Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane

Group 19 - Project Check-in

Our project is the same concept, a music recommendation system, but the implementation has changed. Previously, we mentioned using a Generative Adversarial Network trained on a public music dataset. To be more specific, the user would first select a few songs from a set list. Then, the program will generate a user profile to create a tailored playlist for them. Afterward, the user will give feedback on whether they liked the playlist, and that's how the model would learn. However, now, we're using a model that utilizes the k-mean clustering algorithm. To start, the user takes a survey to specify their music preferences. We gather information on whether they want to include explicit songs, their favorite genre, preference for audio features (e.g. how much danceability they want), and a list of five songs the user likes. Then, we generate music recommendations with three different methods. The first way is to iterate through the user's song list, find the song's cluster, and calculate the cosine similarity between that song and the songs in its cluster. The second way is to calculate the cosine similarity between the user's audio feature preferences and the songs in the dataset. The final way is to calculate the mean vector for the song list and, again, calculate the cosine similarity. We might also find the mean vector's cluster. Also to clarify, we're using three different methods with the hope that they diversify and provide both more general and fine-grained recommendations.

Through our research, we found that k-mean clustering would be more effective than GAN because it has a user-centered approach, which would allow the user to take a survey so we can more efficiently narrow down our results. GAN requires a lot more training, and with the data set we are using, we do not have a lot of user-specific data or associations. Our current data set explores features of different tracks on Spotify. Our goal is to take the songs that the user inputs and find closely related songs based on how similar they are in values. The songs that are most alike and have the least differences in their distribution for each variable are our desired outputs.

A milestone is that we have a working survey. However, it can still be improved, whether that is implementation-wise or just adding more questions. Also, we made and trained the model that uses k-mean clustering. Finally, we can output some recommendations using two of the three methods mentioned earlier. The recommendations are not great, but it's progress nonetheless!

Another milestone was completing our initial exploratory data analysis. We selected several pairs of variables, such as instrumentalness and danceability, and tried to see if the values were concentrated in a specific region. From this, it would be easier to see which variables have a strong correlation and use that information to determine a song most like it. Energy and loudness have a clear exponential correlation. There

seems to be a higher danceability score for the 4/4 tempo, which logically makes sense as dance beats are broken into groups of 4 (or 8). Correlations like these help inform which songs will likely pair together.

Some challenges encountered are struggling to get started, learning the libraries, and figuring out the best method for our specific dataset. Firstly, the professor and TA mentioned clustering algorithms (specifically k-means), so we looked up examples to help us get started. That was extremely helpful as it gave us a starting point to work from. From there, we were able to recognize that an algorithm that focuses on user preferences would be more beneficial to our program. For the second one, I used a lot of online resources (e.g. the library's user guide) and scrutinized the errors. Lastly, it was mainly trial and error and coming to terms with the fact that there is probably no best method.

We are working on a Shiny App to further explore correlations between the data sets. This app will produce visuals that compare artists and genres with their score distributions across specific variables. For instance, the user might want to compare the popularity scores across the different genres. They will first group by the genre and summarize the distribution of popularity scores within that genre.

An anticipated challenge in the coming stages is finetuning our program to give good, or at least decent, recommendations in the limited time left. I plan to address this challenge by experimenting with different filters (e.g. filtering based on a popularity threshold), reevaluating my implementations, and gathering more user data. Since we have very limited user data, we could always add more questions to the survey. In other words, our biggest challenge is figuring out how to give decent recommendations with the user data accessible to us.

As mentioned previously, we obtained some preliminary results by being able to output recommendations with two of the methods. However, they're not good recommendations, as filtering and adjustments to the code still need to be done. The results signify that it's pretty likely that we could figure out how to output decent recommendations by next week. However, outputting good recommendations is questionable.

Throughout this process, it became increasingly clear that understanding the data set is more challenging than anticipated. Identifying the direction that the exploratory data analysis should take and their patterns requires a lot of additional context to understand how these correlations would actually work to inform the model.

In terms of feedback, we would like more direction on how to train our model based on the given data and how to find more patterns that will better guide our model. Our recommendations are not strong yet, so we were wondering if you had any advice on additional steps we can take to fine tune our recommendations.

Contribution and Work Division:

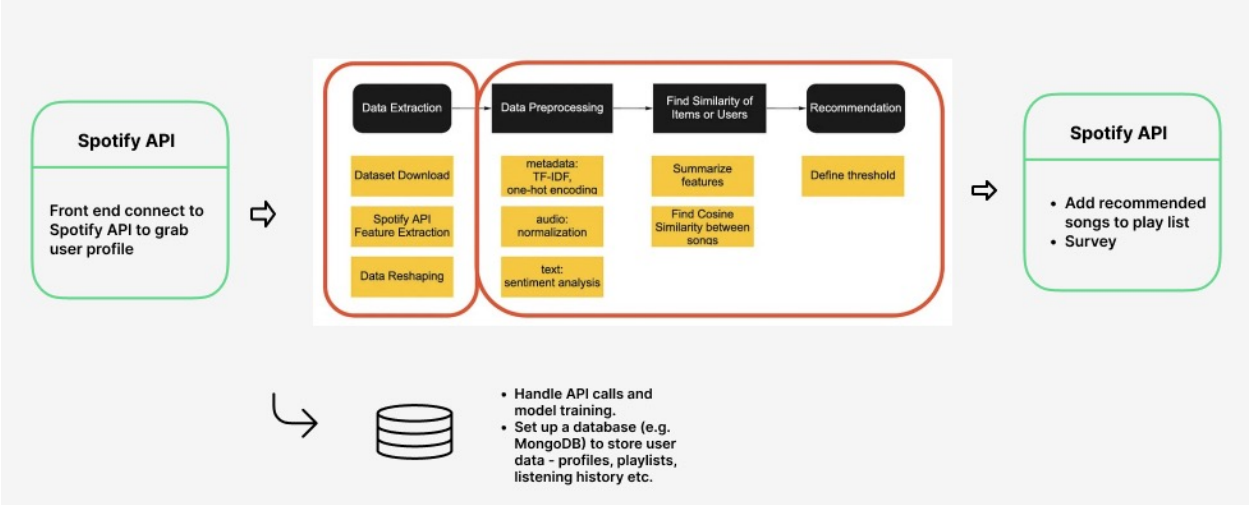
contribution:

5. jaishree
    a. cleaning data
    b. eda
6. vanessa
    a. sample survey
    b. k mean clustering
    c. music recommendation system
7. nico
    a. created GitHub repository
8. yacob
    a. outlined project goals and specifics

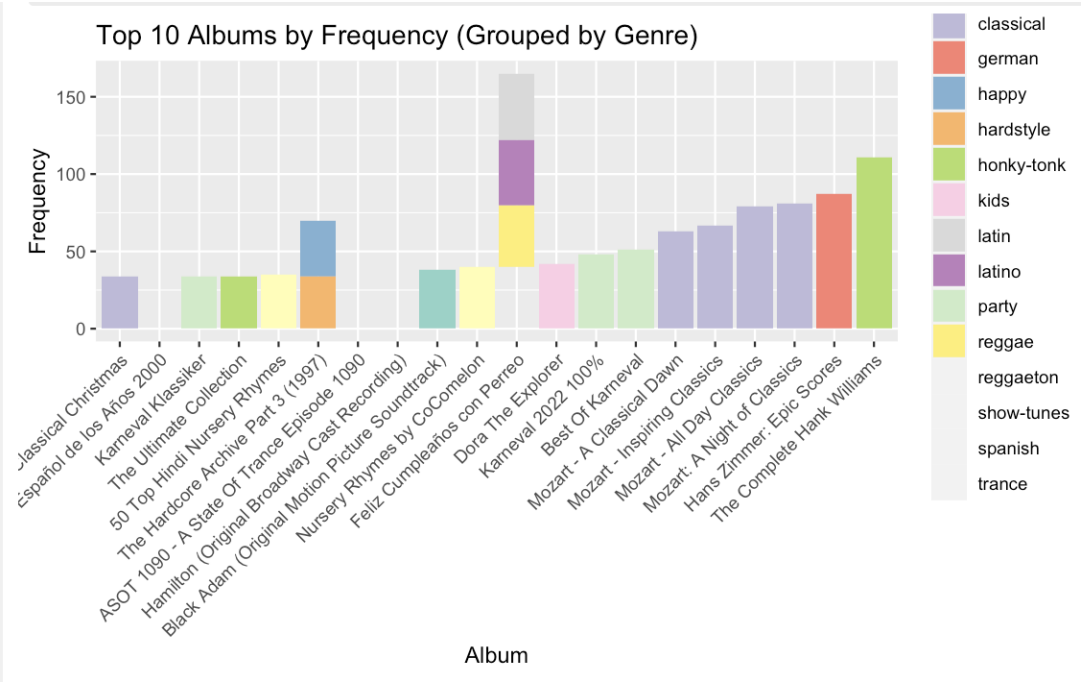
remaining work division:

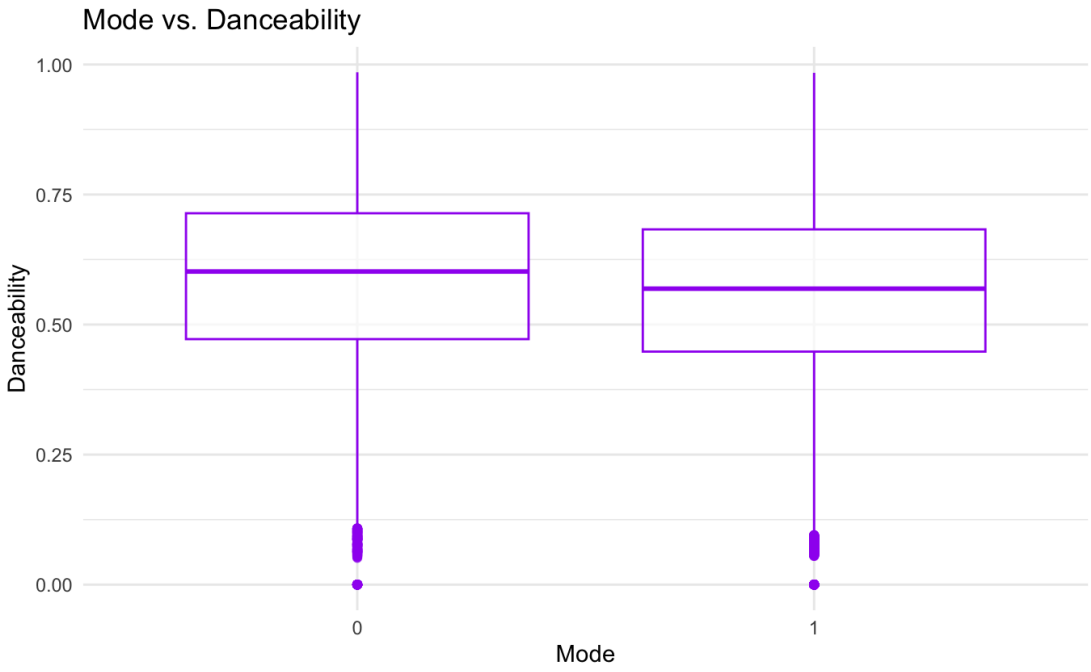
5. jaishree
    a. further EDA to figure out how to fine tune recommendation system
        i. using shiny apps
6. vanessa
    a. finishing the music recommendation system
7. nico
8. yacob
    a. Connecting to spotify API
    b. context-based outputs via collaborative filterin

Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane

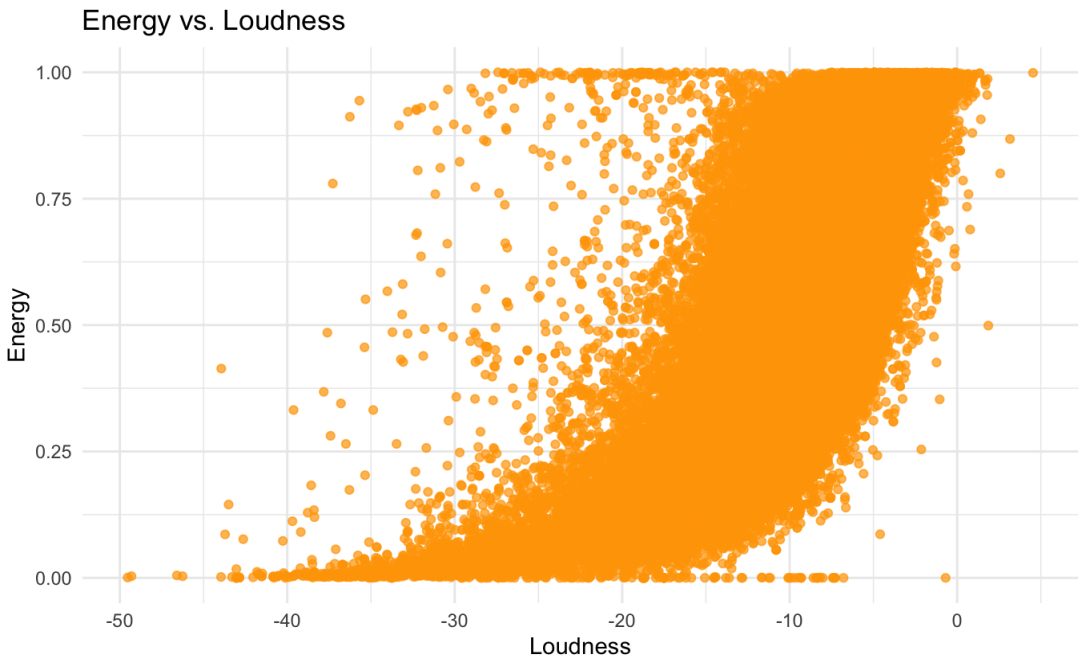


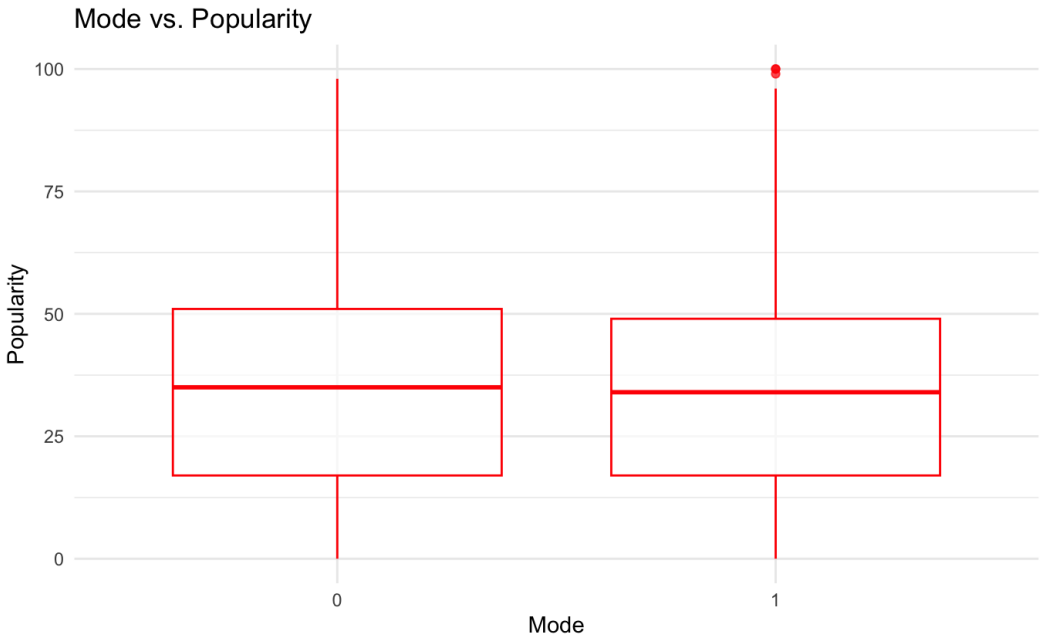
Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane



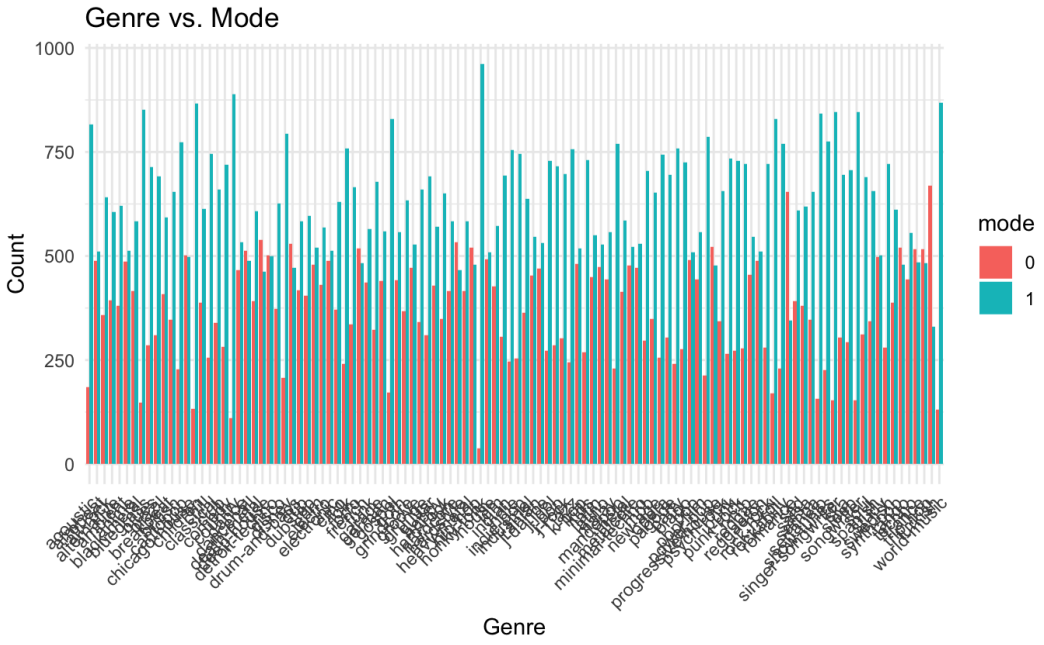


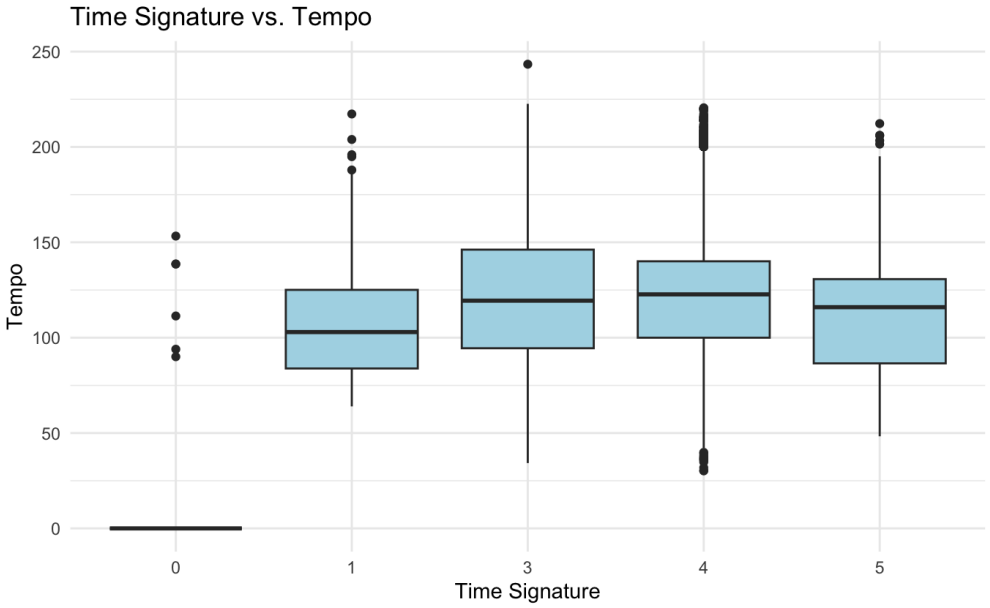
Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane



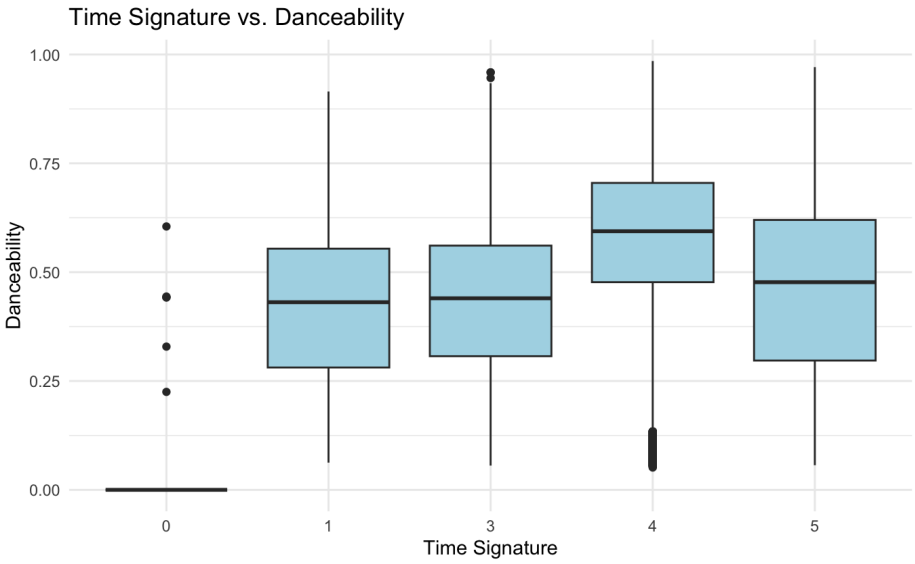


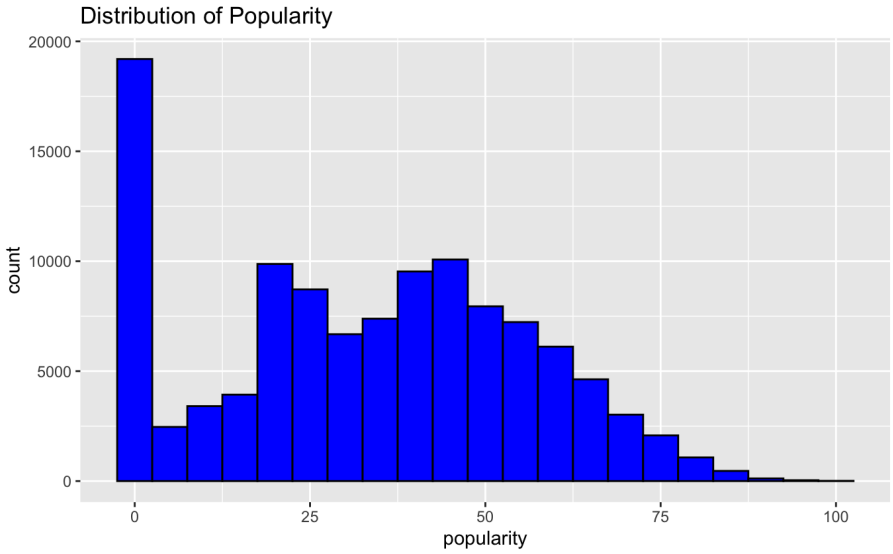
Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane



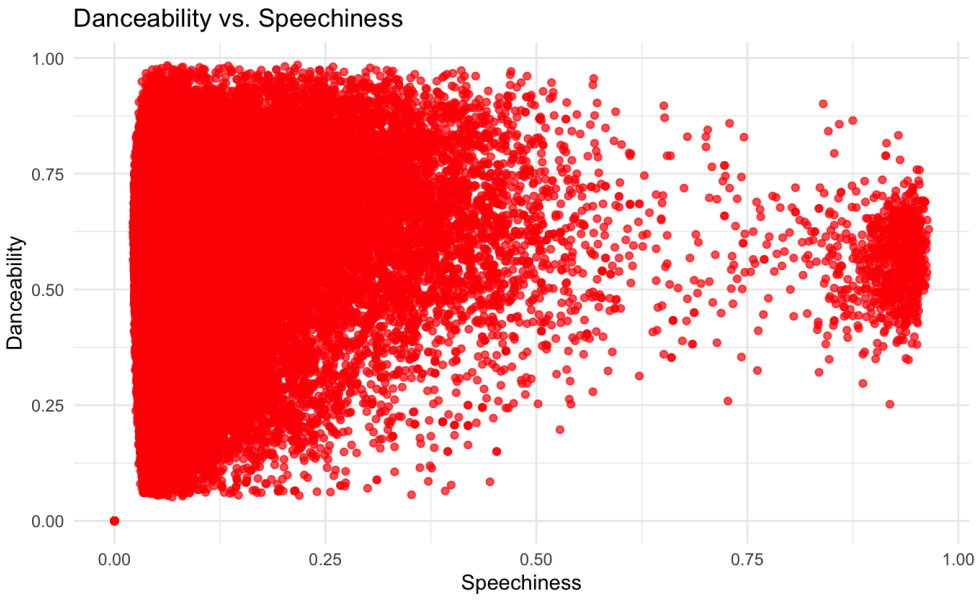


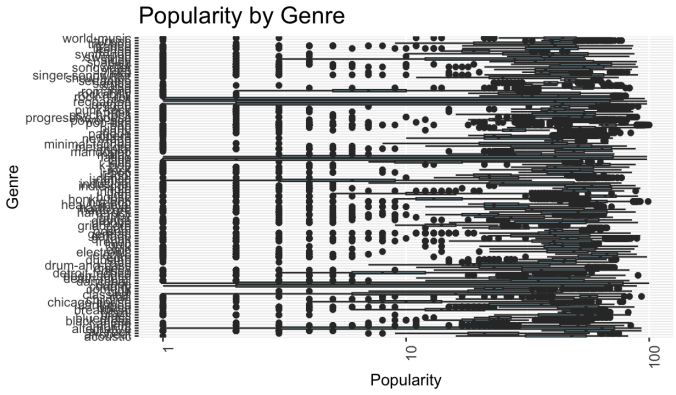
Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane

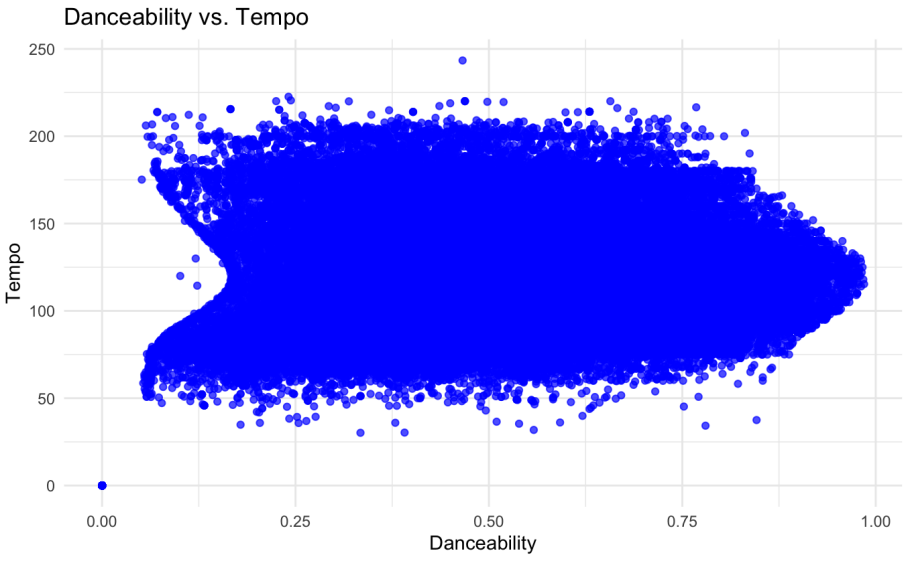




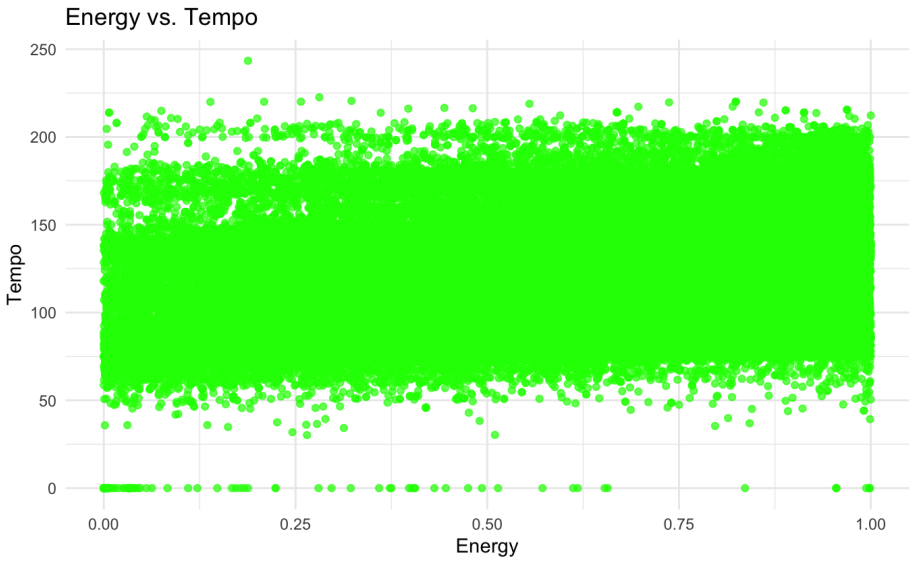
Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane

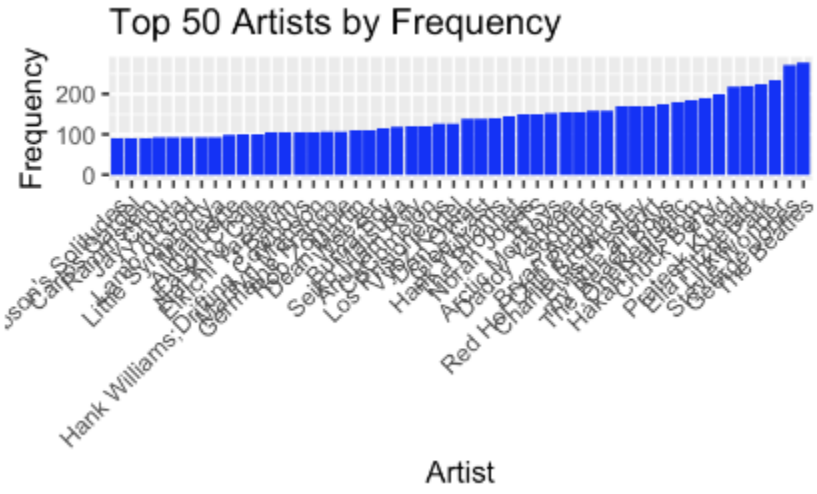


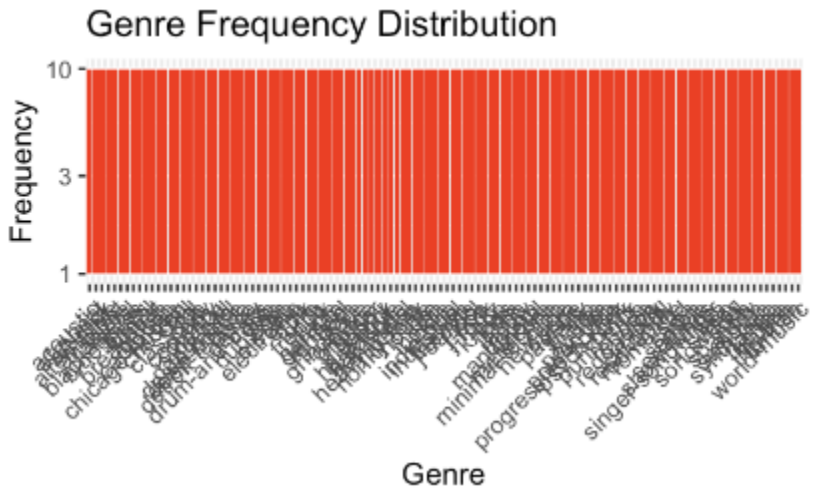




Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane





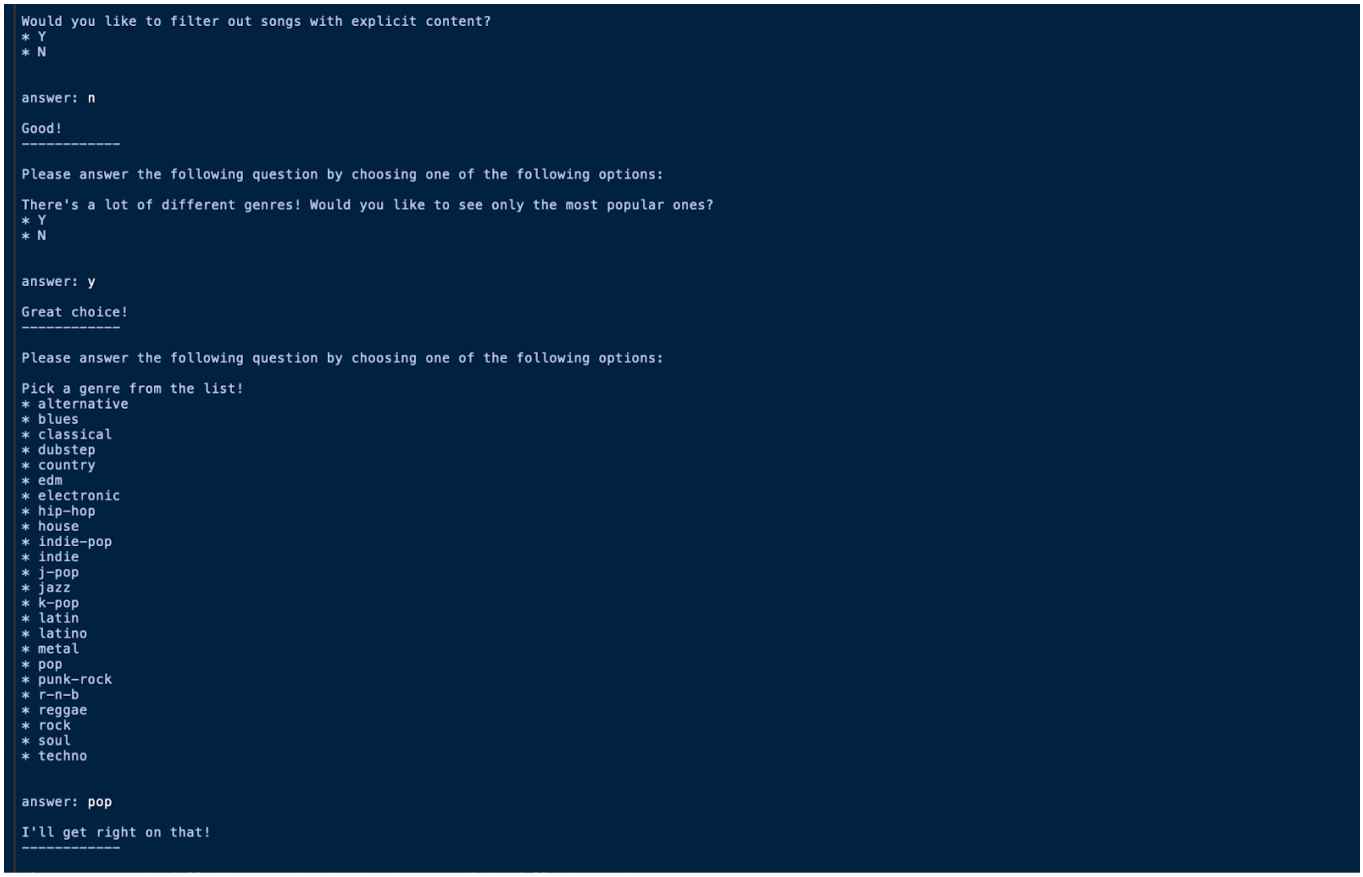


Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane

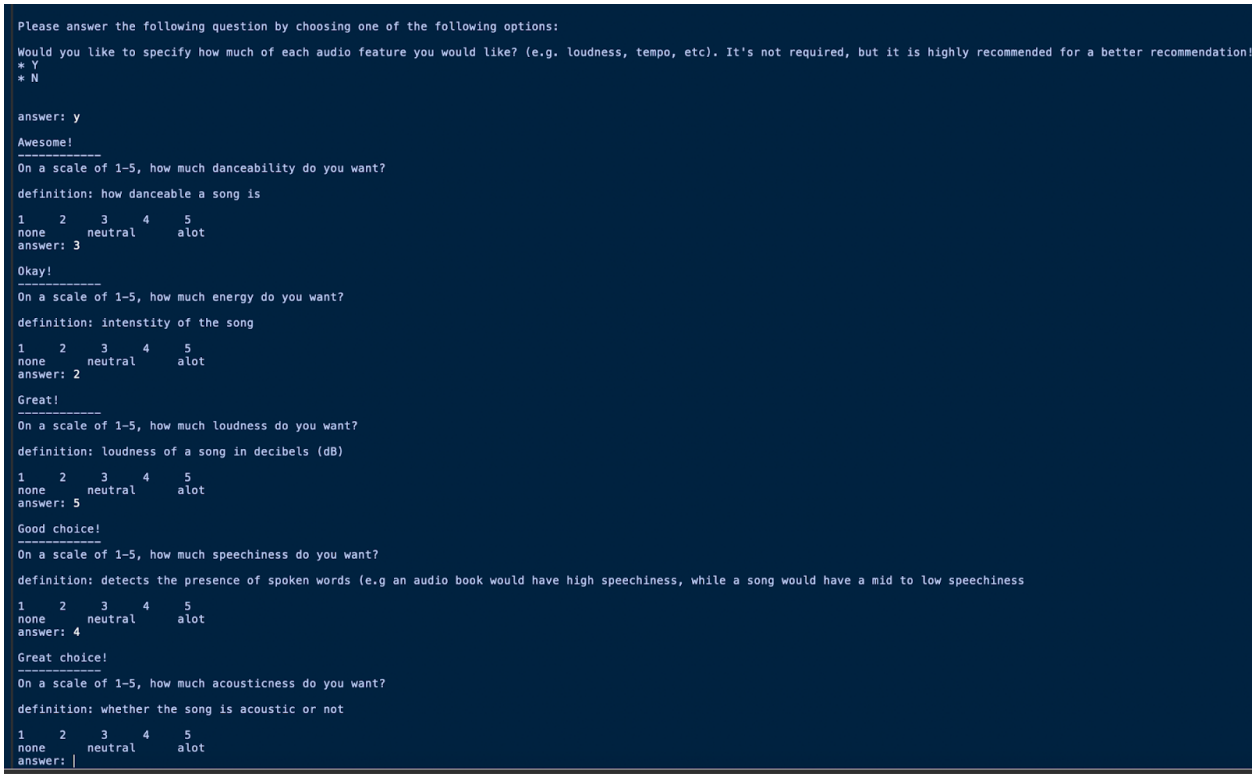
survey:

The first three questions:

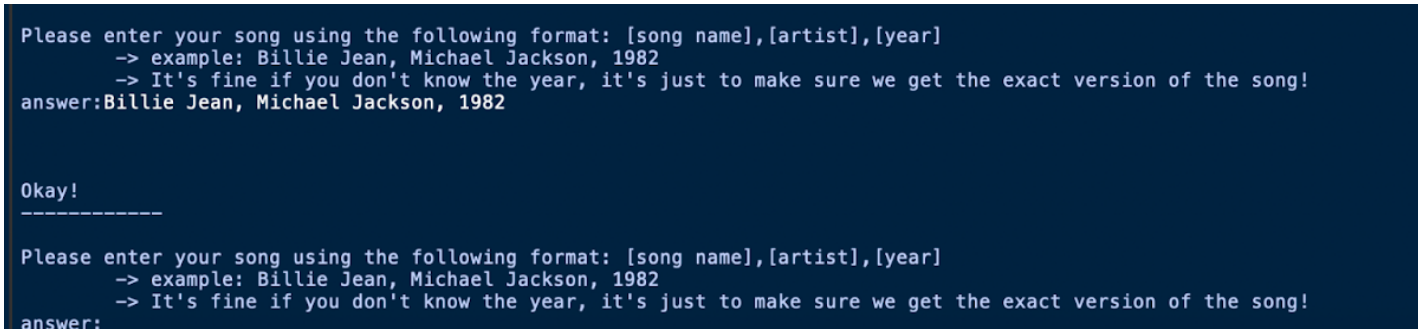
- asks about whether they want explicit songs
- to pick their favorite genre (show only the popular or all the genres in the dataset)



The next question asks the user whether they want to specify how much of each audio feature they want. The screenshot doesn't show all the features.



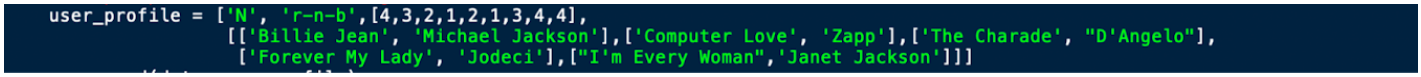
asks the user to input a song five times:



music recommendations:

sample user profile that will be used to make recommendations:

- tried to make it similar genres



method one:

clusters each song in the song list and calculate the cosine similarity between the song and the songs in the cluster

Jaishree Ramamoorthi, Vanessa Garcia, Nico Del Bonta, Yacob Kidane

```
- Heaven Knows I'm Miserable Now - 2011 Remaster by The Smiths
- I Love You Will Still Sound The Same by Oh Honey
- Quiero Decirte by Abraham Mateo;Ana Mena


- Oasis - Kyco x Barkley Remix by The Him;Sorana;Kyco;Barkley
- The Space In Between by Jan Blomqvist




- Addicted by Mishaal;Teqkoi;Ondi Vil
- Easy by DaniLeigh;Chris Brown
- Perfect Pair by Unodavid
- Cut My Hair by Mounika.;Cavetown
- I Like Me Better by Lauv


- Heart Burn by SUNMI
- The Real Slim Shady by Eminem
- Jump Around by House Of Pain
```

method two:

calculates the cosine similarity between the user audio preferences and all the songs in the dataset. will probably incorporate clustering and a popularity threshold

```
Song: Shout to the Lord - Live, Artist: Hillsong Worship;Integrity's Hosanna! Music;Darlene Zschech, Cosine Similarity: 0.9999906763492
Song: Family Life, Artist: New Model Army, Cosine Similarity: 0.9999923330256335
Song: Vem, Artist: Samuca e a Selva, Cosine Similarity: 0.9999887061891666
Song: 恋人 (Original Version), Artist: Masaharu Fukuyama, Cosine Similarity: 0.9999897940563282
Song: The Buffalo, Artist: Christopher Zondaflex Tyler, Cosine Similarity: 0.9999887312439555
Song: Reflections Of My Life, Artist: Marmalade, Cosine Similarity: 0.9999888151129299
Song: Qué te ha dado esa mujer, Artist: Los Freddy's, Cosine Similarity: 0.9999933651235015
Song: Favorite T, Artist: The Lemonheads, Cosine Similarity: 0.9999889085245441
Song: Chaiyya Chaiyya, Artist: Sukhwinder Singh;Sapna Awasthi, Cosine Similarity: 0.999988791286104
Song: Карнавал, Artist: Valery Obodzinsky;Оркестр п/у Вадима Людвиковского, Cosine Similarity: 0.9999885626439037
Song: Torturas de Amor - Ao Vivo em Campina Grande, Artist: Zezo, Cosine Similarity: 0.9999951757904715
Song: Superhéroes, Artist: Charly García, Cosine Similarity: 0.9999887566348656
Song: Mentira - En Vivo desde Puerto Rico, Artist: Gilberto Santa Rosa, Cosine Similarity: 0.9999916730087149
Song: Não Viva Em Vão - Versão Acústica, Artist: Charlie Brown Jr., Cosine Similarity: 0.999993155898378
Song: Come and Talk to Me, Artist: Desmond Dennis;Demarious Cole;Tone Stith;Shade Jenifer, Cosine Similarity: 0.9999899232727901
Song: Fera Ferida - Versão Remasterizada, Artist: Roberto Carlos, Cosine Similarity: 0.9999929341874791
Song: One Night (feat. Leila's Cat Loralie), Artist: Leila Bela;Leila's Cat Loralie, Cosine Similarity: 0.9999883814448991
Song: Golden Brown, Artist: The Stranglers, Cosine Similarity: 0.9999954403669508
Song: Labelled With Love, Artist: Squeeze, Cosine Similarity: 0.9999929192285749
Song: And I Love You So, Artist: Elvis Presley, Cosine Similarity: 0.9999896889546027
Song: High Hopes, Artist: Pink Floyd, Cosine Similarity: 0.9999921865272038
Song: The Crying Game, Artist: Boy George, Cosine Similarity: 0.9999910187907929
Song: That's Good, That's Bad (with Lacy J. Dalton), Artist: George Jones;Lacy J. Dalton, Cosine Similarity: 0.9999925555726769
>>
```