MMA867 Assignment 2 Technical Report

Jacqueline Mak
Student #: 20311028

Fit a logistic regression model using all variables

Before I begin to build a logistic regression model using all variables, I split the the dataset into a training set ‘SongsTrain’ that consists of all the observations up to and including 2009 song releases and a test set ‘SongsTest’ that consists of the 2010 song release. I then excluded some of the variables in the dataset from being used independent variables which are “year”, “songtitle”, “artistname”, “songID” and “artistID” because I want to use only continuous variables in my model.

Below is a logistic regression model using all continuous variables from the Music dataset. In this model, we can say that for a one-unit increase in loudness, we expect to see about 34% increase in the odds of the song to hit top 10 since exp(0.29987940) = 1.34

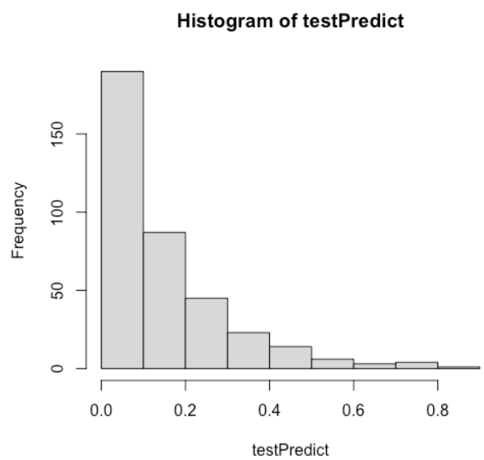
```
Call:
glm(formula = Top10 ~ ., family = binomial, data = SongsTrain)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.9220  -0.5399  -0.3459  -0.1845   3.0770

Coefficients:
                            Estimate   Std. Error z value             Pr(>|z|)
(Intercept)              14.69998823   1.80638746   8.138 0.000000000000000403 ***
timesignature             0.12639483   0.08673566   1.457             0.145050
timesignature_confidence  0.74499227   0.19530526   3.815             0.000136 ***
loudness                  0.29987940   0.02916535  10.282 < 0.0000000000000002 ***
tempo                     0.00036340   0.00169146   0.215             0.829889
tempo_confidence          0.47322705   0.14217401   3.329             0.000873 ***
key                       0.01588199   0.01038950   1.529             0.126349
key_confidence            0.30867509   0.14115620   2.187             0.028760 *
energy                   -1.50214447   0.30992402  -4.847 0.000001254591330988 ***
pitch                   -44.90773986   6.83488314  -6.570 0.000000000050188970 ***
timbre_0_min              0.02315894   0.00425625   5.441 0.000000052933134209 ***
timbre_0_max             -0.33098196   0.02569259 -12.882 < 0.0000000000000002 ***
timbre_1_min              0.00588100   0.00077981   7.542 0.000000000000046437 ***
timbre_1_max             -0.00024486   0.00071524  -0.342             0.732087
timbre_2_min             -0.00212741   0.00112599  -1.889             0.058843 .
timbre_2_max              0.00065857   0.00090658   0.726             0.467571
timbre_3_min              0.00069196   0.00059845   1.156             0.247583
timbre_3_max             -0.00296730   0.00058149  -5.103 0.000000334457039019 ***
timbre_4_min              0.01039562   0.00198505   5.237 0.000000163238506711 ***
timbre_4_max              0.00611050   0.00155029   3.942 0.000080967043288844 ***
timbre_5_min             -0.00559796   0.00127670  -4.385 0.000011614677389716 ***
timbre_5_max              0.00007736   0.00079354   0.097             0.922337
timbre_6_min             -0.01685618   0.00226395  -7.445 0.000000000000096605 ***
timbre_6_max              0.00366807   0.00218950   1.675             0.093875 .
timbre_7_min             -0.00454922   0.00178148  -2.554             0.010661 *
timbre_7_max             -0.00377369   0.00183198  -2.060             0.039408 *
timbre_8_min              0.00391105   0.00285101   1.372             0.170123
timbre_8_max              0.00401134   0.00300298   1.336             0.181620
timbre_9_min              0.00136726   0.00299806   0.456             0.648356
timbre_9_max              0.00160266   0.00243364   0.659             0.510188
timbre_10_min             0.00412631   0.00183907   2.244             0.024852 *
timbre_10_max             0.00582498   0.00176941   3.292             0.000995 ***
timbre_11_min            -0.02625234   0.00369327  -7.108 0.000000000001175988 ***
timbre_11_max             0.01967338   0.00338549   5.811 0.000000006206866068 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6017.5  on 7200  degrees of freedom
Residual deviance: 4759.2  on 7167  degrees of freedom
AIC: 4827.2

Number of Fisher Scoring iterations: 6
```

Predict the popularity of records in the testing set

Please see CSV file called Predicted Popularity of records. The image below on the left is a snippet of the predicted values. For example, the first song has only a 3% chance of hitting top 10.

Predicted Popularity of

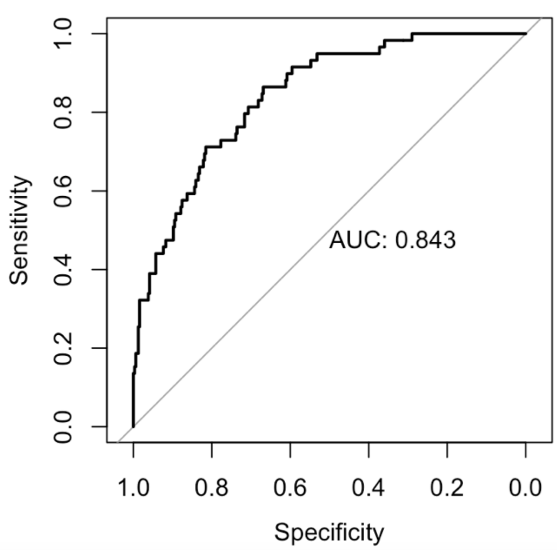
	testPredict
1	0.0328564215706949
2	0.0318610976699904
3	0.0380681345183234
4	0.0660660252609543
5	0.00250255183519714
6	0.0266287723968323
7	0.0544253486124285
8	0.00998124017576679
9	0.0293383885499284
10	0.363778932658368



Using a histogram as shown above to visualize the predicted probability for each song, we can see that the data is heavily skewed. The vast majority of the songs will not hit top 10 and the frequency seems to decrease with a small number of songs that are above 80%. I would recommend to invest in the songs that have a higher probability to hit top 10.

Generate the ROC curve

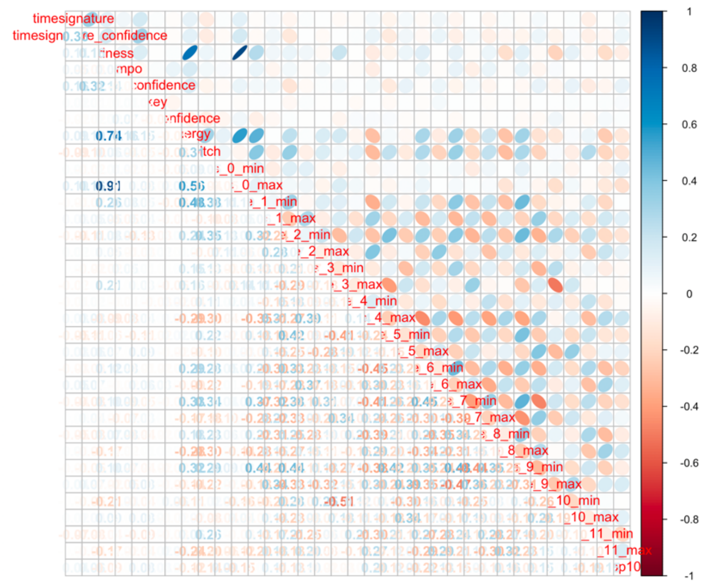
Below is the ROC curve where we can analyze the sensitivity and specificity. This curve depends on the model and it generated by many different thresholds from 0 to 1. The AUC is 0.843; the higher the AUC, the better the model is.



Improve the prediction performance of the model

Apply PCA to Logistic Regression

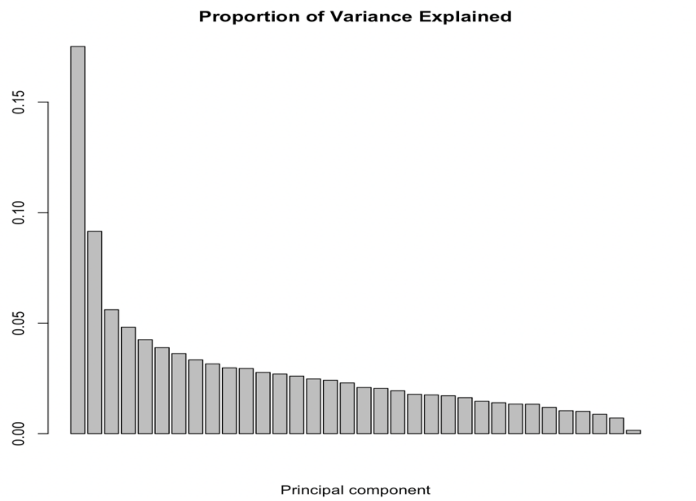
The first approach to improve the prediction performance is to use Principal Component Analysis (PCA). It is a method that captures the important variables in form of components from a dataset where most of the variables are highly correlated (multicollinearity). As you can see from the corrplot below, many of the variables are correlated. PCA also works the best with continuous variables, so I removed the categorical variables. Since PCA is a tool for exploring historical data, I did not split the dataset into training and testing set when exploring the data.



Below is the summary of the results when PCA has been applied. The first principal component only explains 17.52% of the variance. The first 8 PCAs count for 52.2% of the variance, which is approximately half of the dataset.

```
Importance of components:
                          PC1     PC2     PC3     PC4     PC5     PC6     PC7
Standard deviation     2.4404 1.76428 1.38102 1.27952 1.20216 1.15045 1.10995
Proportion of Variance 0.1752 0.09155 0.05609 0.04815 0.04251 0.03893 0.03623
Cumulative Proportion  0.1752 0.26671 0.32281 0.37096 0.41347 0.45239 0.48863
                           PC8     PC9    PC10    PC11    PC12    PC13    PC14
Standard deviation     1.06543 1.03605 1.00636 1.00139 0.97064 0.95745 0.94076
Proportion of Variance 0.03339 0.03157 0.02979 0.02949 0.02771 0.02696 0.02603
Cumulative Proportion  0.52201 0.55359 0.58337 0.61287 0.64058 0.66754 0.69357
                          PC15    PC16    PC17   PC18    PC19    PC20    PC21
Standard deviation     0.91757 0.90621 0.88339 0.8430 0.83448 0.81246 0.77771
Proportion of Variance 0.02476 0.02415 0.02295 0.0209 0.02048 0.01941 0.01779
Cumulative Proportion  0.71833 0.74248 0.76544 0.7863 0.80682 0.82623 0.84402
                          PC22    PC23    PC24    PC25    PC26    PC27    PC28
Standard deviation     0.77223 0.76364 0.74388 0.70576 0.68972 0.67473 0.67318
Proportion of Variance 0.01754 0.01715 0.01628 0.01465 0.01399 0.01339 0.01333
Cumulative Proportion  0.86156 0.87871 0.89499 0.90964 0.92363 0.93702 0.95035
                          PC29    PC30    PC31    PC32    PC33    PC34
Standard deviation     0.63564 0.59373 0.58516 0.54556 0.48955 0.22782
Proportion of Variance 0.01188 0.01037 0.01007 0.00875 0.00705 0.00153
Cumulative Proportion  0.96223 0.97260 0.98267 0.99142 0.99847 1.00000
```

I also checked the weights of the principal components. For example, PC2 is positively related with loudness, if loudness is high then PC2 is high. I then needed to decide the number of components to use for the modelling stage. The answer to this question can be illustrated with the following graph which explains the most variability in the dataset. It is somewhat subjective when choosing the number of components. However, there is a clear break after the 3rd or 4th component because they have much larger variances than the rest of the components. For that reason, so I will choose the first 4 components as predictor variables for my model.

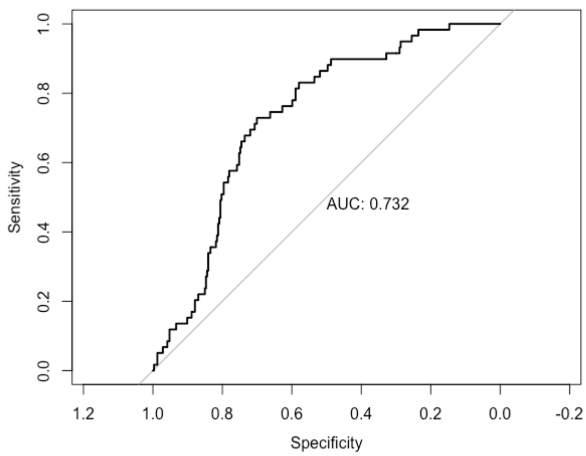


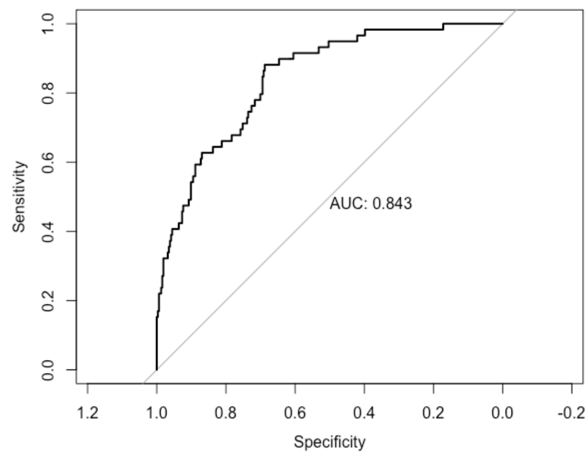
I begin my predictive modelling by building new datasets with the new PCA. As mentioned previously, I chose the first 4 components and I put back the dependent variables ‘Top10’ to the training set. I applied the same fix to the test set and then I fit the logistic regression to the training set and made predictions using the test set. The following is the output of the model.

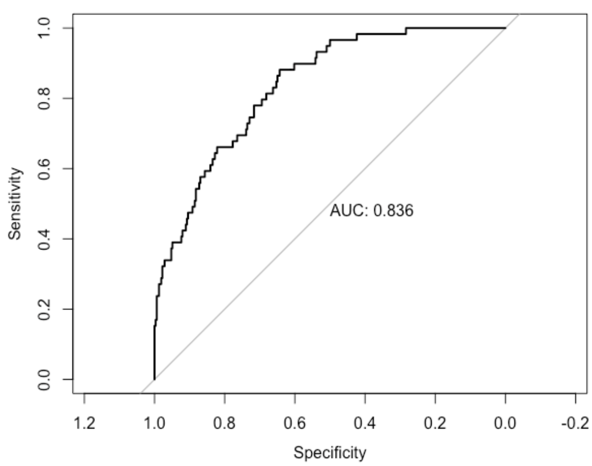
```
Call:
glm(formula = Top10 ~ ., family = binomial, data = training_set_pca)

Deviance Residuals:
    Min       1Q   Median       3Q      Max
-1.7185  -0.6139  -0.4569  -0.2988   2.6275

Coefficients:
                Estimate Std. Error z value             Pr(>|z|)
(Intercept)     -3.33293    0.09656 -34.515 < 0.0000000000000002 ***
`Top10.1 comps` -0.89270    1.64909  -0.541              0.58828
`Top10.2 comps` -0.68677    1.89734  -0.362              0.71738
`Top10.3 comps`  9.41078    3.56890   2.637              0.00837 **
`Top10.4 comps`  1.35896    3.20692   0.424              0.67174
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

(Dispersion parameter for binomial family taken to be 1)

    Null deviance: 6017.5  on 7200  degrees of freedom
Residual deviance: 5515.8  on 7196  degrees of freedom
AIC: 5525.8

Number of Fisher Scoring iterations: 5
```

Below is the ROC (top left diagram) curve generated using the logistic regression with PCA. I compared this output with the base model which is using logistic regression with all variables. Although the AUC obtained by applying PCA has been reduced to 0.732, the dataset only captures approximately 37.1% of the variance. Therefore, PCA has improved the model and eliminated the multicollinearity.

As an experiment, I chose the first 20 components and compared the AUC to the results of the base model. The AUC (top right diagram) is exactly 0.843 as the base model which means that PCA did indeed improve the model because it only captured 82.6% variability in the dataset. However, as another experiment, when I chose the first 25 components, the AUC (bottom left diagram) dropped to 0.836 because the remaining components are somewhat similar and likely to be statistical noise.

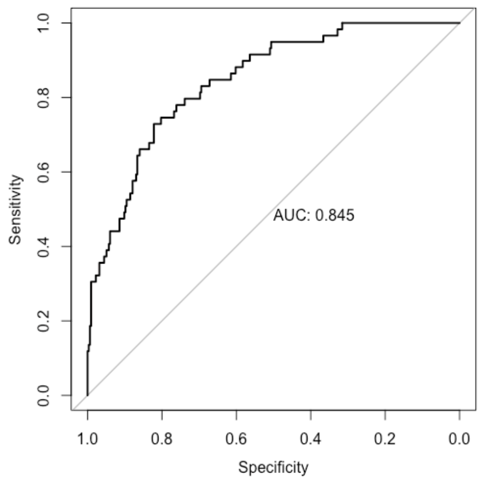






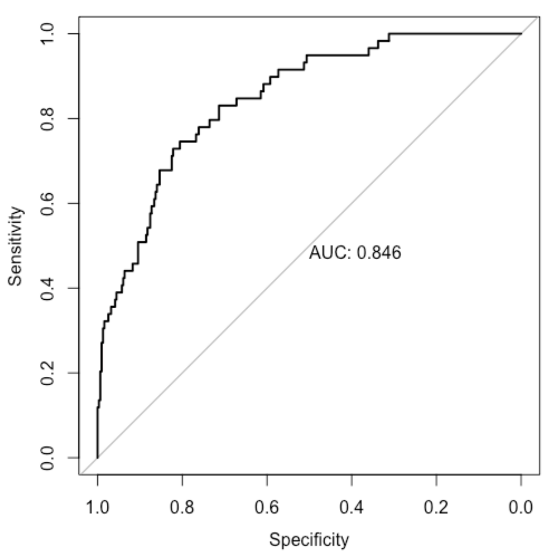
Logistic regression with selected variables & Feature Engineering

The second approach is to run the logistic regression model with selected variables. I used the TTT approach which is going from the top down to filter out the variables. I first ran the model with all of the variables and then only included all of the independent variables that are statistically significant with p-value less than 0.03. I then recorded the AUC from the ROC model and compared it with the base model where all of the variables are included in the model. With the collinear variables, I excluded either one of the pair in the model. For example, 'timesignature_confidence' and 'timesignature' are positively correlated hence I removed the variable 'timesignature'. The AUC resulted from this model is 0.845.



Further, I added an interaction feature between tempo confidence and pitch because it is fair to assume that the pitch of the song is an important variable when combined with the confidence in the estimated beats per minute of the song. I also added an interaction feature between tempo confidence and energy because the overall acoustic energy of the song should have a synergy effect with the confidence in the estimated beats per minute of the song.

Below is the ROC curve for the logistic regression model with significant variables with p-value less than 0.03 combined with feature engineering. The AUC has improved to 0.846 which is the highest when compared to the previous models. For that reason, this is the final model I will proceed with.



Interpretation of all model coefficients of the final model

Below is the final model using logistic regression with selected variables and feature engineering.

```
Coefficients:
                          Estimate  Std. Error z value             Pr(>|z|)    
(Intercept)             15.5940515   1.7062660   9.139 < 0.0000000000000002 ***
timesignature_confidence  0.7905007   0.1885658   4.192 0.000027629330605138 ***
loudness                  0.3008057   0.0283777  10.600 < 0.0000000000000002 ***
tempo_confidence          0.5721554   0.3727839   1.535             0.124829    
energy                   -1.5907116   0.4670027  -3.406             0.000659 ***
pitch                   -31.4835555  13.7001042  -2.298             0.021559 *  
timbre_0_min              0.0233706   0.0041622   5.615 0.000000019667694969 ***
timbre_0_max             -0.3379537   0.0248564 -13.596 < 0.0000000000000002 ***
timbre_1_min              0.0055863   0.0006940   8.049 0.000000000000000835 ***
timbre_3_max             -0.0038252   0.0004995  -7.659 0.000000000000018774 ***
timbre_4_min              0.0100586   0.0018855   5.335 0.000000095640132039 ***
timbre_4_max              0.0066188   0.0013771   4.806 0.000001536370777311 ***
timbre_5_min             -0.0067812   0.0011959  -5.670 0.000000014245553907 ***
timbre_6_min             -0.0171189   0.0021184  -8.081 0.000000000000000641 ***
timbre_10_max             0.0059928   0.0016695   3.590             0.000331 ***
timbre_11_min            -0.0281343   0.0035669  -7.888 0.000000000000003080 ***
timbre_11_max             0.0219488   0.0031826   6.897 0.000000000005327517 ***
tempo_confidence:energy   0.1971874   0.5851917   0.337             0.736146    
tempo_confidence:pitch  -26.3835281  19.2219205  -1.373             0.169884    
```

In the model above, the estimated coefficient for the intercept represents the log odds of a song with the rest the variables being zero. We can start with interpreting the confidence in time signature of the song. The coefficient for 'timesignature_confidence' says that there is 120% increase in the odds of the song hitting Top 10 for a one-unit increase in the confidence of time signature score because exp(0.7905007) = 2.204. For the loudness coefficient, every one-unit increase in the average amplitude of the audio, there is about exp(0.3008057) = 1.35 which is 35% increase in the odds of the song hitting Top 10. For the confidence in the estimated beats per minute of the song (tempo_confidence), if we increase it by one unit, we are expected to see about 77% increase in the odds of the song hitting top 10, since exp(0.5721554). For the energy coefficient, the odds ratio of the song hitting top 10 with an additional unit in energy is 0.204 times lower, since exp(-1.5907116) = 0.20378. For the pitch coefficient, if we increase it by one unit, the odds ratio of the song hitting top 10 with an additional unit in pitch is 2.1225 x10^-14 times lower, since exp(-31.483555).

Since there are many variables that are related to minimum/maximum values in the timbre vector in the model, I will explain 'timbre_0_min' and 'timbre_3_max' coefficients. For the 'timbre_0_min' coefficient, if we increase it by one unit, we are expected to see about 2.4% increase in the odds of the song hitting Top 10, since exp(0.0233706) = 1.024. For the 'timbre_3_max', the odds ratio of the song hitting top 10 with an additional unit in the maximum value in the timbre vector is 0.996 times lower, since exp(-0.0038252) = 0.996.

Lastly, for the interaction variable between tempo confidence and energy, if we increase it by one unit, we are expected to see about 22% increase in the odds of the song hitting Top 10, since exp(0.1971874) = 1.2179. For the interaction variable between tempo confidence and pitch, the odds ratio of the song hitting top 10 with an additional unit is exp(-26.3835) times lower.