SpotifyVA: An Interactive Visual Analytics Dashboard for Exploring Spotify's 2023 Top-Streamed Songs

Carlos Fernández Fernández
Sapienza University of Rome
Engineering in Computer Science
Italy
fernandezfernandez.2247701@studenti.uniroma1.it

Alberto Rivas Casal
Sapienza University of Rome
Engineering in Computer Science
Italy
rivascasal.2247719@studenti.uniroma1.it

Abstract

We present *SpotifyVA*, an interactive Visual Analytics (VA) dashboard designed to help music industry analysts, data journalists, and curious listeners explore and reason about the audio characteristics of the 953 most-streamed songs on Spotify in 2023. The system integrates a t-SNE dimensionality reduction projection, a coordinated scatter plot, and several linked complementary views to support both overview exploration and targeted drill-down. The analytical backbone is triggered by user visual interactions—specifically, a lasso-brush selection over the t-SNE space—providing on-demand statistical summaries and feature comparisons without requiring users to issue explicit queries. The design follows the Visual Analytics mantra *"analyse first, show the important, zoom and filter, then details on demand"* and was evaluated against the exclusion and penalty criteria defined in the course guidelines of the Visual Analytics course at Sapienza University of Rome (Prof. Giuseppe Santucci, A.Y. 2025–2026).

Keywords

Visual Analytics, Spotify, t-SNE, dimensionality reduction, music data, interactive visualization, D3.js, coordinated views

ACM Reference Format:
Carlos Fernández Fernández and Alberto Rivas Casal. 2025. SpotifyVA: An Interactive Visual Analytics Dashboard for Exploring Spotify's 2023 Top-Streamed Songs. In *Proceedings of Visual Analytics – Engineering in Computer Science (VA 2025).* ACM, New York, NY, USA, 5 pages.

1 Introduction

Music streaming platforms generate enormous amounts of data about listener preferences, audio properties, and chart performance. Spotify alone hosts more than 100 million tracks; understanding *what* makes a song popular—or how acoustically similar groups of songs cluster—is a non-trivial analytical task that benefits enormously from visual interfaces.

The dataset for this project is the *Spotify Top Songs 2023* collection, publicly available on Kaggle [1]. It records 953 songs across 24 attributes (AS index = 953 × 24 = 22,872, well within the prescribed range of 10,000–50,000 [2]), covering audio features such as danceability, energy, valence, acousticness, speechiness, liveness, instrumentalness, and tempo (BPM), together with chart statistics across multiple platforms (Spotify, Apple Music, Deezer, Shazam) and metadata (artist, release date, musical key, mode).

The primary **intended users** are music data analysts and A&R professionals who need to identify clusters of similarly-styled tracks, compare audio profiles across genres or modes, and relate acoustic properties to streaming success. A secondary audience includes music enthusiasts who wish to explore the landscape of popular music in a self-directed, visual manner.

The core design decisions are:

- A **t-SNE projection** of eight audio features as the primary exploration space, integrating dimensionality reduction directly into the analysis flow.
- A **lasso-brush interaction** that triggers on-demand analytics on the selected subset without any explicit filtering menu.
- At least **two coordinated views** linked bidirectionally: the t-SNE scatter plot and a parallel-coordinates / radar-chart detail view.
- A **filter panel** (mode, artist search, per-feature range sliders) that refines the active selection progressively.

The rest of this paper is structured as follows: Section 2 positions the work with respect to related proposals; Section 3 describes the dataset and preprocessing pipeline; Section 4 details the dimensionality reduction step; Section 5 documents the design rationale and the visual encoding choices; Section 6 describes the implemented prototype; Section 7 presents the discovered insights (in progress); Section 8 concludes.

2 Related Work

2.1 Visualization of Music Data

Visualization of music data has attracted significant research attention. Wattenberg [6] introduced *The Shape of Song*, an early aesthetic approach to representing repeated structures in musical scores. More recently, *MusicMap* [9] used a genre-based 2-D layout to allow exploration of musical relationships. Both works share our goal of providing an overview of a musical corpus, but focus on structural or genre-level abstractions rather than quantitative audio features as measured by Spotify's API.

2.2 Dimensionality Reduction for High-Dimensional Data

The use of t-SNE [3] for exploratory data visualization is well established in information visualization. Wenskovitch et al. [7] proposed interaction patterns specifically designed for dimensionality-reduction scatter plots, demonstrating that lasso selection with linked views dramatically improves users' ability to relate structure in the low-dimensional projection to the original high-dimensional space—a key design goal of SpotifyVA. Cutura et al. [8] further studied how visual interaction can steer t-SNE parameters, pointing to future extensions of our system.

VA 2025, Sapienza University of Rome
2025.

2.3 Visual Analytics for Streaming and Recommendation Systems

Toward the end of the music analytics spectrum, Bostock et al.'s D3.js ecosystem [5] has enabled multiple interactive music dashboards. The *Every Noise at Once* project[1] and *Spotify's own "Sound Town"* feature both employ scatter plots over audio embedding spaces. However, these are proprietary production systems without published interaction semantics or reproducible analytical pipelines. By contrast, SpotifyVA is fully open-source (GitHub link provided at submission), reproducible, and specifically designed around the Visual Analytics cycle [4] with a clear data-transformation–model-visualization feedback loop.

2.4 Differences from Related Work

Compared to the above, SpotifyVA distinguishes itself by (i) making the dimensionality reduction step *part of the interactive analysis flow* rather than a one-shot pre-processing step, (ii) coupling lasso interaction directly to statistical analytics (not mere highlighting), and (iii) targeting a real, externally-available dataset of contemporary popular music rather than synthetic or licensed corpora.

3 Dataset and Preprocessing

3.1 Dataset Description

The raw dataset (file `spotify-2023.csv`) contains 953 rows and 24 columns after loading. Table 1 summarises the features used in the analysis.

Table 1: Selected features of the Spotify 2023 dataset.

Feature	Type	Description
track_name	Categorical	Song title
artist(s)_name	Categorical	Performing artist(s)
released_year	Numeric	Release year (1930–2023)
bpm	Numeric	Beats per minute (65–206)
danceability_%	Numeric	Danceability (0–97%)
energy_%	Numeric	Energy (0–97%)
valence_%	Numeric	Musical positivity (0–97%)
acousticness_%	Numeric	Acoustic quality (0–97%)
speechiness_%	Numeric	Spoken-word proportion
liveness_%	Numeric	Live-performance probability
instrumentalness_%	Numeric	Instrumental proportion
key	Categorical	Musical key (C–B, Unknown)
mode	Categorical	Major / Minor
streams	Numeric	Total Spotify streams

There are 645 unique artists among the 953 songs, confirming that many artists appear multiple times. The AS index [2] for this dataset is 953 × 24 = 22,872.

3.2 Data Quality Issues and Cleaning

The preprocessing pipeline was implemented in Python and is fully documented in the companion notebook `cleaning-EDA.ipynb`. Four quality issues were identified and resolved.

Type coercion. Three columns (`streams`, `in_deezer_playlists`, `in_shazam_charts`) were stored as strings rather than numbers in the raw CSV. The `streams` column contained one non-numeric value at row 547, which was imputed with the column mean after converting valid values to integers. The Deezer and Shazam columns used comma-separated thousand separators (e.g., `"1,234"`), which were stripped before type conversion.

Missing values. Two columns contained missing values: `key` (50 entries, 5.2%) and `in_shazam_charts` (50 entries). For `key`, missing values were filled with the string `"Unknown"`, treated as a meaningful category (indicating the audio analysis could not determine the key). A binary indicator column `key_detected` was created. For `in_shazam_charts`, missing values were set to 0, reflecting the interpretation that the song was not tracked on Shazam; a boolean `shazam_tracked` flag was added.

Outlier analysis. IQR-based outlier detection (threshold = 1.5) was applied to all numerical columns. Given the domain—popular music features are bounded by definition (percentages) or represent physically meaningful ranges (BPM 65–206)—the decision was made to *keep* all outliers, as extreme values carry genuine semantic content (e.g., a 97% instrumentalness song is genuinely unusual and analytically interesting).

Normalization. Prior to dimensionality reduction, the eight audio feature columns were standardized (zero mean, unit variance) using `scikit-learn`'s `StandardScaler`. This ensures that features on different scales (percentages vs. BPM) contribute equally to the t-SNE layout without any single feature dominating by virtue of its numerical range.

4 Dimensionality Reduction

4.1 Feature Selection

Eight audio features were selected for the projection: `danceability_%`, `energy_%`, `valence_%`, `acousticness_%`, `speechiness_%`, `liveness_%`, `instrumentalness_%`, and `bpm`. Platform statistics (stream counts, playlist appearances) were deliberately excluded from the projection because they reflect market success rather than intrinsic sonic properties; they are available as tooltip and analytical information but should not drive the acoustic layout.

4.2 t-SNE Configuration

t-SNE (t-distributed Stochastic Neighbor Embedding) [3] was applied using `scikit-learn`'s implementation with the following hyperparameters:

Listing 1: t-SNE configuration.

```
tsne = TSNE(
    n_components=2,
    random_state=42,
    perplexity=30,
    n_iter=1000,
    verbose=1
)
X_tsne = tsne.fit_transform(X)   # X: 953    8
     standardized matrix
```

[1] https://everynoise.com

A perplexity of 30 is a standard choice for datasets of this size (roughly the geometric mean of the 91-nearest-neighbor graph computed internally). After 1000 iterations the KL divergence converged to 1.133, indicating a stable embedding. The fixed `random_state=42` ensures full reproducibility.

4.3 Integration into the Analysis Flow

A key requirement of the course specification [2] is that dimensionality reduction must be *integrated into the analysis flow*, not used merely as a pre-processing curiosity. In SpotifyVA this is achieved by:

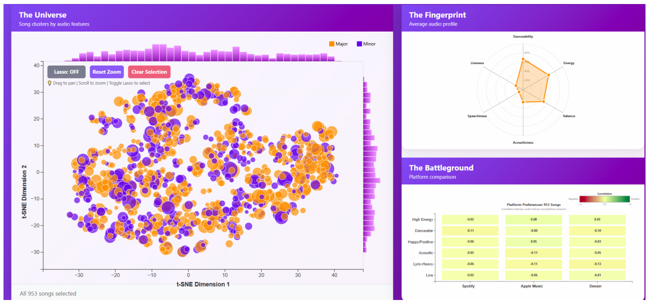
(1) Making the t-SNE scatter plot the **primary navigation space**: all exploration begins here.
(2) The lasso brush on the t-SNE projection immediately **triggers computation** of feature statistics (mean, standard deviation, range) for the selected subset, displayed in the linked detail panel.
(3) Filter interactions (mode, artist, per-feature sliders) propagate back to the t-SNE view, updating opacity to reflect the current active subset— closing the Visual Analytics feedback loop.

The two t-SNE coordinates `tsne_1` and `tsne_2` were appended to the full data frame and exported as both CSV and JSON for consumption by the D3.js front-end (`visualization_data.json`, 953 records).

5 Design Rationale

5.1 Visual Analytics Cycle

The design follows the Visual Analytics cycle of Keim et al. [4]: raw data are *transformed* (cleaning, normalization), *modelled* (t-SNE), *mapped* to visual representations, and *interacted with* to generate knowledge. Figure 1 (placeholder) illustrates how each component of SpotifyVA maps to a stage in the cycle.



	Spotify	Apple Music	Deezer
High Energy	-0.02	0.08	0.03
Danceable	-0.11	-0.00	-0.10
Happy/Positive	-0.06	0.05	-0.03
Acoustic	-0.05	-0.11	-0.05
Lyric-Heavy	-0.06	-0.11	-0.13
Live	-0.05	-0.06	-0.01

Figure 1: The Visual Analytics cycle as instantiated in SpotifyVA. Left: raw data and t-SNE model. Centre: t-SNE scatter plot (primary view) and coordinated detail view. Right: insights derived through user interaction.

5.2 Primary View: t-SNE Scatter Plot

Each song is represented as a circle in the t-SNE 2-D space. Visual encodings:

- **Position** (x, y): t-SNE coordinates encoding overall audio similarity.
- **Colour**: Musical mode—orange for Major, purple for Minor—a categorical encoding using maximally distinct hues that are also distinguishable under common forms of colour blindness.
- **Radius**: Encoded on a sqrt-scale proportional to stream count, allowing viewers to perceive relative popularity within clusters.
- **Opacity**: 0.9 for active songs (in current filter/selection), 0.05 for inactive songs, providing a strong figure-ground separation without removing context.

Zoom and pan (via D3 zoom) are supported to navigate dense regions; all interactions preserve the lasso selection state.

5.3 Lasso Brush — Interaction-Triggered Analytics

Satisfying the mandatory requirement that *at least one computation must be triggered by user visual interaction* [2], the lasso brush (D3 `brush`) enables freeform rectangular selection. Upon releasing the brush, the system:

(1) Identifies selected data objects using **object-reference identity** (JavaScript `Set` of object references, not string keys), correctly handling songs that share a `track_name`.
(2) Computes per-feature statistics (mean, std, min, max) for the selection.
(3) Updates all linked views to reflect the selection.

Using object references rather than track names as identity keys is critical: the dataset contains multiple entries for remixed or re-released songs that share a title, and string-key matching would incorrectly light up acoustically unrelated songs.

5.4 Coordinated Views

Two views are coordinated bidirectionally [2]:

(1) **t-SNE scatter plot** (primary): lasso selection exports object-reference arrays to `FilterState.lassoSelection`.
(2) **Detail / radar chart** (secondary): renders the mean audio profile of the current selection as a radar chart, updating whenever the lasso or any filter changes.

Coordination is bidirectional: clicking an artist in the detail panel highlights her songs in the t-SNE plot, and a lasso in the t-SNE plot updates the detail panel. This satisfies the course requirement of *"at least 2 visualizations coordinated in both ways and interactive"* [2].

5.5 Filter Panel

A persistent side-panel provides:

- **Mode filter**: Major / Minor toggle buttons. Implemented with a cleared state cycle to avoid race conditions where the filter was applied before the previous state was fully reset.
- **Artist search**: Free-text search with autocomplete, restricting the active set to the entered artist(s).
- **Range sliders**: Per-feature mini-sliders (danceability, energy, valence, BPM) for granular subsetting.
- **Badge counter**: Shows the number of currently active songs.
- **Clear all**: Resets all filters atomically to prevent partial-state artifacts.

All filter state is maintained in a central `FilterState` object and propagated through a single `applyAllFilters()` function to ensure consistent rendering across views.

5.6 Color Encoding and Accessibility

Colour choices follow standard encoding principles discussed in the Visual Analytics course module on *Representation-Encoding* [2]: categorical colour palettes are limited to a small number of hues, sequential scales are used for quantitative attributes, and diverging scales are avoided where there is no meaningful midpoint. Legends are provided for every visual channel.

6 Prototype Implementation

6.1 Technology Stack

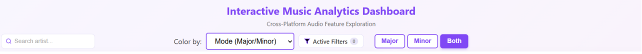
- **Data pipeline**: Python 3.10 with `pandas`, `numpy`, `scikit-learn` (cleaning-EDA.ipynb + dimensionality-reduction.ipynb).
- **Front-end**: Vanilla JavaScript with D3.js v7 for all visualizations; no framework dependencies to keep the codebase portable and inspectable.
- **Deployment**: Static HTML/JS/CSS served via a GitHub Pages repository; no server-side component required.

6.2 Architecture

The front-end is organized into three modules:

- `main.js`: Application bootstrap, data loading (JSON), global state (`AppState`, `FilterState`), and the central `applyAllFilters()` orchestrator.
- `universe.js`: The `UniverseView` class encapsulating the t-SNE scatter plot, zoom/pan, and lasso brush logic.
- `detail.js`: The detail / radar-chart view, updated on every selection change.

6.3 Prototype Screenshots



Interactive Music Analytics Dashboard

Cross-Platform Audio Feature Exploration

Color by: Mode (Major/Minor)

Figure 2: SpotifyVA control bar. From left to right: artist search with autocomplete, colour-by selector (Mode, Energy, Danceability, Valence, Acousticness), active filters badge, and Major/Minor/Both mode filter buttons. Here the Major filter is active (orange), restricting the view to 571 Major-mode songs.

Note: Replace placeholder boxes with actual screenshots before submission.

7 Discovered Insights

In this section of the report, we will aim to identify features and correlations that can help our artists and producers improve the attributes of their songs in order to reach the Top Most Streamed Songs, by visualizing the data and uncovering common patterns and relevant insights.

Insight 1 — Studio Polish

The 2023 streaming landscape is characterized by a definitive trend toward technical studio perfection, as evidenced by the consistent prevalence of low Liveness and Speechiness metrics among top-charting tracks. This data-driven insight reveals a shift away from live, organic recordings in favor of high-fidelity, polished studio environments, while also highlighting a clear market preference for melodic, sung content over spoken-word delivery or dense verbal structures. For analysts, this suggests that the modern listener prioritizes a 'crystal clear' audio experience optimized for high-definition hardware such as AirPods and smart speakers, where ambient noise from live performances or excessive speech can be perceived as distractions. For producers and musicians seeking global fame, these findings serve as a strategic roadmap: by prioritizing pristine technical production and melodic accessibility, they can maximize their commercial viability and ensure their work aligns with the acoustic preferences of today's mainstream consumption habits

Insight 2 — The Synthetic-Organic Trade-off

Through continuous interaction with the t-SNE projection and coordinated views, we identified a robust negative correlation where high Acousticness scores systematically correspond with a marked decrease in Energy and Danceability levels, defining a boundary between intimate and mainstream sonic niches. For producers and musicians aiming for global commercial success, this insight provides critical actionable intelligence: as the current fame formula for the top charts is heavily driven by high intensity rhythmic patterns, relying on acoustic instrumentation may be counterproductive for those looking for maximize streaming potential and viral reach, as it inherently shifts the track's profile away from the high energy characteristics that dominate the 2023 mainstream landscape.

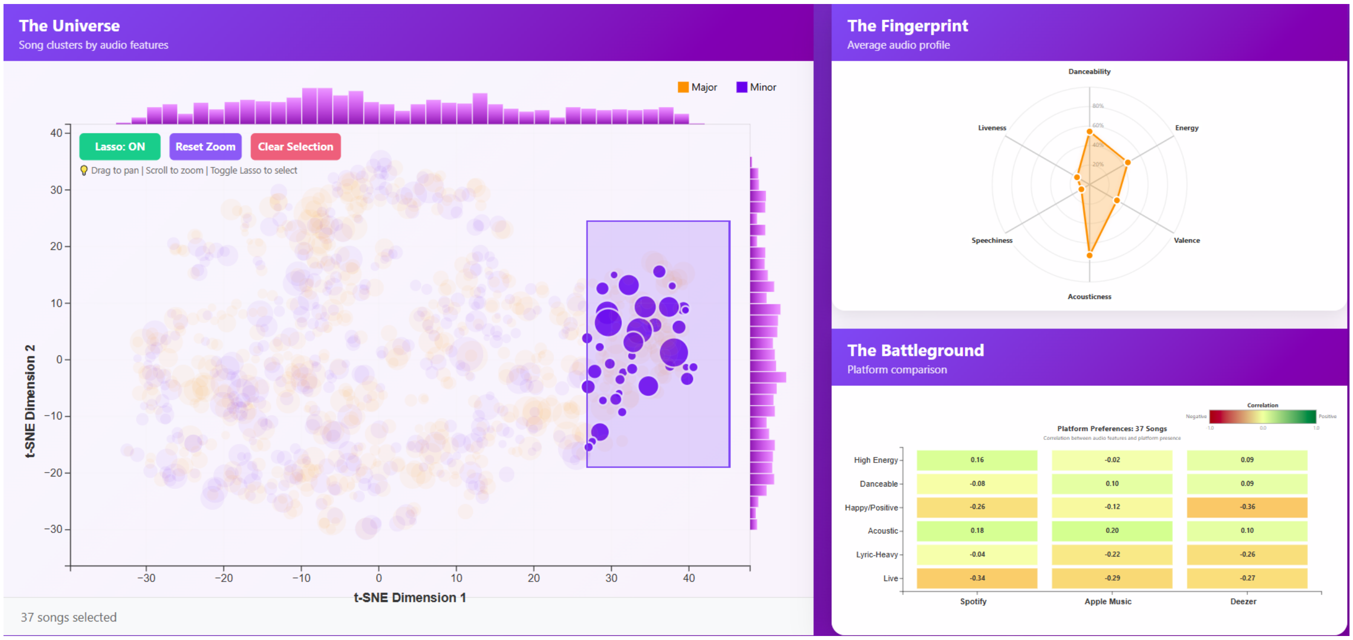
Insight 3 — Some Observations

Success in the 2023 landscape is not random, is mathematically skewed toward high fidelity, synthetic, and melodic productions that prioritize clarity and rhythmic intensity over organic authenticity.

- In the t-SNE embedding, Major and Minor mode songs do not form two distinct, isolated islands. Instead, they appear as interwoven micro-clusters. This suggests that while Mode (encoded by color) provides a tonal mood, it is not the primary driver of acoustic similarity in the 2023 Top Hits. We observed that high energy dance tracks often cluster together regardless of being in Major or Minor mode, indicating that rhythmic structure and production polish are stronger "gravitational forces" in the multidimensional space than the musical scale alone.
- Analyzing Streams (bubble size) across "The Universe" reveals that 2023 hits are not acoustically homogeneous. The visual encoding shows a bimodal distribution: a large concentration of hits in the yellow/light area (low Acousticness) and an equally significant cluster of large bubbles in the dark blue region (high Acousticness). This t-SNE separation confirms that 2023 allows for genuine acoustic diversity, where organic tracks are not outliers but dominant chart-toppers alongside dense electronic productions.

8 Conclusion

We have presented SpotifyVA, an interactive Visual Analytics dashboard for exploring the audio landscape of Spotify's 2023 top-streamed songs. The system satisfies all mandatory requirements specified in the Visual Analytics course at Sapienza University of



	Spotify	Apple Music	Deezer
High Energy	0.16	-0.02	0.09
Danceable	-0.08	0.10	0.09
Happy/Positive	-0.26	-0.12	-0.36
Acoustic	0.18	0.20	0.10
Lyric-Heavy	-0.04	-0.22	-0.26
Live	-0.34	-0.29	-0.27

Figure 3: Lasso selection interaction. A rectangular brush selects a dense Minor-mode cluster in the right region of the t-SNE space (37 songs). Non-selected songs are faded to near-transparency, providing context without distraction. The Fingerprint view (top right) immediately updates to show the selected subset's average audio profile — notably high Danceability and low Acousticness. The Battleground (bottom right) recalculates platform correlations for the 37-song subset, revealing that Live songs in this cluster are negatively correlated with presence on all three platforms.

Rome [2]: it integrates t-SNE dimensionality reduction into the analysis flow, provides at least two bidirectionally coordinated interactive visualizations, and triggers non-trivial analytics from user visual interactions. The preprocessing pipeline ensures data quality and reproducibility, and the entire system is open-source and hosted on GitHub.

Future work will extend the system with: (i) user-steerable t-SNE parameters to allow on-the-fly re-embedding; (ii) temporal views of chart trajectories; and (iii) a recommendation sub-system that identifies acoustically similar songs to a user-selected target.

References

[1] Nidula Elgiriyewithana. Most Streamed Spotify Songs 2023. https://www.kaggle.com/datasets/nelgiriyewithana/top-spotify-songs-2023, 2023. Accessed: February 2026.

[2] Giuseppe Santucci. Visual Analytics – Exam Structure and Rules, Fall 2025. Sapienza University of Rome, 2025.

[3] Laurens van der Maaten and Geoffrey Hinton. Visualizing data using t-SNE. *Journal of Machine Learning Research*, 9(86):2579–2605, 2008.

[4] Daniel Keim, Gennady Andrienko, Jean-Daniel Fekete, Carsten Görg, Jorn Kohlhammer, and Guy Melançon. Visual analytics: Definition, process, and challenges. In Andreas Kerren, John T. Stasko, Jean-Daniel Fekete, and Chris North, editors, *Information Visualization*, pages 154–175. Springer, 2008.

[5] Michael Bostock, Vadim Ogievetsky, and Jeffrey Heer. D3: Data-driven documents. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2301–2309, 2011.

[6] Martin Wattenberg. Arc diagrams: Visualizing structure in strings. In *Proceedings of IEEE Symposium on Information Visualization (InfoVis)*, pages 110–116, 2002.

[7] John Wenskovitch, Ian Crandell, Naren Ramakrishnan, Leanna House, Scotland Leman, and Chris North. Towards a systematic combination of dimension reduction and clustering in visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, 24(1):131–141, 2018.

[8] Rene Cutura, Christoph Sieb, and Michael Sedlmair. Comparing dimensionality reduction methods using data descriptor graphs. In *Proceedings of the Workshop on TDA & Visualization*, 2020.

[9] Kwinten Crauwels. Musicmap: The Genealogy and History of Popular Music Genres. https://musicmap.info, 2019.