**COM-480: Data Visualization**

**Laurent Vuillon**

# The Colors of Songs

Alexander Apostolov
Valentin Garnier
Maina Orchampt-Mareschal

June 4, 2021

# Contents

# 1 Inspiration

Music doesn't have any barriers, it is part of every country and every culture. This is why we wanted to have a project that visualizes songs. We were struggling to find an idea that could be used in order to show music in an innovative, beautiful and insightful way, we didn't want to just have a plot or simple graph, we wanted something more original for this project. We had our *wow* moment when we saw a poster [1] that you can see on Figure 1, which was grouping books by the color of their titles and we decided that we could use the same idea for songs. The poster itself presents all books on the radius of a wheel, each one as a label colored with the respective color found in its title. We took this big wheel as a basic structure but we updated it in order to present it as a vinyl and decided to have more information about each song that accompanies it.


Figure 1: Poster that inspired the project. Books containing a reference to a color a depicted on a wheel.

# 2 Goal

The general goal of our project was to visualize the distribution of colors in the lyrics of popular songs according to the time period, genre and artist. We aimed at showing what are the most used colors in a nice manner and providing an interactive way for the user to manipulate and explore the colors in the songs of their favorite artist, genre or time period. For each song, we wanted to detect the colors it "contains" and for each song with at least one color, we put it on a color wheel with interactive displays. We wanted the user to be able to see details about each song by clicking on it on the color wheel. Also, we added a functionality that allows the user to "unravel" the color wheel into a histogram and see the same data through another perspective. The project targets a wide audience as music is a universal and timeless interest. According to us, it is a great and aesthetically pleasing way to visualize music in an uncommon manner using colors which could be interpreted as many different things in the context of songs and lyrics. On Figure 2, you can see a sketch of the objective visualization we had at the beginning of the project.

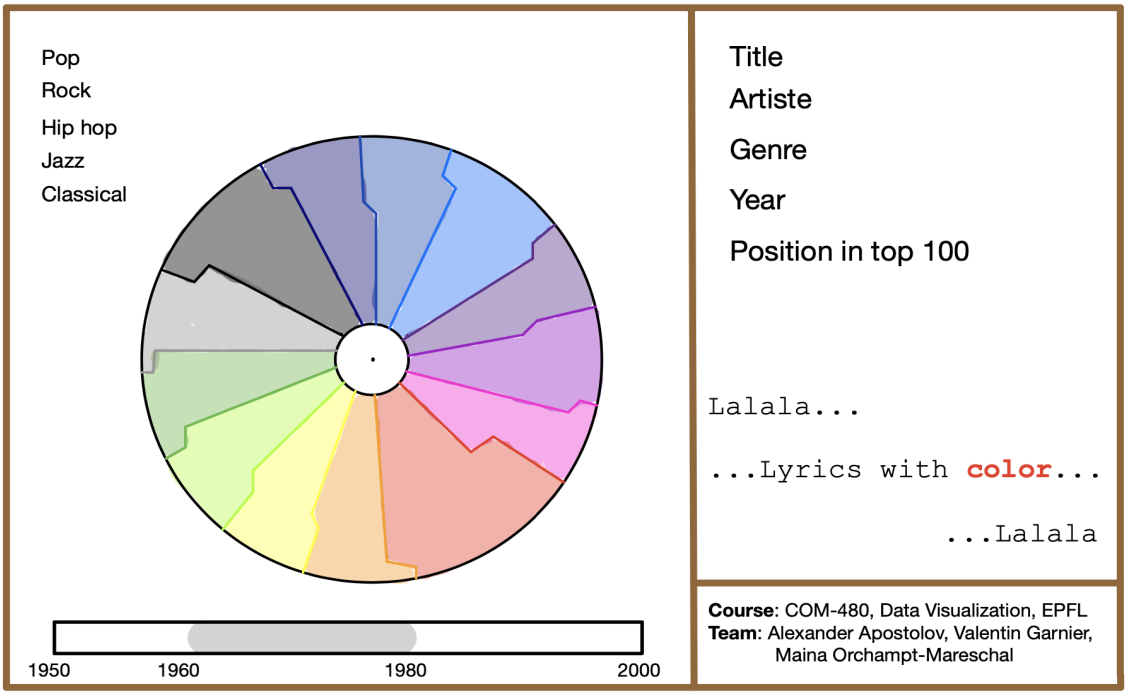


Figure 2: Sketch of the objective visualization

# 3 Cleaning and Preprocessing

## 3.1 Data Exploration

The dataset [2] we used consists of a subset of the top 100 songs from 1950 to 2015. After obtaining the data we performed some data exploration on it. It was already in a good shape. We had around 4000 songs and 1300 artists from 1953 to 2015 with the first years only having a small percentage of the top 100 songs. By looking at the sentiment analysis of the songs on Figure 3, we observed that we had more happy songs!

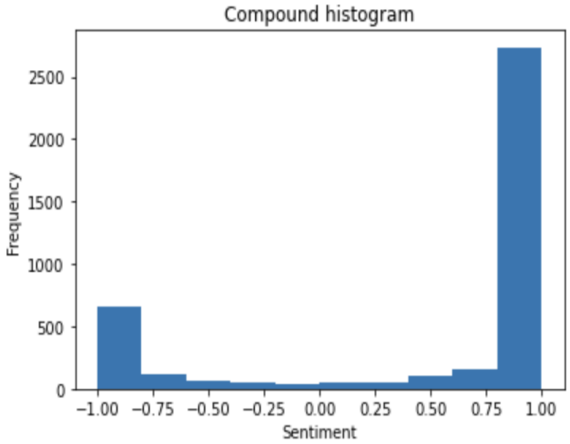


Figure 3: Sentiment analysis of the songs

## 3.2 Detection of Colors

We created a list of 24 colors that we wanted to find in the lyrics. To detect the colors, we simply lowercased the lyrics and split them by words. Then we computed the intersection with the set of colors and stored the result, if the intersection was not empty. We had to make sure not to detect garbage words such as *tired* for red, so we made sure that we do actually detect colors as whole words in the lyrics. As some songs contained more than one color, we *exploded* them into multiple rows. We thus had songs appearing multiple times in our dataset because either they contained multiple colors or they were in the top 100 songs for multiple years. Over the 24 colors we selected, the most represented were blue, black, red and white. In order to have a nicer looking visualization, we added some perturbations to the detected colors. This is done in order to avoid having big chunks of the final visualization having exactly the same RGB.

## 3.3 Genre preprocessing

The original dataset contained a dictionary that mapped all the genres associated to songs to general genres such as Hip-Hop, Pop, Rap...
For each song, we looked at all of its genres and took the first match from the dictionary of main genres, as we thought these main genres would be more useful for a user.

# 4 Main Visualization

## 4.1 Vinyl

The main visualization is a vinyl representing the colors of songs. It represents every song in the dataset as an interactive tile. Drawing the tiles was challenging as we had to make sure to have a similar looking vinyl whatever the selected time range. We had to think about how to deal with arbitrary numbers of songs to draw and decided to draw a potentially partial circle as the innermost circle of the vinyl in order to maintain the overall vinyl shape. The vinyl orders the songs by the hue that we calculated in the dataset.

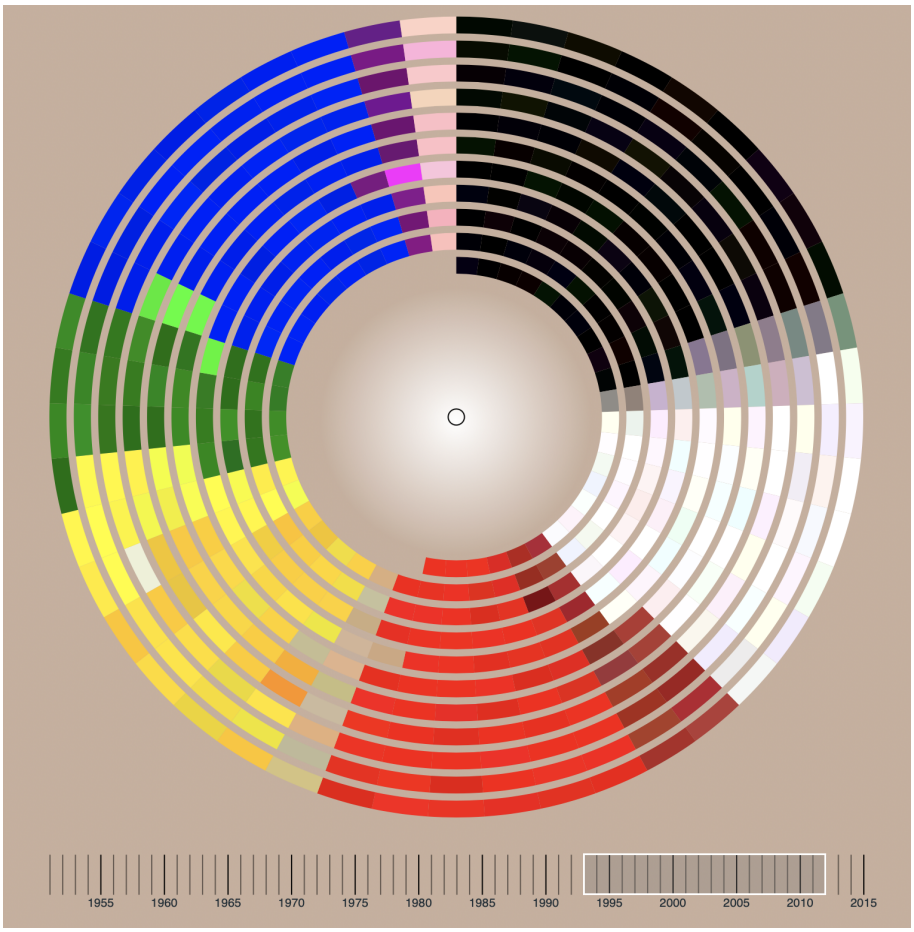


Figure 4: Screenshot of the vinyl

We needed to compute different variables such as the width of the tile, the radius, the space between circles. Every song is represented in the code by a *path* which returns an arc defined by a start angle, an end angle, the inner and the outer radius as well as information related to the song it represents.
Below the vinyl, we added a brush timeline which is fully interactive and allows to select a range on the displayed years. Since we allowed the user to filter on the year, we implemented

a way to redraw the vinyl every time and re-compute the number of circles necessary for the filtered data. Indeed, less songs displayed allow wider tiles. We realized that it was not really user-friendly to go back to the full time range with the default d3 brush, so we added a feature that allows a user to double-click on a selection and it goes back to the whole time selection between 1951 and 2015.

## 4.2 Histogram

We realized that the vinyl visualization might be misleading as it shows the color distribution as an area or angle, depending on how the user looks at it and the tiles have a different area depending on their distance from the center. As we saw in class angles and areas are not well suited for comparisons.

In order to add more value to the visualization and deal with this potential bias, we decided to add a histogram representing the same information through a different perspective. We thus avoid some biases as with a histogram, we can represent all tiles the same way.

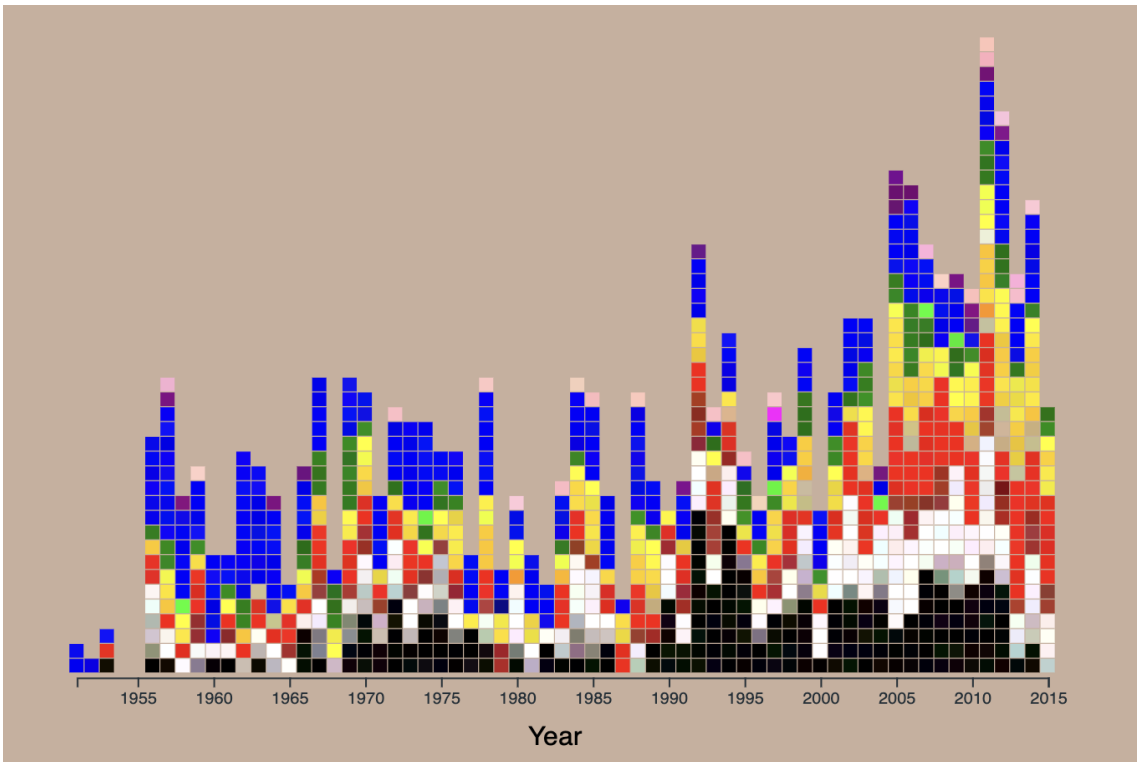


Figure 5: Screenshot of the histogram

Building the histogram required to add some preprocessing of the data first. We had to group songs by year and assign an index for each song in its respecting group. This index represents the y-axis coordinate of the song on the histogram and helps us to position tiles to their right place. Each tile is represented by a *rect* attribute. The height and the width of each tile was challenging to determine because of differences in the screen sizes. Hence we

decided to compute the height and the width relatively to the user's screen (as in the vinyl). We assigned a class to each tile in order to print information regarding the selected song.
We also decided to remove the brush timeline from the vinyl because we realized that filtering the histogram on the year produced irrelevant and not visually pleasing results. Indeed, when filtering on years, the width of each tile had to adapt to the screen which produced very long tiles. Finally, by including a histogram we had to create a way to switch from one visualization to the other. This is what the button at the bottom of the website is for. The challenge here was to keep track of which visualization is currently displayed using a global variable and the *click* event on the button.

## 4.3 Interactive features

Both visualization are fully interactive. First, each tile displays its corresponding song when clicked. From the song information, it is possible to highlight every song from the same artist by hovering or clicking on the artist's name. The user can also listen to the song on Spotify (or only a 30 secods preview if the user is not connected to a Spotify account). For this part, we had to do some preprocessing. Using the Spotify API, we ran for each song a lookup based on the title and the artist and retrieved the first matched URI. Around 50 songs did not give any results in the API, so we had to find the URI by hand.
The song information also includes the lyrics. In the lyrics, we highlighted all colors that are present for that song in the visualization.

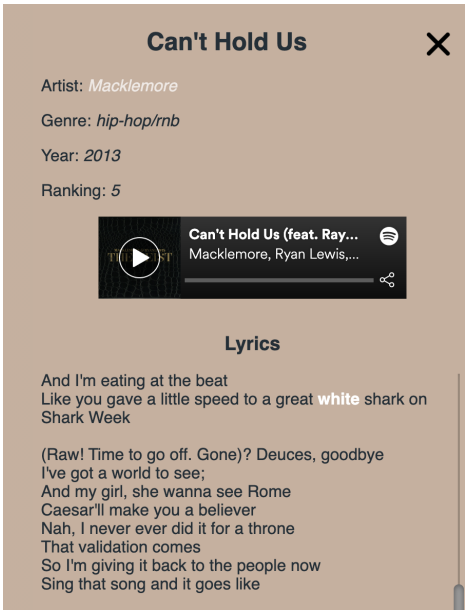


Figure 6: Screenshot of the song information

When the users selects a song or a genre, it is possible to come back to the original graph (histogram or vinyl depending on the current graph) by pressing the key Q (as in Quit) or the exit button at the top right corner.

### 4.3.1 Genres

The left panel of the website provides filters based on genres. For all songs displayed, we propose the five most represented genres.

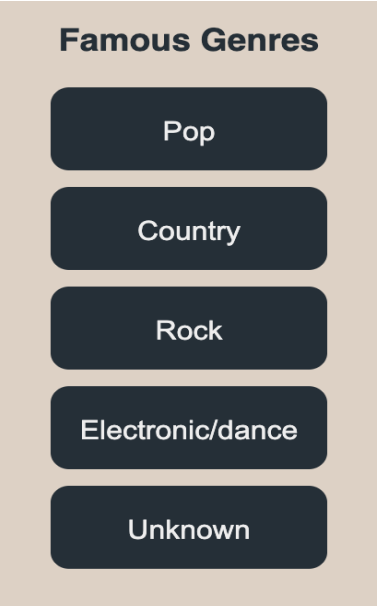


Figure 7: Screenshot of the genre panel

Note that if the time range is changed, the genre filters change as well to reflect the top 5 genre of the current selection. By clicking on a genre, every song that has that genre is highlighted. This last part was challenging since it was necessary to keep track of the genre currently printed with a global variable.
We also linked all tiles to a class representing whether the tile should be highlighted or not based on the genre selected.

## 4.4 Design choices

A difficult choice to make in this visualization was to choose the background color. Because we chose all the basic colors, we had to adapt the background so that each color stands out correctly. For this reason and after trying multiple backgrounds we decided that shades of brown were the most suited for our project. This allowed to also have a clear look on the visualization after selecting a song, when most of the tiles are a bit transparent.

# 5 Discussion

Some extra features that we proposed in the second milestone have not been implemented. For example, we talked about a small popup bubble that displays the song information above the mouse when hovering a tile. We decided to not include it. Indeed, we built this project keeping in mind that we should produce the most simple and uncluttered visualization with as much information as possible. Adding too much features would have produced some undesirable chart junk and after trying the popup bubble idea we realized it was not adding extra information and was not really useful. Moreover, we also had the idea to add a colorblind mode. We thought of many ways to implement it. However, whatever the idea we tried to use, we were left with huge Stroop effects as we were doing a mapping between a color and another color.

# 6 Peer Assessement

We all were very invested in the project since we liked the idea since day one so we globally worked on the project together most of the time. However we each spend more time on different tasks. Alexander was more focused on the vinyl, Valentin on the histogram and Maina on song information and data preprocessing. The design implementations was the product of our three collaborations as well as the left panel with everything related to the genre buttons.

# References

[1] The poster that inspired the project. https://www.wearedorothy.com/products/the-colour-of-books-original-open-edition.

[2] The dataset. https://github.com/kevinschaich/billboard.