UNC Charlotte School of Data Science

DSBA 6211 Final Project

Predicting the Popularity of Songs Released in the 2010s

Cameron Harwood | Yash Jinwala | Lauren Battles | Michael Lewis | Ty Birling

Introduction

The purpose of this project is to understand the different factors that can be attributed to whether a song is a "hit". A hit song in this case is one that has been seen on the Top 100 weekly billboard at least once. For the purpose of this project, we will be focusing on songs after 2010. To conduct research on this, we will look at topic modeling to identify whether there are specific topics that dominate the Top 100 Billboard. We will also create a Logistic Regression and Random Forest model to determine the most significant variables that make up a hit. In addition to this, we will look at the effect of the lyrics on these models which we will implement using Singular Value Decomposition to determine whether they have any impact on the accuracy.

Data Summary

Data Dictionary (Kaggle)

- Target: The target variable for the track. It can be either '0' or '1'. '1' implies that this song has featured in the weekly list (Issued by Billboards) of Hot-100 tracks in that decade at least once and is a 'hit'. '0' implies that the track is a 'flop'.
- Danceability: Danceability describes how suitable a track is for dancing based on a combination of musical elements.
- Energy: Energy is a measure from 0.0 to 1.0 and represents a perceptual measure of intensity and activity.
- Key: The estimated overall key of the track. Integers map to pitches using standard Pitch Class notation. E.g. 0 = C, 1 = C?/D?, 2 = D, and so on.
- Loudness: The overall loudness of a track in decibels (dB). Loudness values are averaged across the entire track.

- Mode: Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. (1 or 0)
- Speechiness: Speechiness detects the presence of spoken words in a track.
- Acousticness: A confidence measure from 0.0 to 1.0 of whether the track is acoustic.
- Instrumentalness: Predicts whether a track contains no vocals.
- Valence: A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track.
- Tempo: The overall estimated tempo of a track in beats per minute (BPM).
- Duration_ms: The duration of the track in milliseconds.

Data Preprocessing

Logistic Regression/Random Forest

1. Removed variables track, artist, uri, sections, chorus_hit, liveness, time_signature
2. Log transformation performed on variables speechiness, loudness, accousticness, instrumentalness
3. Factored variables key, mode, and target

SVD

1. Create a corpus for the song lyrics, remove stopwords and convert to tf-idf dfm
2. Create SVD matrix with 10 variables, then trim down to 6 after evalutation
3. Create our final matrix by multiplying modelSvd$docs * diag(modelSvd$sk)
4. Join this new matrix with original Kaggle set using uri as our ID

Analysis Methods and Results

Logistic Regression

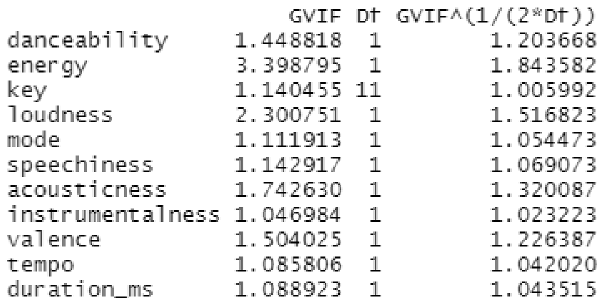
This analysis was completed using R, with packages Dplyr, caret, e1071, car, pROC, and ROCR. To start, we tested the multicollinearity of the preprocessed dataset with danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, valence, tempo, and duration_ms as the independent variables and target as the dependent variable. Next, we set a random seed at 101. Then, we created an ID per row to organize the data. After adding the row ID, we created the training set for the model with a sample of 70% of the data, followed by the test set creation containing 30%.

The training set was then used to run the logit regression model with target as the dependent variable and danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, valence, tempo, and duration_ms as the independent variables.

Finally, we evaluated the model's performance with a confusion matrix and Receiver Operating Characteristic's (ROC) Area Under the Curve (AUC). The missing values were omitted from the test data set for evaluation. The threshold was set to 0.5. Once the Confusion Matrix was established, we plotted the ROC and checked the numerical value for AUC.

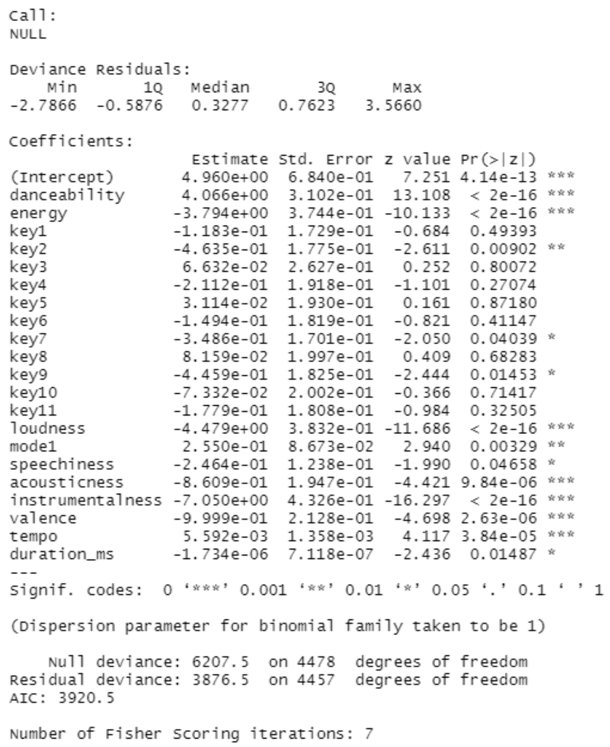
Results

The multicollinearity results showed all variables having a multicollinearity less than 5. A multicollinearity between 5 and 10 indicates high intercorrelations among two or more independent variables in the regression model. High intercorrelation would warrant a variable's exclusion from the model. Considering the results below, none of the variables needed to be excluded.



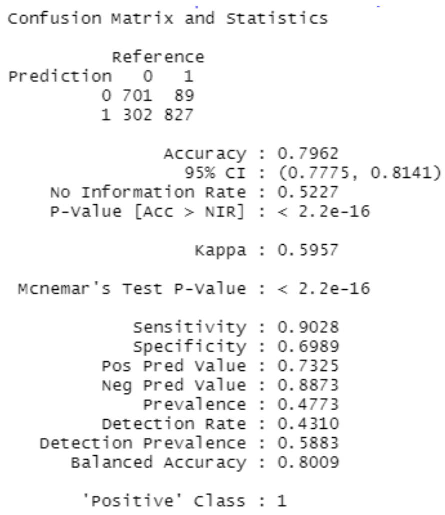
```
                     GVIF Df GVIF^(1/(2*Df))
danceability     1.448818  1        1.203668
energy           3.398795  1        1.843582
key              1.140455 11        1.005992
loudness         2.300751  1        1.516823
mode             1.111913  1        1.054473
speechiness      1.142917  1        1.069073
acousticness     1.742630  1        1.320087
instrumentalness 1.046984  1        1.023223
valence          1.504025  1        1.226387
tempo            1.085806  1        1.042020
duration_ms      1.088923  1        1.043515
```

The logistic regression model results below indicate the significant and insignificant variables in predicting the likelihood of a song becoming a hit or a flop. Based on the p-values with a cutoff of .05 and the coefficients, the variables that had a positively significant impact were: danceability, mode1, and tempo. This means the higher danceability and tempo measures that a song has, the more likely it was to be a hit in the 2010s. This also indicates that songs that belong to the mode1 category are more likely to be a hit. The variables that had a negatively significant impact were: energy, key2, key7, key9, loudness, speechiness, acousticness, instrumentalness, valence, and duration_ms. Indicating that songs with a high measure of energy, loudness, speechiness, acousticness, instrumentalness, valence, and duration_ms were less likely to be a hit in the 2010s. Also, songs that can be put into the key2, key7, or key9 categories were less likely to be a hit in the 2010s.

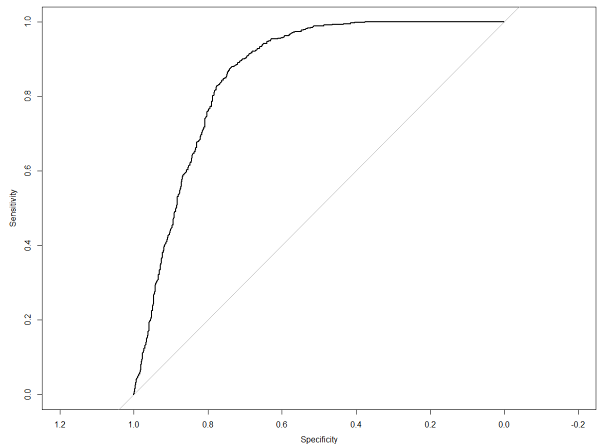


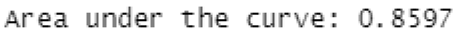
```
Call:
NULL

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-2.7866  -0.5876   0.3277   0.7623   3.5660

Coefficients:
                   Estimate Std. Error z value Pr(>|z|)
(Intercept)       4.960e+00  6.840e-01   7.251 4.14e-13 ***
danceability      4.066e+00  3.102e-01  13.108  < 2e-16 ***
energy           -3.794e+00  3.744e-01 -10.133  < 2e-16 ***
key1             -1.183e-01  1.729e-01  -0.684  0.49393
key2             -4.635e-01  1.775e-01  -2.611  0.00902 **
key3              6.632e-02  2.627e-01   0.252  0.80072
key4             -2.112e-01  1.918e-01  -1.101  0.27074
key5              3.114e-02  1.930e-01   0.161  0.87180
key6             -1.494e-01  1.819e-01  -0.821  0.41147
key7             -3.486e-01  1.701e-01  -2.050  0.04039 *
key8              8.159e-02  1.997e-01   0.409  0.68283
key9             -4.459e-01  1.825e-01  -2.444  0.01453 *
key10            -7.332e-02  2.002e-01  -0.366  0.71417
key11            -1.779e-01  1.808e-01  -0.984  0.32505
loudness         -4.479e+00  3.832e-01 -11.686  < 2e-16 ***
mode1             2.550e-01  8.673e-02   2.940  0.00329 **
speechiness      -2.464e-01  1.238e-01  -1.990  0.04658 *
acousticness     -8.609e-01  1.947e-01  -4.421 9.84e-06 ***
instrumentalness -7.050e+00  4.326e-01 -16.297  < 2e-16 ***
valence          -9.999e-01  2.128e-01  -4.698 2.63e-06 ***
tempo             5.592e-03  1.358e-03   4.117 3.84e-05 ***
duration_ms      -1.734e-06  7.118e-07  -2.436  0.01487 *
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6207.5  on 4478  degrees of freedom
Residual deviance: 3876.5  on 4457  degrees of freedom
AIC: 3920.5

Number of Fisher Scoring iterations: 7
```

The confusion matrix below shows an accuracy score of .796 or 79.6%. 79.6% of the model’s predictions were correct. The Receiver Operating Characteristic’s Area Under the Curve was .859. Indicating the model’s 85.9% ability to distinguish between a song being a hit or a flop.



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 701  89
         1 302 827

               Accuracy : 0.7962
                 95% CI : (0.7775, 0.8141)
    No Information Rate : 0.5227
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.5957

 Mcnemar's Test P-Value : < 2.2e-16

            Sensitivity : 0.9028
            Specificity : 0.6989
         Pos Pred Value : 0.7325
         Neg Pred Value : 0.8873
             Prevalence : 0.4773
         Detection Rate : 0.4310
   Detection Prevalence : 0.5883
      Balanced Accuracy : 0.8009

       'Positive' Class : 1
```



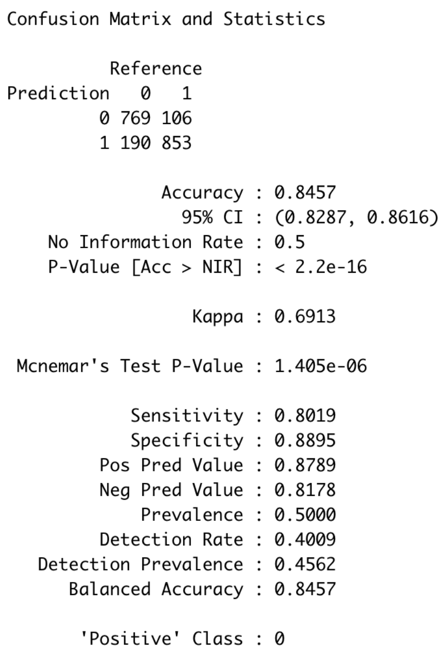


```
Area under the curve: 0.8597
```

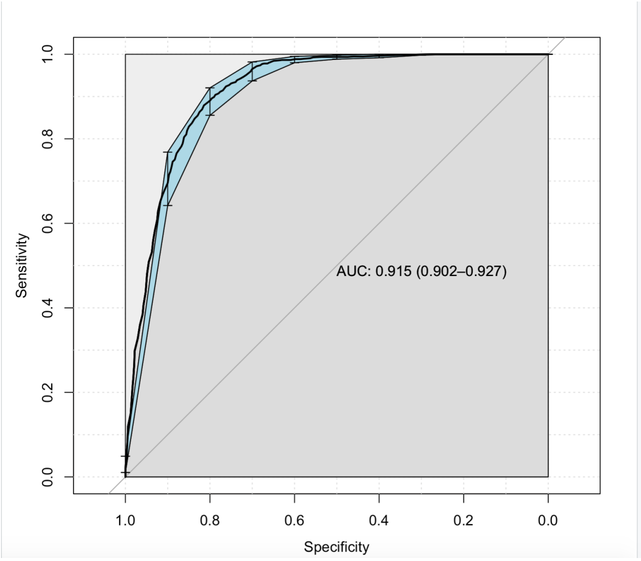
Random Forest

The packages used to create the Random Forest were ISLR, caret, and randomForest. After inputting the preprocessed data into R, we must merge the five new columns from the SVD. The independent variables used include danceability, energy, key, loudness, mode, speechiness, acousticness, instrumentalness, valence, tempo, duration_ms, and Main_Topic (from SVD). The dependent variable was target which is binary and states whether the song was featured once on the top 100 weekly billboard. We must factor the categorical variables target, key, and mode. To ensure similar results on the same dataset, we will set our seed to 101. After this, we can create our test/train split, which randomly selects 70% of the dataset to be trained and the remaining 30% to be the test set. After doing this, we can create two random forest models with ntree = 100.

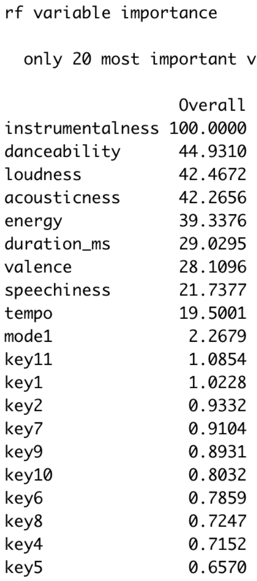
The confusion matrix and ROC curve results for the Random Forest model was as follows:



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 769 106
         1 190 853

               Accuracy : 0.8457
                 95% CI : (0.8287, 0.8616)
    No Information Rate : 0.5
    P-Value [Acc > NIR] : < 2.2e-16

                  Kappa : 0.6913

 Mcnemar's Test P-Value : 1.405e-06

            Sensitivity : 0.8019
            Specificity : 0.8895
         Pos Pred Value : 0.8789
         Neg Pred Value : 0.8178
             Prevalence : 0.5000
         Detection Rate : 0.4009
   Detection Prevalence : 0.4562
      Balanced Accuracy : 0.8457

       'Positive' Class : 0
```



The model without the SVD variables had an accuracy of 84.57% and an area under the curve of .9146 when running the prediction on the test set. The most important variables were as follows:



```
rf variable importance

  only 20 most important variables shown (out of 21)

                  Overall
instrumentalness 100.0000
danceability      44.9310
loudness          42.4672
acousticness      42.2656
energy            39.3376
duration_ms       29.0295
valence           28.1096
speechiness       21.7377
tempo             19.5001
mode1              2.2679
key11              1.0854
key1               1.0228
key2               0.9332
key7               0.9104
key9               0.8931
key10              0.8032
key6               0.7859
key8               0.7247
key4               0.7152
key5               0.6570
```

As we can see instrumentalness was the most important predictor of whether a song will be a hit or not, followed by danceability, loudness, and acousticness. This aligns with the logistic regression as these were all significant variables. The least important predictors were the key variables which were factored.

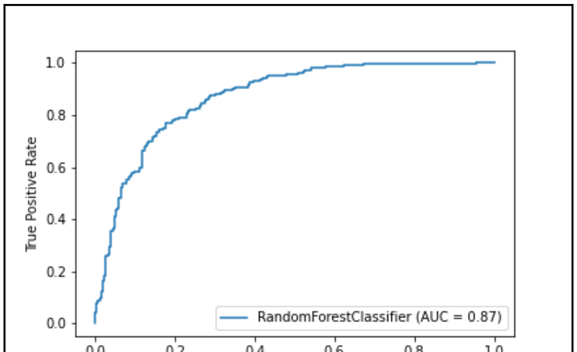
Topic Modeling

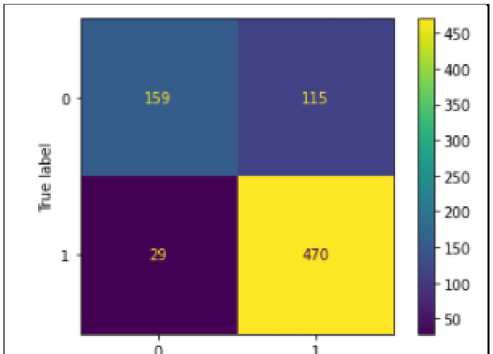
The Genius API was used in conjunction with the BeautifulSoup & Requests modules within Python to pull the lyrics of roughly 4000 songs from the dataset. Using these lyrics, a term frequency matrix was created, with the only stopwords used being those found in the NLTK Stopwords package. Tf-idf was not used as the distribution of words in the songs was in a consistent range, with only 36 songs being deemed outliers. These outliers had more than 1500 words in each song, but were still included in building the LDA model as it seemed unnecessary to exclude them from the dataset. A tunegrid was used to find the optimal parameters of the LDA model, which ended up being 10 categories with a learning decay rate of .5. These topics contained many words that offered no information about the contents of the lyrics and were subsequently added to the list of stopwords. One topic that did prove useful was a topic that contained songs in non-english languages. Songs that fit in this topic were dropped from further LDA models as it seemed inappropriate to include them if we were unable to understand the subject matter of the song. This topic contained less than 100 songs. LDA models were run repeatedly, removing words that contained low information, until we had topics that we felt we were able to accurately interpret. Roughly 50 words were removed. We best described these five topics as:

1. Money/Possessions
2. Heartbreak
3. Love
4. Physical Intimacy
5. Life/Death

The topics were then joined with the Kaggle dataset and songs with no topics (i.e. lyrics were unable to be pulled) were dropped. The same data transformations and dropped variables were used in this model as compared to the previous models. A Random Forest model and Logistic Regression model were run on the new dataset. Below are the ROC curves and confusion matrices for each model.

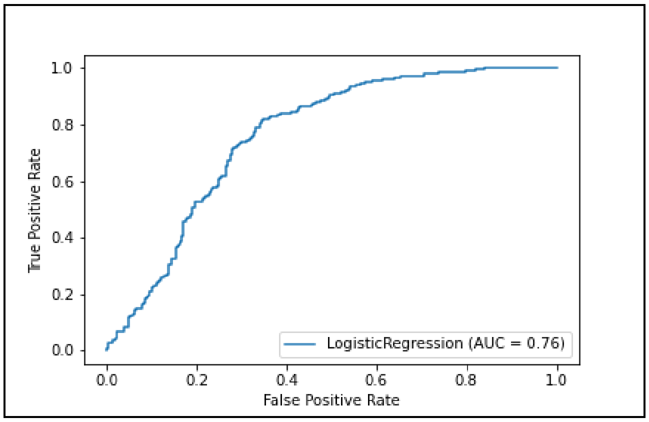
Random Forest

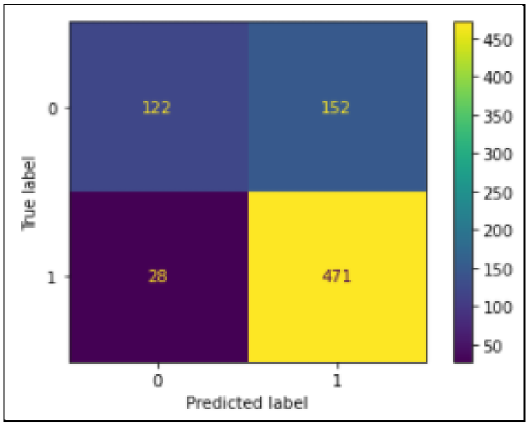




Accuracy:	81.37%
Sensitivity:	94.19%
Specificity:	58%

Logistic Regression





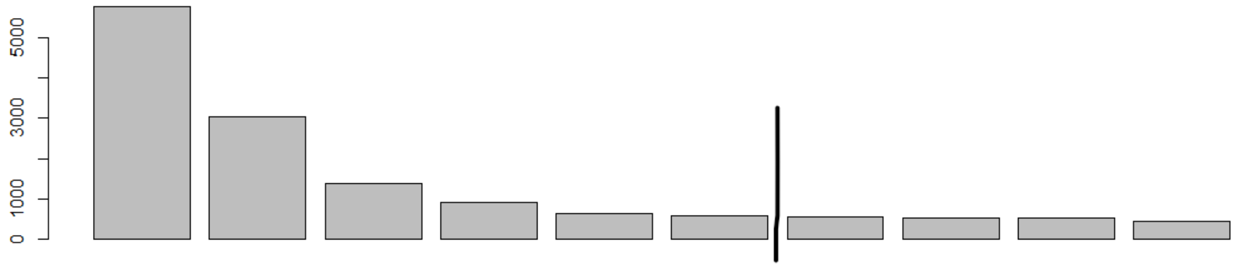
Accuracy:	76.71%
Sensitivity:	94.39%
Specificity:	44.53%

The addition of topics in the models added no predictive power. Topics were the least important factors in both the Random Forest and Logistic Regression models. The accuracy of both models decreases relative to the models fitted without the addition of topics. Sensitivity increases substantially for both models, although it is worth noting that the distribution of target changes from a 50-50 distribution in the baseline model to a 67-33 distribution in favor of 'Hit' in the models with topic modelling. For this reason, the increase in sensitivity makes sense. The takeaway from these models is that the lyrical content of songs does not play as critical of a role in whether a song will be a hit or not compared to the musical components such as instrumentalness. It could also be possible that the topics were not condensed enough to accurately describe the contents of the song, and therefore the topics were too broad to contain predictive power.

Latent Semantic Analytics using Singular Value Decomposition

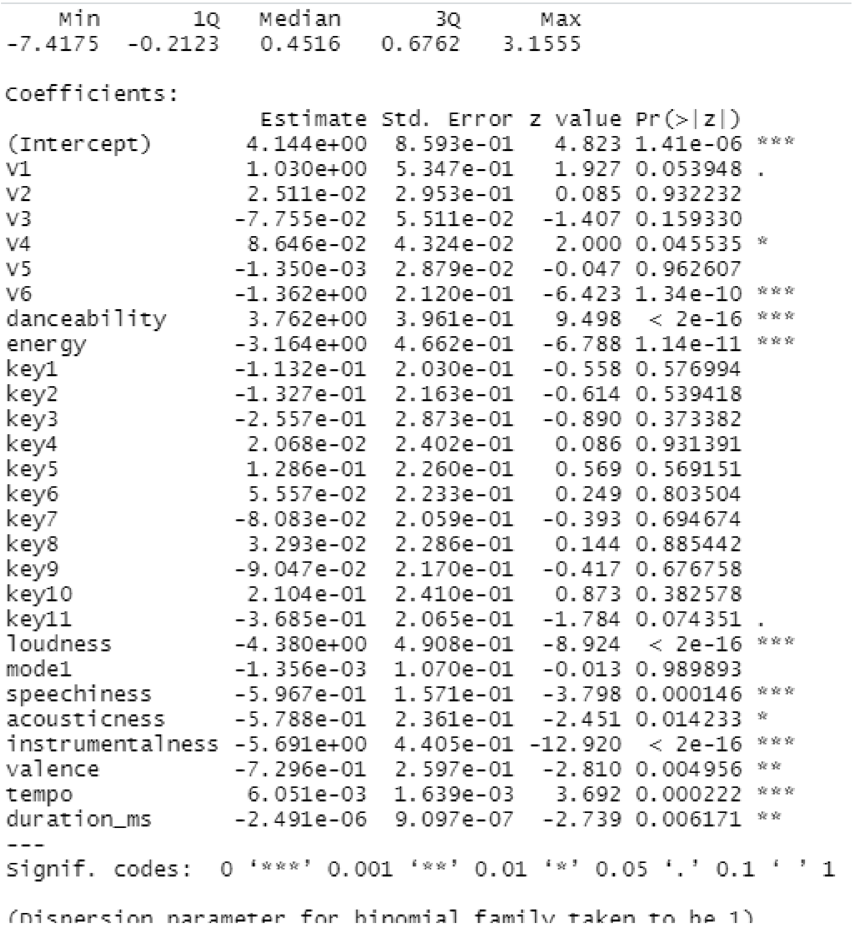
Similar to the topic modeling exercise, we performed another modeling method that incorporated song lyrics into our set of variables. The intent for using the SVD method is to capture enough variance from the lyrics with many fewer variables than a typical dfm, and then include those variables in our model to strengthen it. We found that this method improved our model but only marginally, as we saw increases of roughly 1 percentage points for the logistic model accuracy and a very small improvement in the AUC value as well. Below is a discussion of this process and its results.

Data preprocessing is not much different from our other models. We create a corpus for the lyrics, remove stopwords and punctuation and create a data frequency matrix from which we can work. We then remove words that are infrequent or appear in fewer than 2 documents before converting our DFM into a tf-idf which will take into account the length of songs' lyrics. Now we are ready to convert our tf-idf text model to SVD using R code and we must determine the number of variable dimensions that we want and we select 10 to start. A quick barplot of the "sk" values from the SVD model demonstrates that the majority of the variance is captured in the first several data points.

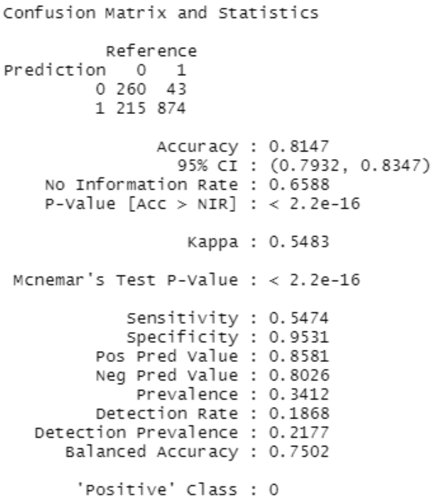


Since we do not gain much additional information beyond the 6th bar/data point, we can cut it off here and only use the first 6 variables from the SVD model to incorporate with our original data set. We prefer to add as few variables as possible without losing too much variance and we are comfortable that 6 variables will be sufficient for the next step after looking at the plot above. We then join these SVD variables with the original Kaggle dataset using the uri field. We know that there will be fewer observations due to NA values since we did not have lyrics for every song.

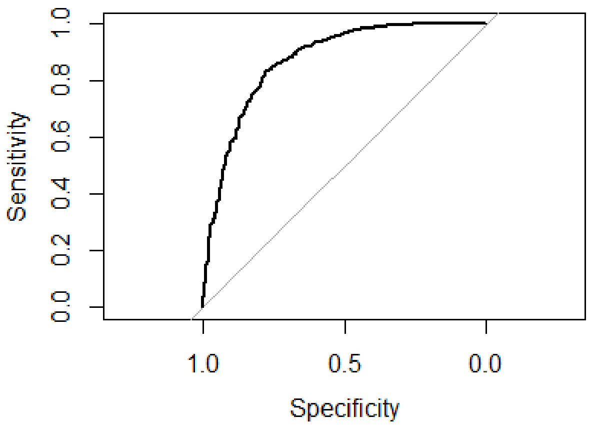
The full logistic regression model experienced the highest accuracy and AUC score. The results for this model can be seen below and the AUC from ROC curve is 0.8721. As mentioned earlier, the results from this model do not differ much from the original logistic regression model which indicates the 6 new variables have minimal effect on the model overall.

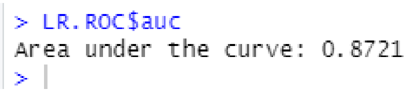


```
    Min        1Q    Median        3Q       Max  
-7.4175   -0.2123    0.4516    0.6762    3.1555  

Coefficients:
                     Estimate Std. Error z value Pr(>|z|)    
(Intercept)         4.144e+00  8.593e-01   4.823 1.41e-06 ***
V1                  1.030e+00  5.347e-01   1.927 0.053948 .  
V2                  2.511e-02  2.953e-01   0.085 0.932232    
V3                 -7.755e-02  5.511e-02  -1.407 0.159330    
V4                  8.646e-02  4.324e-02   2.000 0.045535 *  
V5                 -1.350e-03  2.879e-02  -0.047 0.962607    
V6                 -1.362e+00  2.120e-01  -6.423 1.34e-10 ***
danceability        3.762e+00  3.961e-01   9.498  < 2e-16 ***
energy             -3.164e+00  4.662e-01  -6.788 1.14e-11 ***
key1               -1.132e-01  2.030e-01  -0.558 0.576994    
key2               -1.327e-01  2.163e-01  -0.614 0.539418    
key3               -2.557e-01  2.873e-01  -0.890 0.373382    
key4                2.068e-02  2.402e-01   0.086 0.931391    
key5                1.286e-01  2.260e-01   0.569 0.569151    
key6                5.557e-02  2.233e-01   0.249 0.803504    
key7               -8.083e-02  2.059e-01  -0.393 0.694674    
key8                3.293e-02  2.286e-01   0.144 0.885442    
key9               -9.047e-02  2.170e-01  -0.417 0.676758    
key10               2.104e-01  2.410e-01   0.873 0.382578    
key11              -3.685e-01  2.065e-01  -1.784 0.074351 .  
loudness           -4.380e+00  4.908e-01  -8.924  < 2e-16 ***
mode1              -1.356e-03  1.070e-01  -0.013 0.989893    
speechiness        -5.967e-01  1.571e-01  -3.798 0.000146 ***
acousticness       -5.788e-01  2.361e-01  -2.451 0.014233 *  
instrumentalness -5.691e+00  4.405e-01 -12.920  < 2e-16 ***
valence            -7.296e-01  2.597e-01  -2.810 0.004956 ** 
tempo               6.051e-03  1.639e-03   3.692 0.000222 ***
duration_ms        -2.491e-06  9.097e-07  -2.739 0.006171 ** 
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)
```



```
Confusion Matrix and Statistics

          Reference
Prediction   0   1
         0 260  43
         1 215 874
                                          
               Accuracy : 0.8147          
                 95% CI : (0.7932, 0.8347)
    No Information Rate : 0.6588          
    P-Value [Acc > NIR] : < 2.2e-16       
                                          
                  Kappa : 0.5483          
                                          
 Mcnemar's Test P-Value : < 2.2e-16       
                                          
            Sensitivity : 0.5474          
            Specificity : 0.9531          
         Pos Pred Value : 0.8581          
         Neg Pred Value : 0.8026          
             Prevalence : 0.3412          
         Detection Rate : 0.1868          
   Detection Prevalence : 0.2177          
      Balanced Accuracy : 0.7502          
                                          
       'Positive' Class : 0               
```





```
> LR.ROC$auc
Area under the curve: 0.8721
> 
```

Suggestions

We suggest that more time be invested into finding a source that contains all lyrics in the dataset. The Genius API only contained about ⅔ of the songs in the dataset, most of which were hits, which resulted in a class imbalance when creating models with text-based variables.

Conclusion

We chose this dataset from Kaggle because we are all familiar with music and hit songs from various decades. The variables presented made sense, for the most part, and it was an interesting challenge to determine the influence of each independent variable on our Target. We also realized that we could grab the lyrics for the songs and use text analytics to add another layer to our analysis. Our findings were mixed as detailed above and we learned several valuable lessons from each analysis.

The first major takeaway from our various modeling; feeling and emotion tend to be stronger indicators for a hit song than the lyrics. Before layering in any text analytics into our models we found that we could predict hit songs at a reasonable accuracy using the variables from the Kaggle dataset. The effect of including any SVD variables was marginal at best and they were not as significant compared to other variables. This would lead us to believe we should keep the Text Analytics variables out of our models in order to reduce dimensionality.

We can also learn lessons from our models' results and especially from the original dataset variables which we find to be more descriptive and more influential than the lyrics, as discussed. Variables like Danceability and Tempo are positive significant variables whereas

Energy, Loud and Instrumentalness are negative significant variables. This indicates that there is a certain balance that must be achieved by artists. Songs must be catchy and fun without being too loud or energetic.

References

https://www.kaggle.com/theoverman/the-spotify-hit-predictor-dataset?select=dataset-of-10s.csv