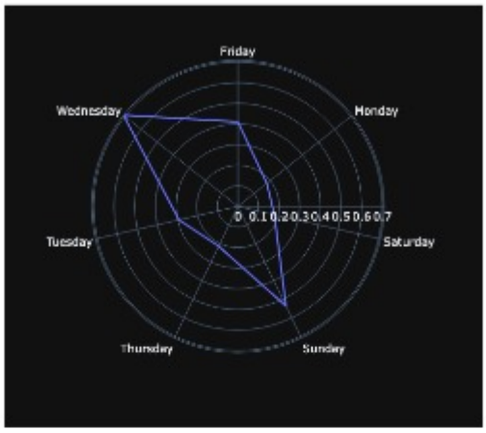
Milestone 2 - Sketches

1- Spider

Description Viz putting danceability, energy and other factors through each day, month, year...



MVP: Just one metric as we did in the mileston 1.

Full version: plot all metrics together and add a selector so the user can choose between: days of the week, month and years

How to achieve: we can use plotly and then add the others sizes. We can add a picker with the following options "days, months, years". We can use plotly

Lectures references

2- Map

Description: Show listening times as world as bubble chart

MVP: fixed window of time.

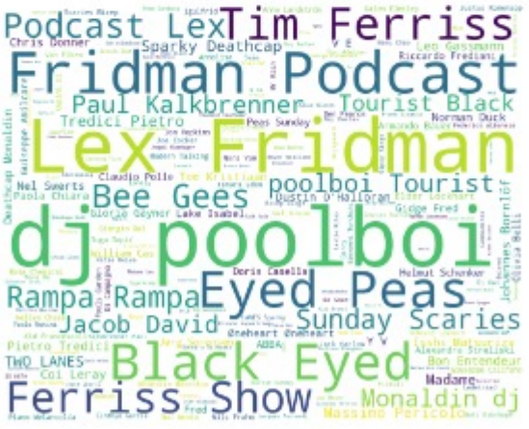
Full version: add a time bar which makes the map interactive through time.

How to achieve: use d3 or other higher level tools relying on it to achieve the map viz and the time bar. We get the location by the Spotify data itself. We could use an IP api to get a more precision location, to be able to point cities, no just countries.

Lectures references: Maps from lecture 9 and how to design one with practical lab maps

3 - Wordcloud artists/tracks

Description: *Wordcloud of most listened tracks and artists*



MVP: fixed window of time.

Full version: add a time bar which makes word cloud interactive through time.

How to achieve: word cloud library in python.

Lectures references: Tag cloud from Lecture 9

4- Most played tracks/artists (+ add filters such as country)

Description: *floating tracks/albums/artists most listened + filters: specific country*

Drake - Do for love

Beyonce - Renaissance

Hamza - Codeine

MVP: fixed window of time.

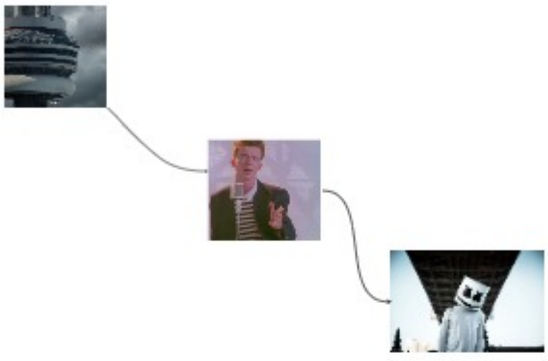
Full version: add a time bar which makes word cloud interactive through time.

How to achieve: JS dictionary that contains per entry frequency and country, then use filtering based on the user selection for country

Lectures references:

5 - Graph tracks relations

Description: *Explore the most frequent user's selection of music after listening to certain music. For instance: User listen Do for love -> Never gonna give you up-> Alone*



MVP: node graph where each node is a song listened to.

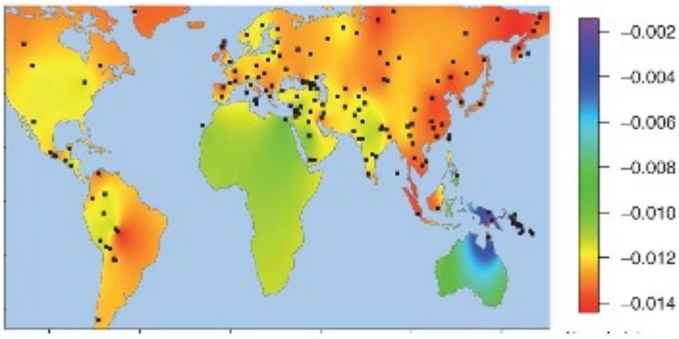
Full version: we want to ideally add interactivity.

How to achieve: use d3.

Lectures references: node link diagram Lecture 10

5 - State of mind

Description: *Plot your emotional/mind state (sad, party, focus. introspective...) in a map, where each color represents a mind state.*



MVP: node graph where each node is a song listened to.

Full version: we want to ideally add interactivity.

How to achieve: use d3, and use music features score to develop emotional state

Lectures references: Maps from lecture 9 and how to design one with practical lab maps

6- Horizontal bar chart animated with time

Description Show time spent on the most listened tracks and animate its sizes and positions in the rank as time goes by. Example: https://observablehq.com/@mbostock/most-popular-programming-languages-2004-2021

MVP: fixed window of time and no animations or time bar.

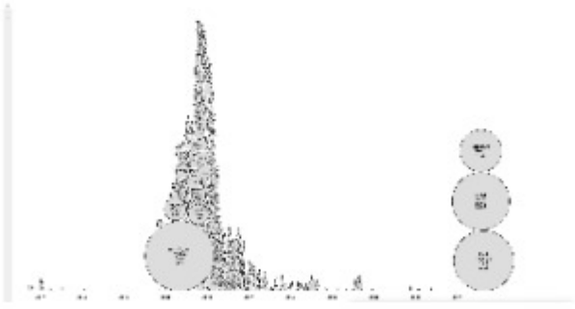
Full version: Add interactive and a time bar for browsing through time.

How to achieve: use d3 or other higher level tools to develop a horizontal bar chart which animates the time played through time.

Lectures references: Stacked bar charts from lecture 11

7 - Genres played through time

Description: *Kind of bubble chart with only an X-axis indicating the time (months). Each bubble represents a genre listened to, and its radius its determined by the time played in milliseconds.*



MVP: Only a bubble chart rendering without interactivity or time, only using a fixed time window

Full version: Add interactive and a time bar for browsing through time.

How to achieve: d3 rendering

Lectures references: