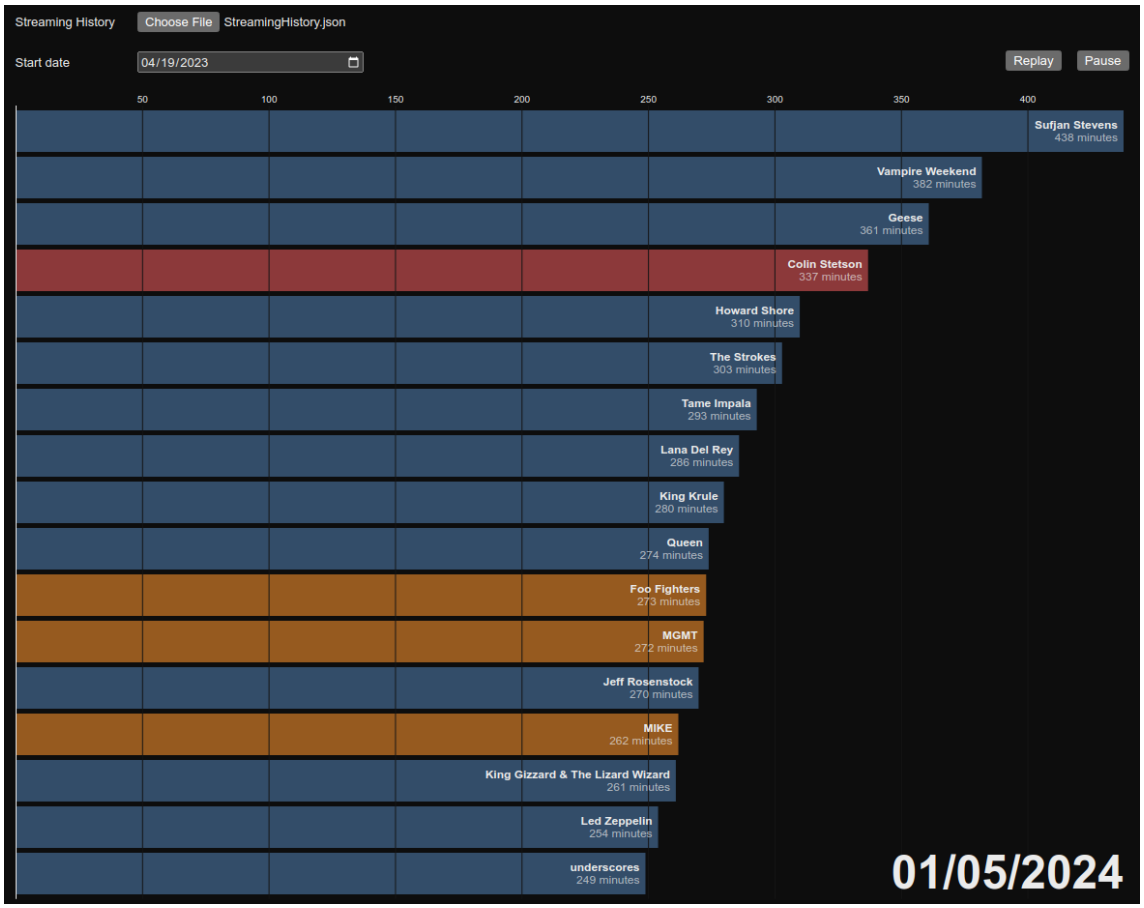

Figura 13: Resultado de la carrera de barras del historial de reproducción.

4. Conclusiones

En este trabajo se ha realizado un análisis visual de un conjunto de datos de música, con el objetivo de obtener información relevante sobre las canciones y artistas más populares. Se han desarrollado visualizaciones interactivas en D3 que permiten explorar la popularidad de los artistas, la evolución de su popularidad a lo largo del tiempo, las características musicales de las canciones y el historial de reproducciones de un usuario concreto.

En el proceso de desarrollo de las visualizaciones se ha explorado el uso de diferentes bibliotecas de visualización, como Vega-Lite, Observable Plot y D3, evaluando las ventajas y desventajas de cada una de ellas. Optando, finalmente, por utilizar D3 para las visualizaciones finales, debido a su flexibilidad y capacidad para crear visualizaciones interactivas.

Las visualizaciones realizadas permiten explorar de forma dinámica los datos, facilitando la identificación de patrones entre ellos. Además, se han incluido funcionalidades interactivas que permiten filtrar y explorar los datos de forma personalizada.

En conclusión, el análisis visual de datos realizado en este trabajo ha permitido obtener información relevante sobre las canciones y artistas en la plataforma. Las visualizaciones desarrolladas han demostrado ser una herramienta eficaz para la exploración y análisis de datos musicales.

Referencias

[1] Michael Bostock, Vadim Ogievetsky y Jeffrey Heer. D^3 Data-Driven Documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301-2309, 2011. DOI: 10.1109/TVCG.2011.185.

[2] The pandas development team. pandas-dev/pandas: Pandas, versión latest, febrero de 2020. DOI: 10.5281/zenodo.3509134. URL: https://doi.org/10.5281/zenodo.3509134.

[3] Mark Raasveldt y Hannes Mühleisen. DuckDB: an Embeddable Analytical Database. En *Proceedings of the 2019 International Conference on Management of Data*, SIGMOD '19, páginas 1981-1984, Amsterdam, Netherlands. Association for Computing Machinery, 2019. ISBN: 9781450356435. DOI: 10.1145/3299869.3320212. URL: https://doi.org/10.1145/3299869.3320212.

[4] Arvind Satyanarayan, Dominik Moritz, Kanit Wongsuphasawat y Jeffrey Heer. Vega-Lite: A Grammar of Interactive Graphics. *IEEE Trans. Visualization & Comp. Graphics (Proc. InfoVis)*, 2017. URL: http://idl.cs.washington.edu/papers/vega-lite.

[5] Mariana O. Silva, Laís Mota y Mirella M. Moro. MusicOSet: An Enhanced Open Dataset for Music Data Mining. Versión v2. Zenodo, junio de 2021. DOI: 10.5281/zenodo.4904639. URL: https://doi.org/10.5281/zenodo.4904639.