Regression Analysis on Track Popularity using a Spotify Dataset

Tristan Dull, Daniel Khan, Aaron Bajorunas, Liam Daly, and Christopher Chen

2024-06-09

Contents

List of Figures

Introduction

What makes a song popular?

There are many different ways to answer this compelling question, and we believe it's best to approach it using data science methods along with statistical techniques. Specifically, we want to determine whether or not certain individual factors contribute to higher levels of song popularity.

In order to answer our primary question, we will be analyzing a real-world Spotify dataset with 2400 observations of 23 different variables.

Analysis Goal

Our main goal is to harness the data available in our Spotify dataset to create a linear regression model. We'll then examine this model and make inferences on track popularity.

Given how our preliminary question is: "what makes a song popular?", our research question is as follows:

Which variables are most significant for predicting track popularity?

Stakeholder Interpretation

Our main stakeholders in this scenario would be music industry executives who are interested in examining factors that may influence track popularity. Given the insights generated by our analysis, they may be able to allocate resources toward improving certain factors for the sake of generating more track popularity.

Data Description

We are using data that was directly extracted from Spotify's API and compiled into a post on Kaggle.com called "Spotify Top Hit Playlist (2000-2023)". The Spotify dataset comprises a total of 2400 observations of 23 distinct variables. Overall, there are 23 columns and 2400 rows. It's organized so that there are 100 observations per year from 'Top Hit' playlists in 2000 to 2023. The main response variable of interest is `track_popularity`. Below is a description of each variable.

Playlist related:

- Playlist url: Categorical character variable for the link to the playlist.
- Year: Categorical double variable for the year the playlist was created.

Track related:

- Track_id: Unique categorical character identifier for specific track.
- Track_name: Categorical character variable for the name of track.
- Track_popularity: Numerical continuous double measurement for track's popularity on a scale of 0 to 100 (technically ordinal but we're classifying it as continuous).

Audio Features:

- Danceability: Numerical continuous double measurement from 0.0 to 1.0 for danceability; describe how suitable a track is for dancing.
- Energy: Numerical continuous double measurement from 0.0 to 1.0; represents a perceptual measure of intensity and activity.
- Key: Numerical continuous double measurement for the key of track.
- Loudness: Numerical continuous double measurement for how loud the track is in dB.
- Mode: Binary categorical double variable indicating major or minor key.

- Speechiness: Numerical continuous double measurement from 0.0 to 1.0 for speechiness: the presence of spoken word in the track.
- Acousticness: Numerical continuous double confidence measurement measurement from 0.0 to 1.0 of whether the track is acoustic.
- Instrumentalness: Numerical continuous double measurement for instrumentalness from 0.0 to 1.0. The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content.
- Liveness: Numerical continuous double measurement from 0.0 to 1.0 for the presence of an audience in the recording.
- Valence: Numerical continuous double measurement from 0.0 to 1.0 describing the musical positiveness conveyed by a track.
- Tempo: Numerical continuous double measurement for beats per minute.
- Duration_ms: Numerical continuous double measurement for track duration in milliseconds.
- Time_signature: Numerical categorical double measurement for time signature (technically ordinal but we're classifying it as continuous).

Album related:

- Album: Categorical character variable for the name of the album.

Artist related:

- Artist_id: Categorical character variable for id for artist.
- Artist_name: Categorical character variable for name of artist.
- Artist_genre: Categorical character variable for genre of artist.
- Artist_popularity: Numerical continuous double measurement for of artist popularity on a scale of 0 to 100.

Please view our raw rmd file for detailed summary statistics.

Exploratory Data Analysis and Visualizations

Data Cleaning

Upon looking through the entire dataset, it became clear that it wasn't fit for regression analysis straight away. We first began our data cleaning process by removing NA's and duplicate entries. We also found the following variables below had far too many categories and were unfit to include in our analysis moving forward.

- playlist_url
- track_id
- track_name
- album
- artist_id
- artist_name
- artist_genre

Visualizations

Response Variable Plots

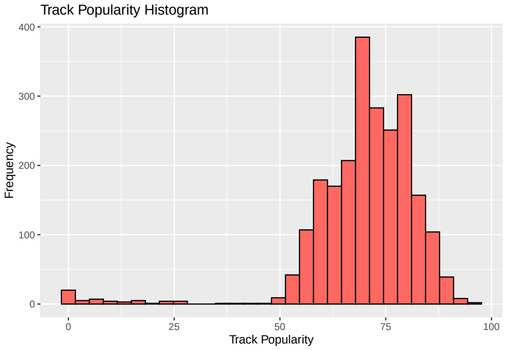


Figure 1: Track Popularity Histogram Before Filtering

The above plot indicates track_popularity is severely left skewed. There are several outlier track_popularity scores lower than 25. These low observations will likely significantly influence our analysis. In order to normalize our response and avoid model instability, we decided to filter out entries with track_popularity scores lower than 45.

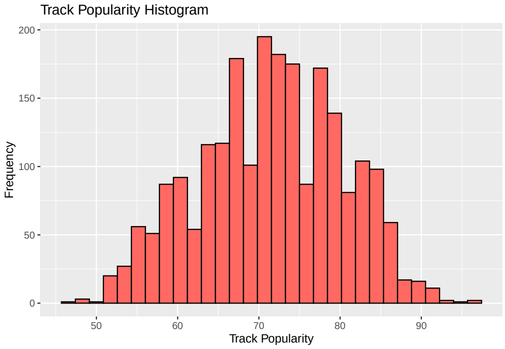


Figure 2: Track Popularity Histogram After Filtering

The distribution now appears to be approximately normal.

Predictor Plots

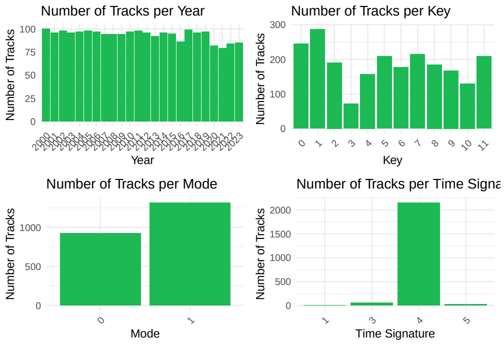


Figure 3: Categorical Predictor Distributions

The above categorical predictor plots indicate fluctuating distributions for the number of tracks represented by each year and each key. There's more tracks of mode "1" and a vast majority of tracks are categorized under time signature "4."

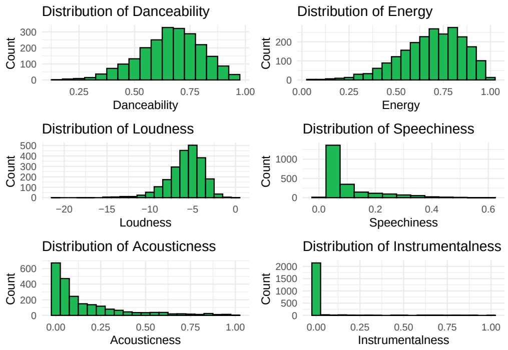


Figure 4: Continuous Predictor Distributions I

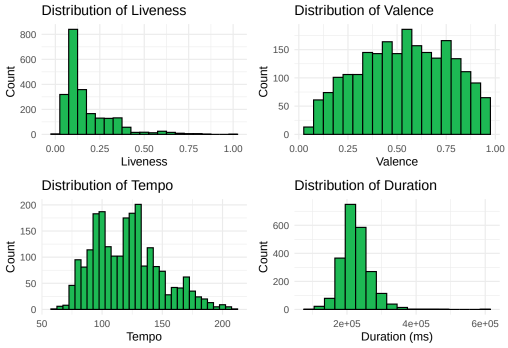


Figure 5: Continuous Predictor Distributions II

Continuous Plots Interpretations:

- Danceability, Energy, and Loudness are all left skewed.
- Speechiness, Acousticness, Instrumentaness, Liveness, and Duration are right skewed.
- Valence appears approximately normally distributed, Tempo has a fluctuating distribution.

Our plots indicate some predictors may be correlated to one another.

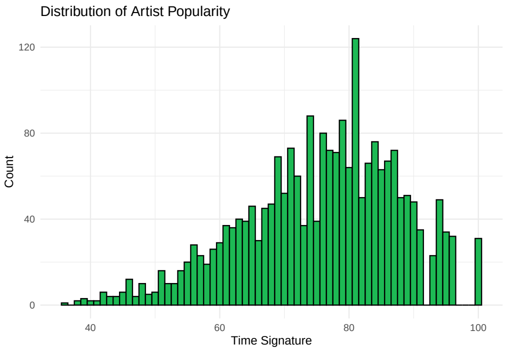


Figure 6: Artist Popularity Distributions

Based on the plot above, artist popularity appears left skewed

Correlation Heatmap

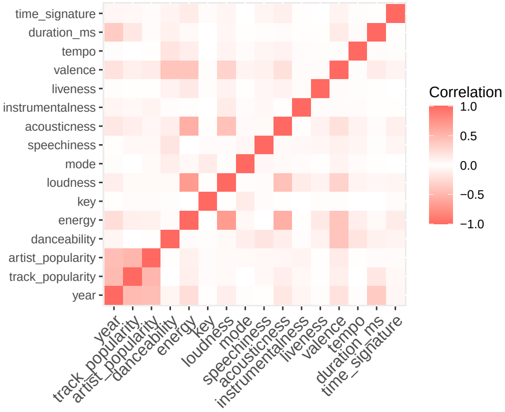


Figure 7: Correlation Heatmap

If predictors are highly correlated to one another, problems may arise and our variance may be inflated.

Our heatmap doesn't indicate any serious problems, but there are darker red squares for the following pairs of predictor variables:

- acoustincess and energy
- loudness and energy

Given this, we decided to specifically examine their correlation coefficients.

```
## Correlation coefficient for acousticness and energy:: -0.5520272
```

```
## Correlation coefficient for loudness and energy 0.6930411
```

These values are highly correlated but they're below the abs(.7) threshold so we decided to keep them.

Regression Analysis

For all code related to our entire linear regression model building process, we encourage you to view our raw rmd file.

Data Splitting

To evaluate and build our models, we'll be splitting spotify into 50% training and 50% test. The purpose of this is to make initial interpretations based on training data and evaluate how our candidate models perform based on unseen test data.

Full Model

```
##
## Call:
## lm(formula = track_popularity ~ year + artist_popularity + danceability +
##     energy + key + loudness + mode + speechiness + acousticness +
##     instrumentalness + liveness + valence + tempo + duration_ms +
##     time_signature, data = train)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -29.2626  -4.9355   0.0089   5.1559  20.9944
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -6.158e+02  8.227e+01  -7.486 1.45e-13 ***
## year               3.338e-01  4.078e-02   8.186 7.37e-16 ***
## artist_popularity  2.575e-01  2.020e-02  12.749  < 2e-16 ***
## danceability       1.810e-01  1.834e+00   0.099   0.9214
## energy             1.571e-01  2.106e+00   0.075   0.9405
## key               -6.884e-02  6.062e-02  -1.136   0.2563
## loudness           1.957e-01  1.477e-01   1.325   0.1855
## mode              -2.592e-02  4.494e-01  -0.058   0.9540
## speechiness       -4.144e+00  2.346e+00  -1.767   0.0776 .
## acousticness       1.590e+00  1.266e+00   1.256   0.2094
## instrumentalness   7.009e-01  2.957e+00   0.237   0.8127
## liveness          -3.905e-01  1.593e+00  -0.245   0.8064
## valence            4.906e-01  1.234e+00   0.398   0.6910
## tempo             -1.734e-05  8.018e-03  -0.002   0.9983
## duration_ms       -5.717e-06  5.724e-06  -0.999   0.3181
## time_signature    -2.472e-01  1.031e+00  -0.240   0.8105
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.211 on 1107 degrees of freedom
## Multiple R-squared:  0.3057, Adjusted R-squared:  0.2963
## F-statistic: 32.49 on 15 and 1107 DF,  p-value: < 2.2e-16
```

Statistically Significant Predictor Coefficients

Out of the 15 total predictors used, only the following three are associated with statistically significant coefficients with alpha level = .10:

- year
- artist_popularity
- speechiness

Artist popularity is by far the most statistically significant predictor since it's associated with the highest t value and lowest p value.

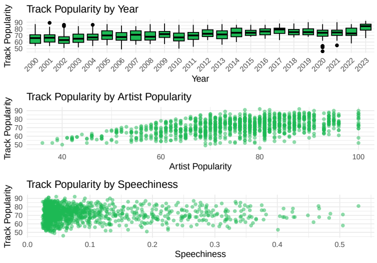


Figure 8: Full Model: Significant Predictors vs Track Popularity

Year and artist_popularity are positively linearly associated with track_popularity while speechiness is negatively associated. On the plot above you can see clear positive trends for year and artist popularity. Speechiness, on the other hand, displays a somewhat unclear trend.

F Statistic

H_0: All model terms are unimportant for predicting track_popularity

H_A: At least one model term is useful for predicting track_popularity

The p value output is statistically significant so reject H_0.

Multicollinearity

```
##               year artist_popularity       danceability             energy
##           1.698634          1.320464           1.557245           2.673691
##                key          loudness               mode        speechiness
##           1.022095          1.941688           1.045820           1.085894
##       acousticness  instrumentalness           liveness            valence
##           1.556087          1.056469           1.045883           1.733217
##              tempo       duration_ms     time_signature
##           1.084505          1.252849           1.052186
```

Variance inflation factors above 10 indicate the presence of multicollineairty, a situation in which multiple predictors are highly correlated to one another. In order to avoid model instability we ideally want to see low VIF scores.

As you can see above, all VIF values are far below 10, indicating multicollinearity is not a concern.

R Squared Statistics

Multiple R squared is .3057 and Adj. R squared is .2963.

- 30.57% of variability in track_popularity can be explained by our full model.
- 29.63% of variability in track_popularity can be explained by our model while adjusting for the number of predictors used.

These R squared values are relatively low, but we believe they're acceptable considering this is real world data based on an art-form.

Full Model Assumptions

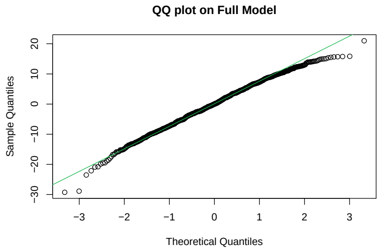


Figure 9: Full Model qqplot

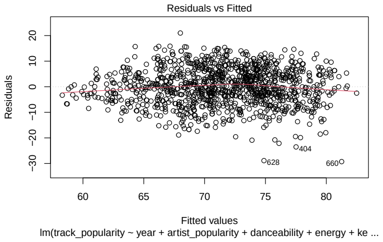


Figure 10: Full Model Residuals v Fitted Plot

```
shapiro.test(m1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m1$residuals
## W = 0.99373, p-value = 0.0001152
```

```
dwt(m1)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.07730377      1.840045   0.006
##  Alternative hypothesis: rho != 0
```

Points on the qqplot deviate from the qqline, and the Shapiro-Wilk test outputs a statistically significant p value. Therefore, our full model violates the normality assumption.

Points on our residuals v. fitted plot appear randomly clustered, and the red line does not significantly deviate from 0. We can conclude linearity and constant variance are approximately satisfied.

The Durbin-Watson p value is statistically significant. This means autocorrelation is present in our model and independence is violated.

Only two out of four total assumptions are satisfied and our R squared value is somewhat low, so our goal moving forward is to determine if we can improve our model in any way.

Model Transformations

We decided to assess the following three transformations on track_popularity:

- square root
- log
- inverse

After testing, we found that none of these transformations contributed to significant improvements in our R squared statistics. They also did not improve any of our assumptions. Our findings indicated that transformations on our response would not lead to any type of significant improvement for our full model. Therefore, we decided to see if we could reduce our model.

Stepwise Selection

We used both-direction stepwise selection to assess whether or not our full model could be reduced. The function output the model with the lowest AIC below.

```
summary(stepwise_model)
```

```
## 
## Call:
## lm(formula = track_popularity ~ artist_popularity + year + speechiness + 
##     loudness, data = train)
## 
## Residuals:
##      Min       1Q   Median       3Q      Max 
## -29.7284  -4.8980  -0.0237   5.2281  21.3861 
## 
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)    
## (Intercept)       -659.11532   70.10846  -9.401   <2e-16 ***
## artist_popularity    0.25356    0.01973  12.850   <2e-16 ***
## year                 0.35440    0.03521  10.065   <2e-16 ***
## speechiness         -4.25351    2.25400  -1.887   0.0594 .  
## loudness             0.15403    0.10632   1.449   0.1477    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
## 
## Residual standard error: 7.192 on 1118 degrees of freedom
## Multiple R-squared:  0.3027, Adjusted R-squared:  0.3002 
## F-statistic: 121.3 on 4 and 1118 DF,  p-value: < 2.2e-16
```

Statistically Significant Predictor Coefficients

Three out of four predictors are associated with statistically significant coefficients at alpha level = .10:

- artist_popularity
- year
- speechiness

Artist popularity is still the most significant predictor but year comes in as a close second.

F Statistic

H_0: All model terms are unimportant for predicting track_popularity

H_A: At least one model term is useful for predicting track_popularity

The p value output is statistically significant so reject H_0.

Multicollinearity

```
## artist_popularity              year       speechiness          loudness
##          1.267174          1.273438          1.008277          1.011298
```

All variance inflation factor values are far below 10, indicating multicollinearity is not a concern.

R Squared Statistics

Multiple R squared is .3027 and Adj. R squared is .3002.

- 30.27% of variability in track_popularity can be explained by our full model.
- 30.02% of variability in track_popularity can be explained by our model while adjusting for the number of predictors used.

Our Multiple R squared value is slightly lower than our full model but our Adj. R squared is slightly higher. This result indicates our stepwise model is better suited for our response with consideration for the amount of significant predictors used.

Stepwise Model Assumptions

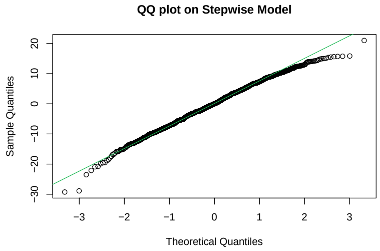


Figure 11: Stepwise Model qqplot

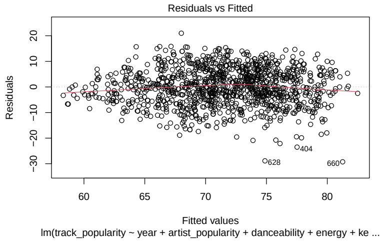


Figure 12: Stepwise Model Residuals v Fitted Plot

```
shapiro.test(m1$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  m1$residuals
## W = 0.99373, p-value = 0.0001152
```

```
dwt(m1)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1      0.07730377      1.840045    0.01
##  Alternative hypothesis: rho != 0
```

Assumption checks indicate linearity and constant variance are still satisfied, while normality and independence are still violated.

With consideration for the improved adjusted R squared statistic, we determined our stepwise model is a valid candidate for our final model.

Model Evaluation and Validations

Cross Validation

Next, we'll use K-Fold Cross Validation testing to assess which model performs better. We'll be comparing our models based on mean squared error: a value that assesses model prediction accuracy. The lower the mean squared error is, the better the model performs for unseen data.

Below are the steps of the our cross validation process:

1. Split our cleaned Spotify data into 10 equally sized subsets.
2. Train both our full and stepwise model 10 times using k-1 folds as the "train" data and the remaining fold as the unseen "test" data
3. Calculate training and test MSE's for each iteration and store the values into vectors
4. Output the average training and test MSE's and assess which model is associated with lower test MSE

For detailed code, please reference our raw rmd file.

```
## Average Training MSE for Full Model: 52.27889
```

```
## Average Test MSE for Full Model: 52.26966
```

```
## Average Training MSE for Stepwise Model: 52.5463
```

```
## Average Test MSE for Stepwise Model: 52.54705
```

On average, our full model is associated with a slightly lower test MSE compared to the stepwise model, indicating that it generalizes only slightly better to unseen data. We'd like to emphasize the extremely small difference in MSE between the two models. There are far fewer predictors in our stepwise model, and the performance difference in predictive power is negligible. Therefore, we'll select our stepwise model as our final model.

Final Model

Below, we'll assess how our stepwise model preforms when fitting it to our entire Spotify dataset.

```
final_model <- lm(formula = track_popularity ~ artist_popularity + year + speechiness +
    loudness, data = spotify)
summary(final_model)
```

```
##
## Call:
## lm(formula = track_popularity ~ artist_popularity + year + speechiness +
##     loudness, data = spotify)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -30.3845  -4.9119  -0.0089   5.1902  21.9840
##
## Coefficients:
##                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)       -725.06354   49.91350 -14.526  < 2e-16 ***
## artist_popularity    0.27402    0.01414  19.376  < 2e-16 ***
## year                 0.38618    0.02507  15.402  < 2e-16 ***
## speechiness         -6.71345    1.62888  -4.122  3.9e-05 ***
## loudness             0.02463    0.07356   0.335    0.738    
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 7.237 on 2241 degrees of freedom
## Multiple R-squared:  0.3318, Adjusted R-squared:  0.3306 
## F-statistic: 278.2 on 4 and 2241 DF,  p-value: < 2.2e-16
```

Coefficient Interpretation

- Every one unit increase in artist popularity induces an increase in track popularity by 0.274 units, when all other variables are held constant.
- Every one unit increase in year induces an increase in track popularity by .386 units, when all other variables are held constant.
- Every one unit increase in speechiness induces a decrease in track popularity by 6.71345 units, when all other variables are held constant.
- Every one unit increase in loudness induces an increase in track popularity by .02463 units, when all other variables are held constant.

Out of these predictors, artist_popularity, year, and speechiness are associated with statistically significant coefficients.

Artist popularity is, again, the most statistically significant predictor coefficient since it's associated with highest t value and lowest p value.

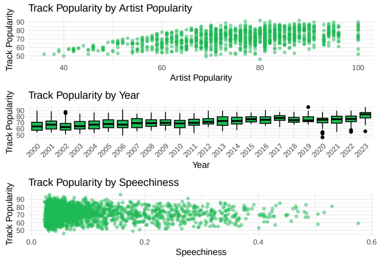


Figure 13: Final Model: Significant Predictors vs Track Popularity using complete dataset

Similarly to our full model, year and artist_popularity are positively linearly associated with track_popularity while speechiness has a negative linear association. The plot above also continues to illustrate the somewhat unclear trend between speechiness and track_popularity.

F Statistic

H_0: All model terms are unimportant for predicting track_popularity

H_A: At least one model term is useful for predicting track_popularity

The p value output is statistically significant so reject H_0.

Multicollinearity

```
## artist_popularity              year       speechiness          loudness 
##          1.254096          1.267964          1.003776          1.016403
```

All variance inflation factor values in our final model are far below 10, indicating multicollinearity is not a concern.

R Squared Statistics

Multiple R squared is .3318 and Adj. R squared is .3306. This indicates that our final model explains more variability in track_popularity compared to any of the models we tested using our training set.

- 33.18% of variability in track_popularity can be explained by our full model.
- 33.06% of variability in track_popularity can be explained by our model while adjusting for the number of predictors used.

These are still relatively low values, but they are still suitable for inferences.

Final Model Assumptions

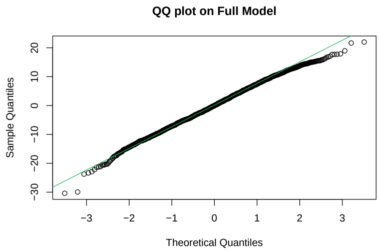


Figure 14: Final Model qqplot

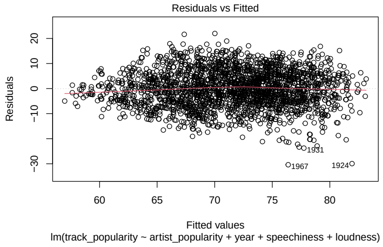


Figure 15: Final Model Residuals v Fitted Plot

```
shapiro.test(final_model$residuals)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  final_model$residuals
## W = 0.99699, p-value = 0.0002174
```

```
dwt(final_model)
```

```
##  lag Autocorrelation D-W Statistic p-value
##    1       0.6239573     0.7498106       0
##  Alternative hypothesis: rho != 0
```

The points on our qqplot still noticeably deviate from the qqline, and our Shapiro-Wilk p value is statistically significant. Therefore, normality is still violated when we fit to our full model.

On our residuals v. fitted plot, the points still appear randomly clustered, and the red line is consistently close to the zero line. This indicates constant variance and linearity are satisfied.

The Durbin-Watson p value of zero indicates autocorrelation, therefore independence is violated.

Final Model Remarks

In consideration of all our findings, we conclude that our final model is the best fit for track_popularity compared to all others we tested. It contains fewer predictors overall, with three out of four being statistically significant. It also has the highest R squared statistics we observed, and best-satisfies two out of four linear regression assumptions.

Conclusions and Discussion

Following our regression analysis, we found that the predictors below are significantly linearly related to track_popularity.

- Artist Popularity
- Year
- Speechiness

Unsurprisingly, artist_popularity turned out to be the most significant predictor for track_popularity. Tracks released by already highly popular artists typically receive high popularity scores. Furthermore, track popularity scores tend to increase with every passing year. This could be attributed to Spotify's growing user-base, or a variety of other factors. Interestingly, speechiness is significantly negatively related to track popularity; meaning songs with a higher degree of spoken word tend to be less popular.

Essentially, we recommend music industry executives to direct their focus toward reducing speechiness when possible, as this negatively impacts track popularity. They should attempt to work with artists who are already popular in order to maximize track popularity.

Limitations

The final model we created faces some important limitations. Two out of four linear regression assumptions are violated. Shapiro-Wilk shows our response residuals are not normally distributed and the Durbin Watson test shows some autocorrelation.

While we encourage preliminary inferences based on our final model, we would not recommend this model for definitive concrete conclusions. Our R squared value indicates only 33.18% of variability in track popularity is explained by our data. This means Spotify track popularity is impacted by many external factors which are not included in our dataset.

Appendix

Other Findings

Genre Analysis

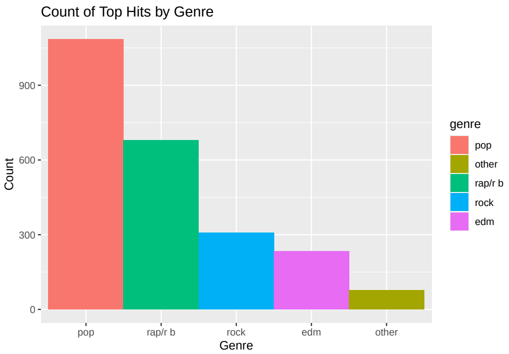


Figure 16: Count of Top Hits by Genre

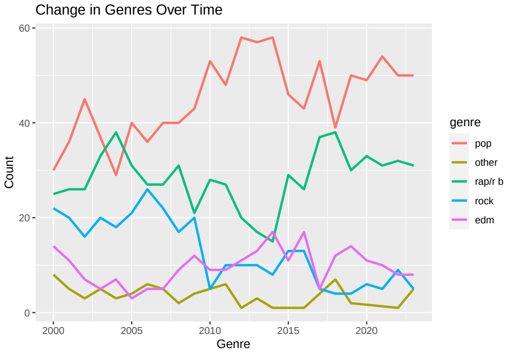


Figure 17: Changes in Genre Over Time

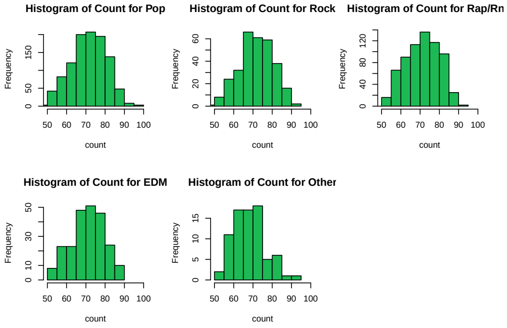
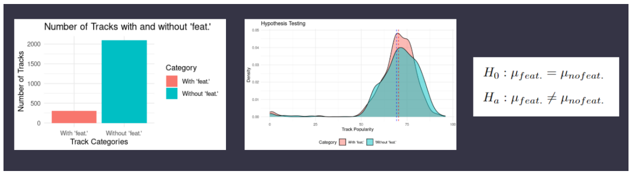


Figure 18: Histograms for Major Genres

The original dataset had 241 different genres and each artist had multiple genres assigned to them. This was categorized into 5 distinct groups in order to get a better look at the trends. These categories are pop, rock, rap/RnB, EDM, and other.

After cleaning up the genres in our data, using this code, we are able to create these visualizations. It's apparent that pop has been one of the most popular genres over the past 24 years since a large number of the top hit songs are pop songs each year and overall. However, the line graph does show that there has been a rise in popularity of rap/RnB in recent years. The figure with histograms also shows that each genre follows an approximately normal distribution when it comes to track popularity.

Features



$$H_0 : \mu_{feat.} = \mu_{nofeat.}$$

$$H_a : \mu_{feat.} \neq \mu_{nofeat.}$$

Figure 19: Feature Visualizations

When determining if there's a difference between popularity of music with and without features, our findings indicate that a song's feature doesn't amplify a musician's popularity.

Authors' Contributions

Tristan Dull: Organized project direction, stepwise regression, initialized final model, plot of actual values with predicted

Liam Daly: Organized slides template, cleaned data, created variable visualizations/correlation heatmap, initialized model building for full model, organized final document template/sections, wrote interpretations.

Christopher Chen: Organized the look of slides, cleaned up genres in the data, created visualizations for EDA and for genre analysis.

Daniel Khan: Filtered music for with and without features, visualized average song popularity and performed hypothesis testing

Aaron Bajorunas: Conducted analysis on song duration, danceability, and their relationship with popularity, and creating visualizations and added cross validation results

Data Availability

https://www.kaggle.com/datasets/josephinelsy/spotify-top-hit-playlist-2010-2022