Project Proposal

Spotify Top 100 Songs of 2010-2019

Brian Karstens and Christian Kleronomos

GitHub Repository: https://github.com/bkarstens24/data-wrangling-project

1. Introduction:

The number of streams a song has on a popular platform such as Spotify can be a good indicator of how the public perceives a specific song. However, is the number of streams for a song, according to Spotify, the only influence on whether the song makes it onto the Top Hits? Using the Web API for Spotify we plan to collect the number of streams for the songs to see if there is any correlation between the number and the popularity column obtained through the Kaggle dataset linked below, looking at the top hits from 2010-2019. To obtain the number of streams we used the Spotify Web API with our individual client id.

2. Data:

In our project, we used three main sources of data. Our first data source was acquired from Kaggle, our second data source was a Spotify API, and our third data source was scraping Spotify for number of streams.

2.1 Kaggle Dataset

The first data source we are using in our project is a comprehensive dataset we discovered on Kaggle[1]. This dataset contains information about the top 100 songs for each year from 2010 to 2019, spanning a full decade of music. In total, the dataset includes 1,000 rows, each representing a song, along with 17 columns that provide various features such as song title, artist, genre, and key musical attributes like tempo, energy, and danceability. By analyzing the features within this dataset, we aim to identify the patterns and characteristics that contributed to these songs achieving their place among the top 100 for their respective years.

2.2 Spotify Web API

[1] https://www.kaggle.com/datasets/muhmores/spotify-top-100-songs-of-20152019?resource=download

Our second data source involves retrieving data directly from the Spotify Web API[2], which serves as an essential bridge between our initial Kaggle dataset and our third data source. Using the API, we plan to gather detailed information about the top 100 songs for each year from 2010 to 2019. Specifically, we will extract the song titles, the artists who created these songs, their current popularity scores (as rated by Spotify's algorithm), each song's unique track ID, and its corresponding URL. The track IDs and URLs are particularly critical, as they will allow us to link the information from the Kaggle dataset with Spotify's live data and enable further processing. These URLs will also be instrumental in the development of our web-scraping code, written in Python, which we will use to retrieve the number of streams for each song. By utilizing the Spotify API, we can ensure that the information is up-to-date and accurate, which is crucial for connecting and validating data across our sources. This intermediate step not only enriches our dataset but also lays the foundation for our analysis of song popularity and streaming trends over time.

2.3 Scrapping Spotify Streams

For our third and final data source, we focused on collecting the number of streams for each song using a combination of the Spotify Web API and a custom web scraping script. While the API provided essential information like track IDs and URLs, our scraping script allowed us to retrieve specific data directly from Spotify's web pages[3]. This process involved navigating through the pages of various artists to extract the total number of streams for each song. For this source, we concentrated on three key pieces of information: the song title, the artist who performed the song, and the total number of streams. The number of streams represents how many times a song has been played by users on the Spotify platform. We believe this metric is a critical addition to our dataset of the top 100 songs from 2010 to 2019 because it reflects listener engagement and long-term popularity. By incorporating this data, we can analyze trends over time, compare a song's initial success to its sustained popularity, and gain insights into factors that contribute to a song's continued relevance in the streaming era. This enriched dataset will help us draw meaningful conclusions about what makes these songs standout hits both at the time of their release and in the years that followed.

All the scrapping was performed in a twenty-four-hour period over the course of December 7th and 8th to make sure the data was not different for any songs, although the difference would have been negligible.

[2] https://spotipy.readthedocs.io/en/2.24.0/

[3] https://open.spotify.com/

2.4 Combining Kaggle, API, and Spotify Streams

Since each of our data sources were stored as separate CSV files, we needed to combine them into a single dataset that could be used for analysis. This required merging all three CSV files into a single dataframe. To begin, we loaded each file into a notebook and examined their structures. The Kaggle dataset contained 1,000 rows and 17 columns, representing the top 100 songs of each year from 2010 to 2019 along with their key features. The Spotify API dataset had 990 rows and 5 columns, including information such as the song title, artist, popularity score, track ID, and URL. Lastly, the Spotify streams dataset included 990 rows and 3 columns, providing details on the song title, artist name, and total number of streams.

For the first merge, we performed an outer join to combine the Kaggle dataset and the Spotify API dataset. The merge was based on two key attributes: the song title and the artist's name. This approach ensured that no data was excluded, even if some songs or artists appeared in only one of the datasets. After this merge, the resulting dataframe contained 1,124 rows and 20 columns, as it included all unique entries from both datasets. For our second merge, we performed a second outer join to merge the Spotify streams dataset with the already merged dataframe. Again, we used the song title and artist name as the keys for the merge. This step produced a larger dataframe with 1,300 rows and 21 columns, reflecting the inclusion of all unique entries across the three datasets.

Since both merges were performed using an outer join, the resulting dataframe contained a number of empty cells and potentially duplicate rows. To address this, we cleaned the data by dropping rows that had missing values in critical columns and removing any duplicate entries. After the cleaning process, our final merged dataframe was reduced to 904 rows and retained all 21 columns. This final dataset served as the foundation for our analysis, providing a complete and well-structured representation of the data from all three sources and can be seen by Table 1 below:

Table 1 Data Dictionary

Column	**Type**	**Source**	**Description**
title	Text	Kaggle	Song’s title
artist	Text	Kaggle	Song’s artist
genre	Text	Kaggle	Genre of the song
year_released	Date	Kaggle	Year the song was released
added	Date	Kaggle	Day song was added to Spotify’s Top Hits Playlist
bpm	Numeric	Kaggle	(Beats Per Minute) - The tempo of the song

nrgy	Numeric	Kaggle	(Energy) - How electric the song is
dnce	Numeric	Kaggle	(Danceability) - How easy it is to dance to the song
dB	Numeric	Kaggle	(Decibel) - How loud the song is
live	Numeric	Kaggle	How likely the song is a live recording
val	Numeric	Kaggle	How positive the mood of the song is
dur	Numeric	Kaggle	Duration of the song
acous	Numeric	Kaggle	How acoustic the song is
spch	Numeric	Kaggle	The more the song is focused on a spoken word
pop	Numeric	Kaggle	Popularity of the song (not a ranking)
top_year	Date	Kaggle	Year the song was a pop hit
artist_type	Text	Kaggle	Tells if artist is solo, duo, trio, or in a band
streams	Numeric	Spotify	How many people listened to the song
Pop_today	numeric	Spotify API	Popularity of the song today (according to Spotify metrics)
Track_id	text	Spotify API	Each songs specific identifier at the end of their URL
url	Text (URL)	Spotify API	Specific URL for each song in the data

3. Analysis:

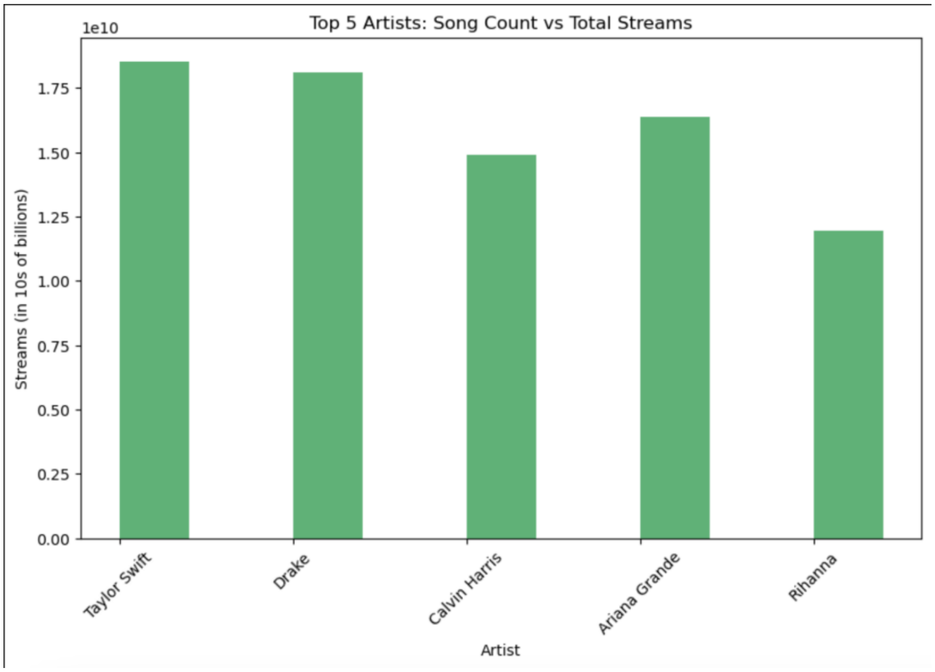

Figure 1 Top Artists Streams

Combining all the artists song streams together over the entire decade, we have found the top 5 artists in the data. Taylor Swift, Drake, Calvin Harris, Ariana Grande, and Rhianna have 19, 18, 17, 13, and 12 songs in the top hits from 2010-2019, respectively. Taylor Swift and Drake have amassed over 18 million streams on their songs that from 2010-2019 were on the top hits and had the most songs to appear in the data.

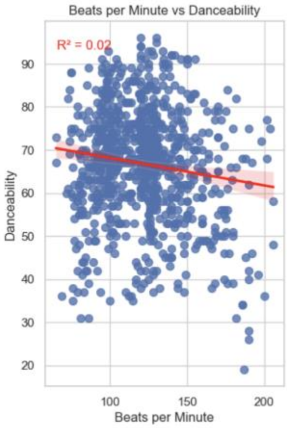

Figure 2 Danceability vs BPM

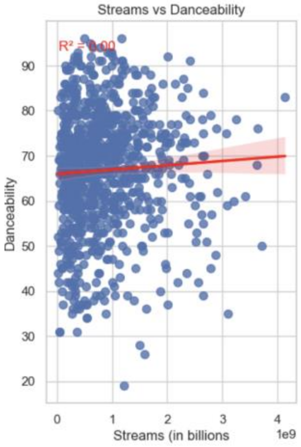

Figure 3 Danceability vs Streams

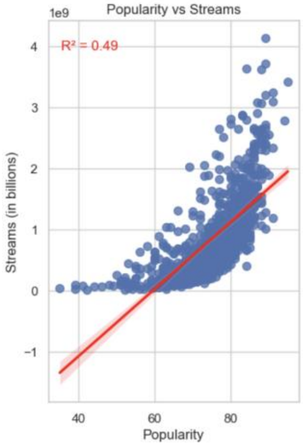

Figure 4 Streams vs Popularity

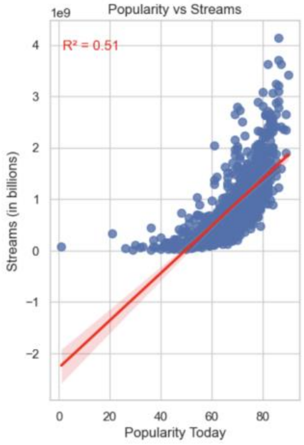

Figure 5 Streams vs Popularity Today

Looking at the above graphs, Danceability is not correlated with beats per minute or the number of streams, with extremely low R values. Both Popularity and popularity today are moderately correlated with the number of streams a song has. Although there is a moderately strong correlation between these variables, they would not make for good predictors, as record labels or artists may have liked them to. Since most people will not have direct access to song streams available on a constant basis due to Spotify limitations, labels may look at historical and current day popularity values. While these can help gauge how well a song is doing on Spotify, it will not help with total streams.

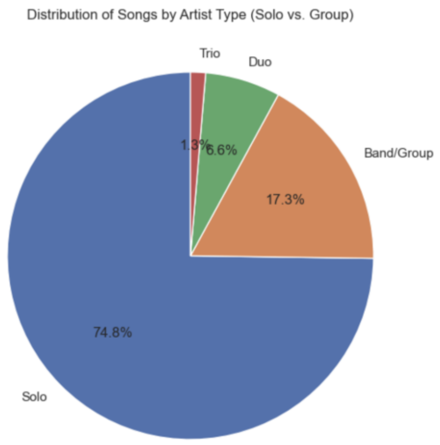

Figure 6 Artist Type Distribution

In looking at all the songs in the data, it was noticeable during the cleaning phase that there was a lot of solo artist songs that made the top hits, and we wanted to explore what the distribution of artist types was in the music. The largest group was solo, followed by Band/Groups, Duos, and Trios, which made sense to us. Being a solo artist does not exclude songs with featured artists. This is something we wanted to explore but could not find a solid way of doing it since some songs will just use the featured artists name in the artist field, others will use ft or FEAT, and there are other ways of doing it. If there was a less error-prone way to categorize a song as having a feature or not, then we could run more correlation tests with streams and songs with/without features.

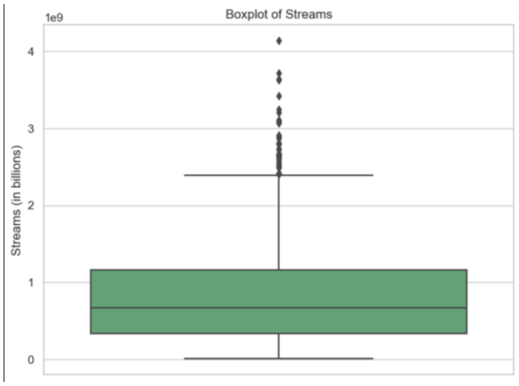

Figure 7 Stream Boxplot

To find out what the average number of streams were, the first and third quartiles, and any outliers in the data we decided to look at a boxplot. Most of the songs (50%) and their streams were between about 400,000 and 1,250,000. There are quite a few outliers above the 2,500,000 streams mark that could be further analyzed to view the other song metrics such as bpm, dnce, and val, speech, etc.

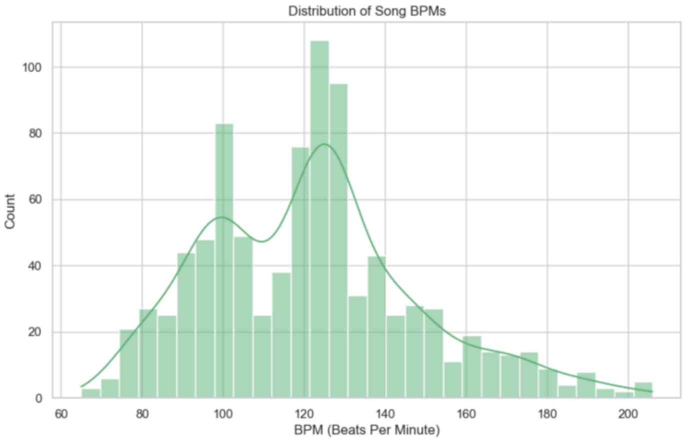

Figure 8 Distribution of BPMs of Top Hit songs

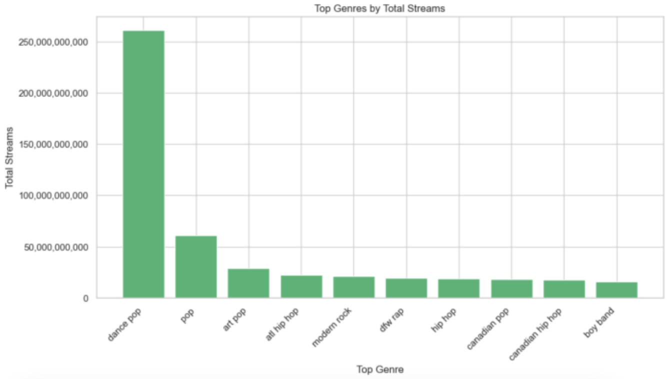

Figure 9 Top Genres by Streams

When looking at the beats per minute of a song and the most popular genres by streams, there appears to be some correlation between the two. Using an article[4], we found that the most common BPM for a dance pop song was around the 110-130 range in the 2010s, which is the most frequent one in the data. With the most popular BPM of the songs being within this 110-130 range, and the most streams coming from the dance pop category, record labels and artists can potentially use this information to their advantage if they just wanted to make a top hit. Streams are accumulated over the lifespan of the song so it may not be the best metric to look at,

[4] https://www.playnetwork.com/blog/2017/03/13/the-changing-bpms-of-club-dance-music/

but using the beats per minute may be an effective way to predict if a song will become a top hit in the future.

Minor Analysis:

We produced a few questions, without visualizations, that we feel are important to address related to the data.

1. How much does each genre appear in the list?
    a. Dance pop was the most popular genre with 332 songs.
2. What is the average danceability of songs from 2010-2019
    a. 66.77
3. What is the average difference between popularity of the song when added to top hits vs today?
    a. -6.88, meaning these songs, on average, are less popular today.
4. Are there any songs that are more popular today? What is the average difference between the popularity for these songs?
    a. There are 41 songs more popular today than when added to the top hits.
    b. The average difference is 3.93.
5. Do songs released in the first half of the decade have more streams than songs released in the second half?
    a. No, songs in the second half of the decade have more streams with 463,600,195,928.
6. What is the average duration of the songs in seconds?
    a. 220.83 seconds
7. Were multiple songs added to the playlist on the same day? Which one is the most popular?
    a. June 22, 2020, was the day with the most songs added with 251 songs.
8. Of songs with popularity > 80 from 2010-2019, what % have a danceability > 80?
    a. 15.81%
9. Did songs have a higher danceability score in the first half of the decade or second? Popularity?
    a. Songs in the second half of the decade had a higher danceability score: 69.34
    b. Songs in the second half of the decade had a higher popularity score: 76.69

4. Conclusion:

While this project looks to analyze artists of the 2010-2019 period, danceability of their songs, release year, and year they made the top hits, based on their popularity and number of streams, we also want to figure out if there is correlation between the number of streams and the Spotify “popularity” rating. To summarize our analysis from above, we will show the results to our questions below:

1. Are there any dominant artists in the list?

   There are 3 closely dominant artists in the 2010-2019 period. Taylor Swift, Drake, and Calvin Harris were the most popular artists, having 19,18, and 17 songs, respectively. This is a good indicator that these artists were putting out songs that the public liked, thus streaming more.

2. Using graphs, is there correlation between beats per minute and danceability? Streams and Danceability? Streams and popularity when added to the top hits? Current day popularity and streams?

   It appears that there is a moderate correlation between the stream variable and the popularity/popularity today variables. While the difference in R values is only .02, the popularity today variable is a better predictor of streams than the popularity variable.

3. Does genre type have any effect on a song making the top hits?

   It is possible that the genre type is a good predictor for making the top hits. This is because 1/3 of the songs in the data are dance pop. The same can be said for artist type being a predictor for making the top hits since 2/3 of the songs are of the solo type.

4. Does this trend continue past the 2010-2019 timeframe?

   This data is not available yet as the decade is reaching the midpoint and there is no top hits playlist yet. When made available it would be interesting to compare the results or combine them to have more data to analyze.
   We found a few different limitations in using both the Spotipy API and scraping Spotify itself. Since we had a set of songs that we wanted to specifically look at to get the track ids from, we had to match get songs based on their artist and title. This did not consider that an artist may have two of the same titled songs released in multiple different years, so we had to manually check that we were getting the correct ones. There were also problems between the Kaggle dataset, and the API, where some songs we are not getting a track id at all, meaning certain songs from the dataset could not be used. We scraped Spotify for all the songs that we got track ids for, but there were a few missing in the end. This forced us to clean the data after merging to drop any duplicate tracks and any missing ones as well. There is also not a very accessible way to retrieve the streams of specific songs in masses, but if there were, larger data could be analyzed.