What makes a song popular?

Image credits to Sergei Bezborodov from Pexels

By Andrew Nguyen, Anne Cuzeau, Samuel Flusche

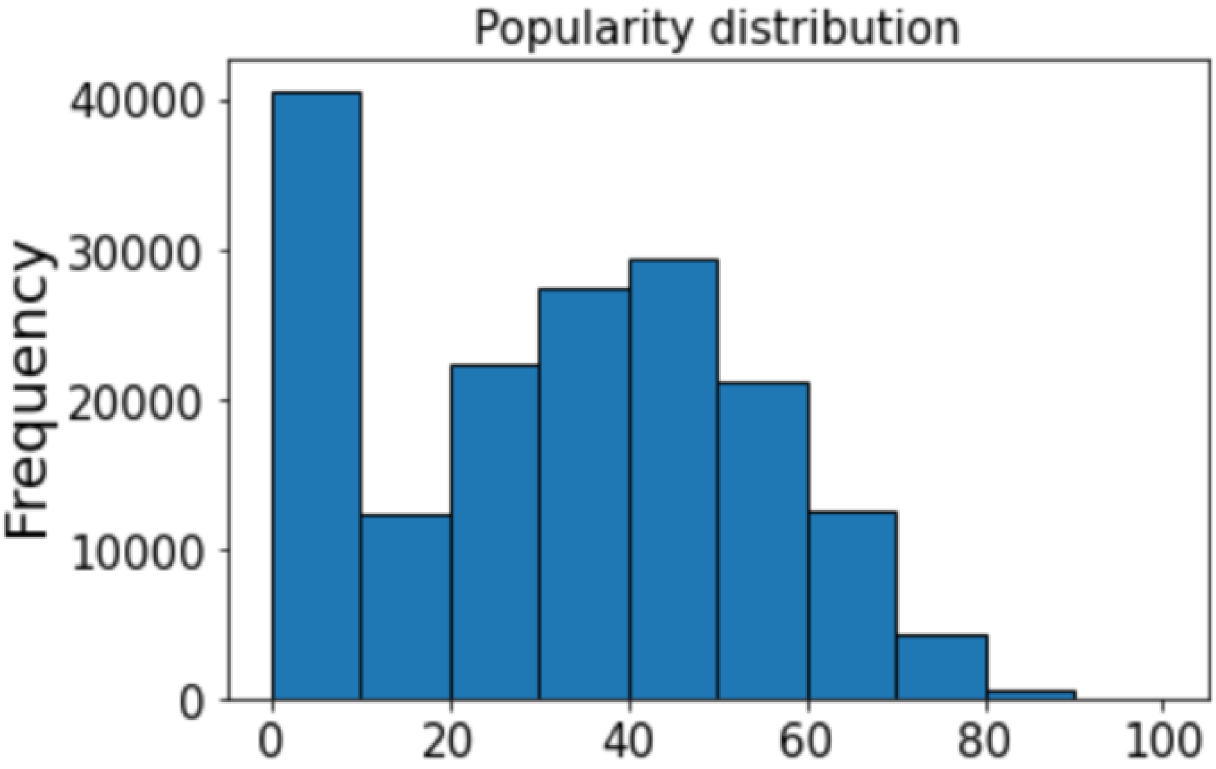
Popular songs don’t just pop out of nowhere, there are factors that cause songs to be popular. Using Spotify’s dataset, the main objective of our project was to get an understanding of which of these factors make a song popular, and use those to create a linear regression model to predict the popularity of any song.

A little bit about our data...

Our data didn't have any missing values, had 170k rows of data and included songs released anywhere between 1921 and 2021. Most columns in our dataset were descriptive statistics for continuous variables. For our analysis, we discarded columns due to inconsistency/redundancy as well as those showing no trends. The discarded columns were: Artist name, release date (which was inconsistent as a date time, sometimes having months/days or just years), tempo, speechiness, mode and key.

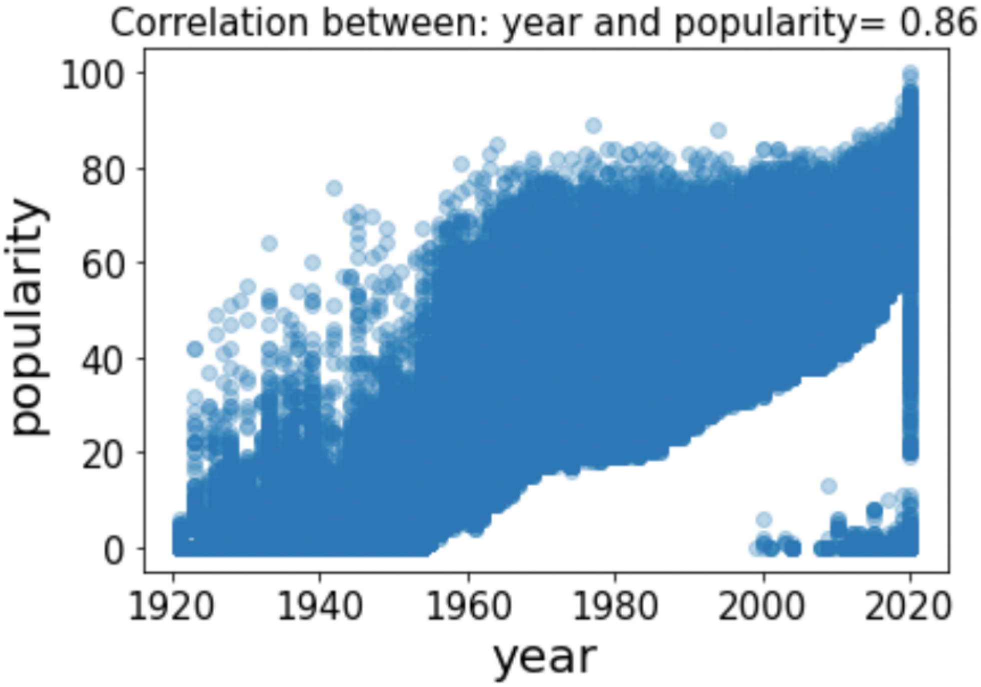
Initial findings

We initially found that many of the factors we were focusing on (such as loudness, energy, danceability, and popularity) showed an upward trend over time.

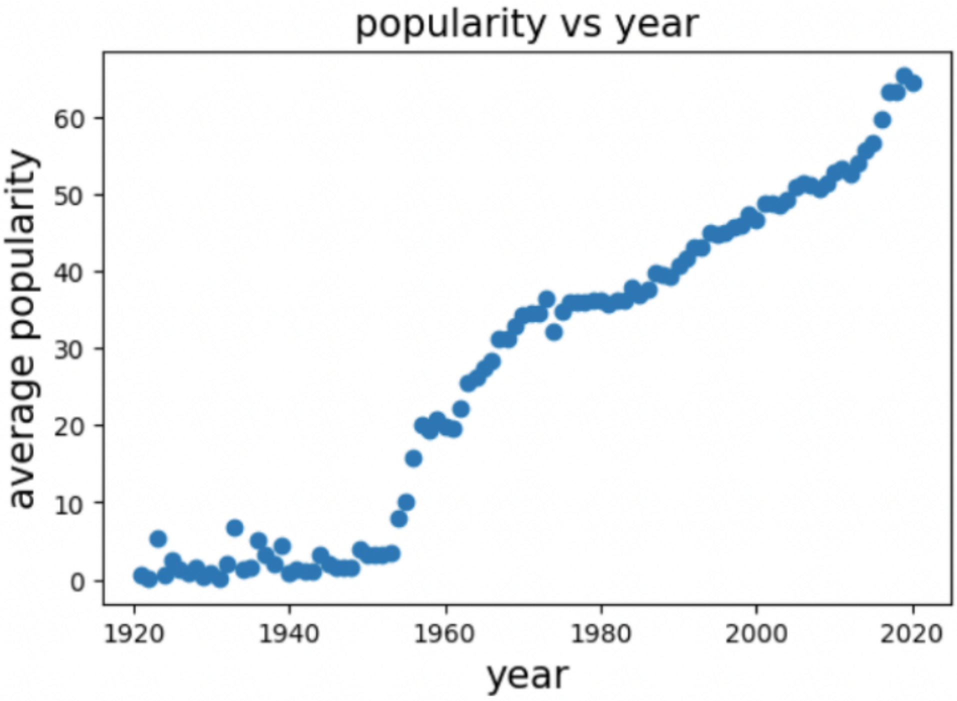


Another observation we made early on was that the distribution of popularity was not even: as we can see in the graph above, many songs (around 16%) have a popularity of zero, and popular songs (popularity of 80+) are very rare.

Popularity and release year: a strong but complex relationship

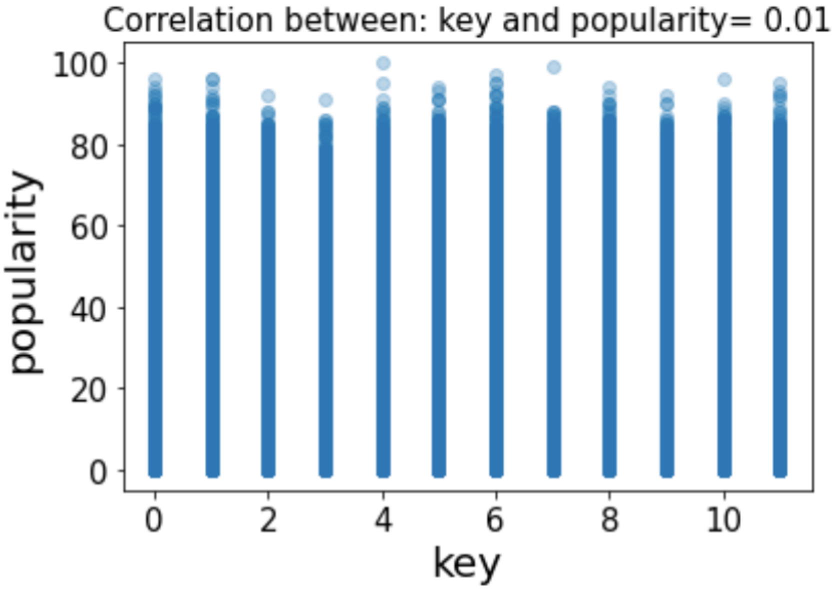


The most linear relationship we observed was between year and popularity. The graph below shows that these two variables have a strong correlation coefficient (0.86), which is close to a perfect linear model (coefficient of 1). This scatterplot also makes it obvious that most of the very popular songs occurred in recent years. We can infer that this phenomenon is most likely due to Spotify's audience: younger and more prone to like newer songs.

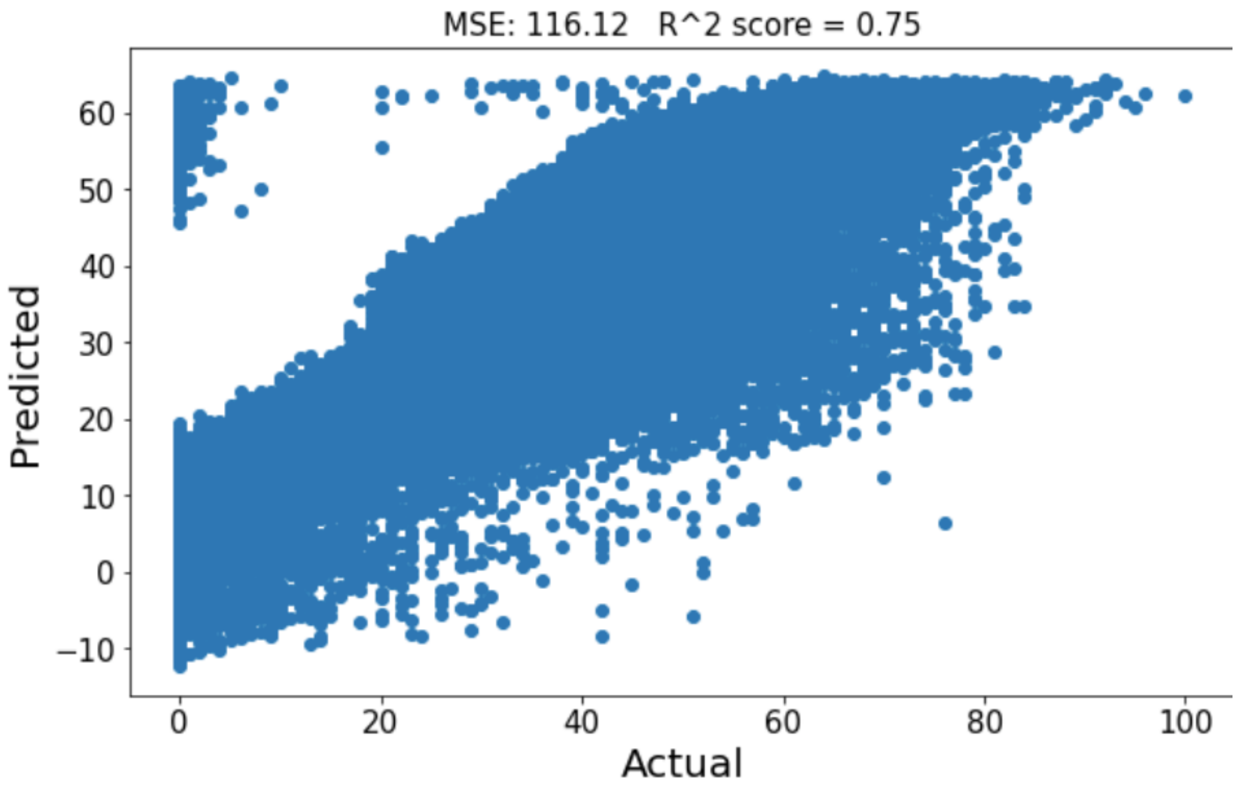


Using the average popularity for each year allowed us to clearly see a fast upward trend.

This triggered an important question for us: how can we ensure that our model will be able to predict older but popular songs?

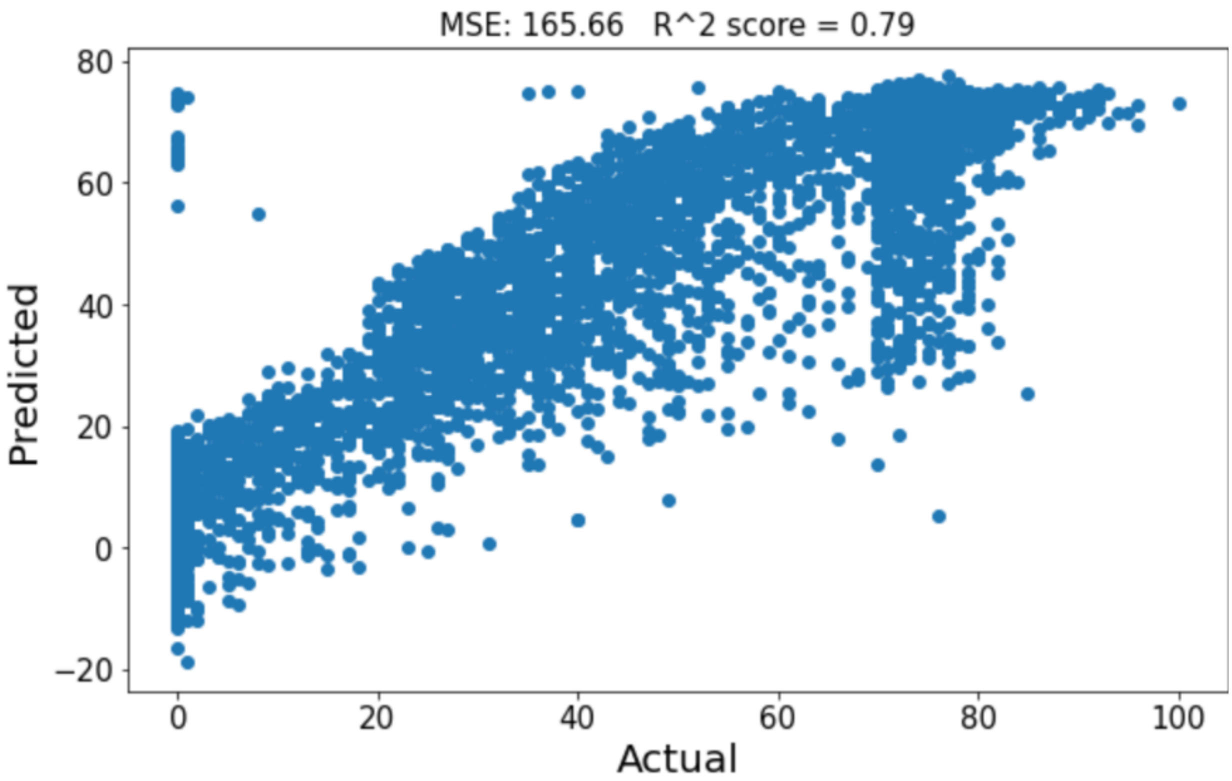


Excluding variables
As we stated earlier, we excluded variables from our model. Our reasoning was that these variables were not showing any leading trends and weren't very decisive with our linear regression models.



Our baseline model
Our initial (baseline) linear regression model included all the variables which weren’t previously dropped. We used a 70/30 train/test split. As expected, our model showed room for improvement: it was unable to predict any songs with a popularity above 60, and had a r-squared value of 0.75.

Improving our model: balancing
To improve our baseline model and get it to be able to predict popular songs, we started by balancing the data. To do this, we took the number of songs that had a popularity greater than or equal to 70, and then randomly selected the same amount of songs with a popularity between 70 and 30 and the same for 30-0.



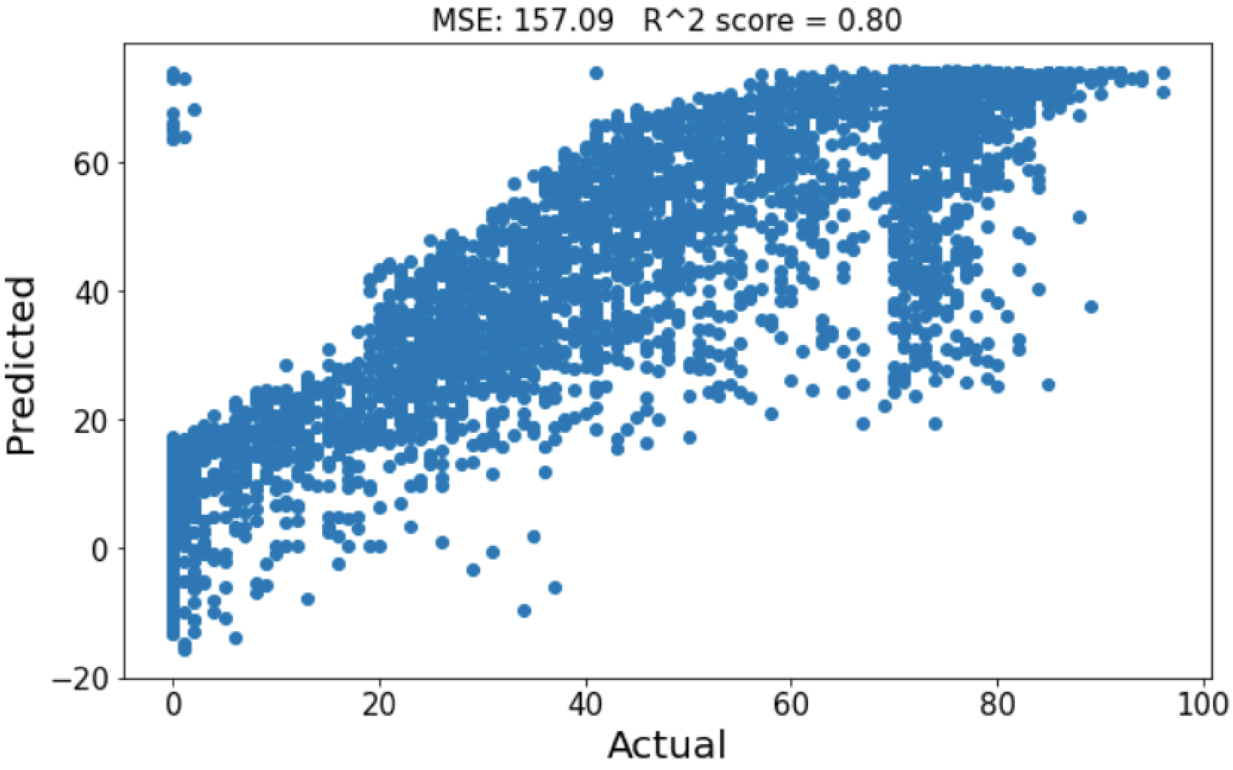
As we can see in this graph, the model relying on balanced data was able to predict popularity values up to 80 (as opposed to 60 for our baseline model). The R^2 value was better as well, however our mean squared error shot up. Looking at the graph, it is easy to see why: our model made ‘costly’ mistakes at 0 (predicting negative values) and at 70 (predicting values ranging from 20 to 80).

Improving our model: a closer look at certain variables
Next, we looked into acousticness. At first, it looked like popular songs had a lower acousticness. However, after further analysis, we saw that the relationship between popularity and acousticness was changing over time: older songs with lower acousticness were more likely to be popular, while newer songs with higher values showed a higher popularity.

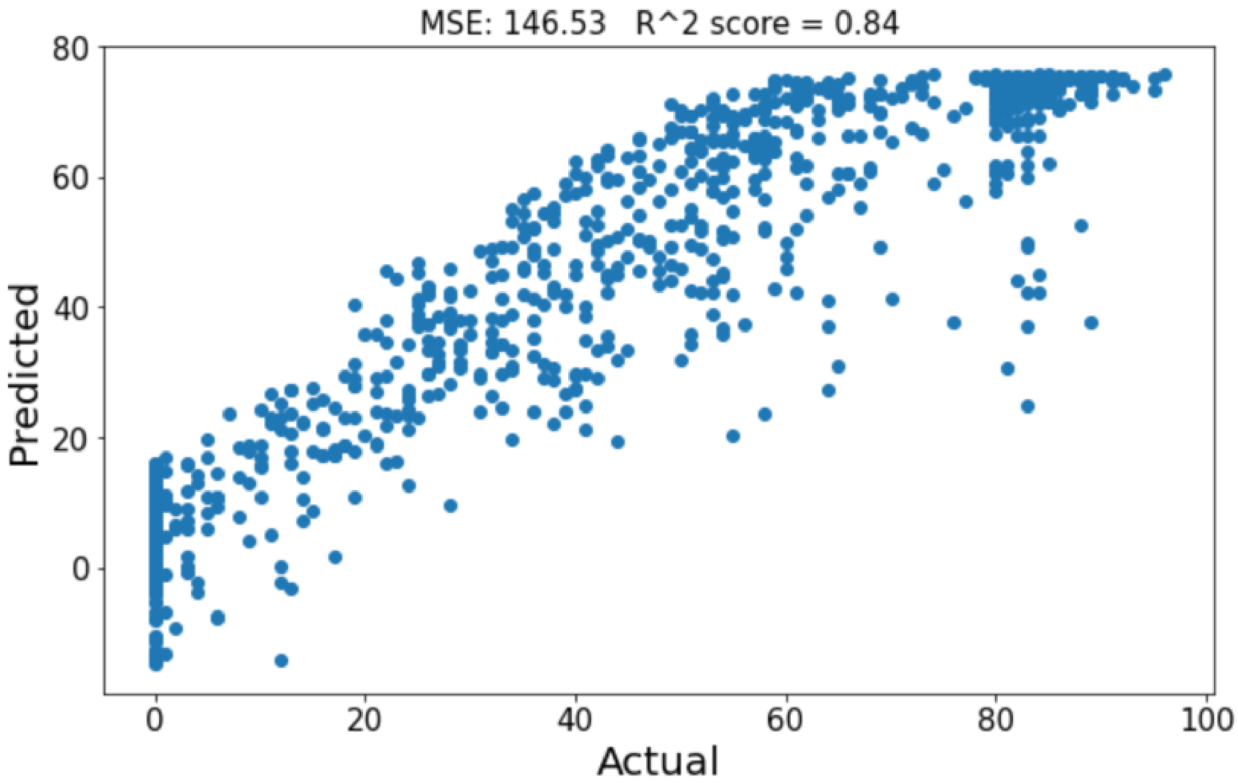
We also considered energy and loudness. Unfortunately, after de-trending those, popular and unpopular songs ended up having overlapping same ranges regardless of their release year. Similarly, popular and unpopular songs shared the same range for duration and danceability, regardless of the release year.

As for instrumentalness, it is trending down, but lower instrumentalness consistently resulted in a higher popularity.

Putting it all together



For our next attempt, we chose to use the coefficients and variables discussed above as well as release year and balanced data. This resulted in getting a higher r^2 value (hence, a better fit), but also a higher MSE: our model was still making mistakes at 0 and 70.



For our final attempt at predicting popularity, we decided to revamp our balancing by using narrower bins using all songs with a popularity above 80, and random samples between 79-50, 49-20, and less than 20. This smaller train/test set allowed us to lower our MSE (fewer high value mistakes were made, but we did have fewer data points). We were also able to achieve a higher R^2 score, which was our best fit compared to previous models. However, this model was still incapable of predicting any song with a popularity above 80.

Conclusion

To conclude, our last model was currently our best model due to a lower MSE, highest R^2 value, both of which translates to a more linear relationship. There were also fewer mistakes for high values. And as mentioned above, it is still not accurately predicting extreme values (high and low).

Lastly, looking at our model’s highest 20 mistakes by taking the difference between predicted popularity and actual popularity allowed us to notice that almost all the 'missed' songs were unpopular songs with a popularity around 0 but were predicted to be between 65-80.

Fun fact: our model is not a fan of Christmas classics: the song “White Christmas” has an actual popularity of 78 but was predicted to have a popularity of 5.