Analyzing Popular Music Through Spotify and Visualization

Team 157 - Arianna Schmid, Ellen Khachatryan, Marcos Micheloni, Margaret Pires-Fernandes

I. Introduction and Literature Review

There are few platforms today that create the capability to explore the visual nature of why popular music is in fact popular. In this project, Team157 aimed and succeeded to build such a platform that is both interactive and can continue to be built upon. This platform is intended for music enthusiasts, aspiring artists, and those in the music business to better understand what factors make music successful.

The music industry today primarily exists in the digital space. The IFPI 2018 reported that the revenue acquired from digital and streaming content in 2018 went up 20% in 2018 [1]. This is almost 60% of the total recorded music revenues [1]. Music recommendation systems are often based on popularity approaches [2]. These approaches generally don't take into account the cultural or regional differences that can vary from the global music mainstream [2]. Over the past few years, big data has started to play into how users engage with music content [1]. Efforts to optimize profitability impact the artists, labels, and streaming platforms [1]. Music data analysis as a field is growing and utilizing tools like prediction and recognition, classification, audio analysis, and recommendation [3]. There are key metrics that can affect song popularity, including energy, valence, and duration [4]. Additionally, loudness, genre, and danceability can impact a song's popularity [5]. Machine learning techniques can support analysis to understand fan behavior and artist similarities [6]. Due to the size of the industry and scale of the business, attempts to model and predict song popularity have become increasingly important [7]. Additionally, the Spotify business model has reached across just the music industry, highlighting the importance of the company [8]. Music recommendation algorithms have been developed and expanded to include user data, system logs, and additional factors outside of song popularity with a marginal improvement [9]. Still, there aren't any interactive platforms that give users access to prediction model results as well as the ability to listen to songs and explore the nature of what they are listening to. This motivated our team to create a user-friendly interactive platform that can be used for not only research and analysis, but also entertainment.

II. Methods

A. Data Collection

Spotify does not readily provide data on the top songs by year. Our team needed to figure out another means of obtaining a data frame containing the top songs from 2012 to 2022. We turned to another reliable source for popular songs: The Billboard Top songs. This record chart, recognized as the prevailing standard in the U.S. music industry, is released by Billboard magazine. The rankings on the chart are determined by considering sales, streaming figures, and radio airplay. So how do we programmatically access this data? It turns out that Wikipedia has the tables we need for each year. For example, for the year 2012, this link provides a 3-column table with the top Billboard songs that year: https://en.wikipedia.org/wiki/Billboard_Year-End_Hot_100_singles_of_2012. Figure A in the appendix shows us the first 3 rows as a reference. In the url link, if you replace the year "2012" portion with another year, you get the same data for that year. Therefore, this url serves as the perfect basis for scraping the data we need: the top songs from 2012 to 2022.

Using pandas, BeautifulSoup, and iterating through each year, we created a data frame called top_100_songs.csv with 1096 rows and 4 columns consisting of the year, Billboard Number rank, song title, and artist name. Next, we need to obtain the variables that the Spotify API provides for each song in our top_100_songs data frame. Thus, we need to merge our current frame with Spotify API's 12 audio features (e.g., danceability, energy, valence). Ideally, we would use the song title, artist combination as a key. However, this approach doesn't guarantee a match due to variations in strings which would lead to a lack of standardization within the data. A more reliable method to match our song title to the correct song in Spotify was determined: first, manually create a public Spotify Playlist for each year. Each playlist contains the top songs for that year. Though tedious, this step ensures we have a "URI" for each song. This is a unique identifier that Spotify automatically assigns to every artist, album, and track. This ensures a perfect match when we access the API using the top billboard songs we scraped independently. These same playlists also ended up being embedded into our final visualization, providing entertainment and novelty to our platform. Next, we imported the following Python libraries: argparse for parsing command-line arguments, pandas for data manipulation and analysis, and spotipy for interacting with the Spotify API.

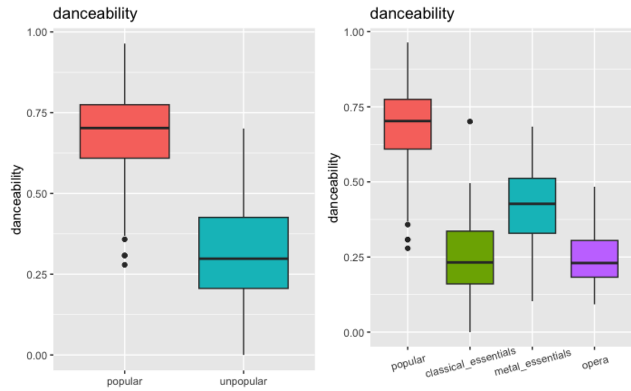
Our program then defines a Spotify playlist link and extracts its URI. Recall we have 11 total playlist links we manually created. We then retrieve the audio features for each specified track using the sp.audio_features() method. A data collection loop then iterates through each playlist link, extracting and storing information about each track in a dictionary (my_dict). In the end, we have our final data frame with the columns: year, rank, name, artist, URI, danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, liveness, valence, tempo, and duration. Figure B in the appendix shows an example of what this frame looks like. At this point, we believed we were ready to proceed with analysis, statistics, and visualizations. During analysis, however, the team decided to incorporate an "unpopular" songs data set, since the 12 audio features appeared uniform across the top songs for each year– which makes sense. This is further discussed in the Innovations section. After creating the unpopular songs playlist, the team had more quality data to analyze and build classification models. See the process flow chart in the Appendix.

B. Experiments / Evaluation

An initial time series analysis was conducted on the dataset following the trends of loudness, speechiness, duration, energy, danceability, valence, and tempo. Figure C in the appendix shows that across the 10 year span the factors that have the least variability are tempo, valence, danceability, duration, liveness, and loudness. Some additional trends included popular songs becoming shorter, less energetic, and less loud over time. The genre of popular songs were 80% pop, demonstrating consistency in the creation of pop music.

While it may be intuitive that pop music will be consistently popular over time, even if the makeup of the song changes with the times and trends, it is important to analyze unpopular songs. The comparisons between popular and unpopular songs can be seen in Figure D in the Appendix. Danceability is the factor that most affects the variance of popular versus unpopular music. The chart below demonstrates the unpopular category as a grouping of classical, metal, and opera essentials. These elements in music are generally categorized as unpopular, and clearly lack in the danceability factor. Energy is another interesting variable that demonstrates a slightly different pattern than danceability. With energy, popular songs are typically more energetic than unpopular songs, however, there is a threshold in which popular songs do not become too energetic. This was an interesting and unexpected discovery for the team. The loudness and tempo factors have less clear variance and do not appear to be a meaningful predictor of popularity or unpopularity. The duration also provides the trend that popular songs are generally shorter in length.

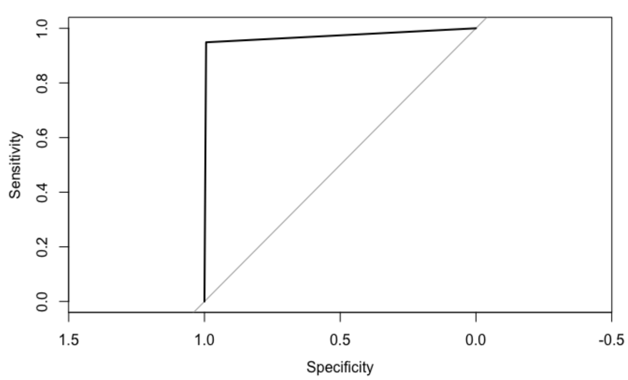
Modeling captured three main groups/approaches. The first set of models included all variables, then reduced down to those suggested in the literature (loudness, valence, danceability), and finally tree based models for assessing variable interaction and importance. All models were cross validated. Statistical outputs are included in the Appendix.



Popular vs. Unpopular Songs

All variables - Logistic Regression The deviance goodness of fit test had a p-value of ~1 meaning the Ho cannot be rejected (good fit). Differing thresholds were attempted to define the labels, with marginal impact to model accuracy. The test data accuracy was 0.9588. See output in Figure E.

All variables - Support Vector Machine The test data accuracy for the SVM model was 0.9794 and demonstrated nearly perfect classification accuracy. See output in Figure F. The figure below demonstrates the sensitivity-specificity plot.



All Variables SVM Sensitivity and Specificity Plot

All variables - K Nearest Neighbors The optimal K value was found to be 3 with a test accuracy of 0.8807.

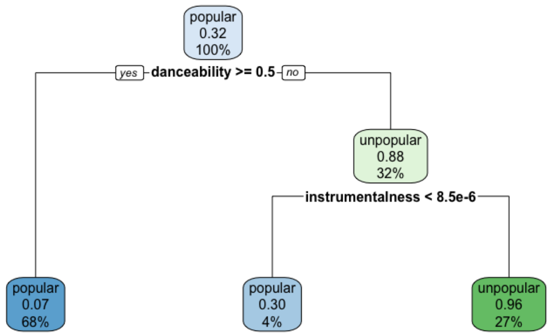
Main Variables - Logistic Regression The results of this logistic regression was very similar to that of the one conducted with all variables included. The test data accuracy was slightly worse at 0.9300. See output in Figure G.

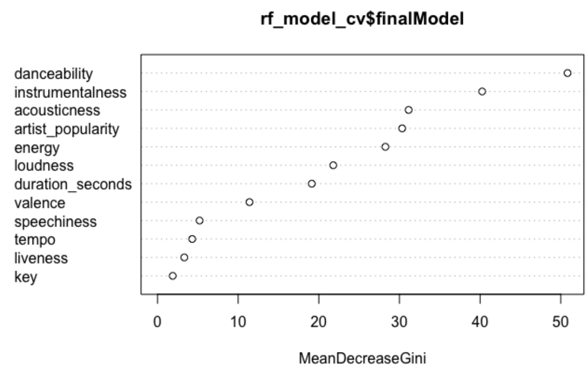
Main Variables - Support Vector Machine The test data accuracy for the SVM model was 0.9465 and demonstrated nearly perfect classification accuracy. See output in Figure H. The figure below demonstrates the sensitivity-specificity plot.

Main variables - K Nearest Neighbors The optimal K value was found to be 3 with a test accuracy of 0.9136.

Decision Tree From the decision tree below we see that danceability is the best predictor of song popularity.

Random Forest We see from the random forest the variables we expect to have the biggest impact on song popularity score the highest in the variable importance plot. Danceability has the highest value followed by instrumentalness and acousticness.





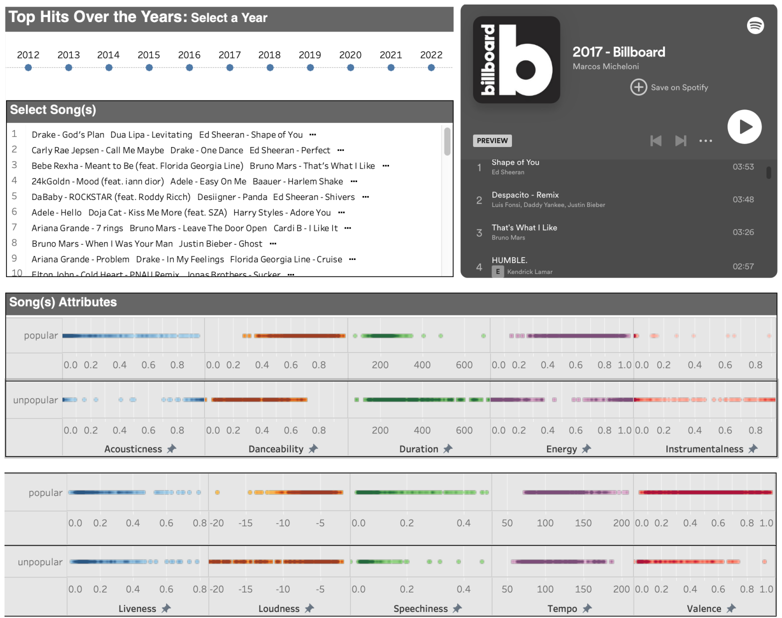
Decision Tree and Random Forest Output

C. Visualization

The scope of our platform will provide a means to see our classification model results as well as explore the components that make up popular songs– all while being able to listen to them. During this project some team members needed to learn Tableau for their jobs and decided that learning how to use Tableau for this project would be a worthwhile endeavor. The D3 visualizations were started and could be considered as future extensions of the project– in particular, when incorporating the “prediction of popularity” component using our model results. At its current state, the user will be able to see an assessment of song attributes, as well as overall differences between popular and unpopular music (ignoring year for unpopular). They will also be able to select specific years and/or songs and view their differences. For example, density plots displaying differences in “danceability” or “tempo” between popular music versus unpopular music.

The final Tableau dashboard is shown below. The dashboard highlights top songs over the past 10 years and includes the list of songs in the order they were ranked on the Billboard charts. The time bar at the top left enables the user to select a specific year they want to explore. When that year is selected, the songs and playlist are available. The Spotify playlist is embedded into the dashboard and enables the user to listen and compare tracks as they explore the features associated with each song. The features of both popular and unpopular songs are highlighted in the bottom section of the dashboard. When a year and song are selected the dashboard automatically filters each section and shows the relevant information.

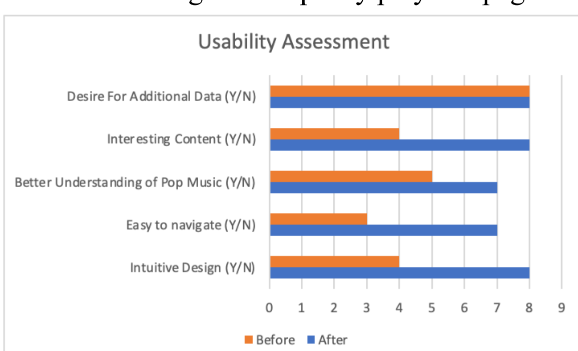
The implementation of the filtering caused many challenges in creating this dashboard. Each of the features of each song and year are independent variables, and a creative solution with calculated fields was identified to group them for visualization.



Final Tableau Dashboard

Usability Evaluation

There were more than 50 interactions of the Tableau visualization and gathering user feedback was a way to continuously improve the final output. A Google Form was distributed to friends of Team 157 alongside a copy of the packaged Tableau workbook. The packaging of the Tableau workbook ensured that there would be no issues with accessing the data that creates the visuals. The form included 10 questions regarding overall look and feel, navigation, and content. The survey was conducted twice to highlight improvements and implementation of user feedback. The results of the Yes/No questions are highlighted below.Specific recommendations from the usability assessment that were implemented included the URL linking to the Spotify playlists page and the linking of the years to the songs.



Usability Assessment Data

D. Innovations

Our team introduced advancements in the realm of data collection, problem resolution, and the creation of visuals that effectively convey a comprehensive overview of the data. To begin with, we see that even though information and visualizations about music and top songs from the past decade is accessible online[10], there is currently a lack of an interactive website that allows users not only to browse the top songs for a selected year but also to explore the details and components associated with these songs. Our team aims to address this by utilizing Tableau to create an interactive page, enabling users to choose a year and examine the audio features of specific songs from that period. Our work also builds on utilizing components of the framework for frequent pattern mining established by K.E. Barkwell et al [11]. The innovative approach tackles an aspect of frequent pattern matching to assess what factors are contributing to popularity. In addition, the team faced the problem of merging our billboard data with data from the Spotify API. A one-to-one match was needed, but this was not the case given variations in Spotify search results. The decision to manually create playlists, in order to guarantee a specific URI, and therefore achieve a one-to-one match, was an innovative solution to this problem.

Analyzing the top billboard songs revealed that there are minimal differences to be found in songs that are already popular, as mentioned in section A. This indicated to the team that more quality data was needed, in order to see true indicators of popularity. As indicated in the findings of Taren-Ida Ackermann et al [13] that classical music and heavy metal generally rank low on popularity, our team decided to incorporate such "unpopular" data into our models. This development of the computations provides the ability to assess how the algorithms created are working as compared to what has been seen in literature. After including an "unpopular_songs" dataset consisting of classical music, opera, and heavy metal, we started to see patterns of what makes music popular more clearly.

We built a series of both parametric and nonparametric models including logistic regression, KNN and SVM, using different sets of predictors. Incorporating these results into our visual platform would have been our next step and an added innovative bonus to our platform, but time constraints did not allow us. Still, our methods and end product establishes an innovative platform making a novel expansion of Spotify through its API.

III. Results

The first few models include all variables in our dataset. Not surprisingly, since we're accounting for a lot of variables, we get classification performance: 96% for logistic regression, 98% for SVM, and 88% for KNN. Note that the accuracy metrics are based on a test holdout set. We've also tuned parameters such as the `k` value and the `c` cost parameter through validation. It is clear that danceability repeatedly is the factor that is most impactful for a song rising to the top of the billboard charts.

Based on the usability assessment with 10 users, improvements were seen in almost every category. Meaningful improvements were made to the dashboard that provided a more valuable experience to the user group, for example, making the charts wider so users don't need to scroll horizontally. Additional recommendations included incorporating a filter for each attribute, so users can view one at a time. In addition, incorporating playlists for "unpopular" songs by year.

IV. Conclusion/Limitations

All team members have contributed a similar amount of effort. The challenge will be to find ways of modeling the data such that we focus solely on audio features, i.e. disregarding information about artist popularity. Tableau is an extremely useful platform, however it does have some limitations. There is less flexibility in the content and formatting of what can be embedded on the dashboard.

Extensions of this project should incorporate our classification model results into the visual, where users can input song names and retrieve whether the song is/will be popular. The "unpopular" data frame can also be updated to include songs from each year. The group members who needed to work with Tableau at their jobs feel much more comfortable using the tool after this project and have been able to make meaningful contributions due to what was learned in this class.

APPENDIX

List of songs on *Billboard*'s 2012 Year-End Hot 100 chart[2]

No.	Title	Artist(s)
1	"Somebody That I Used to Know"	Gotye featuring Kimbra
2	"Call Me Maybe"	Carly Rae Jepsen
3	"We Are Young"	Fun featuring Janelle Monáe

Figure A. Wikipedia Table with Billboard Data for Scraping

	year <int>	rank <int>	name <chr>	artist <chr>	danceability <dbl>	energy <dbl>	key <fctr>	loudness <dbl>	speechiness <dbl>
1	2012	1	Somebody That I Used To Know	Gotye	0.864	0.495	0	-7.036	0.0370
2	2012	2	Call Me Maybe	Carly Rae Jepsen	0.783	0.580	7	-6.548	0.0408
3	2012	3	We Are Young (feat. Janelle Monáe)	fun.	0.378	0.638	10	-5.576	0.0750
4	2012	4	Payphone	Maroon 5	0.739	0.756	4	-4.828	0.0394
5	2012	5	Lights	Ellie Goulding	0.703	0.731	8	-6.283	0.0311

Figure B. Final Dataset Example

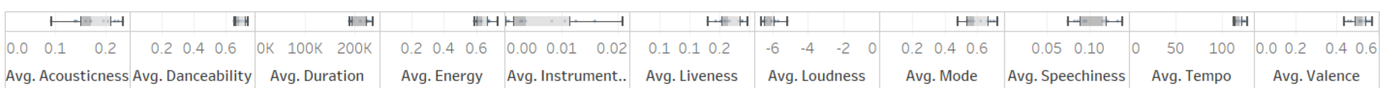
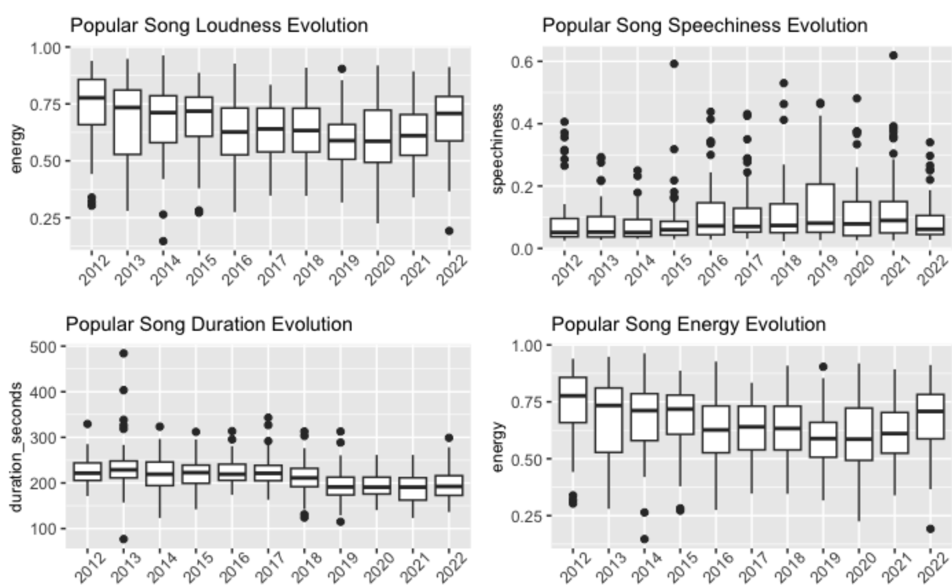


Figure C. Time Series Plots: We find that across the 10 year span the factors that have the least variability are tempo, valence, danceability, duration, liveness, and loudness.



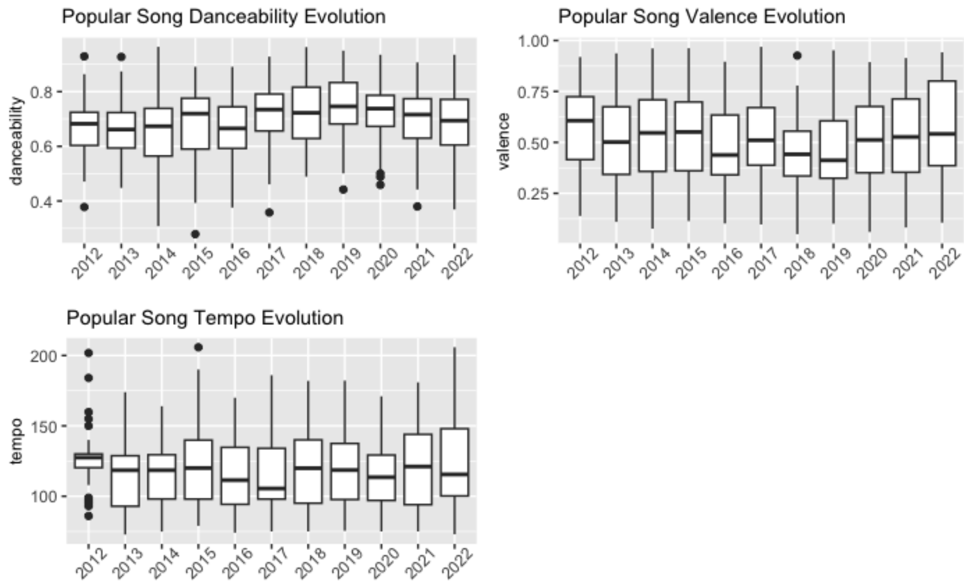
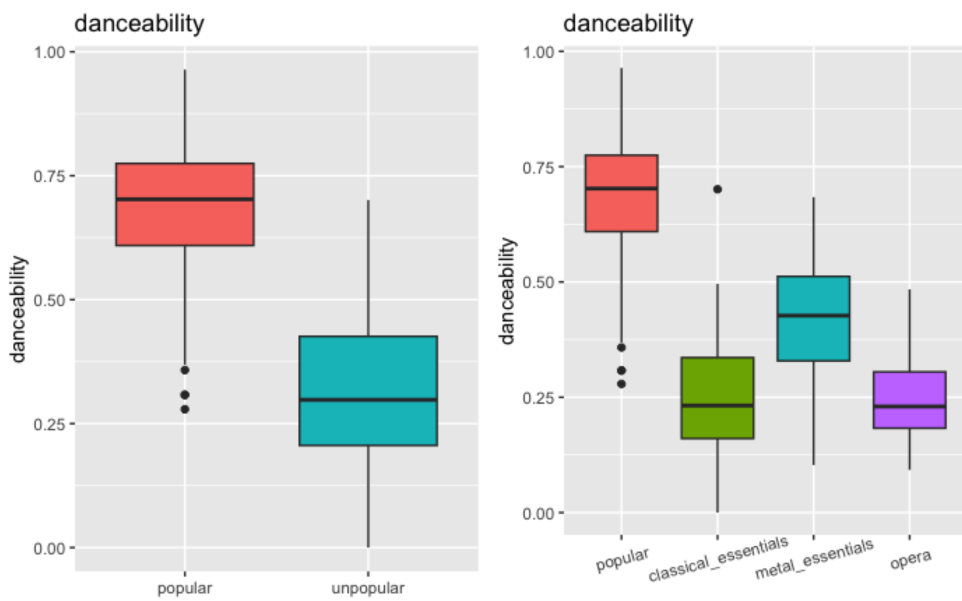


Figure D: Popular vs Unpopular Songs



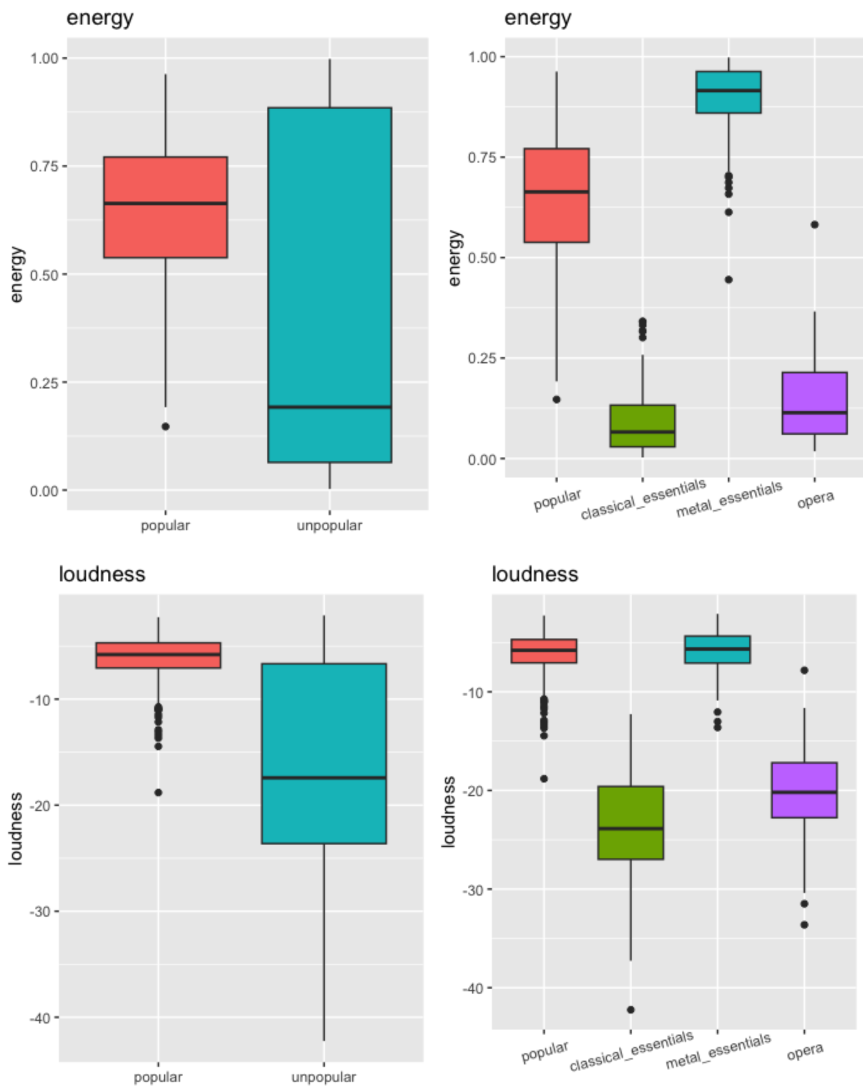

energy
energy
1.00
0.75
0.50
0.25
0.00
popular
unpopular
energy
energy
1.00
0.75
0.50
0.25
0.00
popular
classical_essentials
metal_essentials
opera
loudness
loudness
-10
-20
-30
-40
popular
unpopular
loudness
loudness
-10
-20
-30
-40
popular
classical_essentials
metal_essentials
opera

Figure E: All Variables Logistic Regression

```
Call:
glm(formula = group ~ ., family = binomial(), data = for_logreg)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.2427  -0.0002   0.0038   0.0351   3.1707

Coefficients:
                    Estimate Std. Error z value Pr(>|z|)
(Intercept)         5.833820   4.510901   1.293 0.195917
danceability       20.552934   4.258302   4.827 1.39e-06 ***
energy            -21.479491   4.475901  -4.799 1.60e-06 ***
key                 0.127746   0.108631   1.176 0.239609
loudness            0.663825   0.165478   4.012 6.03e-05 ***
speechiness         9.899627   7.629936   1.297 0.194469
acousticness       -6.029318   2.368924  -2.545 0.010922 *
instrumentalness   -8.077728   2.474385  -3.265 0.001096 **
liveness           -3.554704   2.585600  -1.375 0.169191
valence            -2.470925   1.937845  -1.275 0.202278
tempo               0.019849   0.013304   1.492 0.135728
artist_popularity   0.136644   0.031994   4.271 1.95e-05 ***
duration_seconds   -0.022870   0.006323  -3.617 0.000298 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 713.981  on 567  degrees of freedom
Residual deviance:  70.836  on 555  degrees of freedom
AIC: 96.836

Number of Fisher Scoring iterations: 10
```

Figure F: All Variables SVM

```
Call:
best.svm(x = group ~ ., data = train, cost = seq(0.1, 100, 0.1), kernel = "radial")


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  11.4

Number of Support Vectors:  103
```

Figure G: Main Variables Logistic Regression

```
Call:
glm(formula = group ~ loudness + valence + danceability, family = binomial(),
    data = for_logreg)

Deviance Residuals:
     Min        1Q    Median        3Q       Max
-2.53617  -0.04473   0.12658   0.30166   2.45631

Coefficients:
             Estimate Std. Error z value Pr(>|z|)
(Intercept)  -5.99615    0.93510  -6.412 1.43e-10 ***
loudness      0.17711    0.05244   3.378 0.000731 ***
valence      -1.60651    0.96575  -1.663 0.096217 .
danceability 16.55425    1.78337   9.283  < 2e-16 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 713.98  on 567  degrees of freedom
Residual deviance: 223.49  on 564  degrees of freedom
AIC: 231.49

Number of Fisher Scoring iterations: 7
```

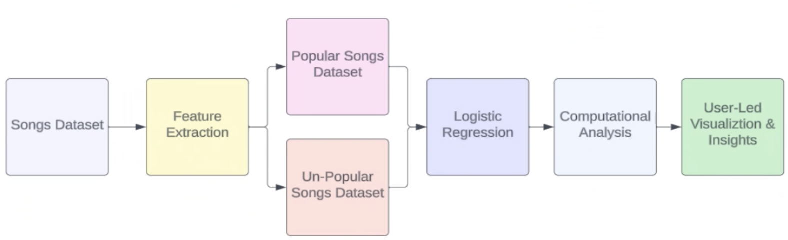
Figure H: Main Variables SVM

```
Call:
best.svm(x = group ~ loudness + valence + danceability, data = train, cost = seq(0.1, 100, 0.1), kernel = "radial")


Parameters:
   SVM-Type:  C-classification
 SVM-Kernel:  radial
       cost:  11.5

Number of Support Vectors:  112
```

Process Flowchart



REFERENCES

[1] Omar Hujran (2020) Big Data and its Effect on the Music Industry, The 3rd International Conference on Software Engineering and Information Management, DOI:10.1145/3378936.3378957

[2] Bauer, C., & Schedl, M. (2019). Global and country-specific mainstreaminess measures: Definitions, analysis, and usage for improving personalized music recommendation systems. *PLOS ONE*, *14*(6). https://doi.org/10.1371/journal.pone.0217389

[3] Gupta, Shubhanshu. (2014). Music Data Analysis: A State-of-the-art Survey. Dhirubhai Ambani Institute of Information and Communication Technology

[4] Sciandra M, Spera IC. A model-based approach to Spotify data analysis: a Beta GLMM. J Appl Stat. 2020 Aug 10;49(1):214-229. doi: 10.1080/02664763.2020.1803810. PMID: 35707796; PMCID: PMC9042099

[5] Kim, J. (2021, October 29). *Music popularity prediction through data analysis of Music's characteristics*. International Journal of Science, Technology and Society. https://www.sciencepublishinggroup.com/journal/paperinfo?journalid=183&doi=10.11648%2Fj.ijsts.20210905.16

[6] Chryssanthacopoulos, J. (2020, December 15). *Advanced Music Analytics using machine learning*. Medium. Advanced music analytics using machine learning

[7] Interiano, Myra (2018), Musical trends and predictability of success in contemporary songs in and out of the top charts, The Royal Society, v.5(5). PMCID: PMC5990848; PMID: 29892348

[8] Rasmus Fleischer (2021) Universal Spotification? The shifting meanings of "Spotify" as a model for the media industries, Popular Communication, 19:1, 14-25, DOI: 10.1080/15405702.2020.1744607

[9] Sun P. Music Individualization Recommendation System Based on Big Data Analysis. Comput Intell Neurosci. 2022 Jul 5;2022:7646000. doi: 10.1155/2022/7646000. PMID: 35837215; PMCID: PMC9276508.

[10] Khulusi, R., Kusnick, J., Meinecke, C., Gillmann, C., Focht, J. and Jänicke, S. (2020), A Survey on Visualizations for Musical Data. Computer Graphics Forum, 39: 82-110. https://doi.org/10.1111/cgf.13905

[11] K. E. Barkwell et al., "Big Data Visualisation and Visual Analytics for Music Data Mining," 2018 22nd International Conference Information Visualisation (IV), Fisciano, Italy, 2018, pp. 235-240, doi: 10.1109/iV.2018.00048

[12] Spotify Developer Policy. https://developer.spotify.com/policy

[13] Taren-Ida Ackermann, “Rationales and functions of disliked music: An in-depth interview study,” 2021 Max Planck Institute for Empirical Aesthetics, Frankfurt am Main, Germany. PMCID: PMC8846515