Ben Psaila - psaila.b@husky.neu.edu DS 2000

Eric Weiss - weiss.eri@husky.neu.edu Professor Strange, Dr. Lee

Spotify Projections

Problem Statement and Background

Our group wished to explore the characteristics of the most popular songs both domestically and internationally. To observe these songs and the variability of their compositions, our group decided to utilize Spotify's playlists 'USA Top 50' and 'World Top 50.' Specifically, by utilizing these playlists, we wished to draw conclusions on the correlation between the most popular domestic songs and those internationally. We set out to answer how correlated popular songs in America were to their international counterparts, determining the weight of the American music opinion relative to the rest of the world.

The data we used came from CSV sources online and a Spotify API, which allowed us to access any playlist we wanted to gather information. The caveat with the API was we were limited to only pull 100 tracks from any given playlist, so we could only use the API for smaller playlists and not all of our data.

Introduction and Description of the Data

One of the main sets of data we used came from Spotify's Web API. We were able to set it up through their website using their app dashboard system. After we created the 'app', Spotify gave us our Client ID and Secret ID, then we set up our redirect url. Using Anaconda, we were able to make them environmental variables. With those 3 items set up as environment variables,

we were able to effectively use the API through Spotipy, which is a python based library for using Spotify's Web API. This was an extremely useful tool in both learning and using the API to extract data.

For the data we needed to extract each track off of a playlist and be able to pull a set of characteristics from each of these tracks. The set of characteristics looked like this: ['id', 'rank', 'popularity', 'danceability', 'energy', 'loudness', 'speechiness', 'acousticness', 'instrumentalness', 'liveness', 'valence', 'tempo', 'duration_ms']. These characteristics from each song allowed us to compare the values against one another and eventually come to conclusions about what elements are more prominent in the most popular songs.

As for why analyzing this song data is important, a look into the lucrative music industry can help find our answer. We wanted to use Spotify's characteristics of songs to create a projection on how popular any given song would be, simply using its characteristics as inputs. This will be an interesting look into the music business as a whole as they try to create songs that fit specific schemes/metrics to do well on these streaming charts. This could be used as a tool by those in the music industry, which globally brings in $19 billion USD, to help find what aspects of a song generally make it do well and hit top charts.

Methods

We pulled our CSV data directly from the Spotify Organizer and used pandas to clean it up. The data from the Spotify APIs were formatted as JSON-formatted data, which at first was confusing to sift through, however when separated using the correct keys, became more clear. For the playlist_tracks function, data came out as a tremendous mess of dictionaries, when the

'items' key was separated, however, it was clear that there was simply a list of songs with many characteristics attached. We were able to use a for loop to extract important information, like the 'id', 'popularity', and 'rank' of a song in the list. We did this for each song, and then used the 'id' and input that back into the Spotify API using an audio_features function from Spotipy to find out characteristics about each song. We put each song in a list with all of its characteristics, and had each of those lists within a larger list for the entire playlist.

```
for i in range(len(world['items'])):
    lst = []
    lst.append(world['items'][i]['track']['id'])
    lst.append(i+1)
    lst.append(world['items'][i]['track']['popularity'])
```

After cleaning the data to obtain a list with the id's, popularity, and characteristics of each song in any given playlist, we turned that into a data frame through pandas to be able to do calculations and statistics with the data.

	id	rank	popularity	danceability	energy	loudness	\
0	0VjIjW4GlUZAMYd2vXMi3b	1	88	0.514	0.730	-5.934	
1	3PfIrDoz19wz7qK7tYeu62	2	80	0.794	0.793	-4.521	
2	24Yi9hE78yPEbZ4kxyoXAI	3	94	0.770	0.724	-5.484	

In terms of visualization, we employed pandas and matplotlibs. Seen below, the visualization of USA rankings against world rankings was hard to interpret. In seeing this, we added an extra visual aid of a y=x line to better represent the correlation of the data.

To better work with the USA and world data sets, we used pandas to merge them.

```
import matplotlib.lines as mlines
import matplotlib.transforms as mtransforms

fig, ax = plt.subplots()
line = mlines.Line2D([0, 1], [0, 1], color='red')
transform = ax.transAxes
line.set_transform(transform)
ax.add_line(line)
```

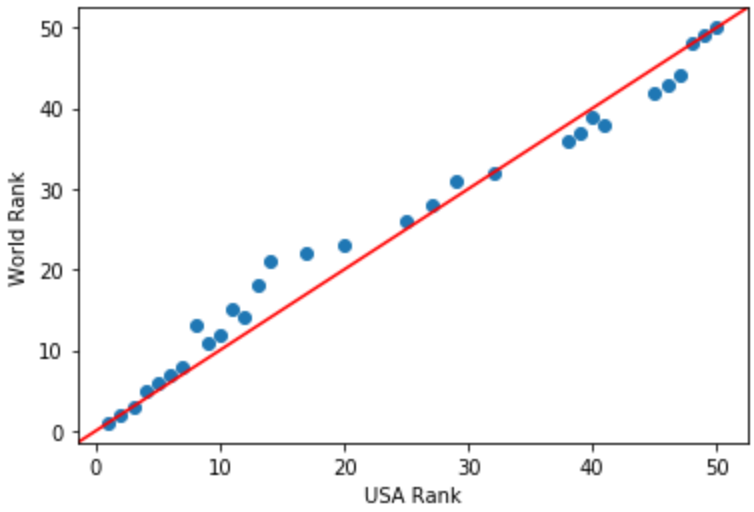
```
merged = pd.merge(usa_data, world_data, on='title')
merged = merged.drop(merged.iloc[:,2:14], axis=1)
```

From there, we were better able to run our projection testing using sklearn's modeling tools.

```
from sklearn.model_selection import train_test_split
X_train, X_test, y_train, y_test = train_test_split(
                                   X, y, random_state=1, test_size=.5)
```

Results, Conclusions and Future Work

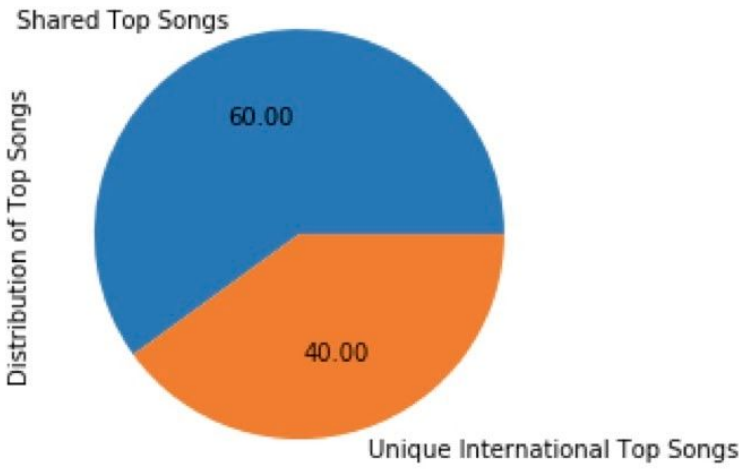
Our initial piece of visualization came in the form of a scatter plot. Employing the merged USA and world data sets, we plotted world ranking against USA rankings of top songs. The plotted songs are only those that appear in both 'World Top 50' and 'USA Top 50'. Below is the visualization.



The line appears through the y=x axis to denote where songs are ranked the same between the USA and the world rankings. Songs that fall above this line do so only in top rankings. From this graph, we can see the international influence of American taste on music. These tracks are ranked higher, and therefore more popular, in the U.S., meaning that American

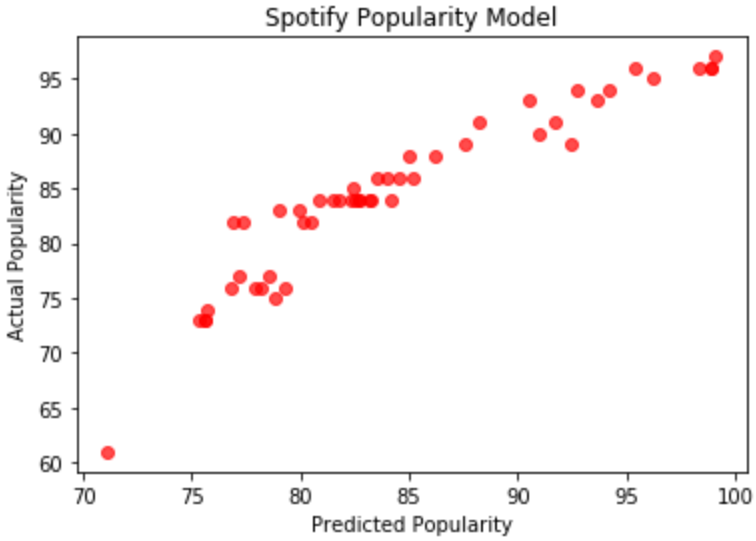
music preference drives international taste. This is extremely interesting, as according to Spotify only 29% of monthly active users reside in North America, with an undisclosed percentage living in America. Conversely, we also see tracks lower in the top rankings below the y=x line. While a smaller deviation than songs at the top of the list, this data could also support the previous point. Songs more popular internationally are unable to break the top of the chart as American listeners are not as exposed to them.

Next, we looked at the distribution of top songs, specifically how many were uniquely on the 'World Top 50' rather than on both charts. Below is the visualization.



We found here that a majority of the 'Top 50' playlists were shared, with only 40% of songs appearing in 'World Top 50' being unique to that playlist. Once again, the powerhouse of American taste driving the international music industry is supported.

Lasly, we wanted to be able to predict a song's popularity in the USA. To train our model, we used 'USA Top 50' and tested the model with the songs found on 'World Top 50'. While this is not training the model based on every song choice and rather popular songs right



now internationally, we wanted to see how popular tracks would be in the USA that had shown promise internationally. Below is our visualization.

With a r^2 of .87 and RMSE of 2.69 (relative to a 1-99 popularity score), the model predicted each song's popularity quite well. Overall, we have found that American music taste has incredible weight in the international music industry. Going forward, we would hope to further develop our predictive model, expanding its usability across all songs and strengthening its predictive value.

A shortcoming of this project was the limited scope of the data we used to tackle our problem. Given more time, we would create larger datasets to use our API on. We would also be able to further determine better variables to measure popularity and strengthen the correlations we found during examination. The predictive model could also be improved with more time and stronger training data.