

# Evolution of R&B

## Introduction

Our project aims at explaining the evolution of *R&B*. This term was first introduced in the late 1940s as *Rhythm and Blues* and has known some evolution to become what we now call *R&B* even though it feels very different to what it originally refers to. We wrote an article that includes some small interactive plots as well as a big visualization in the form of a scatter plot, where we can see how the subgenres of *R&B* evolved.

## Target audience

Our project is not aimed to nobody in particular, although we admit that it would be more interesting to people who like to listen and read about music history. It is more informative than it is useful. The article should be readable no matter how much knowledge you had on this particular subject already. The whole project can also be used to discover the roots of different subgenres of *R&B* and to discover new songs that had some influence on those subgenres.

## Leading idea

Our original idea was to evaluate lyrics' positivity across time and different genres of music. We took a list of songs from Billboard top 100 songs for each year. We were able to find the top 100 songs on different Wikipedia pages, so we could easily scrape this list. We used the Genius API to get the lyrics from all of these songs. We could then try to evaluate the positivity of those lyrics and this is what caused problems and why we changed our project. We planned on using natural language processing techniques to be able to give a positivity score from -1 to 1, and we tried using the Vader sentiment analysis tool which is really used for evaluating tweet, but unfortunately did not give good results for lyrics. We tried splitting the lyrics by groups of 4 lines and averaging the score for each group, or even doing this on each line in the lyrics, but the analysis tool often gave back results that were either very positive or very negative. Taking some specific songs that we knew in the dataset, we tried to see if the result was anywhere close what we thought would be the score of such lyrics, which it was not. The sentiment analysis tool is not supposed to understand the subtleties of song lyrics, which can include poetic formulations, that could mean nothing if not in context of the song. We had the example of "Here Comes the Sun" by the Beatles, the eponym line gave a neutral response, obviously because commenting on the weather as a statement is pretty stale, but if we as humans analyze it as lyrics, we can get the idea that this is a very positive thing. The Vader sentiment analysis is good when using simple words like good, bad. All of this meant that our project idea was going nowhere, so we had to find something new.

When scraping the songs from Wikipedia, we scraped their respective genres as well, but we wanted to generalize them so that we had a dozen of different genres ranging from *Rock* to *Hip-Hop*, passing through *Jazz*, *Metal*, *Disco*, *Funk*, etc. We had the list of genres we found on the Wikipedia pages of the song, and we mapped them to those more general genres. We did this with our knowledge, and if we found a style that we did not know, we went and listened to the songs to find which genre we would label them. From what we knew, we simply put *R&B* in the *Hip-Hop* genre, because this is what *R&B* of today comes the closest to. To our great surprise, when we plotted the curves of lyrics positivity by year and by genre, we saw that there were some data points for *Hip-Hop* in the 1950s and 1960s. After looking a bit why this happened, we saw that this came from the *R&B* genre that was originally closer to *Blues* or *Rock'n'Roll*, and that this term changed meaning quite a bit across time. Once we knew our original project idea was going into a wall, we figured it could be interesting to describe the evolution of *R&B* across time. This was a good idea because we could have a story to tell, starting from the origin of *Rhythm and Blues* in the 1940s to *Contemporary R&B* and *Alternative R&B*, see their difference, and try to explain the shifts of trends.

## Related Works

We did not find a related work per say, but during our research for the article, we came across some timelines that could be considered as related. The first one being <https://www.tiki-toki.com/timeline/entry/730116/History-of-Alternative-RB/> and the second one <https://www.tiki-toki.com/timeline/entry/567212/RB-Timeline/>. They are only close to our project because we include a timeline and we talk about *R&B*.

# Implementation

## Datasets

We could not find a dataset that had exactly what we wanted, because it was too specific, so we had to build one of our own. We let go of the idea of using Billboard lists after talking with the teacher because he told us that a lot of visualizations was using these kinds of set of songs, and he wanted us to do something of our own. We needed to find a list of songs that could go under the R&B genre, for all years between the 1940s to today.

Finding this was also too specific, so we had to do research on what genres was labeled as subgenres of R&B, and we came up with a list, which included some genres such as *Jump Blues* (for the oldest one) or *Alternative R&B* (for a more recent example). In the mean time, we gathered some information about the genres, like how they were introduced or when they peaked, to have a little story to include in the visualization (even if we originally did not know how we would include such a story at first). We found a website called [rateyourmusic.com](http://rateyourmusic.com) that listed released albums, singles, compilations and so on by music genres, where the genres were really what we were looking for. We then proceeded to scrape the lists to gather a small database of songs.

We now had to find a way to see how different songs differ, according to their style. The easiest way we found we could compare songs was with the features that were available in the Spotify API. These were originally provided by Echo Nest, a company that was later bought by Spotify, and consisted of small analysis of songs. With the Spotify API, you can request a track's features, and it gives you back a JSON file with information about the song key, its time signature, its mode (major or minor), its tempo and 7 more "obscure" features, namely valence, energy, speechiness, danceability, acousticness, instrumentalness and liveness. Here are the descriptions of these terms as provided on the Spotify API website:

**Valence:** *A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive (e.g. happy, cheerful, euphoric), while tracks with low valence sound more negative (e.g. sad, depressed, angry).*

**Energy:** *Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy. For example, death metal has high energy, while a Bach prelude scores low on the scale. Perceptual features contributing to this attribute include dynamic range, perceived loudness, timbre, onset rate, and general entropy.*

**Speechiness:** *Speechiness detects the presence of spoken words in a track. The more exclusively speech-like the recording (e.g. talk show, audio book, poetry), the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, either in sections or layered, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks.*

**Danceability:** *Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.*

**Acousticness:** *A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.*

**Instrumentalness:** *Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0.*

**Liveness:** *Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live. A value above 0.8 provides strong likelihood that the track is live.*

From the description, we could already have some idea as to which features will be more important to set apart the different subgenres, and which will not so much, for example, most songs are sung, and the speechiness is not often close to 0.

There were still some subgenres missing such as *Early R&B* (which was rather important for our story) from what we could get from [rateyourmusic.com](http://rateyourmusic.com), so we also scraped another website called [allmusic.com](http://allmusic.com). It worked similarly to [rateyourmusic.com](http://rateyourmusic.com) so the adaptation did not take too much time.

Either way, we now had a list of songs that could be labeled by a subgenre, and to which we attributed features, we had to find what we wanted to show, and more importantly how.

## Visualization

We first had ideas about simply plotting the evolution of features by genre and by year, to see that, on one hand, the genres tend to not have the same values for the features, and on the other hand, the subgenres may have themselves evolved across time. This was okay, but we thought that the result would not be very interactive. So, we had to review how we wanted to plot the results. We had an idea of a scatter plot where we would plot each song a point, color-coding the points depending on the subgenre, the coordinates of the point being two of the Spotify features. Since we wanted to see the evolution of the subgenres across time, we quickly had the idea that we needed some sort of timeline that we could interact with, that will change the year from which we pick the songs which came out this year, and that would redraw the plot. Figure 1 is a sketch of what we originally had in mind. The red arrows on the timeline represented years where we wanted to include some additional trivia about R&B such as a change of name for a style, the appearance of a new style, or the release of an album that had a great influence on a subgenre.

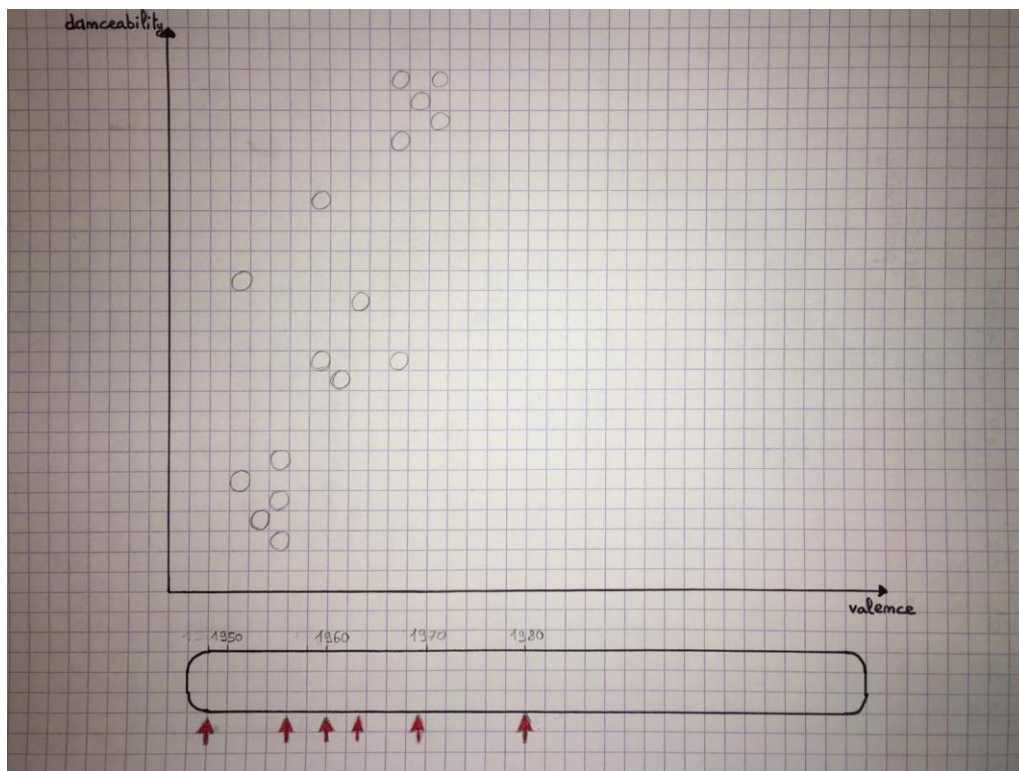


Figure 1. Original sketch for our scatter plot with the timeline

For the first milestone of the project, we digitized the sketch by redoing it on [draw.io](https://draw.io) and adding some other ideas. The first thing is that we say which year we are plotting on top of the graph. We also wanted to show which song the point corresponded to, so we had the idea of a card popping up when we went over the point with the mouse. The card would contain the picture of the album, the artist name and the title of the song. We also put what we figured would be a way to select the features, namely a drop-down menu. The additional trivia was also put on the right of the vis, with a link to the source of the information. The resulting draw.io file is shown on figure 2.

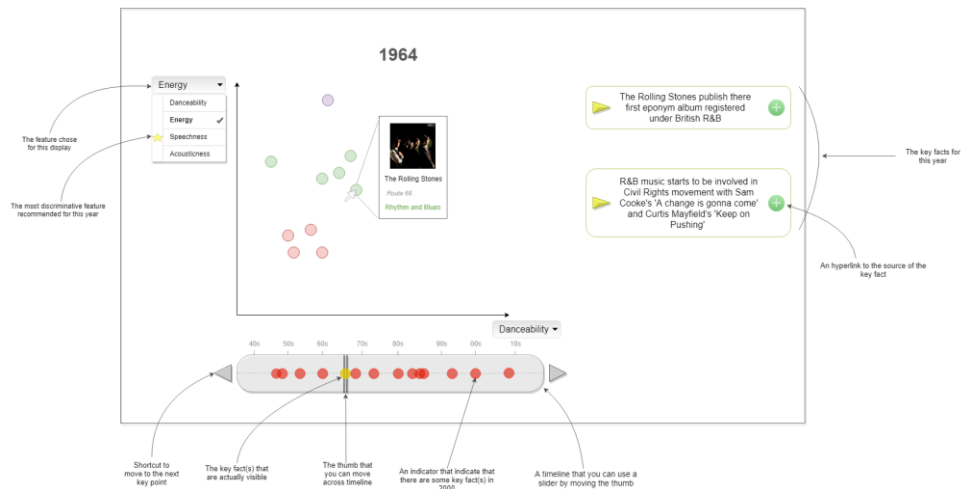


Figure 2. Digitization of our scatter plot idea

The idea began to be stable, so we started implementing it with Javascript and D3 as we learnt during the course's exercises. The first implementation was done without the timeline, but it was done so that linking it to a timeline would be easy (for example, there was a function to change the year of the plot, that would be called by the timeline, which was implemented but not used at first). The plot also had no way to change the features on the axes, although it was doable in the code. We had a HTML page with only a SVG inside, where we would draw everything. If we hovered a point, then the card would show up as expected. The information cards were statically drawn on the right. As we can see on figure 3, we experienced with a dark background at first.

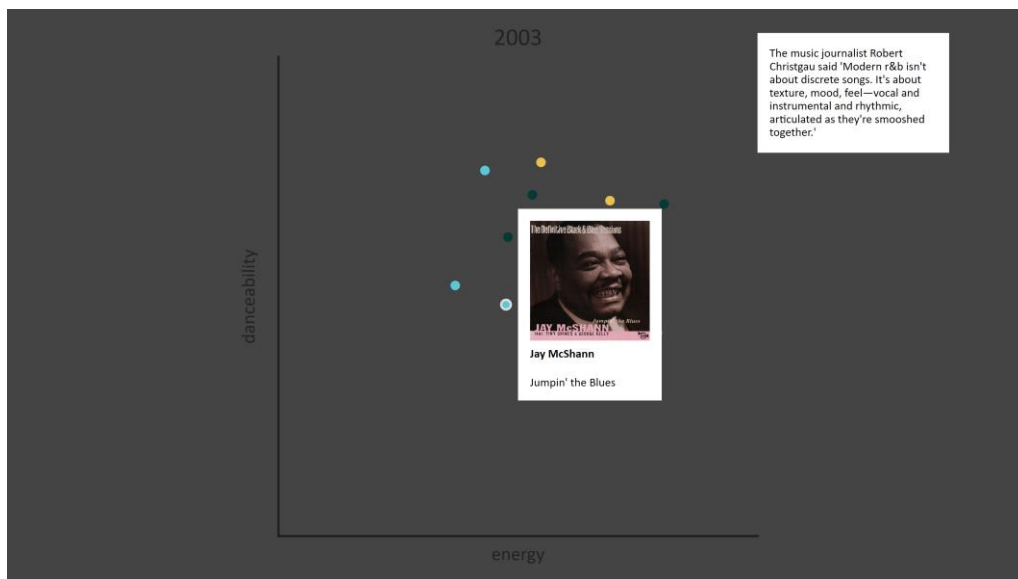


Figure 3. First implementation of our scatter plot with Javascript and D3, focusing on a song

An updated implementation can be seen in figure 4, and it has some small differences with the original plot. There is a small key of the top-right of the plot that was hoverable and that would show the cards with the information on the right. We reverted to a white background because the colors were showing up better on this background. We also added the small legend on the bottom left.

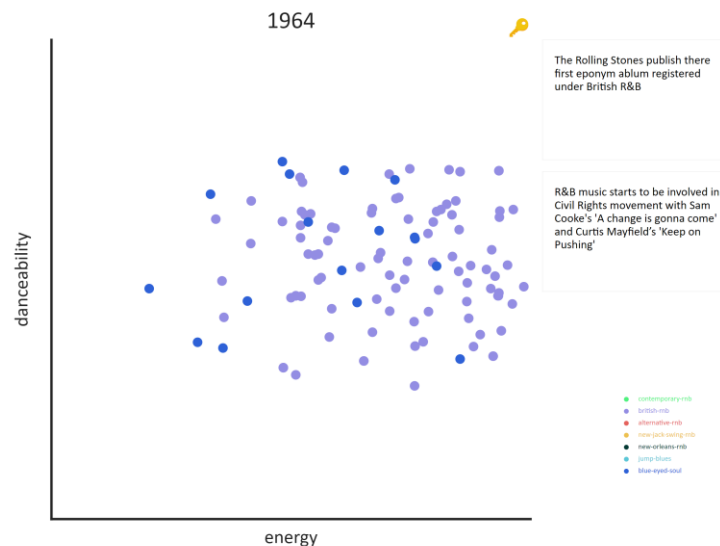


Figure 4. Updated scatter plot, focusing on the key

A rather important change would then come as we did not originally think of it. The Spotify API includes a preview of (almost) all songs which lasts 30 seconds. We thought it might be a good idea to include this preview somewhere, so that people could have an idea as to why the point was at this place in the graph, for example hear what consisted of a high energy value. We had the idea to transform the points when focused into little play buttons, where pressing on it plays the preview, and re-clicking pauses it, and defocusing it stops it. This added some interactivity that was pretty intuitive. This step of the visualization can be seen in figure 5. We also added a way to change the features with the two arrows we can see next to the labels. The legend was changed so that it only shows genres that are present on the plot for this year. Another major change was that the information card on the right was now an HTML div with a paragraph inside, customized with CSS. Up until now, we had the card in SVG and drawn by hand, one rect, and text that we had to wrap (which was not that easy). It was easier to do it with HTML and CSS because the card automatically fits the content, which was not the case when we drew it in the SVG. We show in figure 6 the display of the webpage.

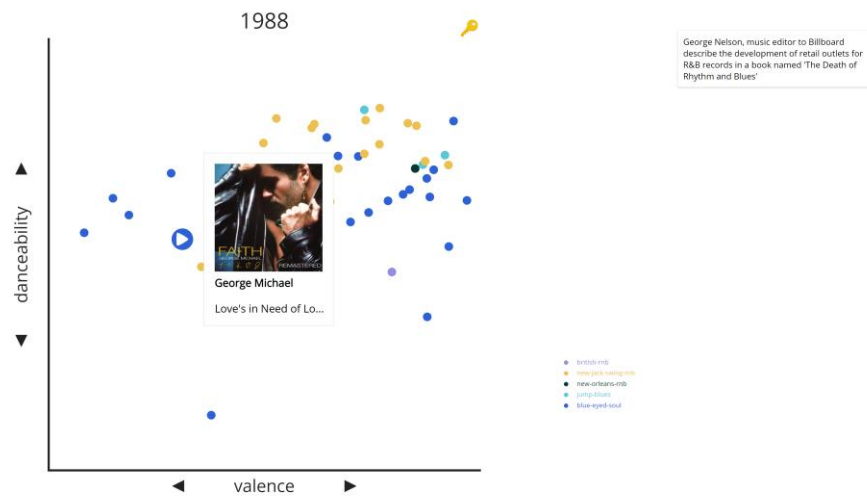


Figure 5. Further updated scatter plot with little play button

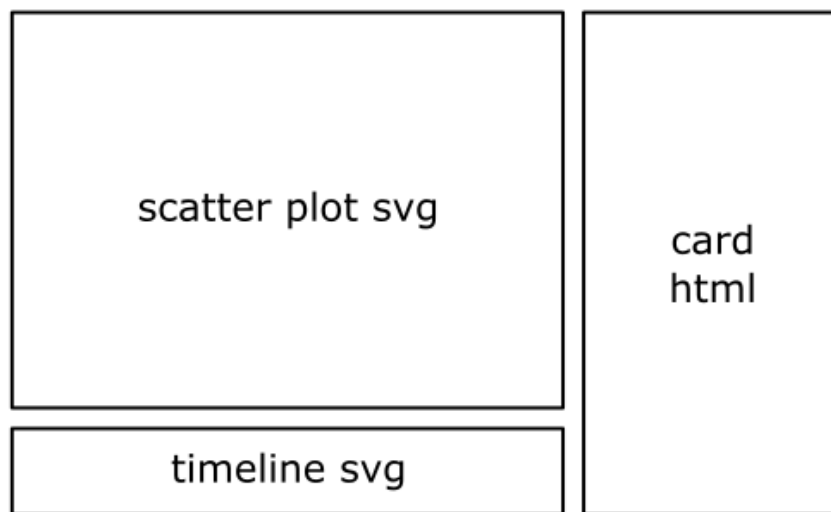


Figure 6. Display of the webpage

The plot was beginning to look good, so we wanted to add the other big part of the visualization, namely the timeline. The timeline is a big slider with elements showing that there is information about the year they are in. The first version of the timeline had crazy colors just because we did not spend time to make it look good, but we rather focused on the implementation so that it was working correctly, which it did. Some small changes included the color of the arrows to change the features, they are light gray when unfocused, and gray when hovered on, so that we understand that we can click on them. We also changed the color for the information panel on the right to pink, because we



wanted to put it a little forward. Obviously, we realized that pink was of poor taste, as we can see in figure 7.

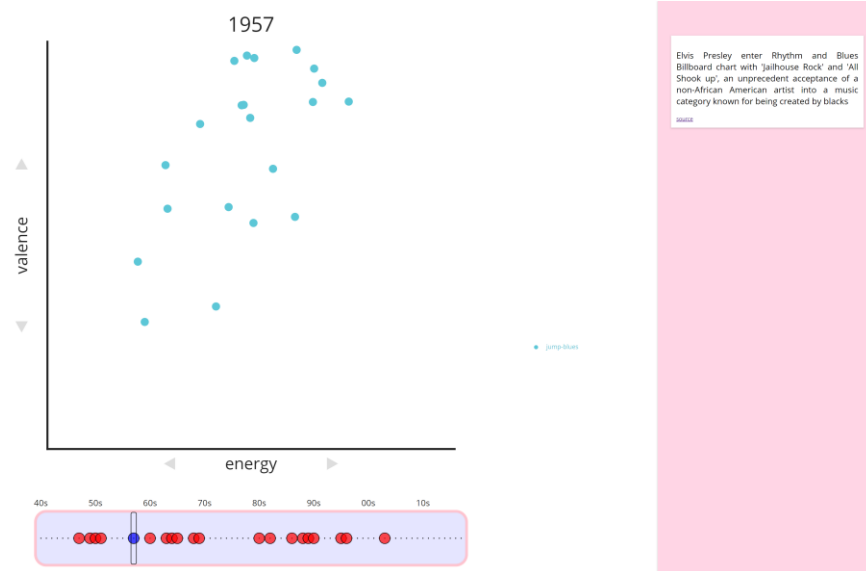


Figure 7. Integration of the timeline with the scatter plot

We had to do something with the timeline, as it was really bad looking. We had another draw.io where we exposed 2 ideas, one simple where we only changed the colors in the timeline, one where we used images of small vinyls instead of the dots. We can see the draw.io result in figure 8.

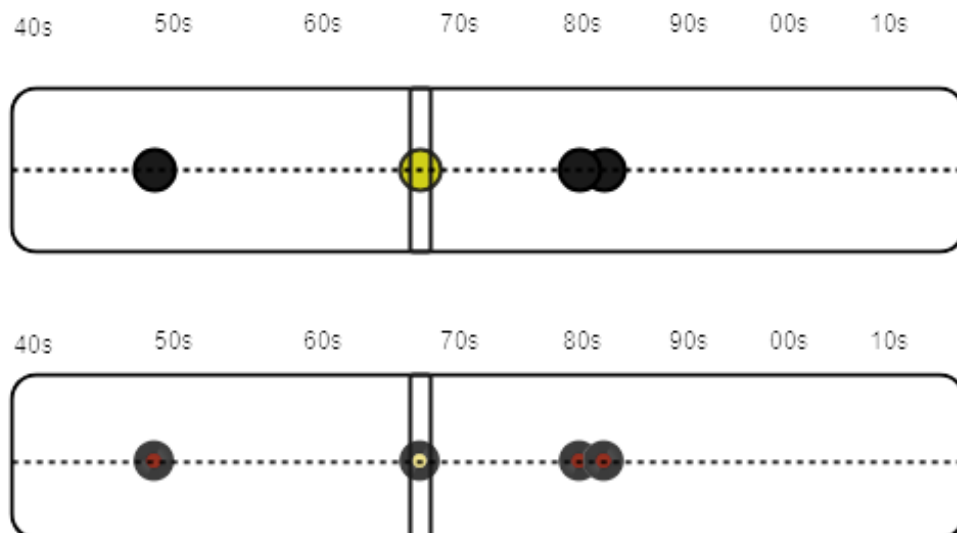


Figure 8. Sketches of redesigns of the timeline

At this point we felt like our visualization was getting bland (apart from the bright pink on the right side), so we figured the second sketch could be bringing some nice balance. We added the small images of vinyls in the information card to reinforce the link between the two. A small change was to draw the legend a bit closer to the actual plot because it felt kind of wrong where it was. The look of our last implementation of our plot is shown in figure 9.

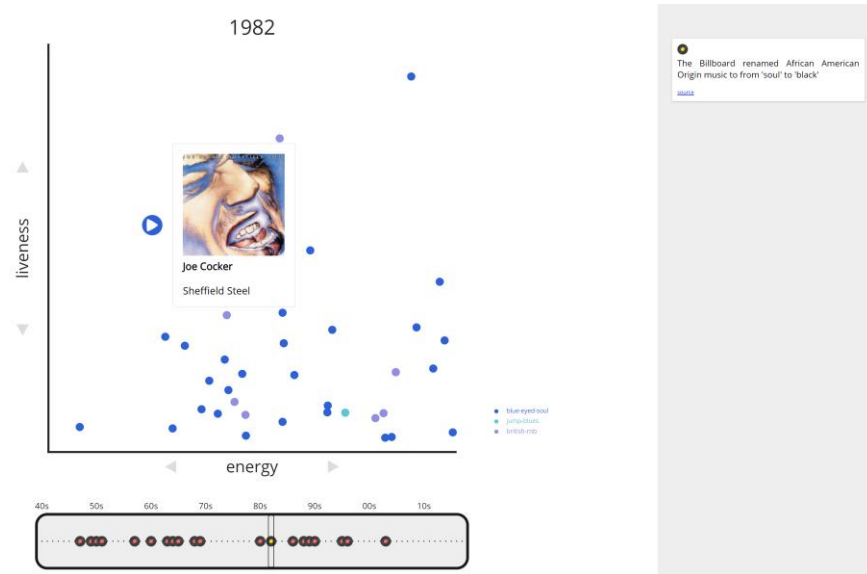


Figure 9. Final implementation of the scatter plot with the timeline

We came to the point where we had a visualization that we did not know how to improve, but we were not really satisfied with it either. The biggest problem of our vis is that we could not easily tell the story we wanted from the beginning.

We thought about having a small tutorial when arriving on the website. We wanted a way to show how the vis worked as well as the most interesting things we could explain with the vis. In the course, we talked about such as technique called the Martini glass where we first have the user do what we want, constraining him to do operations that we decided beforehand. This could have been done in two ways, either we suggested him to click on an interesting year and to choose interesting features, or we could have a automated script that did it for the user, so that the user does not have to do anything while the script was doing it thing. Each choice was pretty equivalent, but we were restrained on what we wanted to say for the story, such that we did not pollute the vis by adding loads of information everywhere.

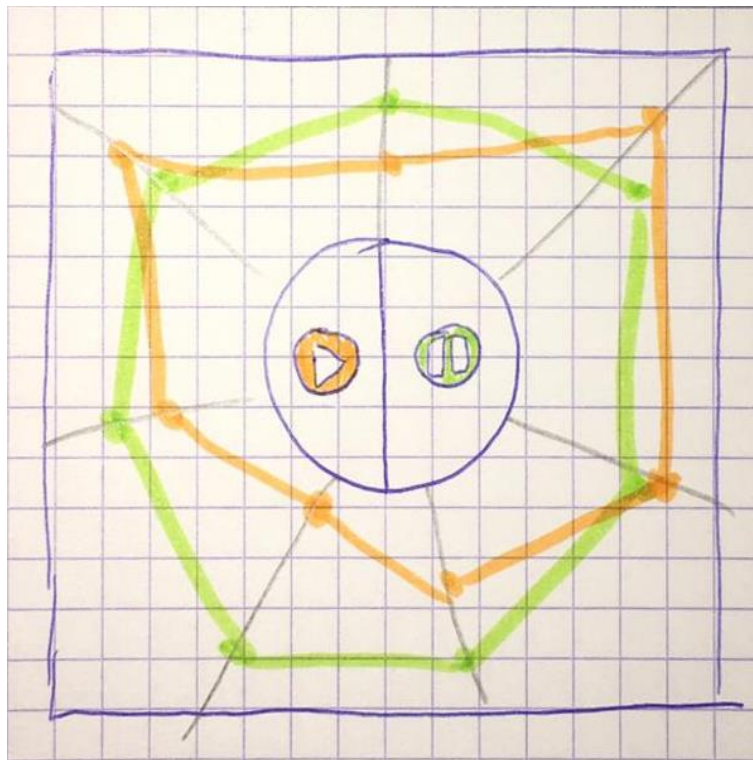
We talked to the teacher who told us that we should write an article instead of having this big, empty visualization. He told us to have an article like we could find on the New York Times, but we could insert some interactive plots inside the story to support our points. This meant that we could tell the story without polluting anything, we could get other small visualizations (like those interactive plots) inside the story without having to click

anywhere, and we could keep our scatter plot for the end of the article, when people have enough knowledge about R&B to be able to look for stuff themselves.

The whole website for the project would then simply be a narrow-columned article with some pictures and plots, which was pretty easy to design. We could do everything with HTML, and we added some SVG when we wanted some plots or visualizations.

We added 2 visualizations for the website, the first being simple and the second amore complex. The first visualization is the preview of some songs we talk about in the article. We have the play buttons as in the graph, next to the cards, so that the user can listen to songs we talked about, and he can also familiarize with the environment (since the cards and play buttons are the same). This small vis is shown in figure 10 [include preview](#).

The second added visualization is an idea that we got late, but we thought might be interesting to have. We started by sketching what we had in mind. The goal was to be able to compare 2 albums with the Spotify API features. We though that a circular plot could be good, the sketch for what we had in mind in included in figure 11.



*Figure 11. Sketch of album comparison visualization*

The design changed a bit during the implementation. The final product for this vis is an interactive plot, where you can see the values of the features for an album (the values are aggregated on the songs). We put a vinyl image as background because we it looked good with the central circle images. When we focus on an album, we see all the values at once,

as well as the name of the artist and album. Figure 12 shows what it looks like when unfocused and when focused.

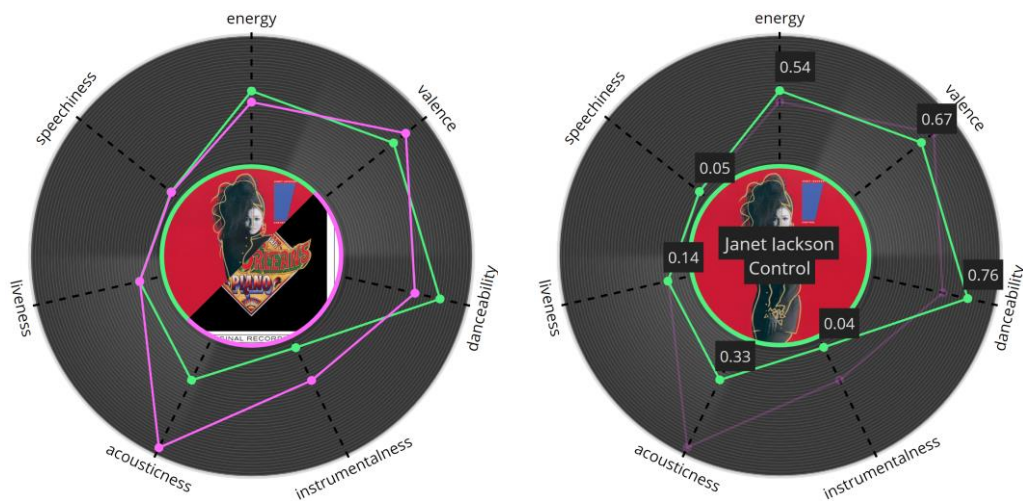


Figure 12. Visualization for album comparison, unfocused and focused versions

## Final product

Our final product is then a digital article that could featured composed of mostly text, but sprinkled with visualizations to help tell the story of the evolution of R&B. It is available on [thibaudperret.github.io](https://thibaudperret.github.io) but there is a preview in figure 13 **include preview**.

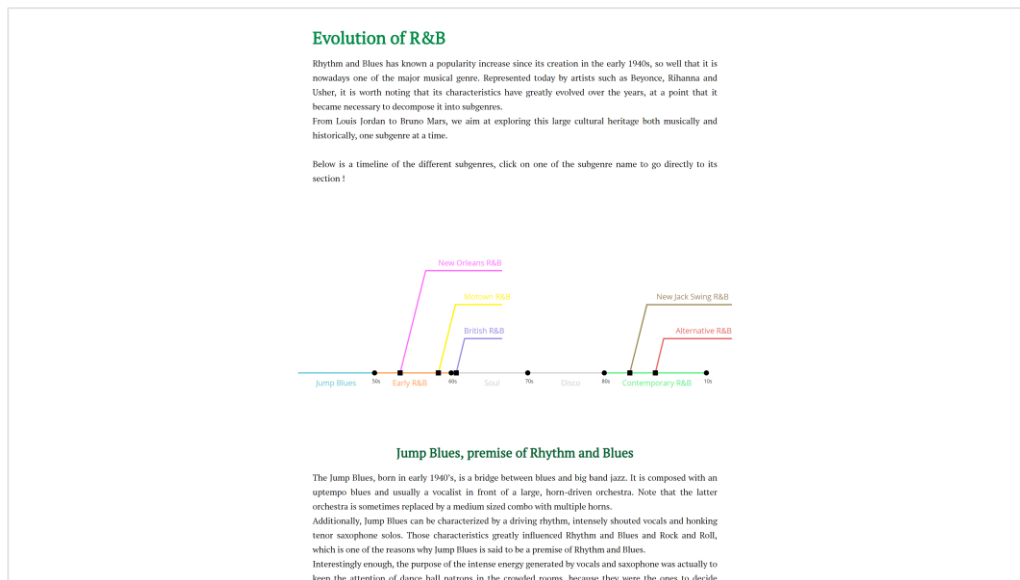


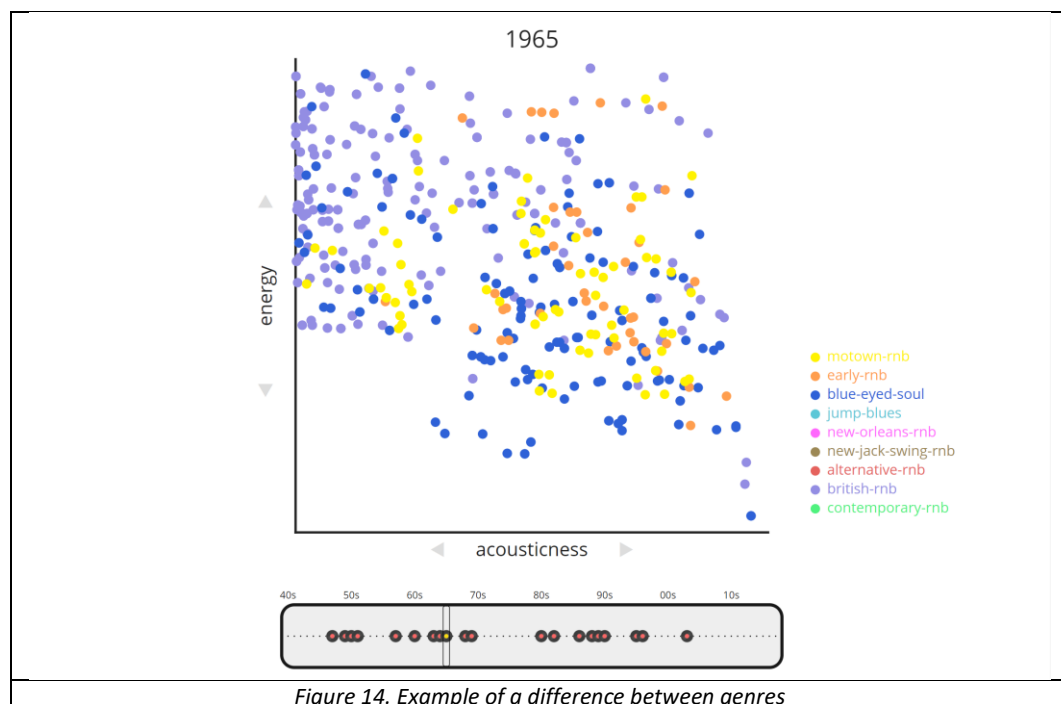
Figure 13. Preview of our article

## Evaluation

Our project was not supposed to answer a problematic, rather it was supposed to help tell a story. We wanted to see if the Spotify API features could help us in this way. The big scatter plot has examples where the intuitive feeling we get about different subgenres is really conveyed. We can see such an example in figure 14. We see that *British R&B* is more energetic than *Blue Eyed Soul*, but way less acoustic, because we get all kinds of electric instruments which do not contribute to the acousticness feature. *Early R&B* is also energetic and more acoustic, which sets it apart as well. Finally, *Motown* is pretty similar to *Early R&B* for those features.

The album comparison visualization as in figure 12 also shows a difference between two albums of different genres. We see that Janet Jackson's album (which is *Contemporary R&B*) is less acoustic and instrumental than Professor Longhair's *New Orleans R&B* album. However, it is more danceable to, which goes along pretty well with the story we tell.

Overall, our visualizations are great support to a story even though it does not answer any new questions.



## Peer assessment

### Thibaud

Preparation: We did not prepare before meetings per say because we are very often together, so we talked a lot even though we were not working.

Contribution: Yes, everybody contributed the same amount of work. We split tasks and were able to work in parallel very efficiently.

Respect for other's idea: Yes, very much respect.

Flexibility: We were flexible because we debated, and since we were a group of 3, we could vote by majority.

### **Lucas**

Preparation: The meetings did not require any special preparation, we simply had our own topics to work on when we weren't together and tried to progress as much as possible.

Contribution: We contributed the same to the project, the work load was well balanced.

Respect for other's idea: There was no hierarchy in the group. We did not really study data visualization before the course, thus we all had the same amount of experience and our ideas were very worth sharing.

Flexibility: We are a flexible group, we changed the project direction once and we were all open to debate.

### **Raphaël**

Preparation: We didn't prepare at all before team meetings as we were constantly speaking about the subjects and the project ideas we had, we weren't really preparing the meetings, but the meetings were follow-up of the talk we had during the week.

Contribution: We all three contributed equally to the project and each member had its strength, like more music background or more aptitudes in website development, so the work load was balanced.

Respect for other's idea: The three of us added its contribution in term of ideas, and some have been accepted other rejected. But good ideas have been encouraged like the circular plot idea from Lucas that we all trusted in.

Flexibility: There were a lot of flexibility in the group and the three of us did some concession about the initial ideas we had of the visualization would be, so the team work was really pleasant.