Data Modeling

Spotify

By: Bianca, Rob, Steve, Jithu

01

About the Project

Overview

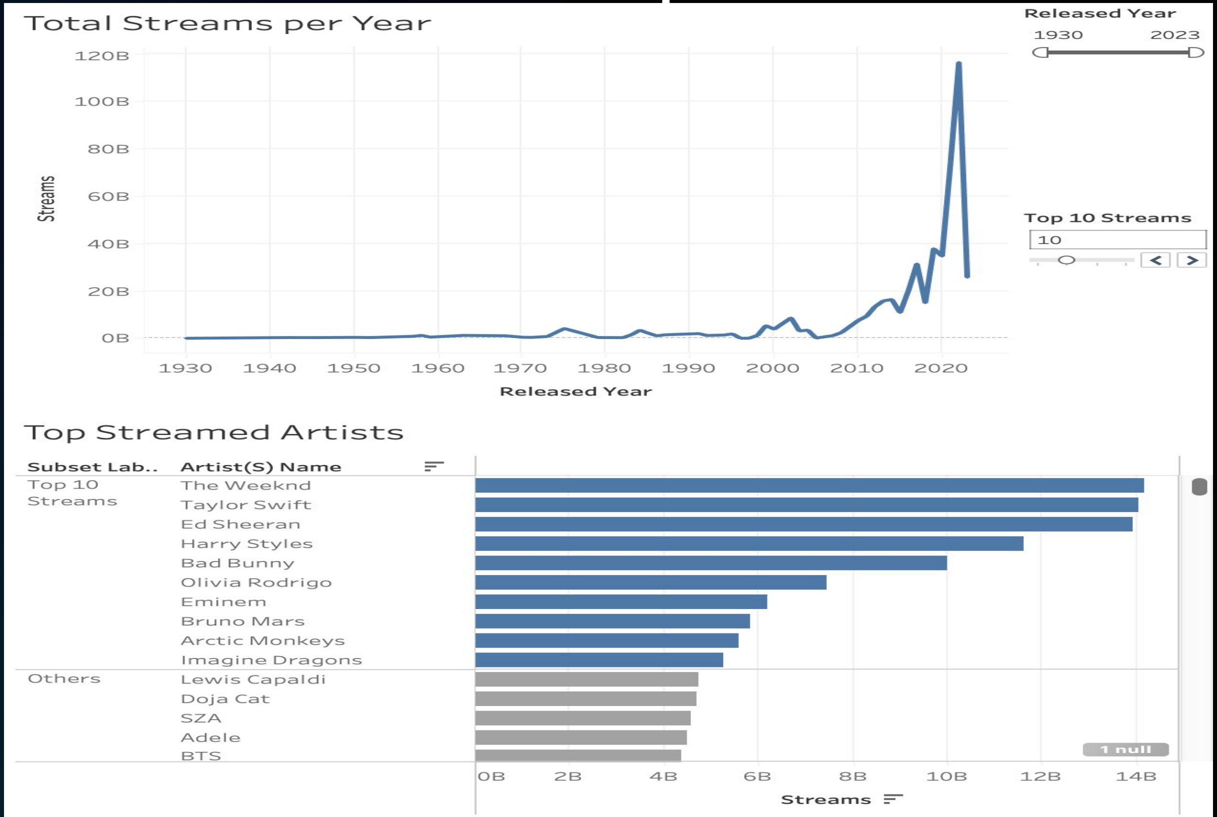
- The music market is extremely competitive with artists always trying to make the next viral song and get the most streams. Our group aims to find a way to see if we can develop an algorithm that can give artists a competitive edge.
- Based on the date it was released, bpm, energy, danceability, key, mode, dancebility%, valence%, energy%, acousticness%, instrumentalness%, liveness_speechiness% we want to predict the following:
    - How many Spotify streams?
    - Is it in spotify's charts?

02

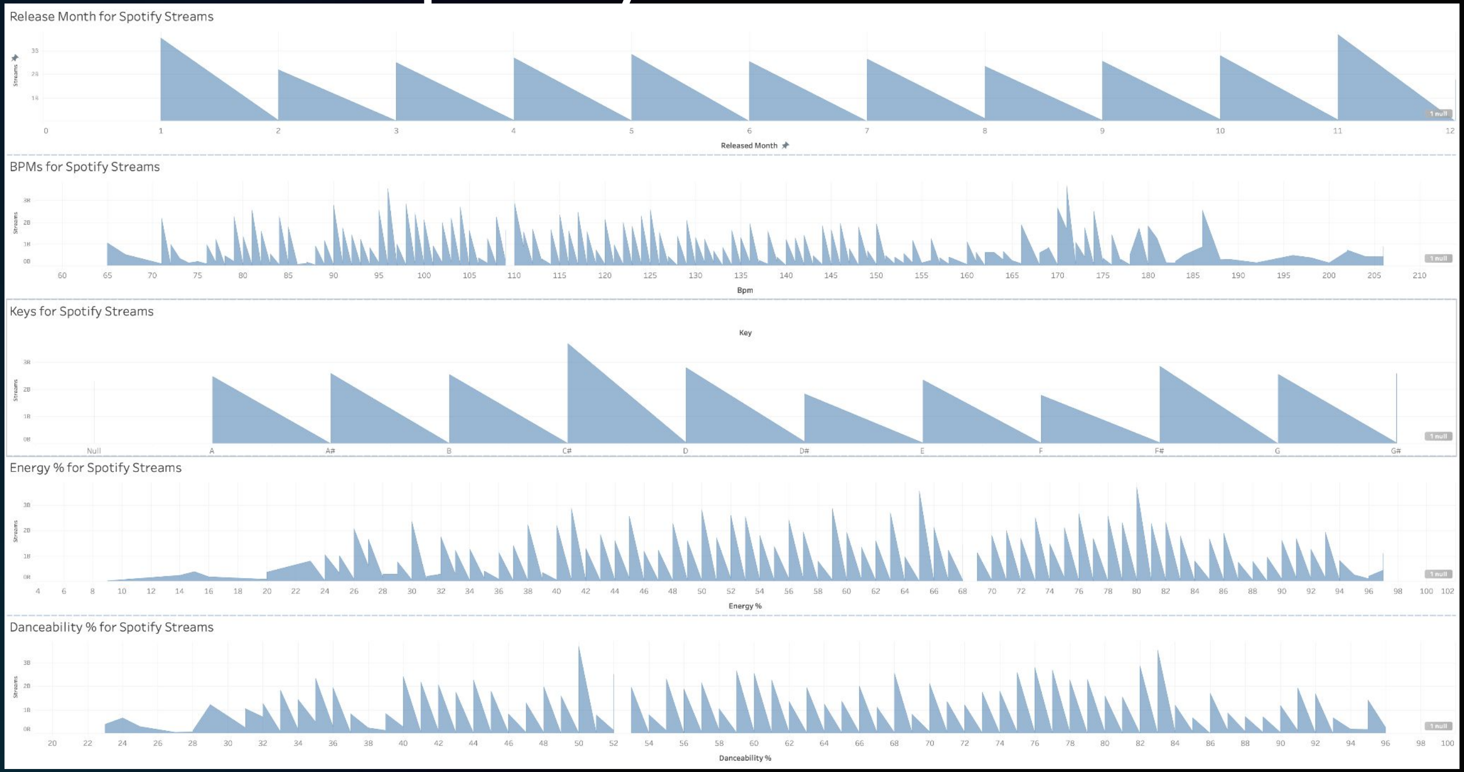
Spotify Data Visualizations

Curated via Tableau & Seaborn

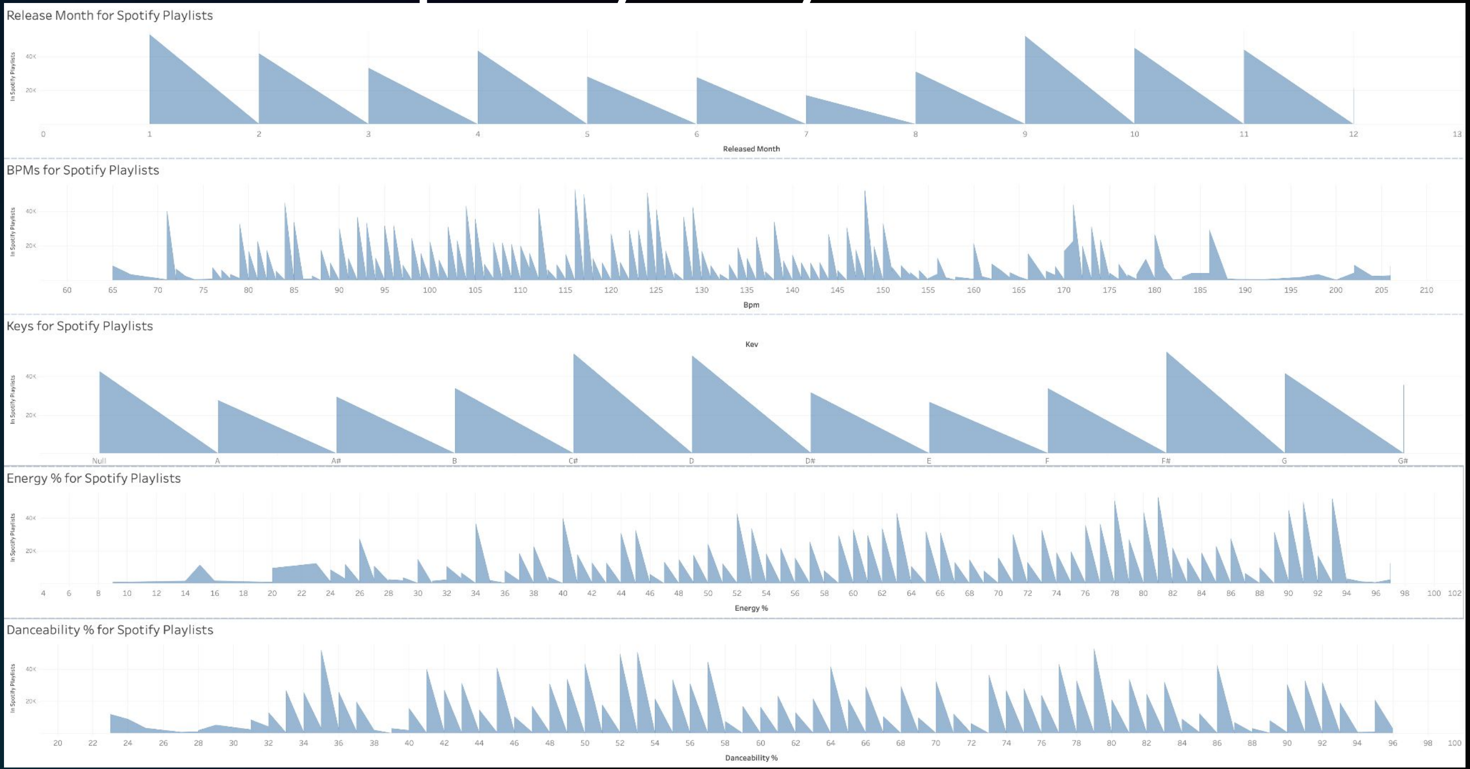
Total Streams and Top Artists



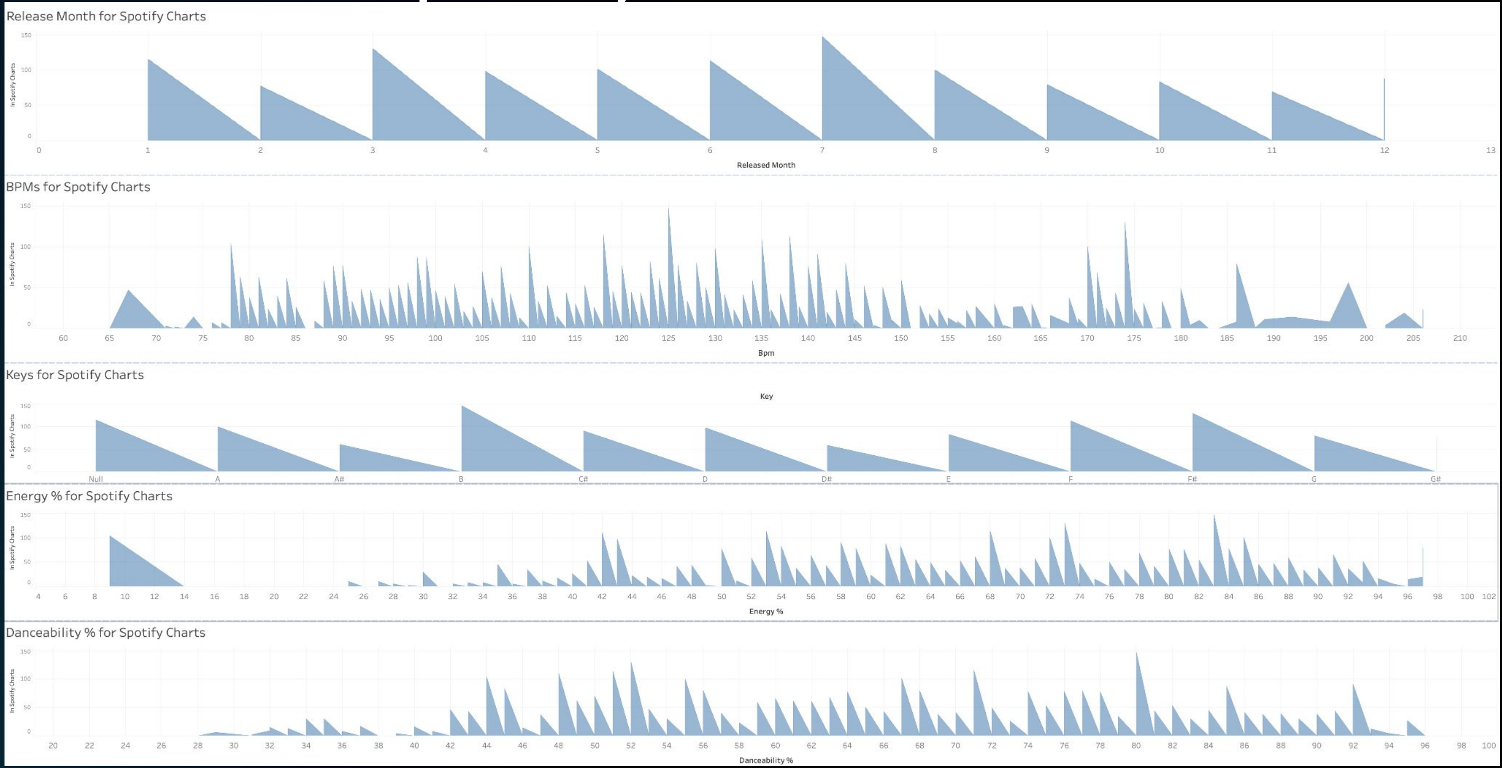
Stats for Spotify Streams



Stats for Spotify Playlists



Stats for Spotify Charts

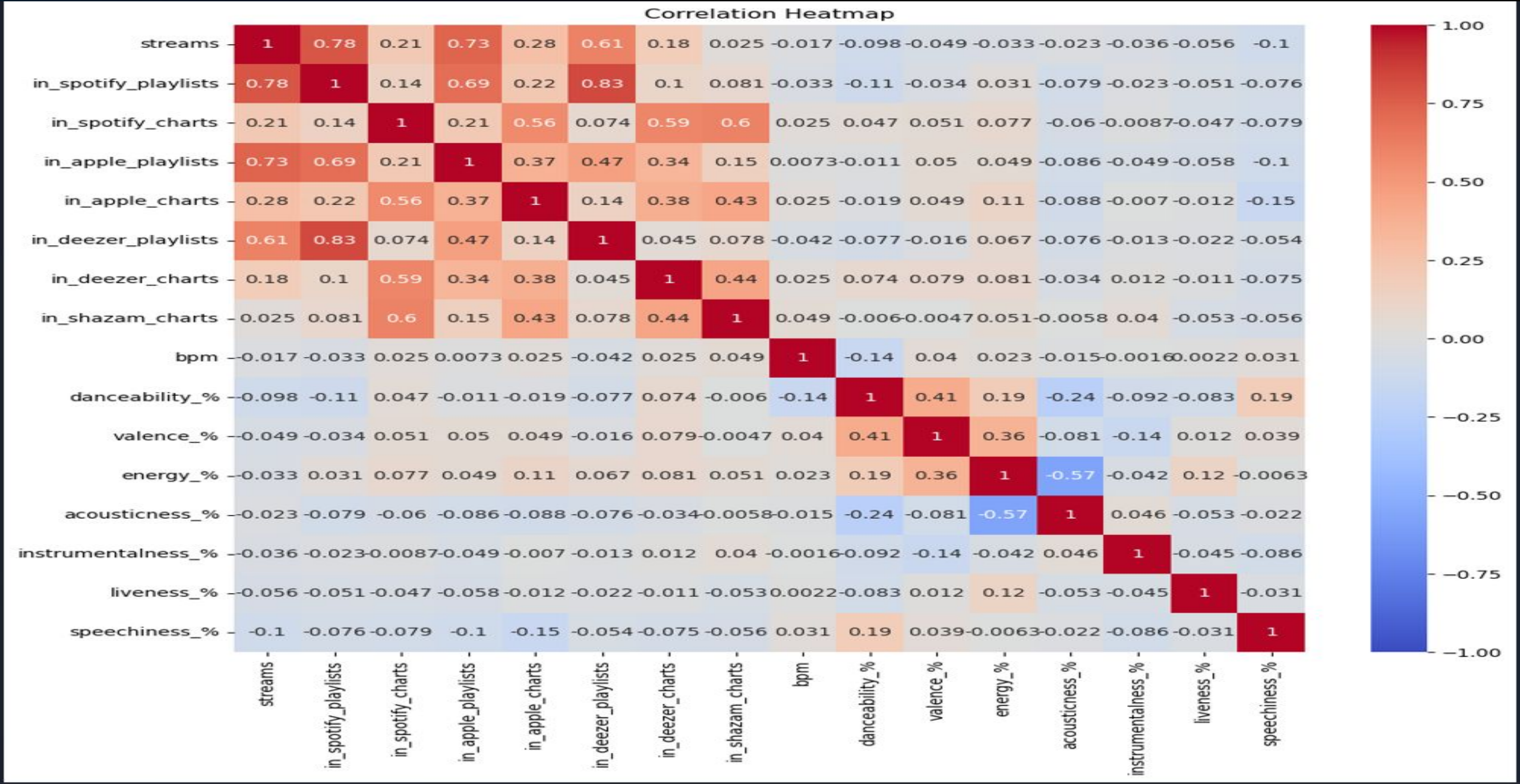


Stats: Overview

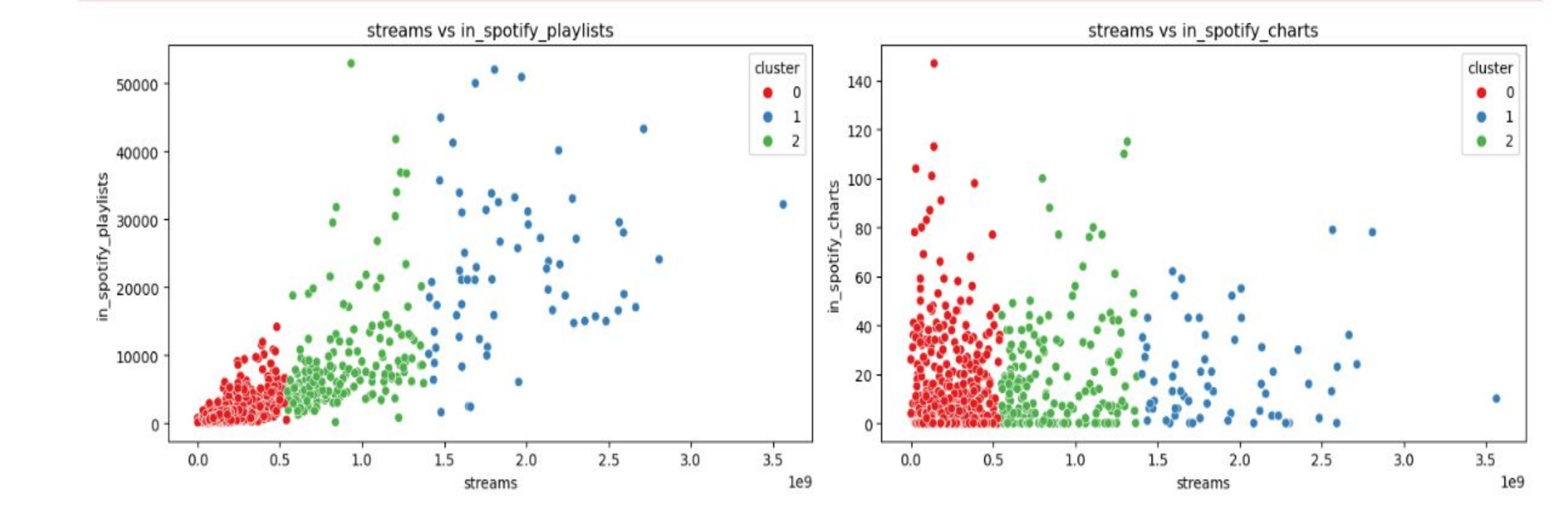
- Release Month
  - Most Streamed Release Month: November
  - Most Playlisted Release Month: January
  - Most Charted Release Month: July
- BPM
  - Most Streamed BPM: 171
  - Most Playlisted BPM: 116
  - Most Charted BPM: 125
- Keys
  - Most Streamed Keys: C# (C Sharp)
  - Most Playlisted Keys F# (F Sharp):
  - Most Charted Keys: B (B Major)
- Energy %
  - Most Streamed Energy %: 80%
  - Most Playlisted Energy%: 81%
  - Most Charted Energy %: 83%
- Danceability %
  - Most Streamed Danceability %: 50%
  - Most Playlisted Danceability %: 79%
  - Most Charted Danceability %: 80%

- Most Spotify Streamed Song:
  - Blinding Lights - The Weeknd
- Most Spotify Playlisted:
  - Get Lucky - Pharrell
- Most Spotify Charted:
  - Seven - Latto, Jung Kook
- Top 5 Streamed Artists
  - The Weeknd
  - Taylor Swift
  - Ed Sheeran
  - Harry Styles
  - Bad Bunny

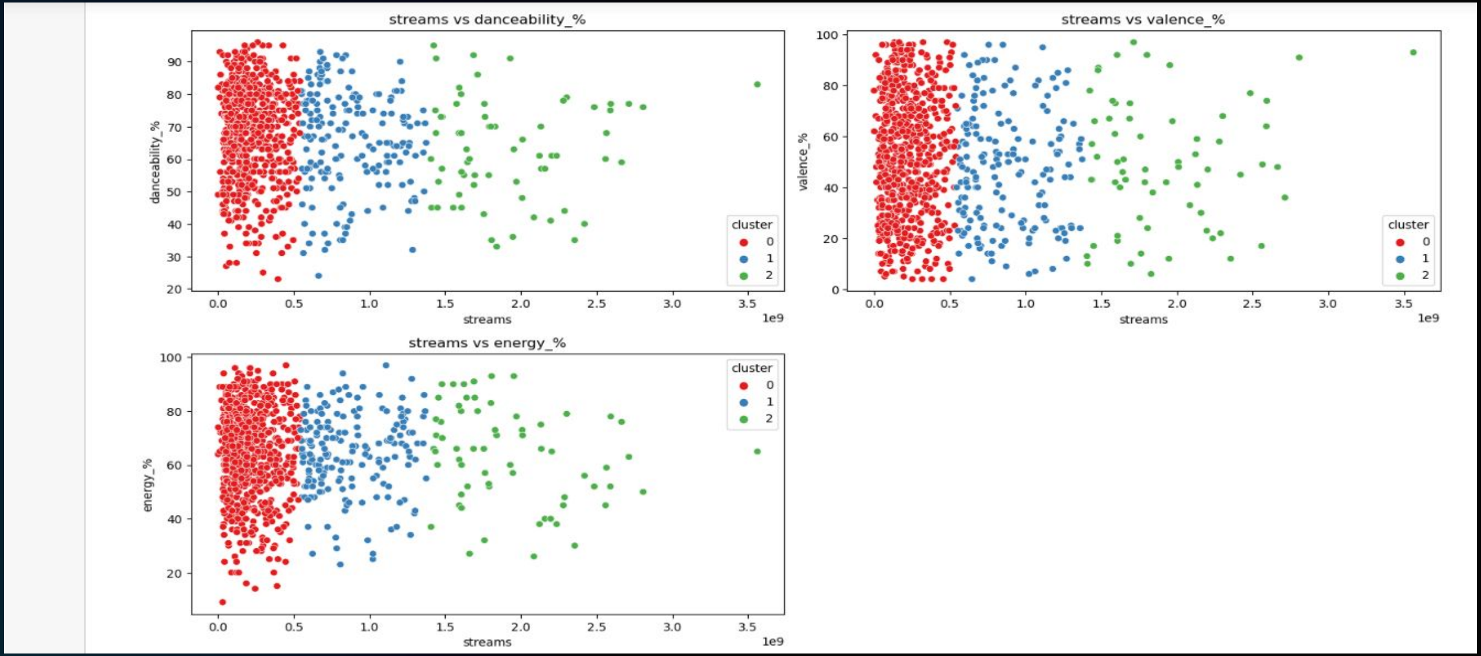
Seaborn: Correlation Heatmap



Seaborn: Cluster Pairplot



Seaborn: Cluster Pairplot



03

Data Model Implementation

Linear Regression, Random Forest, & More

What methods are we using to Predict?

- As mentioned before, we are trying to predict Streams and if the song made it to spotify’s charts and where it charted.
- First we decided to remove all of the variables that was not a Float or integer.
- Converted the streams from object to int64
- Then we tried to predict the two fields mentioned above With Linear Regression and Random Forest

```
RangeIndex: 953 entries, 0 to 952
Data columns (total 24 columns):
 #   Column                Non-Null Count  Dtype
---  ------                --------------  -----
 0   track_name            953 non-null    object
 1   artist(s)_name        953 non-null    object
 2   artist_count          953 non-null    int64
 3   released_year         953 non-null    int64
 4   released_month        953 non-null    int64
 5   released_day          953 non-null    int64
 6   in_spotify_playlists  953 non-null    int64
 7   in_spotify_charts     953 non-null    int64
 8   streams               953 non-null    object
 9   in_apple_playlists    953 non-null    int64
 10  in_apple_charts       953 non-null    int64
 11  in_deezer_playlists   953 non-null    object
 12  in_deezer_charts      953 non-null    int64
 13  in_shazam_charts      903 non-null    object
 14  bpm                   953 non-null    int64
 15  key                   858 non-null    object
 16  mode                  953 non-null    object
 17  danceability_%        953 non-null    int64
 18  valence_%             953 non-null    int64
 19  energy_%              953 non-null    int64
...
 22  liveness_%            953 non-null    int64
 23  speechiness_%         953 non-null    int64
```

Linear Regression

- Used this model to find if there is relation between streams/charts (dependant variable) to all the other variables (independent variables)

	Predictions (charts)	Actual (charts)
256	11.181750	13
373	14.773431	0
791	10.716496	0
39	10.601805	28
619	10.774929	0

	Predictions (Streams)	Actual (streams)
604	3.444988e+12	956865266
559	1.958588e+12	421040617
750	3.531927e+12	1023187129
353	1.874600e+12	372476382
311	1.593008e+12	449701773

Random Forest

- Wanted to use a model that is the "opposite" to linear regression and is robust against non-linear data to see if there is improvement here

	Predictions (charts)	Actual (charts)
256	6.590	13
373	10.842	0
791	7.588	0
39	14.988	28
619	9.056	0

	Predictions (Streams)	Actual (Streams)
604	2.277614e+09	956865266
559	2.195415e+09	421040617
750	2.277614e+09	1023187129
353	2.320015e+09	372476382
311	2.288452e+09	449701773

Which is Better?

- Overall - Both of them are not great but random forest is slightly better, this is backed up by how accurate the mode "thinks" it is using the score method on the test data.
    - Streams: 97% Random Forest, 65% Linear Regression
    - Charts: 90% Random Forest, 6% Linear Regression
- While Random Forest is closer to the actual value in magnitude, it tends to cluster its predicted number and does not deviate much

04

Other Considerations

For Future Exploration & Understanding

Future Considerations

- Include Artist Name for modeling & predictions to be likely more accurate
- Further understanding on Spotify % of
  - Danceability %
  - Valence %
  - Energy %
  - Acousticness %
  - Instrumentalness %
  - Liveness %
  - Speechiness %
- Track YoY changes in statistics for developing further modeling & identifying YoY trends
- Knowing the length of the song will add the to model's robustness

05

Appendix

Links, Glossary, & Final Details

Links

- Dataset Link:
    - https://www.kaggle.com/datasets/zeesolver/spotfy
- Tableau (Public) Link:
    - https://public.tableau.com/views/SpotifyDataVisualizations/Story1?:language=en-US&:sid=&:display_count=n&:origin=viz_share_link
- GitHub Link:
    - https://github.com/jithu-ann/Project-4

GitHub Detail

- GitHub uploads and final repository have been completed
  - free of unnecessary files and folders
  - appropriate .gitignore in use
  - contains a README file

Data Set Glossary

- track_name: The name of the track.
- artist(s)_name: The name(s) of the artist(s) who created the track.
- artist_count: The number of artists associated with the track.
- released_year: The year when the track was released.
- released_month: The month when the track was released.
- released_day: The day when the track was released.
- in_spotify_playlists: Indicates whether the track is included in Spotify playlists.
- in_spotify_charts: Indicates whether the track is present in Spotify charts.
- streams: The total number of streams the track has accumulated.
- in_apple_playlists: Indicates whether the track is included in Apple Music playlists.
- in_apple_charts: Indicates whether the track is present in Apple Music charts.
- in_deezer_playlists: Indicates whether the track is included in Deezer playlists.
- in_deezer_charts: Indicates whether the track is present in Deezer charts.
- in_shazam_charts: Indicates whether the track is present in Shazam charts.
- bpm: Beats per minute - a measure of tempo in music.
- key: The musical key in which the track is composed.
- mode: Indicates whether the track is in a major or minor key.
- danceability_: A measure of how suitable a track is for dancing.
- valence_: The musical positiveness conveyed by a track.
- energy_: The perceived energy of a track.
- acousticness_: A measure of how acoustic a track is.
- instrumentalness_: A measure of whether a track contains vocals.
- liveness_speechiness_: A measure of presence of live elements or spoken words in a track.

Thank You!