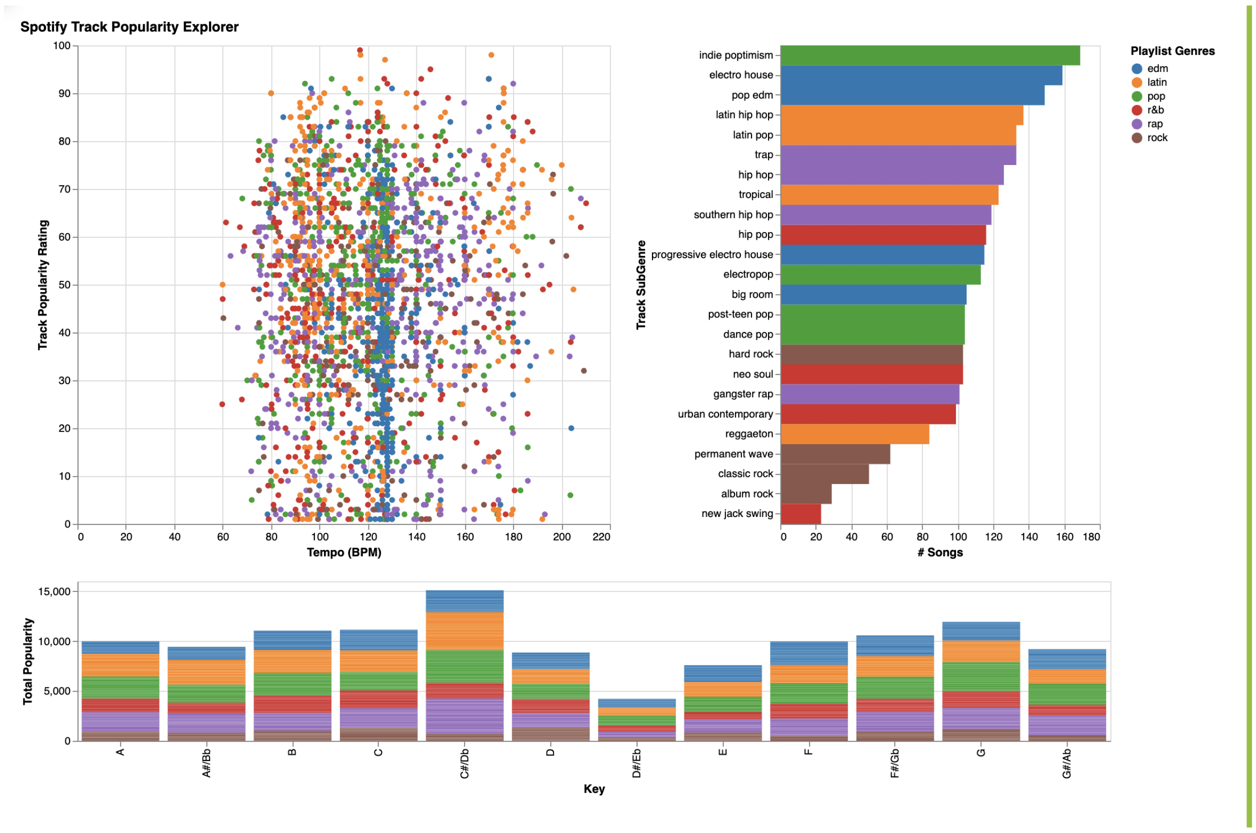
COMP40610 Visual Exploration Tool Design Document - 20456522

Student Name: Alex O’Donnell

Student Number: 20456522

Title: Spotify Track Popularity Explorer

Screenshot:



Dataset overview:

This is an adaptation of the dataset provided by Tidy Tuesday (link included below). The original dataset provides in depth details and statistics on over 30,000 songs collected from Spotify using the `spotifyr` , a package that makes it easier to collect your own data or general metadata of songs from Spotifys API.

> https://github.com/rfordatascience/tidytuesday/blob/master/data/2020/2020-01-21/readme.md

The dataset contained features of each track including Track Name, Track Artist, Track Key, Track Tempo, Track Genre, Track Sub Genre, and many more interesting features that I decided not to use so I could focus on highlighting the correlation between the original features I mentioned.

I used my own python script to carry out some pre processing steps before tackling my assignment. The python script is included in the overall project folder. Firstly, I merged the `Track Name` and `Track Artist` features so I could enhance my ToolTip within my charts so that it would display both the name of the song as well as the artist that performs it when hovering above the entries in the graphs. Secondly, for simplicity I cut the size of the dataset down to 10% of the original size because the original dataset contained over 30,000 rows. However, the fraction of the dataset of which we are using is adjustable and is highlighted within my python script. Finally, I remapped the `Key` feature of each track so that keys were now represented by a String rather than a number (i.e C instead of 1, etc), changing the data from numerical data to ordinal data.

Design Considerations

Overall Goal: My overall goal enable the exploration of the correlation between the tempo (Beats Per Minute), Key (Key Signature), and Popularity (Popularity Rating) of a track on Spotify.

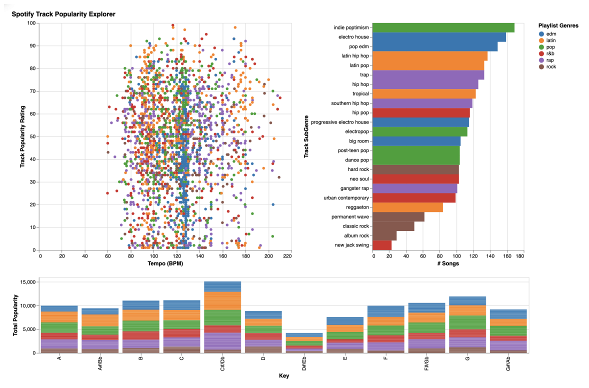
Scatter Plot: This is a scatter plot shows the correlation between the popularity of a song and its tempo. The tempo of a song corresponds to the overall speed of the song, and is measured in beats per minute (BPM). Each song is represented by a dot the is coloured according to the genre of music the track falls under. The range of genres are all stated in the legend. There are 6 distinct colours representing 6 distinct genres, which is a recommended quantity of distinct colours when dealing with colour encoding. This approach makes it easy to identify the correlation between the tempo of a song and its popularity, while simultaneously highlighting the role in which the songs genre plays in this. For example, its clear from the high density of blue on the 120-130 BPM mark that the average EDM song in this dataset falls in around the 125 BPM mark, but the most popular EDM songs in this dataset seem to fall between 160 and 180 BPM mark, achieving significant popularity values between .85 and .95. Despite being very helpful, informative, and easy to read, a common disadvantage of using a scatter plot for this type of data is that it is prone to over-plotting. My solution to this was to trim my dataset by 90% using my python script prior to plotting the graphs. This was crucial because over plotting scatter plot in this case would make it very difficult to firstly read the results efficiently, but also make it difficult to identify a particular plot that you are attempting to inspect solely, which is a feature I have enabled using Tool Tips. Its clear from the results that these encodings and channels for this dataset are both effective and expressive, because all of the data is encoded and the information in the chart can be readily decoded by the reader.

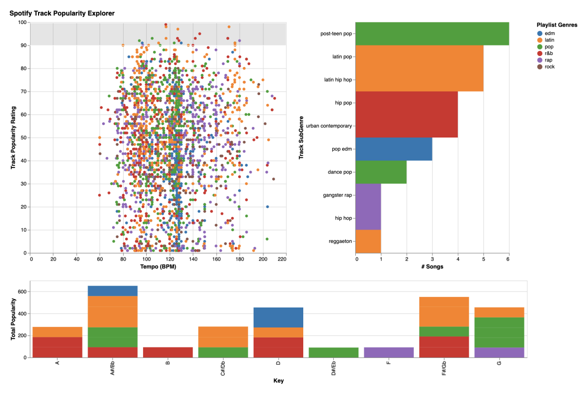
Bar Chart: The purpose of this bar chart is to add more depth of understanding as to why the tracks highlighted on the above scatter plot may be popular or unpopular. Instead of highlighting the correlation between the popularity of a track and its tempo, this bar chart showcases the correlation between the popularity of a song and the key in which it is in (In music theory, the key of a song refers to the set of pitches or notes that a piece of music is centred around). The X axis represents the range of keys, and the Y axis represents the total popularity sum generated by the corresponding keys. Similar to the scatter plot, the songs are also colour encoded according to the genre of the song. This approach, like the scatter plot makes it easy to identify the correlation between a songs key signature and its popularity, offering a more complete understanding to the user as to the possible underlying factors that may contribute to the popularity of a track on Spotify. For example, its clear from the bar chart that the key C#/D Flat generates the highest amount of popularity among Spotify users, whereas D#/E Flat generates the lowest amount of popularity. With the power of the colour coding by genre added to this bar chart, we can also see that there is quite an even distribution of keys among Rock genre tracks, whereas it is clear that C#/D Flat seems to be the most popular key for Latin genre tracks. I also implemented the same feature as the scatter plot in highlight any track & artist tuple you would like with your cursor for enhanced inspection. Instead of using a bar chart I could have used a line chart to represent the key/popularity data just as clearly, but I decided to use a Bar Chart so that I could add the colour encoding per genre for additional information visualisation. An advantage of using a bar chart in this case is clearer separation between categories, which is important because its vital we distinguish the difference between different key signatures being used in this case.

Histogram: The purpose of this histogram is to show the distribution of sub genres within parent genres stated in the legend. There are 6 parent genres, and each parent genre has 4 corresponding sub genres, and the histogram accurately depicts the distribution of sub genres across the the scaled down version of my dataset (As I mentioned previously, I trimmed the original dataset as it was extremely large). The sub genres are also colour coded by the parent genre so that it is easier to understand for the user which sub genre belongs to which parent genre. Its clear from the histogram that `New Jack Swing` , from the `R&B` genre has the least amount of occurrences in this dataset, whereas `Indie Popitism` from the `Pop` genre has the most occurrences, closely followed by `Electro House` and `Pop House` from the `EDM` genre. This could have been implemented using a Pie Chart either, which would have visualised the data just as efficiently, but I felt a Histogram was better suited as it is easier to compare the size of the bars beside each other rather than attempting to determine the size of segments in a pie chart, especially because I ordered them in ascending order of quantity of songs in the histogram. The ease of comparative visualisations for different quantities is a common advantage of using histograms to visualise your data.

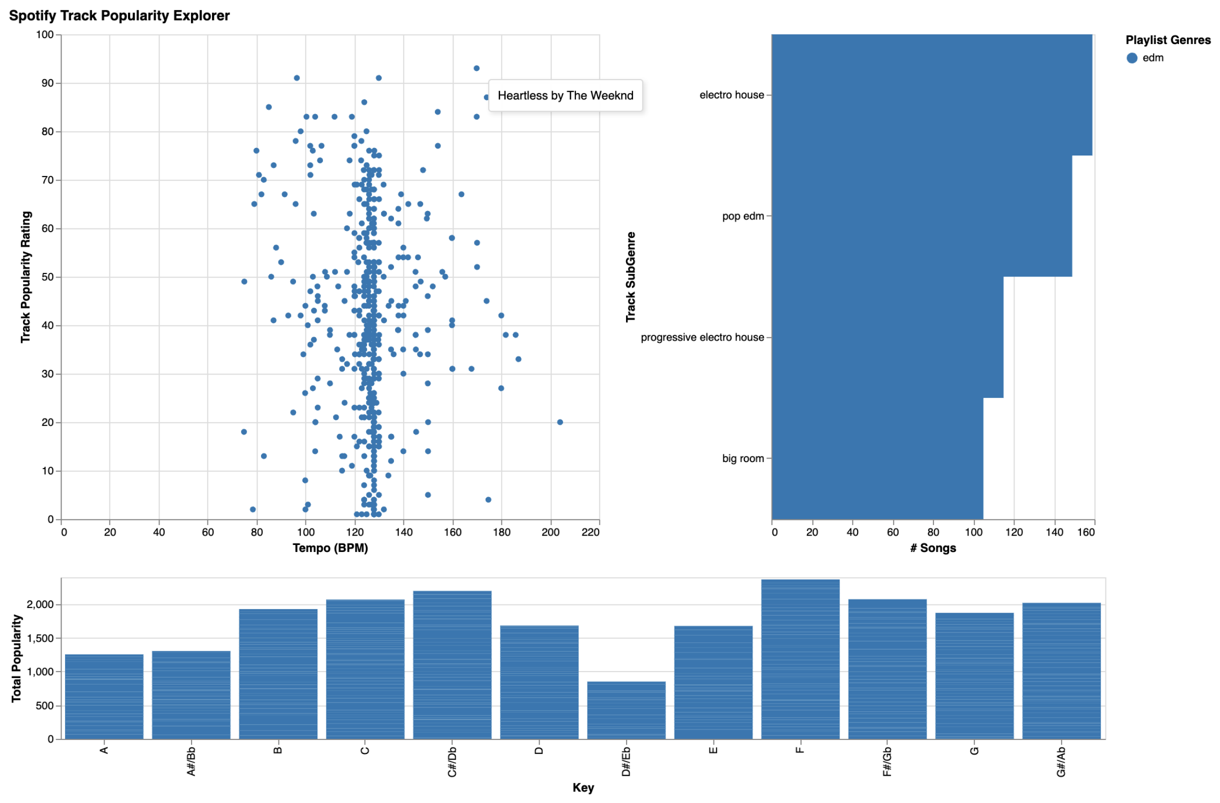
Interaction Consideration: The interaction approach I chose to implement is cross filtering, which in this case means the user can manually select smaller subsections of the scatter plot which will dynamically affect the other 2 corresponding graphs. Originally, all three graphs are tailored to visualising the data from the entire trimmed dataset, but when a user selects a smaller subset on the scatter plot, the bar chart and the histogram will dynamically readjust to the specified area. For example, the user could select the 90-100 percentile range of popularity across all tempos on the scatter plot, and the histogram and bar chart will transform to represent the data within the specified range, revealing that `Post Teen Pop` is actually the most popular sub genre in the highest popularity range, and A#/B Flat is now the key that generates the most popularity, which are significantly different results to when visualising the entire trimmed dataset. Furthermore, the user can then drag their selection around the scatter plot to analyse different sections with the exact same sized range of specification. This visualisation tool is also capable of isolating different genres on all three of the graphs for closer inspection by selecting the genre you wish to isolate on the legend. With these interaction techniques, users can specifically analyse what keys and tempos are most common in specific popularity ranges.

Insight: Some compelling observations have come to light for me after further interaction with this visualisation tool. For example, `indie poptimism` from the `Pop` genre is the most common sub genre on in this trimmed dataset as it has the most occurrences, yet it doesn't appear whatsoever in the 90-100 Popularity Rating. In fact, the top range of popular songs according to this scatter plot is dominated by `Latin` and `R&B` sub genres. This is interesting, because it could imply that Spotify might be oversaturated with `indie poptimism` tracks despite them not being particularly popular among users compared to other sub genres in this list.

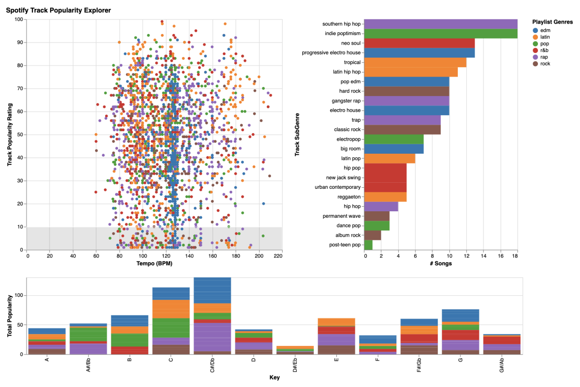


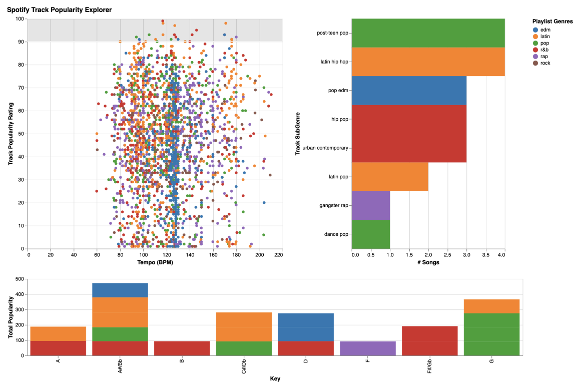


Another thought-provoking observation I made is that the vast majority of `edm` tracks hover around the 120-130 BPM mark, and F is the most popular key signature among `edm` tracks in this trimmed dataset also. However, the most popular `edm` track in this dataset is "Heartless" by The Weekend, which is 170 BPM and is written in A#. Perhaps a contributing factor of the success of this track by The Weekend could have been influenced by the fact it quite literally is composed significantly differently to the average EDM track on this dataset.



One final observation I'd like to highlight is the clear importance of key signature when it comes to the popularity of a song. As a music fan and artist myself, some keys sound nicer than others, especially when it comes to mainstream music. Notice in the first screenshot below how D# is the least popular key across all music genres even in the lowest segment of popularity rating, 0-10. Now in the second screenshot below, notice how D# does not appear at all as a selected key in the 90-100 popularity rating range. In contrast to this, its near neighbour C# performs highly in both of these ranges of popularity.=


Spotify Track Popularity Explorer
Track Popularity Rating
Tempo (BPM)
Track SubGenre
southern hip hop
indie poptimism
neo soul
progressive electro house
tropical
latin hip hop
pop edm
hard rock
gangster rap
electro house
trap
classic rock
electropop
big room
latin pop
hip pop
new jack swing
urban contemporary
reggaeton
hip hop
permanent wave
dance pop
album rock
post-teen pop
# Songs
Playlist Genres
edm
latin
pop
r&b
rap
rock
Total Popularity
Key
A
A#/Bb
B
C
C#/Db
D
D#/Eb
E
F
F#/Gb
G
G#/Ab


Spotify Track Popularity Explorer
Track Popularity Rating
Tempo (BPM)
Track SubGenre
post-teen pop
latin hip hop
pop edm
hip pop
urban contemporary
latin pop
gangster rap
dance pop
# Songs
Playlist Genres
edm
latin
pop
r&b
rap
rock
Total Popularity
Key
A
A#/Bb
B
C#/Db
D
F
F#/Gb
G