Do Popular Songs Endure?

Final Report

Prasoon Rai , Sushant Ojal
110921754 , 110944445
{prrai,sojal}@cs.stonybrook.edu

1 Introduction

The long time endurance of a song is a challenging problem to solve. The aim of this project is to determine if there is a way to predict the long term endurance of a song, and to predict based on a set of recent observations if a song will indeed prevail for a significant span of time. Although its intuitively difficult to know if a song will stay, we have many indicators of what's most likely to stick namely and most notably, enormously popular music that's meant something to somebody. There are other factors such as those of songs which have heralded a huge shift in popular taste, along with the artists who produced them.

In the midterm report, we outlined a baseline linear regression model that was improved upon using Ridge Regression to predict the endurance score, which acts as an indicator of the current popularity of a song. We accumulated more than 25 features for over 25000 songs, analyzed the accuracy of the baseline model, and finally discussed other areas such as demographics and release date distance to charts to be explored towards an all round depiction of a song's characterization.

In the final phase of the project, we discuss over advanced model, that considerably improves our prediction accuracy. We describe an additional set of features being utilized, a detailed analysis of the endurance score formulation and most importantly, try to answer the question: **What makes an initially unpopular song, popular today? (and vice versa)**. The repository of our project can be found at: `https://github.com/prrai/Data_Science_Do_Popular_Songs_Endure`.

2 Data Collection

Till the progress report, we had scraped the following websites to obtain our feature set.

1. **Universal Music Database(UMD):** We had used this website to get weekly Billboard top 100 songs and other important features.

2. **Spotify:** We had used *Spotipy*,to obtain features which analyse the track, as well as information on the popularity of the track.

3. **Youtube:** We used YouTube's python API to query a song for its total viewcount as an estimate of its popularity.

4. **Billboard.com:** We had used this website to obtain the top 200 albums and artists per week.

5. **What-song.com:** We had scraped data of more than 25000 unique artists to get the exact counts of every song that was featured in a movie or a TV show.

6. **Awardsdatabase.oscars.org:** We used this site to get the list of every song that has ever received an Oscar award in the best song category.

7. **Grammy.com:** We were able to obtain every artist that has received a Grammy lifetime achievement award. We also scraped out the number of nominations and wins across all categories for every artist that features in over song database.

In addition to the above sources, we used the following sources for newer features:

1. **Wikipedia:** We wrote a Wikipedia scraper to obtain the release date for each of the songs in our database.

2. **United States Census Bureau:** We use the demographics data available by the United States Census Bureau to find out the percentage of population which is between the age group of 15 to 30 for the years between 1960 to 2013. This value is then appended as a feature in correspondence to when the song made it to the billboards. A more detailed explanation is provided in Section 3.

3. **Spotify:** We wrote a scraper for Spotify yet again to obtain the current popularity of an artist based on the number of streams of his/her songs.

3 Model Features

This section highlights the features we have used to build out training set. We had the following set of features during our project progress stage:

Title	Artist	Entry Date	Total Weeks
TopSongArtist	TopSongArtist10	TopSongArtist100	Peak Position
Total Weeks	Duration ms	Energy	Acousticness
Key	Liveness	Loudness	Tempo
Valence	Artist_grammy_nominations	Movies_TV_feature	Oscars_won
Artist_lifetime_grammy	Artist_grammy_wins	Mode	Speechiness
Entry Date	Peak Position	Danceability	

We have described each of these features along with their sources in the previous reports. In addition to these we added the following set of features:

1. **Artist Popularity (Source: Spotify)** The popularity of the artist. The value will be between 0 and 100, with 100 being the most popular. The artist's popularity is calculated from the popularity of all the artist's tracks.

2. **Days_before_charting (Source: Wikipedia)** We extracted the release date of each of the songs in our database from Wikipedia. This feature is the value obtained from the difference (in number of days) from the time the song was release to the time the song first entered the Billboard Top 100 charts. This feature is an indication of the *Late Bloomers*, and should be important in deciding if a song will remain popular over time.

3. **Age_Percentage_15_30 (Source: United States Census Bureau)**: People have a tendency to latch on to the music of their youth, and a higher percentage value of young adults during a certain era might indicate that the song will endure. We obtained the percentage of people in the age group of 15 to 30 years during all the years from 1960 to 2013, and added this as a feature corresponding to the year the song made it to the charts.

We therefore ended up with a total of 30 features for songs which have been popular from 1960 to 2013. As mentioned in the previous reports, we have removed the songs which were released after 2013, since there is an increase in both the YouTube view-count as well as number of streams on Spotify simply by the virtue of them being recent. After this, we were left with 20228 distinct songs in our database.

4 Estimation of Endurance Score

To estimate the present day popularity of any given song, we calculate an *Endurance Score* of a song. For this, we have taken into consideration, the following two features:

1. **Popularity on Spotify**: The popularity of the track. The value will be between 0 and 100, with 100 being the most popular. This value is determined by Spotify, and is calculated by algorithm and is based, in the most part, on the total number of plays the track has had and how recent those plays are.

2. **YouTube Viewcount**: The number of times a song has been viewed on YouTube. This value is the average of the first two most viewed videos of the song.

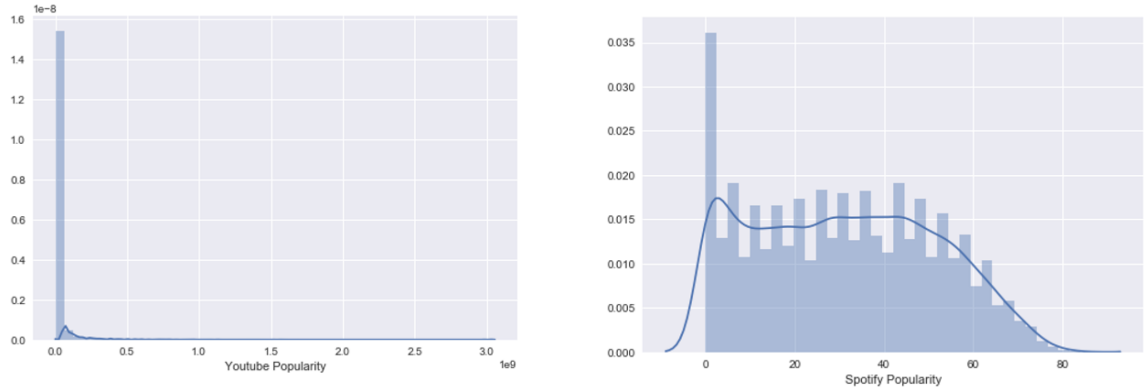


Figure 1: The distribution of the popularity values on Spotify and Youtube.

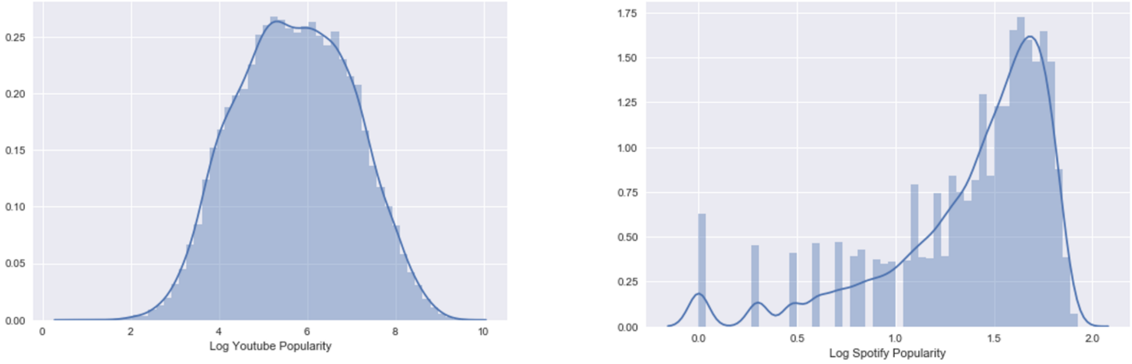


Figure 2: The log of the distribution of the popularity values on Spotify and Youtube.

Taking the log of both the distributions of Spotify and Youtube, we observe that the resulting distributions resemble a normal distribution. (left skewed in case of Spotify). We then made sure that each of the popularity values lie in the same range (0 - 20). This was necessary to add them in a meaningful way. The distribution of the endurance score then, results in the form of a left-skewed distribution.

For future use, we assume songs which have an overall value of less than 0.6 as unpopular, between 0.6 and 0.8 as that of medium popularity and greater than 0.8 as highly popular.

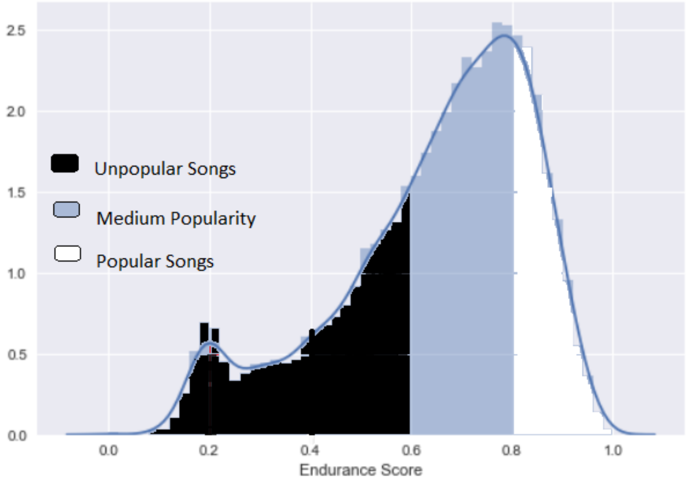


Figure 3: Distribution of the endurance score and the popularity categorization. Songs with values less than 0.6 are considered unpopular, between 0.6 and 0.8 as that of medium popularity and greater than 0.8 as highly popular.

5 Prediction Models

5.1 Baseline Models:

We approached the problem with two baseline models. The following section discusses them.

5.1.1 Data Processing:

Before applying the model, we cleaned up the data. This comprised of removing rows that have missing several entries, filling empty column entries with median values, normalization of data and filtering out songs after the year of 2013.

5.1.2 Linear Regression Model:

As a baseline, we implemented linear regression model.

- Linear regression works by solving a linear equation where coefficients are estimated from the training data with intention to minimize residual sum of squares between difference in the training and test data. In scikit-learn, the coef_ contains a list of all the coefficients, whose number would be equal to the total features considered during modeling.

- Like any regression model, linear regression seeks to fit the data along a curve, in this case a straight line. The performance is gaged by observing the root mean square error (lower is better) and the coefficient of determination (closer to 1, the better).

5.1.3 Ridge Regression model:

As an extension to the linear regression, we use ridge regression to further improve the accuracy of the model.

- Ridge regression introduces a trade off by allowing a bias in the coefficients but decreasing the variance of the parameters. So although, we know that the predictions are off by a little amount, the variance is less and overall the prediction is more reliable as compared to the OLS.

5.2 Advanced Prediction Model:

We worked on Random Forest Regressor, Decision tree Regressor and Gradient Boost Regressor (GBR). We finalized GBR as our advanced prediction model after empirically determining the tuning parameters for maximum prediction accuracy for the current task.

5.2.1 Gradient Boost Regressor (GBR):

- Gradient Boosting Regressors (GBR) are ensemble decision tree regressor models. Gradient boosting regressors are a type of inductively generated tree ensemble model, typically decision trees. At each step, a new tree is trained against the negative gradient of the loss function, which is analogous to the residual error.

- The advantages of GBR are:
  1. Ability to handle data of mixed types, allowing heterogeneous features.
  2. High predictive power.
  3. Robustness to outliers in output space through robust loss functions.
- Mathematical formulation:
  1. GBR uses the following additive model:

     $$F(x) = \sum_{n=1}^{N} \lambda_n h_n(x)$$

     Here, the $h_n(x)$ is the basis function, also known as a weak learner.
  2. GBR builds the additive model in a forward stage-wise fashion:

     $$F_n(x) = F_{n-1}(x) + \lambda_n h_n(x)$$

  3. At each stage, the decision tree $h_n(x)$ is chosen to minimize the loss function L given the current model F_{m-1} and its fit $F_{m-1}(x_i)$

5.2.2 Model Parameters:

We used the following parameters in our model:

1. **n_estimators:** 760
2. **max_depth:** 4
3. **min_samples_split:** 3
4. **learning_rate:** 0.026
5. **loss:** huber

5.3 Model evaluation

We observed that our models perform reasonably well at the baseline. We use r2-score and root mean square error as metrics for estimating the accuracy. The results are as below. The ridge regression slightly improves the model's accuracy. Finally, the advanced model considerably improves the accuracy. We discuss more on interesting findings and evaluations for the advanced model in the subsequent section.

Model	**R2 score**	**RMSE**
Linear Regression	0.5245	0.1542
Ridge Regression	0.525	0.1541
Gradient Boosting Regression	0.6589	0.1121

6 Results

In this section, we provide detailed analysis on the performance of the advanced GBR model. We show that our model, works well in all possible scenarios that a song's popularity can offer. Following cases are possible:

1. The song was popular when it was released and is popular currently.
2. The song was not popular when it was released but is popular now.
3. The song was popular when it was released but is not popular now.
4. The song was not popular when it was released and is not popular now.

Out of the above 4 cases, we discuss on 2 and 3, which are especially interesting.

1. For case 2 above:
    - To filter out candidates that are popular now, we apply the threshold criteria: endurance score should be more than 0.80
    - We filter in 2 ways to get songs that were popular when they released:
        (a) The peak position in Billboard Top 100 for the song was between 75 and 100.
        (b) The time it took for the song to enter Billboard Top 100 from its release date was more than 45 days.
    - In this way, we are able to look at the songs that fall in the case 3 bucket. We perform the filtering on the predicted data. This enables us:
        (a) To compare the difference between predicted and expected values for the endurance score, thereby allowing us to gauge the efficiency of the model.
        (b) To analyze the features that characterize the songs, thereby enabling us to answer: **Which songs were not popular in their release time, but are now, and more importantly, why?**
2. For case 3 above:
    - To filter out candidates that are not popular now, we apply the threshold criteria: endurance score should be less than 0.65
    - We filter in 2 ways to get songs that were popular when they released:
        (a) The peak position in Billboard Top 100 for the song was between 1 and 25.
        (b) The time it took for the song to enter Billboard Top 100 from its release date was less than 3 weeks.
    - In this way, we are able to look at the songs that fall in the case 2 bucket. We perform the filtering on the predicted data, which enables us:
        (a) To compare the difference between predicted and expected values for the endurance score, thereby allowing us to gauge the efficiency of the model.
        (b) To analyze the features that characterize the songs, thereby enabling us to answer: **Which songs were popular in their release time, but are not now, and more importantly, why?**

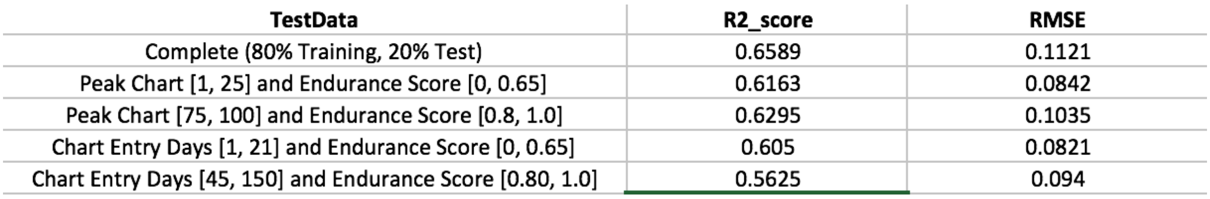
6.1 Model Performance

We tested the model for:

1. A train-test split of 80% - 20% on the entire data.

2. Cases 2 and 3 as described above, each filtering data as per two criteria mentioned above.

6.1.1 Performance metrics:



TestData	R2_score	RMSE
Complete (80% Training, 20% Test)	0.6589	0.1121
Peak Chart [1, 25] and Endurance Score [0, 0.65]	0.6163	0.0842
Peak Chart [75, 100] and Endurance Score [0.8, 1.0]	0.6295	0.1035
Chart Entry Days [1, 21] and Endurance Score [0, 0.65]	0.605	0.0821
Chart Entry Days [45, 150] and Endurance Score [0.80, 1.0]	0.5625	0.094

Figure 4: R2_score and RMSE observed for various test data combinations.

We observe that the model performs with same accuracy even for case 2 and case 3, for both the filtering methods.

6.1.2 Performance scatter plots:

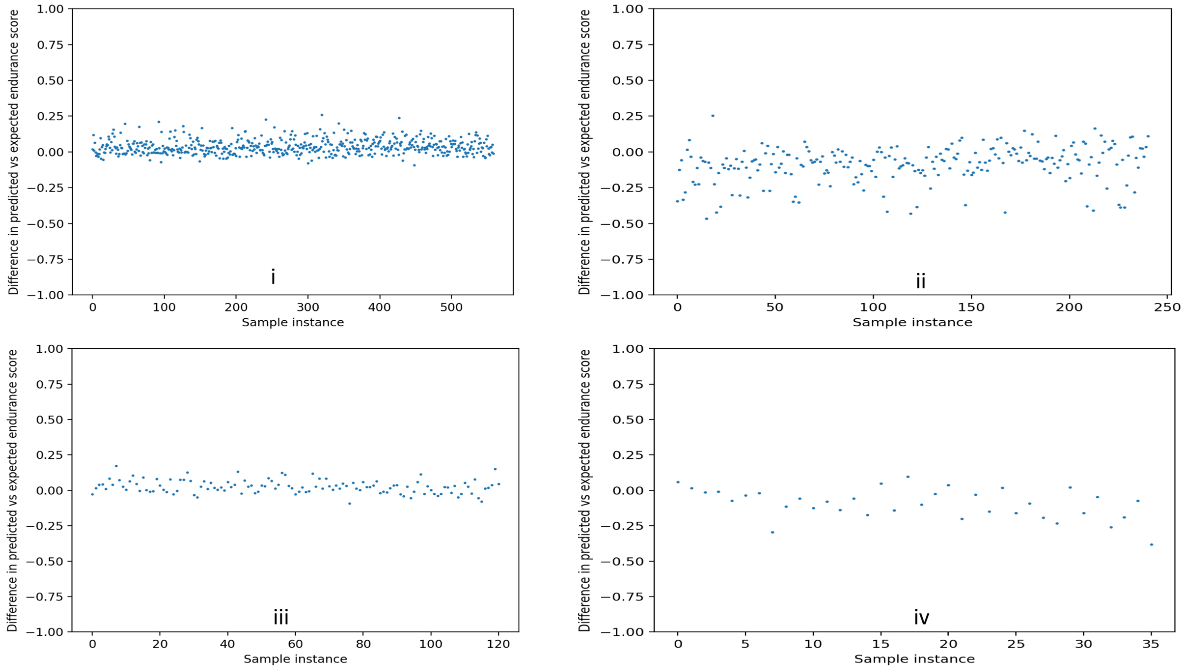


Figure 5: Scatter plots of differences in predicted and expected endurance scores

- The four scatter plots above capture the difference between predicted and expected endurance scores. The values clustering around 0 indicate that for a large number of test cases, the predicted and expected values are very close, indicating a high prediction efficiency.

- The four experiments above are as follows:

    1. 5.1 is for the case where the songs in the sample were unpopular when they were released (have peak chart position between 75 and 100), but are currently popular (have a high endurance score (≥ 0.80)).
    2. 5.2 is for the case where the songs in the sample were popular when they were released (have peak chart position between 1 and 25), but are currently unpopular (have a low endurance score (≤ 0.60)).
    3. 5.3 is for the case where the songs in the sample were unpopular when they were released (took between 45 and 150 days to enter billboard top 100 after their release date), but are currently popular (have a high endurance score (≥ 0.80)).
    4. 5.4 is for the case where the songs in the sample were popular when they were released (took between 1 and 21 days to enter billboard top 100 after their release date), but are currently unpopular (have a low endurance score (≤ 0.60)).

6.2 Feature Analysis

We try to analyze the features which cause the songs which were not popular when they were release to be popular now (and vice versa). We plotted the distributions of different features and found some interesting patterns which we will enumerate shortly. We consider four sets of features of each of the songs which we found to best highlight why a song behaves unexpectedly over the long run:

1. Artist popularity
2. Duration of Song
3. Audio Features of the Song (acousticness, danceability, energy and instrumentalness)
4. Audio Features of the Song (Speechiness, Liveness, Valence)

6.2.1 Case 2: Songs which were not popular when release but are popular now

This case (Figure 6 and Figure 7) considers songs which were unpopular at the time of release either due to the fact that they took a long time to make it to the charts or because they attained a poor peak position in the charts, but yet managed to become popular over the long run.

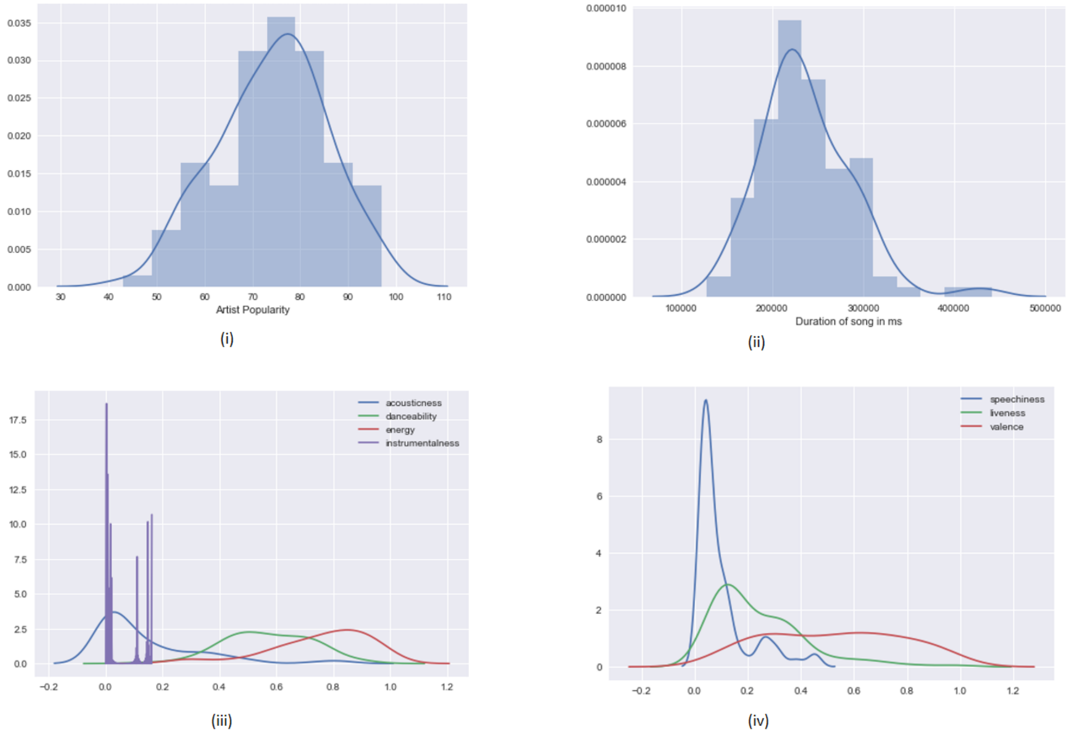


Figure 6: Distribution of features of songs which are popular now but were unpopular on release. (peak chart position between 75 and 100)

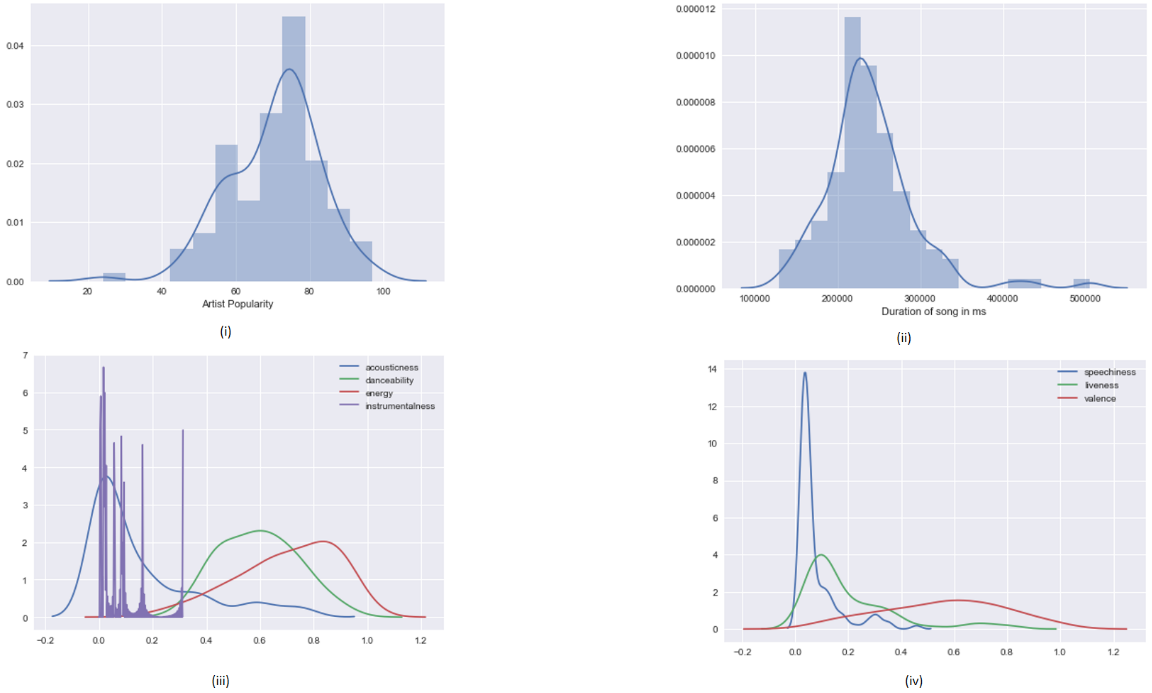


Figure 7: Distribution of features of songs which are popular now but were unpopular on release. (took between 45 to 100 days to enter the billboard top 100 after their release date).

The following is a description of each of the four feature sets:

1. Artist popularity: As is evident from the distributions, the artist popularity has a major role in deciding if a song will prevail, irrespective of its immediate performance on release. The distribution peaks around a popularity value of 75 - 80. Therefore, even if a song performs poorly on its release, if the artist is popular it is too soon to rule out its longevity.

2. Duration of the song: An interesting observation was that the distribution of duration of songs have a duration of about 200 to 250 seconds in both of the cases.

3. Acousticness, Danceability, Energy and Instrumentalness: The songs which prevail tend to have very low acousticness, high energy and low instrumentalness. In other words, vocal songs with high intensity prevail more.

4. Speechiness, Liveness, Valence: Speechiness indicates less musical notes, liveness indicates if a song was played live while valence describes the cheerfulness of a song. We observe that a song doesn't necessarily have to be euphoric to prevail. But it is indeed essential for it to be more musical and not performed in front of a live audience.

6.2.2 Case 3: Songs which were popular when released but are not popular now

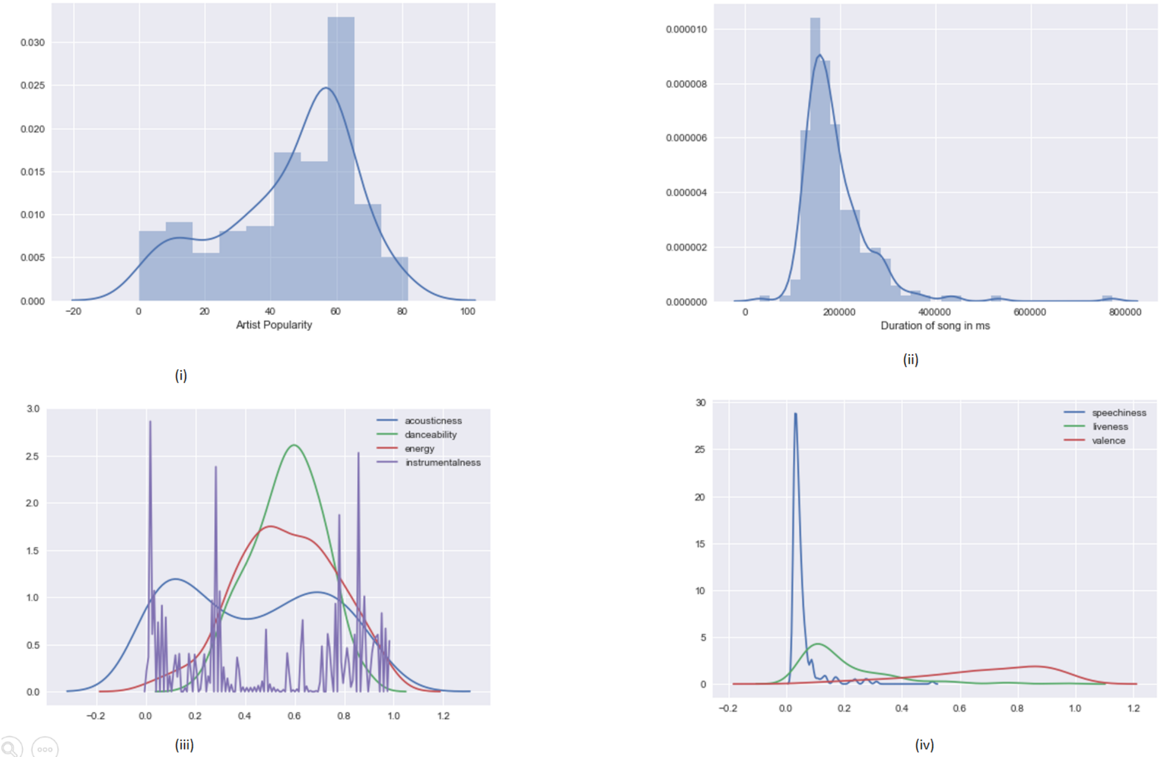
For this case (Figure 8 and Figure 9), along with describing the general trend in the above mentioned feature categories, we also highlight the differences which exist between songs of Case 2 and Case 3:

1. Artist popularity: We observe from the figures that an artist doesn't necessarily have to be very popular for his songs to top the charts immediately on release. The distribution of artist popularity for this category peaks around 60, which is significantly lower than the range of 75 - 80 we observed in the previous section.

2. Duration of the song: Interestingly, we observe that the duration of these songs is lower than the ones in the previous case. The distribution peaks at values around 175 seconds. So although a shorter song may have a fair chance of gaining popularity immediately on its release, its chances of prevailing over time are not so high.

3. Acousticness, Danceability, Energy and Instrumentalness: We observe that songs which top the charts immediately upon release but do not make it in the long run, have no definite distributions of their acousticness and instrumentalness. This is distinctly different from the previous case when both of these values were less. These songs have a lower energy/intensity as compared to the songs in the previous case. The danceability has a similar distribution in both the cases.

4. Speechiness, Liveness, Valence: All of these three features have similar distributions in all the cases. It can be said that a more musical song which was not sung in front of a live audience has a higher chance of gaining popularity either at the time of release or at a later date.



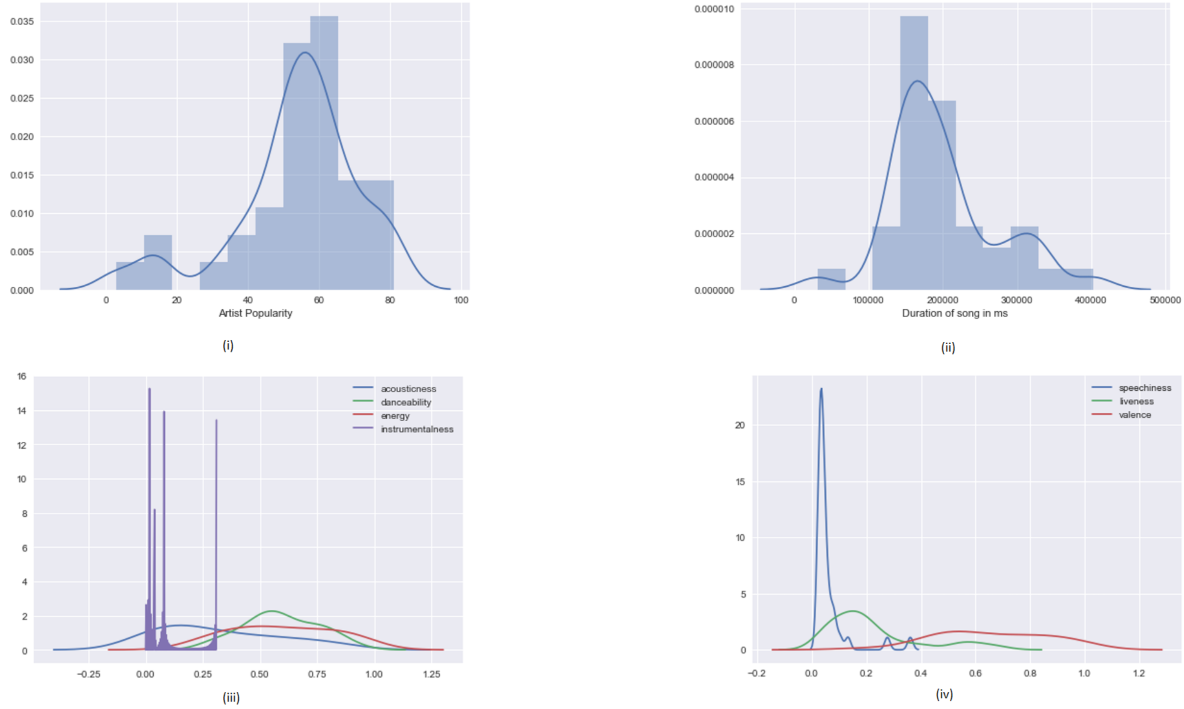
(i)

(ii)

(iii)

(iv)

Figure 8: Distribution of features of songs which were popular(peak position in charts between 1 and 25) when released but are currently unpopular.



(i)

(ii)

(iii)

(iv)

Figure 9: Distribution of features of songs which were popular(took between 1 and 21 days to enter the billboard top 100) when released but are currently unpopular.

7 Conclusion and future work:

In this project, we tried to quantify the endurance of a song in the current time frame through a scoring mechanism. We built a prediction model that reasonably predicts the endurance score for various popularity trends that a song could offer, based on its features encompassing its audio characteristics like loudness, tempo, dance-ability, etc, demographic information about the most active listeners, popularity of the song artist, awards and movie/TV references and more.

The model was able to achieve around 66% accuracy overall. We observe the potential of future work in this project that can further improve the prediction capabilities:

- Perform text analysis on the song lyrics to understand pop culture references, which could potentially make a song popular in the current time frame.
- Perform analysis on the genre of the songs by correlating the year of release and the popular genres in that time.

In conclusion, we would like to say that the long time endurance of a song is a challenging problem to solve, involving high uncertainty due to the dynamics of changes ranging from music tastes to politics and historical events to festivals.