Cindy Khuu

IST 652 Scripting for Data Analysis

Final Project

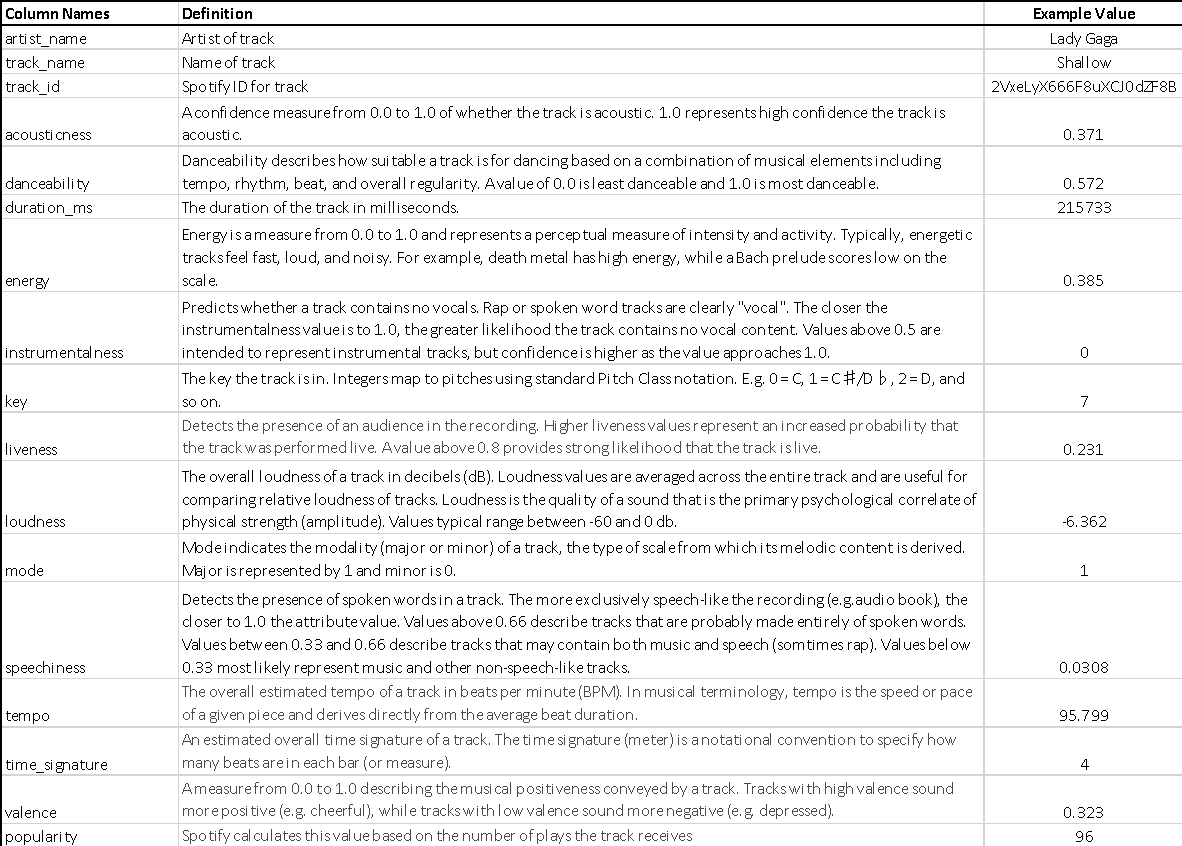
6/6/19

**Spotify Tracks’ Popularity & Audio Features Analysis**

**Data**

1. **Spotify Audio Features Dataset**

**Description:** The main dataset is called “Spotify Audio Features” from Kaggle, which includes about 130,663 tracks with audio features collected from Spotify Web API in April 2019. Each row represents one type of unit, which is a track with track name, artist name, its audio features, and popularity score. The track can range from actual songs, live music recording, to podcasts. Below is a data definition table for further explanation of fields in the dataset.



**Data Processing**: The following steps were executed to get the data cleansed into a workable pandas dataframe:

* Read data into pandas dataframe and renamed column name fields
* Replaced any N/A or missing values in numeric fields with mean value for that field using a defined user function
* Removed track\_id since it was not relevant or needed
* Applied lower case string function to column fields artist\_name and track\_name
* Added new column called “popularity\_attribute” by bucketing each song into a popularity type (popular, average, less\_popular) based on popularity score.
  + Popular = popularity score >= 50, which is 13.1% of the dataset or 1751 tracks
  + Average = popularity score >=25 and popularity score <50, which is 31.5% of the dataset or 41,130 tracks
  + Less\_Popular = popularity score < 25, which is 55.4% of the dataset or 72,382 tracks

**Data Source Link:** <https://www.kaggle.com/tomigelo/spotify-audio-features>

1. **Billboard’s Top Hot 100 Charts**

**Description:** As a supplemental dataset, I web scraped Billboard’s website for their Top Hottest 100 Music Chart, which lists their high-ranking songs based on interactions with the song from track downloads, radio plays, and online streams. This will be compared with top 100 popular songs from the Spotify dataset to understand how close they match in order to validate the Spotify popularity score. I chose to scrape this chart from April 13, 2019 to keep it as close to the time when the Spotify data was posted. There were 100 records that were scraped. Each row represents one type of unit, which is a song with the artist name and rank on the chart. Rank of 1 represents the top song on the charts while rank of 100 represents bottom of the 100 music chart.

**Data Processing:** The following steps were executed process the semi-structured data into a working pandas dataframe:

* Web scraped html website using BeautifulSoup and using separate lists to capture artists name, track name, and rank on the charts, and then transformed the 3 separate lists into a single pandas dataframe
* Applied lower case string function to the artist name and track name in order to compare them with Spotify dataframe
* Used outer join function to determine how many of the Billboard tracks matched with the Spotify dataframe based on artist name and track name. Note: Some tracks had the same track name with different artists
* Manually sifted through the Billboard tracks that did not appear to match with any tracks from the Spotify dataframe and then used replace value function on Billboard Billboard dataframe to ensure these tracks matched the Spotify tracks it was intended to match including the artist name. Note: Billboard and Spotify have different manners of naming the tracks and artist name for the same referring track

**Data Source Link**: <https://www.billboard.com/charts/hot-100/2019-04-13>

**Analysis: Methods & Questions**

Purpose of this analysis is to understand how the audio features and popularity of the tracks in the Spotify dataset relate to each other and highlight common patterns discovered. With a focus on Spotify’s popularity, I will attempt to predict if a song will be popular or not based on the audio features associated in the Spotify dataset.

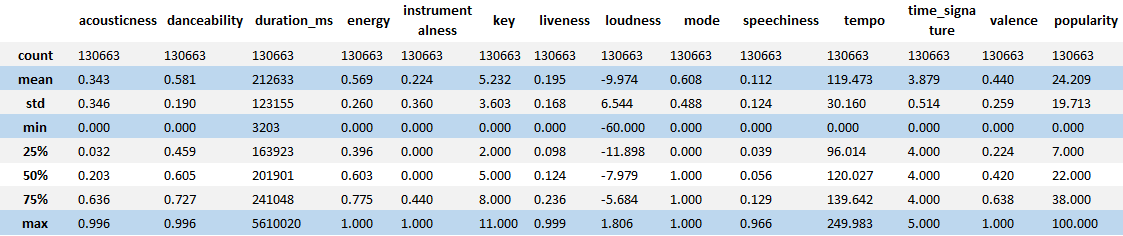
Overview of methods of analysis are:

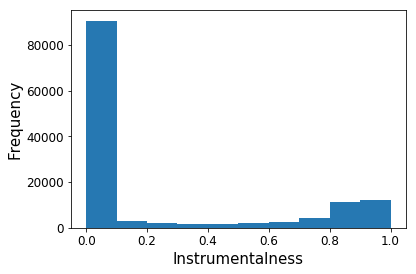
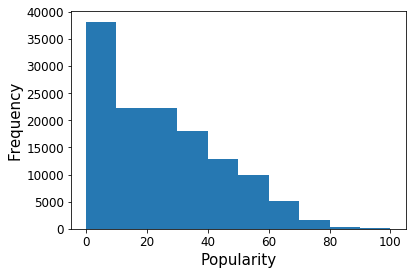
* Use of summary statistics (including calculated measures such as match rate and prediction accuracy) to compare field or unit of analysis
* Use histogram distributions to understand dataset
* Use of correlations to define relationships
* Use modeling algorithms to find best model with highest accuracy

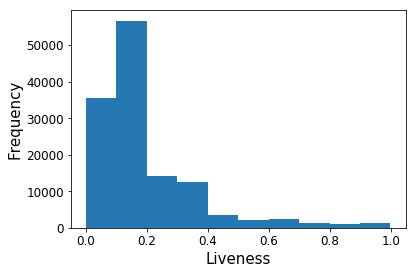
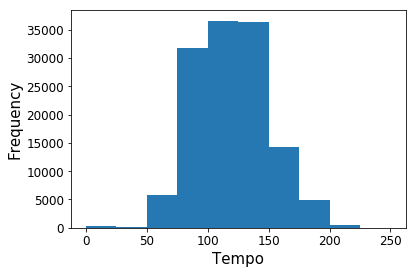
Note: Song attribute fields will refer to the popularity score and the 13 audio features (danceability, loudness, energy, valence, liveness, speechiness, time signature, duration\_ms, mode, key, acousticness, instrumentalness, and tempo) from the Spotify dataset when mentioned.

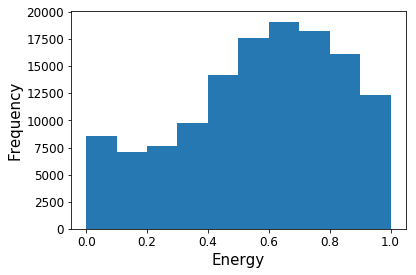
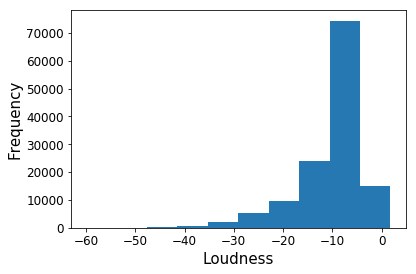
**Questions:**

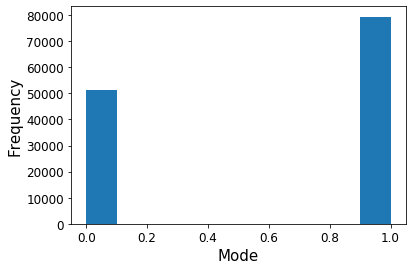
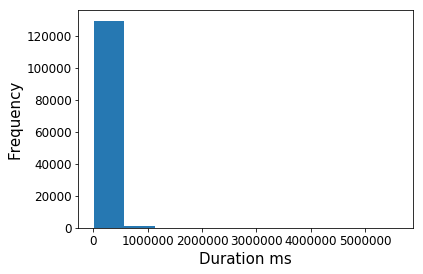
1. What’s the distribution and summary statistics of all the tracks for each audio feature and popularity score?
   1. Unit of Analysis: song attribute fields (audio features and popularity score)
   2. Method of Analysis: Used summary statistics using describe() function and distribution using histograms to understand mean and range of all the tracks for each audio feature and popularity score
   3. Output: One summary description output and 14 histogram charts showing distribution of the audio features and popularity score.

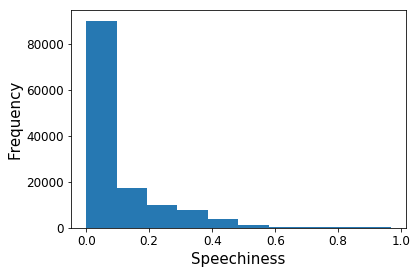
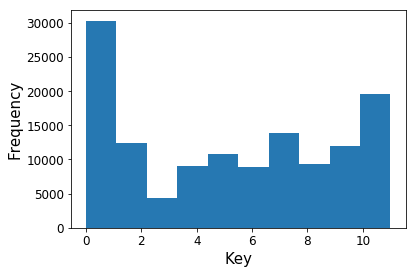


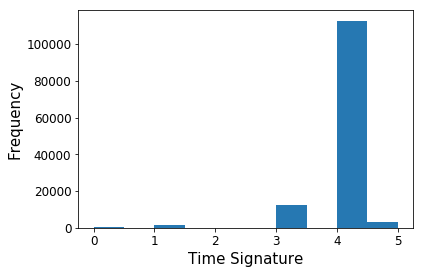
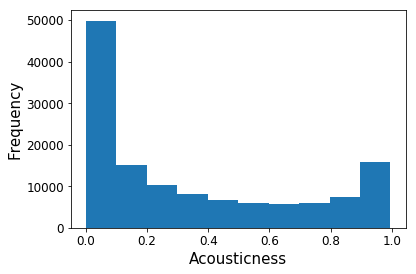


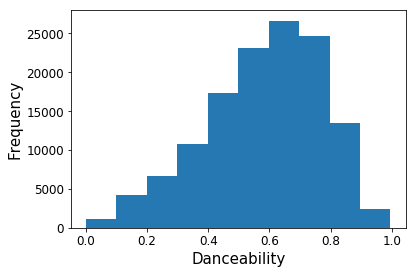
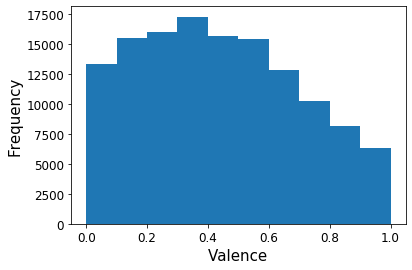








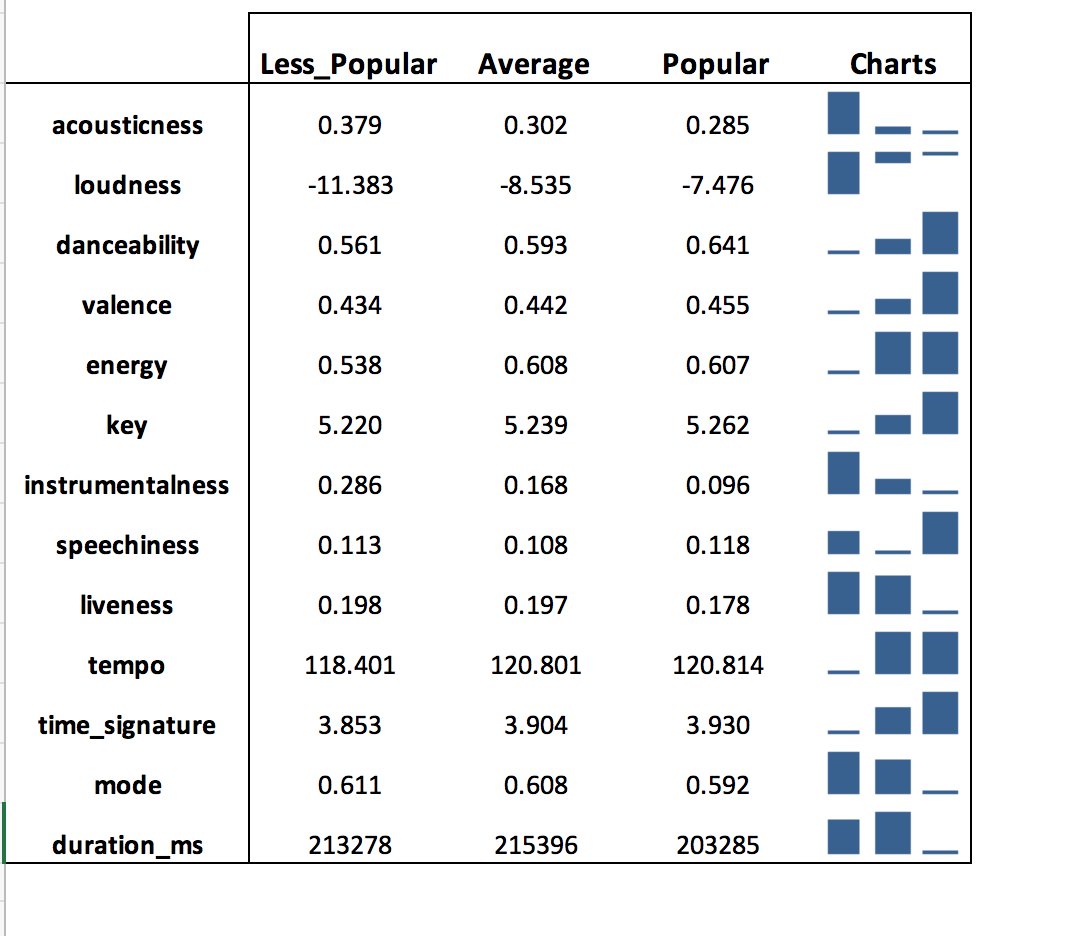
 

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  |  |  |  |  |  |  |  | **loudness** | **mode** | **speechiness** | **tempo** | **time\_signature** | **valence** | **popularity** |

1. Can we validate popularity score on Spotify dataset with external source? How many tracks on Spotify’s top 100 popular tracks match the Billboard’s Top Hottest 100 music charts?
   1. Unit of Analysis: tracks
   2. Method of Analysis: Calculate how many tracks on the Billboard Top Hot 100 Music Chart list matches the tracks on the top 100 popular Spotify tracks. Join the two dataframes to determine how many tracks match based on track name and artist name, keeping all records from the top 100 popular Spotify list. Summarize the match rate.
   3. Output: Write output file called “join\_top100DF.csv” with 100 rows showing the top 100 Spotify tracks with all of the associated audio features and popularity score plus header. There should be one additional column showing the Rank from the Billboard Music Chart showing a numerical value if the corresponding Spotify track matched the Billboard Music Chart. If the track didn’t match, then there would be a blank value under this column for it. After manually calculating the number of tracks in the Spotify list that matched the Billboard list, there was a 51% match rate.
2. For each audio feature and popularity score of all tracks in the Spotify dataset, what’s the relationship amongst each field using correlations? Which fields have strong correlations?
   1. Unit of Analysis: song attribute fields (audio features and popularity score)
   2. Method of Analysis: Use correlations to understand and compare relationships of each audio feature and popularity score
   3. Output: A correlation heatmap showing correlation values between 14 song attribute fields (audio features and popularity score).



1. For each defined popularity type, what is the average value of each audio feature of all the tracks within each popularity type? How does each popularity type compare to each other?
   1. Unit of Analysis: popularity types
   2. Method of Analysis: For each popularity type, compute average value of each audio feature of all the songs for each popularity type using the mean () function, and compare results for each popularity type.
   3. Output: Output file “spotify\_questionpoptype.csv: with 3 rows representing the popularity types and 13 columns representing the audio features and popularity type names plus a header; below is dataset transposed with sparkline charts added to each audio feature.



1. What type of classification algorithm can we use to find the highest prediction accuracy to predict popularity type using all the audio features as input variables?
   1. Unit of Analysis: models classifying popularity type
   2. Method of Analysis: Split data into 80% training data and 20% testing data. Use K-Nearest Neighbors, Multi Logistic Regression, Support Vector Machine, Random Forest, and Neural Network on the training dataset using popularity type as the Y variable and the 13 audio features as the X variables; then calculate the accuracy rate of the models using the testing dataset. Compare the accuracy rate of the models to find the best model based on the highest accuracy rate.
   3. Output: Accuracy rate of the five different types of models

|  |  |
| --- | --- |
| **Models** | **Accuracy** |
| K-Nearest Neighbors | 50.43% |
| Multi Logistic Regression | 55.69% |
| Support Vector Machine | 55.99% |
| Random Forest | 56.37% |
| Neural Network | 55.42% |

**Program Description**

This program imports Spotify dataset and creates a pandas dataframe called spotDF using additional data processing steps such as renaming column fields, replacing missing values with the mean, and conforming string data type fields for artist\_name and track\_name using lowercase string function. Then, summary statistics output, histogram charts, and correlation matrix heatmap are created for the spotDF dataframe. Next, it buckets the tracks into different popularity types: popular, average, and less\_popular based on the popularity score by adding a new column to spotDF dataframe using a user-defined function. Then, it summarizes all the audio features for all the tracks in the dataframe by popularity type using describe () function.

Next, it retrieves from the html link to get the Billboard Top Hot 100 Music Chart, by searching for the defined tags and retrieves the artist name, track name, and rank on the charts into three separate lists, which is then combined into a single pandas dataframe called billboarddf. Data cleansing occurs by attempting to conform artist name and track name to those in the Spotify dataset, using lower case string function and manually replacing values. Then a new dataframe is created for top 100 popular Spotify tracks, where it is joined with the clean billboarddf, keeping all records of the top 100 popular spotify songs, and then writes output file.

Finally, the program runs through five classification models using the y variable as the popularity attribute type and the x variables as the audio features and prints out the accuracy for each model.

**Observations**

* Popularity is skewed to the right with a smaller number of tracks with high scores
* Tracks in the dataset are mostly recorded music and not live recording, audio books, or speeches
* 51% match rate for the top 100 Spotify songs with Billboard Top Hottest 100 Music Chart is good considering total number of tracks that had to be scored in Spotify, which is over 130K songs
* For popularity score, there appears to be weak positive correlations with danceability and loudness while weak negative correlations with instrumentalness and acousticness. Although these correlations are between 12%-24% in absolute value, these might be considered moderate for this dataset and industry. Overall, these mentioned are the strongest from the other audio features.
* There are strong relationships amongst audio features such as: acousticness has strong negative correlations with loudness and energy audio features. Loudness has strong positive correlation to energy and danceability.
* Comparing mean values of audio features by popularity type, there are a few more distinct audio features that show a pattern than others where the less popular popularity type has lowest mean for the audio feature, average popularity type as middle value of mean, and popular popularity type has the highest mean or vice versa. Some distinct audio features are: Danceability – popular popularity type has higher value than others, Instrumentalness – less\_popular popularity type have higher value than others, Loudness - u less\_popular popularity type have less than others, Acousticness- less\_popular popularity type have less than others
* Best classification model was Random Forest with prediction accuracy of 56.37% out of the five different classification algorithms, but relatively all were within 10 percentage points from each other.

**Conclusions**

Spotify popularity distribution is skewed with more tracks with lower scores, making popular tracks very rare. Looking at relationships between audio features and popularity score, the audio features have strong relationships amongst each other while weaker relationships between popularity score and audio features based on typical correlation value.

Comparing audio features with strongest correlations to Spotify popularity score and those that appeared with more distinct with popularity type, there are 4 features determined to have a pattern and stronger relationship with popularity than the others: Loudness (Positive), Danceability (Positive), Acousticness (Negative) , Instrumentalness(Negative). Although we identified some of the more distinct audio features, the other audio features are still relevant in the classification model since we have also identified other relationships amongst themselves.

Based on the analysis, I was able to use all the audio features to predict the popularity type using five types of classification algorithms, where Random Forest had the best prediction accuracy of 56.37%, which is not that great of a prediction accuracy given that I can randomly guess a track will be “less\_popular” and have a 55.40% chance I will be accurate, so I was only able to predict 0.97% better than always guessing “less\_popular”.

Overall, it is possible to try to predict if a track will be popular based on audio features but there are various ways audio features can be composed to make a track which don’t always translate to more plays which is they key driver to popularity score algorithm used by Spotify.