Music Data Mining using Audio Features Extracted from Spotify

Sandy Benito, Antoine Carr, Reinaldo Sanchez-Arias, Ph.D
St. Thomas University, School of Science, Miami Gardens, FL

ST.THOMAS UNIVERSITY School of Science

Introduction

When it comes to music, we all have a unique, peculiar style that characterizes us. What lies behind the songs that appeal to us? In some way, are they similar? Particularly, can we group songs just by the way they sound? A variety of measurement techniques used in science gather data that often times consists of many more variables per sample than the number of samples present. Principal Component Analysis (PCA) and clustering help reduce the dimensionality of such variables, improving visualization so that one can observe where specific, for instance, audio features lie on a chart when comparing those variables. Crucial information might be gathered by analyzing the evolution of music over decades through audio features. In this project, an exploration of the different features that represent songs is performed and a study of the power of data mining and clustering techniques is presented.

Data Collection

The case study for data analysis was performed using Rstudio [1], Spotify's API, along with the `rvest` , `tidyverse` and `spotifyr` packages [2-4]. The `rvest` package allows to extract data from a web page (web scraping); `tidyverse` is used for data transformation and cleaning; and `spotifyr` pulls song audio features from the Spotify Web API (Application Programming Interface: `https://developer.spotify.com/documentation/web-api`). This API accesses user related data, which for the purposes of this project was organized via playlists for a Spotify account created for the project. Playlists were created individually and include the top 10 summer hits from the years 1958 through 2017. Similarly, we gathered data from Grammy award winners from the years 1959-2018, including albums or songs from different categories.

```r
# build list of dataframes
myrecords <- vector("list", length = m)
for (j in 1:m ) {
  # get tracks from playlist
  bb_tracks <- get_playlist_tracks(bb_my_pl[j, ])
  # extract audio features from playlist
  audio_features <- get_track_audio_features(bb_tracks)
  # inner join by `track_uri` to include song metrics and meta-data
  df_join <- inner_join(audio_features, bb_tracks, by = "track_uri")
  # add column `year` to the data frame
  df_join$year <- str_extract(bb_my_pl[j, ][1], "\\d++") %>%
                    as.numeric()

  myrecords[[j]] <- df_join
  # get next item in playlist
}
```

Fig 1: Sample R code snippet used to create a data frame containing tracks auto features (e.g. valence, duration, danceability) and other meta-data associated to the song (e.g. album name, album cover image, year)

Data Science with R

Programming was essential in the development of this project. R [1] is an open source language widely used in the data science community, with focus on statistical data analysis, data visualization and machine learning methods. During this project, tools from the `tidyverse` package [2] were used for data wrangling and data visualization with the help of RStudio, an open source integrated development environment (IDE) for R.
In particular we used the `ggplot2` package for *data visualization*, `dplyr` and `stringr` for *data transformation* and summaries, and `rvest` for *web scraping*. Additionally we used tools from the `tidytext` package [5] for text processing and encoding.

RStudio ggplot2 dplyr stringr

Fig 2 : RStudio logo. RStudio makes R easier to use. It includes a code editor, debugging and visualization tools. Logos for the ggplot2, dplyr, and stringr packages.

Principal Component Analysis and Clustering

Exploratory Data Analysis

We performed basic exploratory data analysis (EDA) to better understand the distribution of the songs audio features extracted from Spotify. The histograms below show the ranges of the duration, tempo and loudness of the different Billboard Summer Hits in the past 60 years.

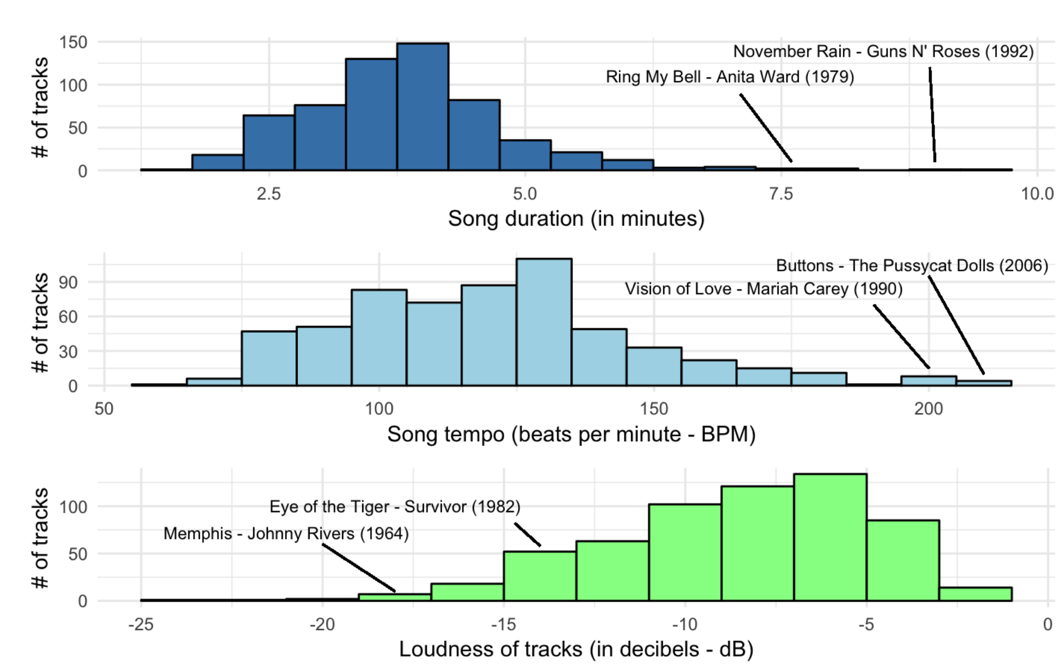

Fig 3: Average song duration, tempo, and loudness, respectively, of Billboard's Top 10 summer hits between 1958 and 2017.

All songs in the dataset for Billboard summer hits were grouped by decade, in order to analyze any significant changes in the different characteristics of songs that made it to the Top 10 list.

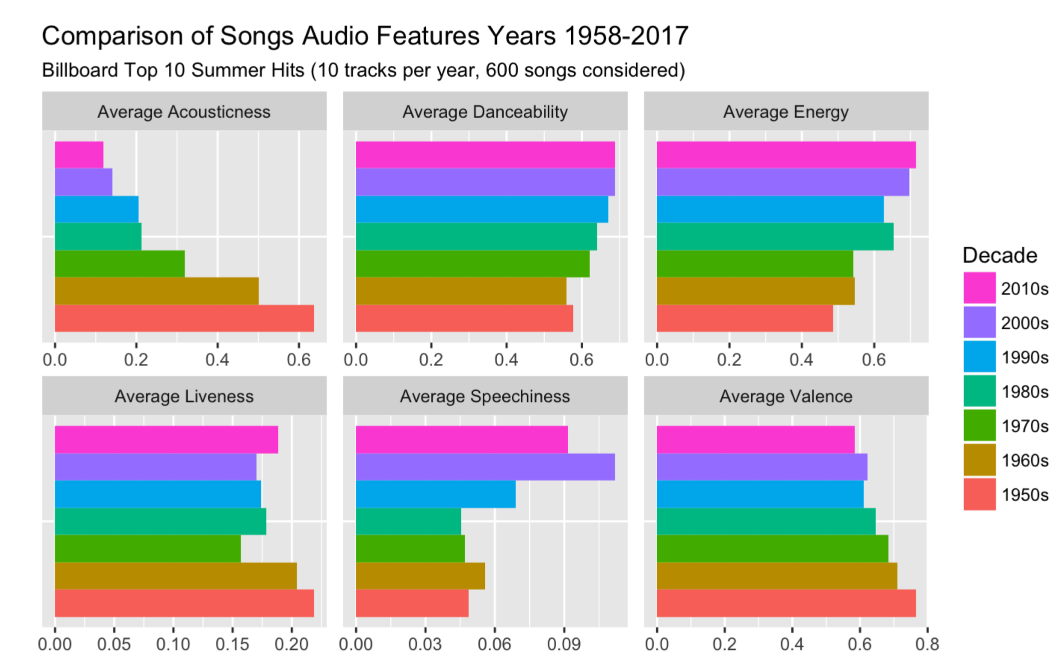

Fig 4: Comparative analysis of the average of each audio feature per decade. Notice that all summer songs considered had a low level of speechiness (< 0.1). Higher levels of acousticness in the 60s and 50s compared to summer hits after 2000.

Danceability	The appropriateness of a song to be danced to using a combination of different elements; tempo, rhythm, stability, beat strength, and general regularity. A value of 0.0 denotes the song to be least danceable, while a value of 1.0 denotes a song to be most danceable.
Energy	Variable used to describe the perceptual measure of intensity and activity, with ranges from 0.0 to 1.0. A high energy song typically sounds fast, loud, and noisy.
Loudness	Measured in decibels (dB). These loudness values are averaged throughout the entire track and are useful for comparing relative loudness. The range of loudness is from -60dB to 0
Speechiness	Detects weather or not a song has spoken words as well as how much. The more speech driven a track is, such as a talk-show podcast, commentary, etc. the closer to 1.0 the attribute value is. Anything below 0.33 represents a track where the vast majority is mainly music with little to no spoken words.
Acousticness	A confidence measure on a scale of 0.0 to 1.0. If a track has high confidence the track is represented closer to 1.0 where as if it has low track confidence it is represented closer to 0.0.
Instrumentalness	A measure of whether the song lacks vocal content or not. Vocal content is defined as being words or raps, but excludes the small words such as "ahh" and "ooh" as they are considered to be instrumentalness. Therefore, if the instrumentalness value is close to 1.0 then the song lacks vocal content.
Liveness	Feature that predicts whether the song is being performed live in front of an audience. After detecting whether the presence of an audience exists or not, a song is given a value from 0-1 in which a value closer to 1 indicates an audience is present.
Valence	Associated with the "emotion" of the song so a song can be described as being happy and having positive valence or sad and having negative valence. Spotify specifically has a music expert that categorizes sample songs by their valence which then they use machine-learning to apply it to other songs. It is measured from 0.0 to 1.0, songs that are closer to 1 are considered to be "happy/joyful" songs.
Tempo	Measured by the BPM of a song which stands for beats per minute. In other words, it measures the pace of a song or part of the song "and derives directly from the average beat duration". These values can range from 36 to 240 or 0.15 and 1 (by dividing by 240).

Principal Component Analysis (PCA)

A song is comprised of numerous variables that accounts for its rhythm, how fast it is, the loudness of it, among several others. Principal Component Analysis, better known as PCA [6], is a dimensionality reduction technique that allows for optimized visualization of high dimensional data by projecting key variables (components) that contribute to the highest variance.

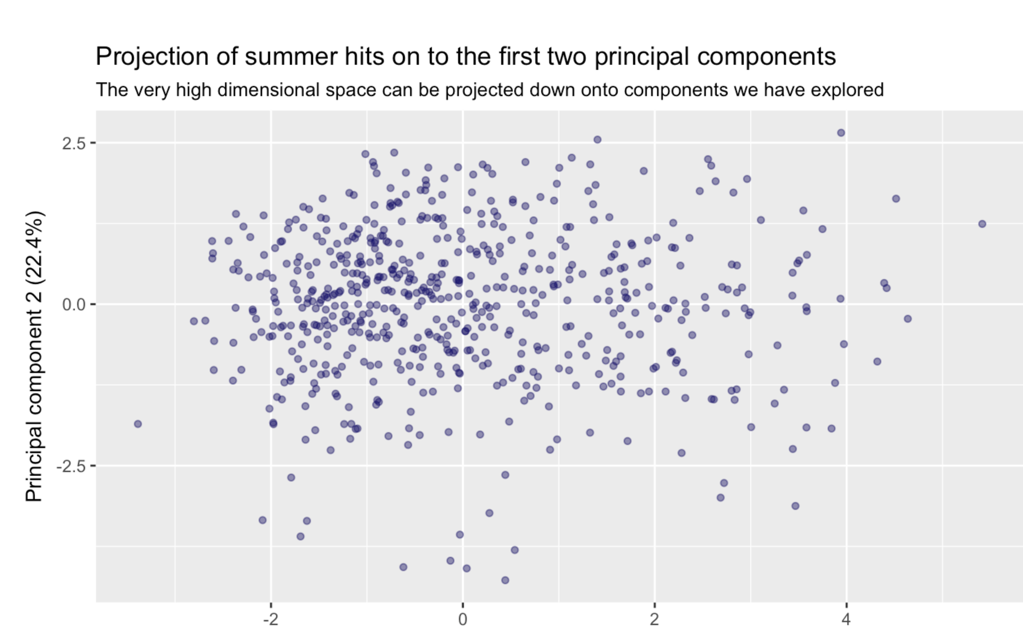

Fig 5: Acousticness and energy contribute the most to the 1st principal component; tempo and danceability contribute the most to the 2nd principal

Clustering

Clustering methods deal with finding similarities and structure in a collection of data points. We used K-means clustering [7], a distance-based method, that consists of creating clusters so that the total intra-cluster variation is minimized. In k-means clustering, each cluster is represented by its center (i.e, centroid) which corresponds to the mean of points assigned to the cluster.

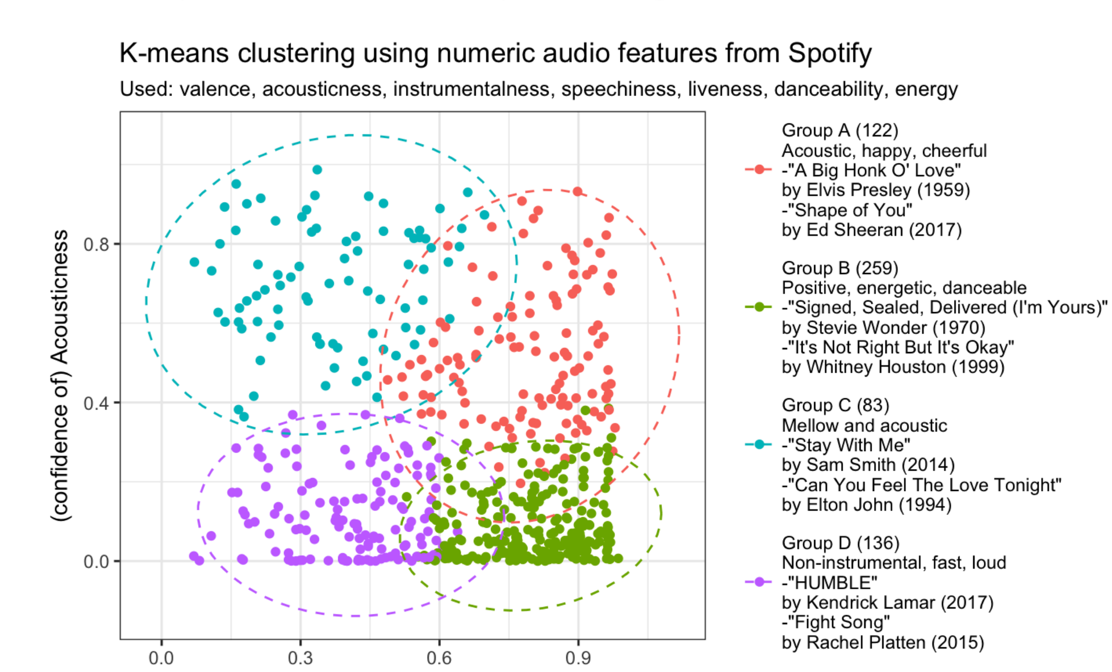

Fig 6: Song grouping based on emotions determined by audio features. Audio features in the interval [0, 1] were considered to cluster 600 different Billboard summer hits.

Trends

Data for Grammy Awards winners of different categories was collected using tools from the `rvest` package to perform web scraping from the Grammys official website (`https://www.grammy.com/grammys/awards`). Below the changes in different audio features are shown for the winning albums in the category of "*Best Alternative Music Album*" in seven consecutive years.

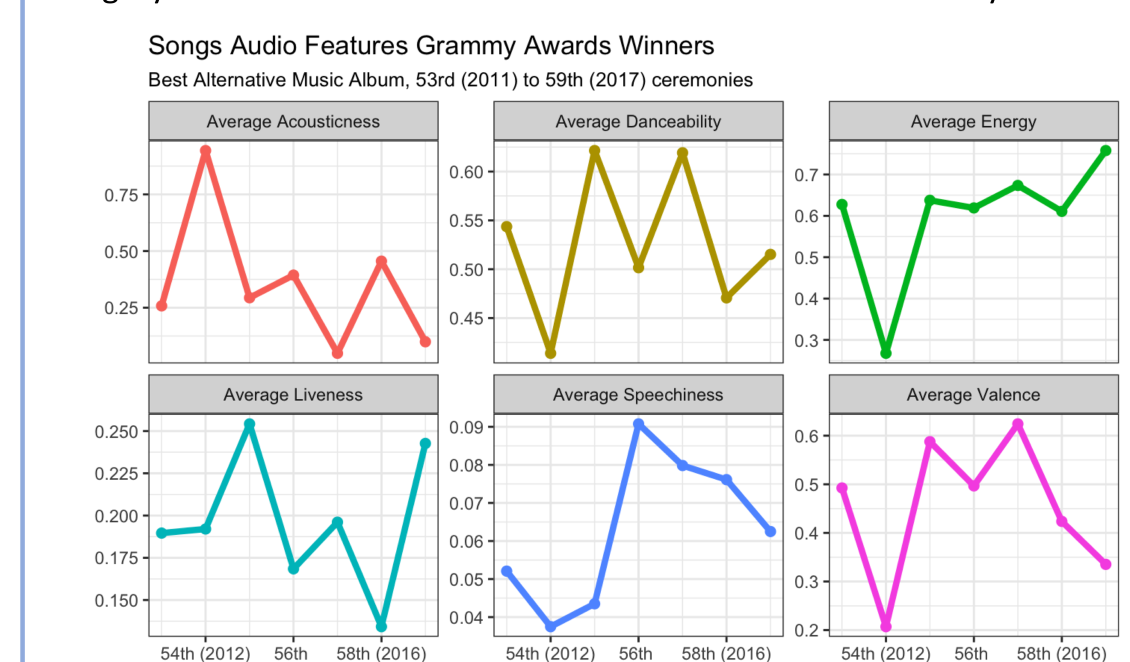

Fig 7. Winner album in 2012 has the highest acousticness: Bon Iver, an American indie folk band. Winner of 59th has Highest energy reported for *"Blackstar"* by David Bowie (winner in 2017) shows high levels of energy. Winner of the 55th edition has high levels of danceability: *"Making Mirrors"* by Gotye, a Belgian-born Australian multi-instrumentalist and singer-songwriter. In that album, the single *"Somebody That I Used to Know"* was the winner in the category of *"Record of the Year"*

The data we used for the Grammy awards winner analysis can be found at:
`https://github.com/reisanar/datasets/blob/master/all_grammy.csv`

Conclusions

The clustering technique developed four well-defined clusters in the acousticness/valence coordinate system (as shown in Fig. 7). This demonstrates which points are closely related to each other as well in the higher dimensional space and represents a structure in the data considered in this study. The analysis of different audio features helps understand patterns and changes in the music industry. This information can be used for the design of modern recommendation systems and predictive models.

References

[1] R: free software environment for statistical computing and graphics.
`https://www.r-project.org/`
[2] `tidyverse`: an opinionated collection of R packages designed for data science.
`https://www.tidyverse.org/`
[3] `spotifyr`: Pull Track Audio Features from the 'Spotify' Web API
`https://cran.r-project.org/web/packages/spotifyr/index.html`
[4] `rvest`: Easily Harvest (Scrape) Web Pages
`https://cran.r-project.org/web/packages/rvest/index.html`
[5] `tidytext`: Text Mining using 'dplyr', 'ggplot2', and Other Tidy Tools.
`https://CRAN.R-project.org/package=tidytext`
[6] Ringner, M. (2008). What is principal component analysis? Nature biotechnology, 26(3):303-304
[7] James, G. (2017). An introduction to statistical learning: With applications in R. New York: Springer.

Acknowledgements

This work was made possible by the support of the School of Science at St. Thomas University (STU). We also want to highlight the support of our research partner Acxel Vega from Miami Dade College, for always being present when needed, and the Faculty and Staff at the School of Science at STU. This project was supported, in part, by U.S. Department of Education grant award P03C1160161 (STEM SPACE). Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the respective funding agency.