Tommy Cissel
Max Fisher
Richard Rasheed
CSC 272
4/29/2019

When, Where and What did I just hear?:

An Analysis of Music from Spotify based on time period, location, and genre.

I. Introduction

Music has always been an interest for many, however the idea of an analysis of data from various types of music may seem out of the realm of possibility. In our project, we took the various attributes that Spotify uses to classify music and attempted to predict where the track would be popular, when a track would be popular, and which genre the track fell into. We have analyzed different categories of Spotify playlists, such as ones with the top songs from different decades, and the top 50 most popular current songs by country. Along with that, created a model using Weka to predict what genre of music a song is based off of the attributes that Spotify gave us. We looked at Country, Jazz, Hip-Hop, Classic Rock, and Dance playlists which are compiled of the top music in each of those genres. While the models we created struggled to accurately predict the correct continent of popularity, or decade in which songs were popular, it did a far better job of predicting the genre each song identifies under. In this report, we use Spotify API data to make these predictions, along with discussing how we acquired our data, analysis and what we needed to do to prepare it for analysis, along with discussing our results and making conclusions based off of these results.

II. Datasets, Datasets, Datasets

Our datasets come from the Spotify API (https://developer.spotify.com/), which required significant cleansing and processing before it was ready for analysis. The dataset has several characteristics to use as attributes in our analysis, including Acousticness, Danceability, Energy, Instrumentalness, Key, Liveness, Loudness, Mode, Spechiness, Tempo, Time Signature, and Valence. Acousticness, Danceability, Instrumentalness, Liveness, Speechiness, and Valence are indicators that are on a scale of zero to one. We have three different datasets, all comprised of the audio features from songs from Spotify.

Genre Dataset:

- This data set has 341 instances and it is compiled of 5 spotify created genre specific playlists that updated frequently. Each playlist is based on one of the genres we were interested in, that is Hip-Hop, Classic Rock, Country, Jazz, and dance. These genres create the only nominal attribute for the dataset, as all the other attributes are numeric and from the audio features from the songs.

Continent Dataset:

- This dataset has 1050 instances and is a combination of 21 different Spotify playlists that were created by Spotify. The playlists are all 50 of the top songs currently listened to on spotify in a particular country. The playlists are updated almost everyday by Spotify. The dataset in addition to the audio features has a nominal attribute that distinguishes which continent's top 50 playlist the song came from. We grouped the different Country playlists into ones by their continents because of the similarities of the data for countries close to each other. The countries we used were United States, Canada, Mexico, El Salvador, Brazil, Argentina, Chile, Columbia, Uruguay, Japan, Hong Kong, Philippines, Vietnam, Thailand, United Kingdom, France, Germany, Spain, Finland, Romania, and Italy. These were all put into groups based on the continent they are located in to create the nominal attribute of location which were North America, South America, Asia, or Europe. Africa is not present because there weren't enough spotify playlists of top 50 songs in African countries.

Decade Dataset:

- This dataset has 494 instances and is a combination of 5 different Spotify playlist that were created by Spotify. The playlists were about 100 songs each and were playlists from the 70s, 80s, 90, 00s, 10s. Thus in addition to the audio features, there is a nominal attribute that distinguishes what decade the song is from.

III. Dataset Preparation

Getting our datasets was pretty tricky. The Spotify API is very popular and well known so there are some existing softwares that allow the API to be easily accessed using Python. Spotify is the name of the software we used when developing the program that gave us our data in a readable format from the Spotify API. Our Python code had to be able to authorize a user to access the Spotify API using a token and then extract specific information from the desired areas that we needed for multiple songs. Once the data was accessible and able to be read we changed the code to be able to output an entire playlist worth of data in a .arff style data instance, thus allowing us to use many different playlists. This task was not a short or easy one, as creating a program that was able to check someones Spotify credentials and authorize them access to all of the API data took a lot of effort and precision. Then to be able to only extract a couple of things needed in our data set out of the hundreds of pieces of information that the Spotify API has for each track and then finally after all that being able to produce an .arff formatted data instance was a heavy load. After converting all of our data into Weka, we selected only the most relevant attributes and ones that would be most predictive, leaving out ones such as speechiness, which is a binary indicator which as it gets closer to one, is a higher indicator of whether or not the track is purely speech. We left out variables like this because it is exceptionally unlikely for popular songs to be purely speech, which would just add a non-predictive, unnecessary variable to the data. We were left with nine attributes, these being:

Attribute	Description	Type
Danceability	A range of how well a song can be danced along to. As danceability gets closer to one, a song is considered to be more danceable and as it gets closer to zero, it is considered to be less danceable. One might assume that at least in present day music	Numeric (values from 0 to 1)
Energy	As this variable gets closer to one, the higher the energy of the song is. This could be correlated with danceability because higher energy songs usually have a faster tempo and are more danceable.	Numeric (values from 0 to 1)
Key	The key that the song is played in (if one is not detected, it is defaulted to 0)	Numeric

Loudness	Average decibels over the course of the song.	Numeric
Acousticness	Measures whether or not a track is purely acoustic. As it approaches one, that represents a higher confidence that a track is acoustic.	Numeric (values from 0 to 1)
Instrumentalness	Similar to that of acousticness. As it approaches one, that represents a higher confidence that a track is purely instrumental, with no vocals.	Numeric (values from 0 to 1
Tempo	Beats per minute in the song.	Numeric
Valence	Represents the overall happiness of a song. As it is closer to one that represents a sad song, while it approaches five that represents a happier song.	Numeric (from 1 to 5)
Duration	Represents the duration of the song measured in milliseconds.	Numeric
Genre	The style of music that the song is. This is a class attribute. This attribute was only in the Genre dataset.	Nominal {Jazz, Country, HipHop, Classic Rock, Dance}
Decade	The decade the song was released in. This is a class attribute. This attribute was only present in the Decade	Nominal {70s, 80s, 90s, 00s, 10}

	dataset.	
Location	The continent the song was found to be popular in. This is a class attribute. This attribute was only present in the Continent dataset.	Nominal {Europe, North America, South America, Asia}

As all of our data, aside from the class attribute, was numeric, we had to discretize the attributes to prevent overfitting in the model, and to make sure the numeric attributes are nominal. This is because certain classification methods don't work with numeric attributes in the dataset. For discretization, we used a bin number of ten for every model in order to make it consistent. Then we had to seperate the two files into a training data to actually create the model and then an untouched test data set to test our model. This is needed to be done so that we can see if our model actually works for an unseen data set. In order to make two separate files, one of training data and one of test data, we needed to do some reasonable complicated data preparation. We had to first discretize all of the data, then use the unsupervised instance randomize filter, then lastly use the resample unsupervised instance filter to take out 80% of the data and put it in one document for training, then put the other 20% of data in another document for testing. We did this with each data set; the genre, continent, and decade. Next we will look at each dataset and the three different methods we used to analyze them.

IV. Data Analysis and Results

To Analyze this data, we used the .arff files in weka to make insights on the data easily and accurately. We analyzed this data with three different classification learning methods, Naive Bayes, which gives us the probability of the class attribute given an instance. It treats each attribute as independent of each other and then finds the probability of the class attribute being a certain output based on all the inputs and there frequency within the dataset. We also used OneR, which takes the one attribute that is the most predictive of the class attribute, and uses that to predict the class attribute in each instance. The last one we used for each dataset was Prism, which creates rules for the model, such as “if danceability >.85 and energy >.9, Genre = Dance” While this worked pretty well, it left many instances unclassified, with up to almost 42% with one dataset. This can be attributed to the fact that not every instance is going to be able to have a rule that works for it, and with more possible class attributes, this makes this more likely.

Genre

Naive Bayes	Correctly Classified - 66.667%	

```
=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
  5  0  1  1  0 |  a = HipHop
  0 18  2  5  1 |  b = ClassicRock
  1  1 10  1  0 |  c = Dance
  0  3  4  3  0 |  d = Country
  0  2  1  0 10 |  e = Jazz
```

The Naive Bayes algorithm yielded an overall accuracy of 66.67% on the Genre dataset. In 69 instances, this was best at classifying Jazz and Dance Music, both having accuracies of 10/13, or 76.92%. It was worst at classifying Country music, with an accuracy of 3/10, or 30%.

OneR	Correctly Classified - 43.47%	Duration was most predictive attribute

```
=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
  2  1  4  0  0 |  a = HipHop
  2 11  9  0  4 |  b = ClassicRock
  1  1 11  0  0 |  c = Dance
  0  0 10  0  0 |  d = Country
  1  6  0  0  6 |  e = Jazz
```

Using the OneR algorithm shows which one attribute is the most predictive for the class attribute. This model used duration as the most predictive attribute. As you can see, this model had an accuracy of 43.49%, and best predicted Dance music with an accuracy of 11/13, or 85% and worst predicted Country music with an accuracy of 0/10, or 0%.

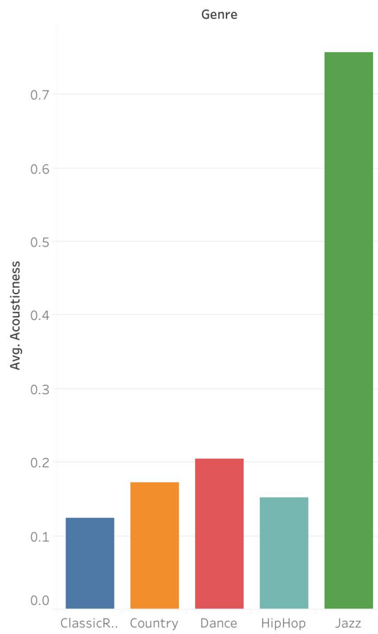
Prism	Correctly Classified - 46.3768%	

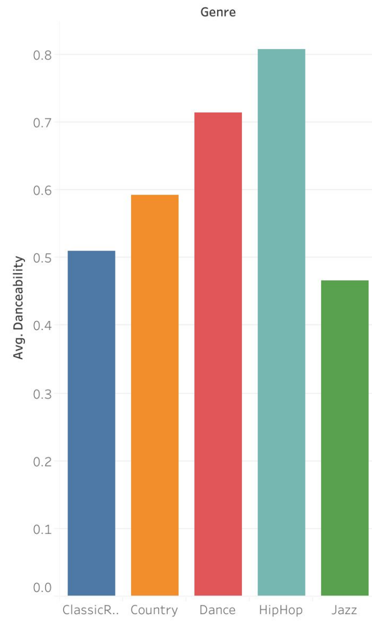
```
=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
  2  1  2  0  0 |  a = HipHop
  1 15  2  4  0 |  b = ClassicRock
  0  2  6  1  1 |  c = Dance
  2  3  0  2  0 |  d = Country
  0  3  0  0  7 |  e = Jazz
```

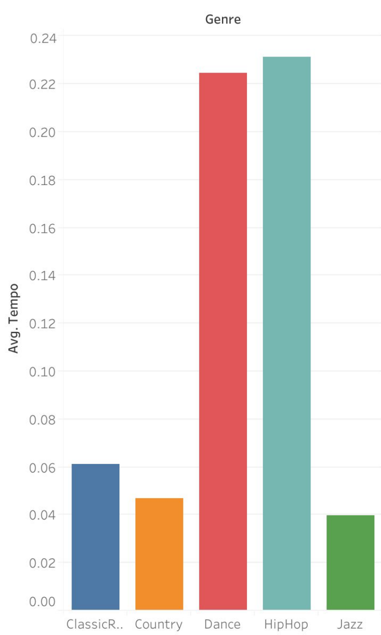
For the Prism algorithm, there is an overall accuracy of 46.37% with 31.88% incorrectly classified instances and 21.74% of instances unclassified. The best individual accuracy was for Jazz music with 7/10 or 70%. This model is the worst at predicting country music, with an accuracy of 2/7, or 28.57%.

Overall, the Naive Bayes algorithm best classified the Genre dataset, while the OneR algorithm worst classified the Genre dataset. We also saw while creating our model that when there were only two class attributes and they were both genres of music that are exceptionally different, such as Hip Hop and Classic Rock, there was a far higher accuracy because the attributes were farther apart in values, which makes it easier for the programs to distinguish between genres.

We also created some graphs that will help to visualize our data and to see some different correlations between the data that we wouldn't get to see through classification learning







These three graphs came from Tableau and were some of the more intriguing ones that we created from the genre data set. You can see that from the 1st graph on the left that as expected, Jazz has the highest average Acousticness. The middle graph and rightmost graph had the most insight between dance and Hip Hop. The attribute of danceability should go hands down to the Dance genre of music but the highest average Danceability goes to hiphop, and as correlates with high danceability is a high tempo, where Hip Hop also scores higher than Dance.

Continent

Naive Bayes

Naive Bayes	Correctly Classified - 41.9048%	

```
=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  0 20  4 10 |  a = NorthAmerica
  6 30  5 11 |  b = SouthAmerica
  5  2 28 15 |  c = Asia
  6 17 21 30 |  d = Europe
```

For the Continent data set the Naive Bayes method yielded a 41.90% overall accuracy. In 210 instances, it also best classified South America, with an accuracy of 30/52, or 57.69%. It worst classified North America, with an overall accuracy of 0/34, or 0%.

OneR	Correctly Classified - 35.2381%	Danceability was the most predictive attribute

```
=== Confusion Matrix ===

  a  b  c  d   <-- classified as
  0  8  3 23 |  a = NorthAmerica
  0 19  6 27 |  b = SouthAmerica
  0  3 23 24 |  c = Asia
  0 26 16 32 |  d = Europe
```

For the Continent dataset the OneR classification method gave an overall accuracy of 74/210, or 35.24%, using danceability as the most predictive attribute. It best classified Asia with an accuracy of 24/50, or 48% and worst classified North America with an overall accuracy of 0/34, or 0%.

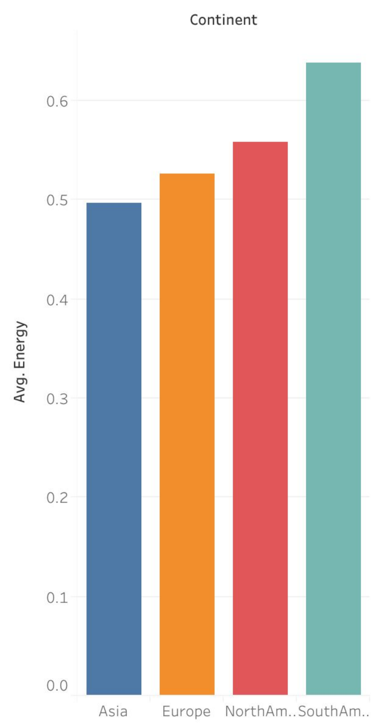
Prism	Correctly Classified - 30.9524%	

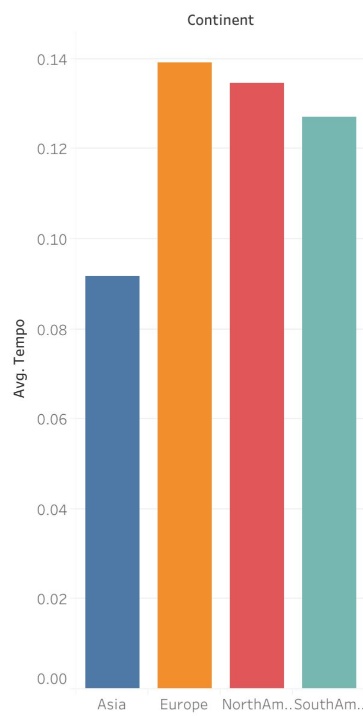
```
=== Confusion Matrix ===

  a  b  c  d   <-- classified as
 20  7  0  0 |  a = NorthAmerica
 27 12  3  3 |  b = SouthAmerica
 19  0 25  1 |  c = Asia
 27  9 13  8 |  d = Europe
```

The Prism algorithm showed an overall accuracy of 65/210, or 30.95%, with 51.90% of instances incorrectly classified and 17.14% of instances unclassified. It was best at classifying North America, with an accuracy of 20/27, or 74.07%. It was worst at classifying Europe, with an accuracy of 8/57, or 14%.

Overall, the Naive Bayes method was the best classifier for the Continent dataset, with an accuracy of 41.9048%, and OneR had the worst with an accuracy of 30.9524%. With the continent dataset, it makes sense that the classifiers could've had some trouble, as there is certainly some overlap between countries on each top 50 chart because countries like the United States and Canada, or some English speaking European countries listen to the same music.

We also created some graphs that will help to visualize our data and to see some different correlations between the data that we wouldn't get to see through classification learning in Weka.





These two graphs we created in Tableau showed two interesting aspects of the Continent data. South America has the high avg. Energy for its popular music, this does make sense to what we have heard and seen from South America as all the excitement and music came from the World Cup that was in Brazil, the music there is very high energy. The graph on the right to us sort of fit the stereotypical European music perfectly. There popular music has the highest average tempo than the other continents and this only lead us to thinking about European night clubs and the fast past music and the thumping of the club. It was kinda funny seeing that there was correlation for that.

<u>Decade</u>

Naive Bayes	Correctly Classified - 35.3535%	

```
=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
  8 10  3  3  0 |  a = 70s
  2 10  1  4  4 |  b = 80s
  1  3  4 10  3 |  c = 90s
  6  0  3  4  5 |  d = 00s
  3  1  1  1  9 |  e = 10s
```

For the Decade dataset, the Naive Bayes method gave an overall accuracy of 35/99, or 35.35%. It also best classified music from the 10s, with an accuracy of 9/15, or 60%. It worst classified music from the 90s, with an accuracy of 4/21, or 19.05%.

OneR	Correctly Classified - 36.3636%	Energy was the most predictive attribute

```
=== Confusion Matrix ===

  a  b  c  d  e   <-- classified as
 16  0  0  3  5 |  a = 70s
  5  0  0 10  6 |  b = 80s
  5  0  0 10  6 |  c = 90s
  5  0  0 10  3 |  d = 00s
  2  0  0  3 10 |  e = 10s
```

The OneR classification method has an overall accuracy for the Decade dataset of 36/99, or 36.36%, marginally better than that of Naive Bayes. It best classified songs from both the 70s and the 10s, with accuracies of 16/24 and 10/15, or 66.67%. It worst classified songs from the 80s and 90s, with both with accuracies of 0/21 and 0%.

Prism	Correctly Classified - 21.2121%	

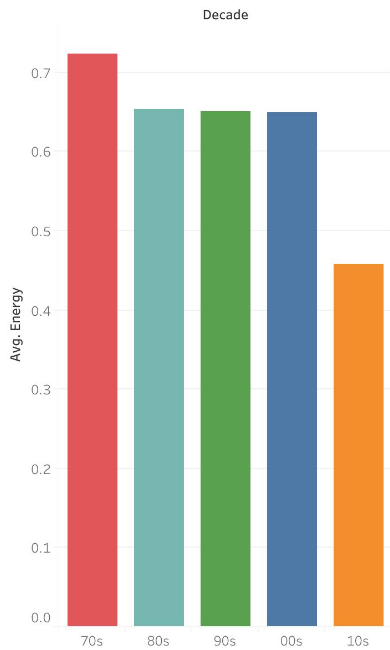
```
=== Confusion Matrix ===

 a b c d e   <-- classified as
 7 5 0 0 0 | a = 70s
 3 3 0 3 2 | b = 80s
 2 3 2 3 0 | c = 90s
 4 1 5 3 2 | d = 00s
 0 1 3 0 6 | e = 10s
```

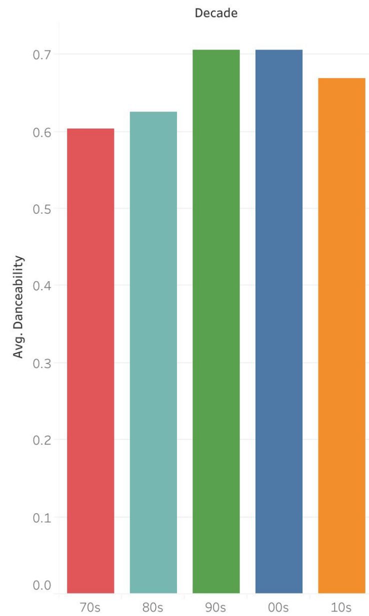
Lastly, for the Decade dataset the Prism classification method gives an accuracy of 21.21%, with 37.37% of instances incorrectly classified and 41.41% of instances unclassified. It best classified music from the 10s, with an accuracy of 6/10, or 60% and worst classified music from the 90s with an accuracy of 2/10, or 20%.

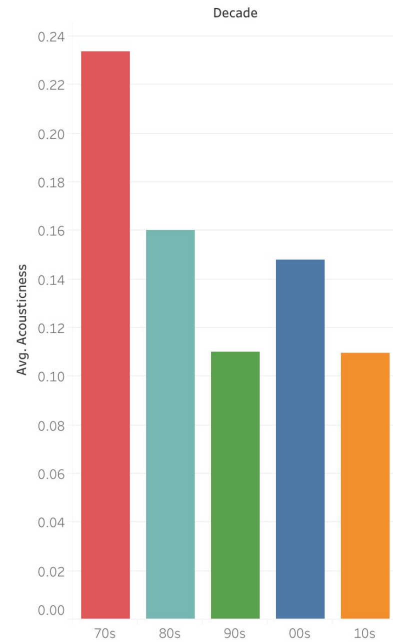
Our model had the most trouble classifying songs by different decades. Unlike the first two, OneR had the best accuracy of 36.3636%, and Prism had the worst with an accuracy of 21.2121%. For these models, it makes sense that when there are classification errors, they would be close in decade to one another, as opposed to one farther away. This is because there is a greater difference in music styles from the 70s to the 10s than there is from the 00s to the 10s.

On average, we were able to see that Naive Bayes was the best at predicting Genre, Continent of Popularity, and Decade, with an average accuracy of 47.9751%. Prism had the worst classifying accuracy, with an average of 32.8471%. These models likely best classified the genre dataset because music from the same genres is more common to be similar in the attributes we analyzed these songs for, while music popular in different decades, or on different continents can be of any genre, therefore likely has a wider range for each attribute. Some of the mistakes on certain songs can be attributed to a lack of test instances, because even though we used an 80/20 ratio of training to test data, one of the sets of test data only had 69 instances.

We also created some graphs that will help to visualize our data and to see some different correlations between the data that we wouldn't get to see through classification learning in Weka.







These graphs show the changes from the 70s to present day in music for the attributes of Energy, Danceability, and Acousticness. The decline in Acousticness over the years makes sense as technology has improved so more beats and tunes are made using a computer so less Acoustic sounds, but what was interesting with that is Energy also has decreased over the years. We would've thought that with the advancement in digital beats and making electronic music, the sounds of the music would be more energetic and highly paced, but it has shown to have the exact opposite trend. Then what seems to bother me is the slight increase in Danceability, even though there is a decline in Energy. These two attributes by common sense seem to be directly proportional to each other but in this case they are not.

V. Conclusions

The main purpose of our project was to see whether or not we would be able to accurately predict genres, continent of popularity, and decade of release for some of the most popular songs given the data collected from the Spotify API. During our data preparation we used Python code to be able to access the Spotify API and then format the data into something that is compatible with Weka. After this, we created a training set of each dataset, and a test dataset. These three datasets were Genre, Continent, and Decade. We then used Naive Bayes, OneR, and Prism to analyze each dataset and see how accurate a classification model would be.

Overall, many of the algorithms were only marginally better than if we had just randomly guessed, with our worst one being the Prism algorithm for Decades, at a 21.2121% accuracy with five attributes to guess from, while the best method was Naive Bayes for Genres, with an accuracy of 66.67%. We discovered that our model was best at predicting Genres of music, while it had the most trouble predicting the Decade of popularity. While we had hoped for better, more consistent results, this makes sense because it makes the most sense that the construction of songs would vary the most in different Genres, while within the same Genre the structures would carry the most similarities. After conducting all of these tests we were able to conclude that while these models struggled at distinguishing similar attributes from one another, they would excel far more if there were only two class attributes that weren't greatly similar, like Hip Hop and Jazz, or 10s and 70s. We had high hopes for the Naive Bayes algorithm having success in our analysis, but Naive Bayes needs every attribute to be independent of each other. This could have caused us problems in our analysis because even though the attributes are different from each other, audio features such as tempo, danceability, and energy could be all connected with each other because those three especially are associated together in music that we like to dance to. This could be why Naive Bayes did not perform as well as we had hoped.

The struggles we had with our models being able to accurately predict the class attribute were disappointing and did not align with what we originally thought for the project, but it did give us some insight on how Spotify does what it does and how that is not simple at all. The Continent one was interesting to see because one would think that the dataset would a big difference in music listened in say North America versus Asia, but in all reality and when looking closely back at the playlists, people all over the World seem to listen to a lot of the same music so it would be hard to determine where a song might be most popular. Then for the Decade dataset, we tend to think of each Decade having very distinct music from each other but from our attributes our model couldn't accurately predict even though many people today could hear a song and be able to predict what decade it was from. As well the same thing goes for genre, it seems that a person could predict both decade and genre better than our models did, which really does intrigue me that there is more to the uniqueness of music other than just numbers we can get from a machine learning software and that is what we think was very intriguing about this project.

VI. Appendix

Spotify:
https://www.spotify.com/us/
Spotify API:
https://developer.spotify.com/documentation/web-api/
Spotipy:
https://spotipy.readthedocs.io/en/latest/
Decade Dataset Playlists:
All Out 70s
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DWTJ7xPn4vNaz?si=JkHAGD6qQXiQ3yW0joCkzw
All Out 80s
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DX4UtSsGT1Sbe?si=jqu3gGWaTgWIRtgx4hw93g
All Out 90s
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DXbTxeAdrVG2l?si=huAa1MN7RM2dpgPbolCWiw

All Out 00s
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DX4o1oenSJRJd?si=bA7i_rMwR46vVKJo_Qr1EA
All Out 10s
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DX5Ejj0EkURtP?si=Uqb9lAxqTTqkpYHOuEOf5A

Continent Dataset Playlists:
United States Top 50
https://open.spotify.com/playlist/37i9dQZEVXbLRQDuF5jeBp
Canada Top 50
https://open.spotify.com/playlist/37i9dQZEVXbKj23U1GF4IR
Mexico Top 50
https://open.spotify.com/playlist/37i9dQZEVXbO3qyFxbkOE1
El Salvador Top 50
https://open.spotify.com/playlist/37i9dQZEVXbLxoIml4MYkT
Brazil Top 50
https://open.spotify.com/playlist/37i9dQZEVXbMXbN3EUUhlg
Argentina Top 50
https://open.spotify.com/playlist/37i9dQZEVXbMMy2roB9myp
Chile Top 50
https://open.spotify.com/playlist/37i9dQZEVXbL0GavIqMTeb
Colombia Top 50
https://open.spotify.com/playlist/37i9dQZEVXbOa2lmxNORXQ
Uruguay Top 50
https://open.spotify.com/playlist/37i9dQZEVXbMJJi3wgRbAy
Japan Top 50
https://open.spotify.com/playlist/37i9dQZEVXbKXQ4mDTEBXq
Hong Kong Top 50
https://open.spotify.com/playlist/37i9dQZEVXbLwpL8TjsxOG
Philippines Top 50
https://open.spotify.com/playlist/37i9dQZEVXbNBz9cRCSFkY
Vietnam Top 50
https://open.spotify.com/playlist/37i9dQZEVXbLdGSmz6xilI
Thailand Top 50
https://open.spotify.com/playlist/37i9dQZEVXbMnz8KIWsvf9

Genre Dataset Playlists
Dancehall Official
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DXan38dNVDdl4?si=htOOokOET-CjhiDHk5TBbg
Jazz Classics

https://open.spotify.com/user/spotify/playlist/37i9dQZF1DXbITWG1ZJKYt?si=hqZWe0gTRzSdF3ebD-j8Dw
Hot Country
https://open.spotify.com/user/spotify/playlist/37i9dQZF1DX1lVhptIYRda?si=x7uzLxVfRw6GLkFDIvTZww
Rap Caviar
https://open.spotify.com/playlist/37i9dQZF1DX0XUsuxWHRQd
Classic Rock Drive
https://open.spotify.com/playlist/37i9dQZF1DXdOEFt9ZX0dh