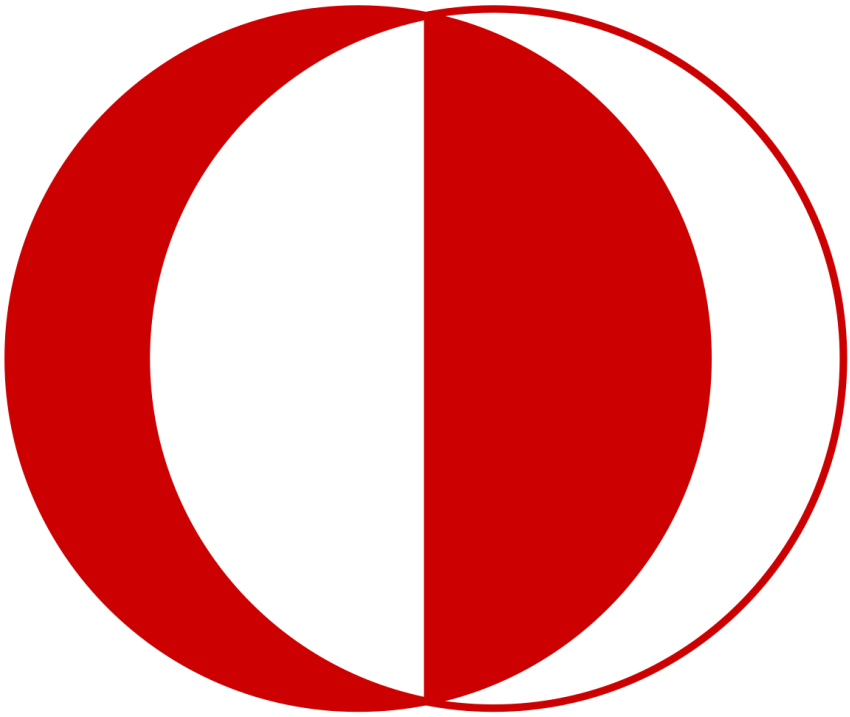

Spotify & YouTube Data Research

PROJECT REPORT SUBMITTED
IN FULFILMENT OF THE REQUIREMENTS FOR THE COURSE
STAT 412 – Statistical Data Analysis

DEPARTMENT OF STATISTICS OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

Efe Uslu

March 2024

ABSTRACT

The dataset gives information about identifying variables that create and index for each of the danceability, instrumentalness and properties like these for tracks on YouTube and Spotify. The data is preprocessed and analyzed according to the research questions.

1. Introduction

The data has been used to determine the possible components that affect streams. The aim is to find and use the most comforting variables for these said responses after doing EDA and data tidying. Some machine learning algorithms will be used to determine the possible effects of some variables on the overall popularity of a song. For these methods RStudio application is used. Also, creating graphs is done through this application.

1.1. Data description and tidying

The data is collected through APIs of YouTube and Spotify, and the following variables from the 27 variables provided by the dataset are used: Album type, Music property indexes, License status, Official video status. There are also 20718 observations in the dataset.

Upon closer examination it was determined that every data type was correct, the format was same throughout the dataset (including the column names), unnecessary columns regarding the analysis were dropped and finally, outliers were removed because they were usually a result of an unwanted situation (i.e. remixes, kid's songs, podcasts). Also, a log transformation was implemented on the response variable, "Stream" as stream count was too high for popular songs. To make it easier to interpret the variables, descriptive graphs are provided below.

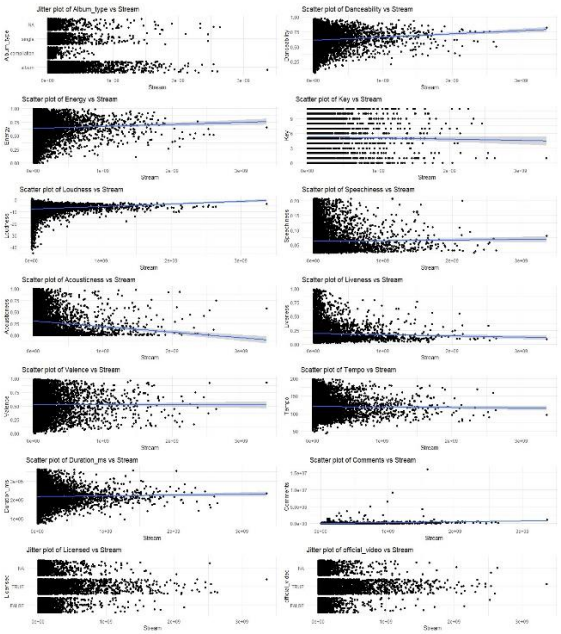

Figure 1: Indexes against the response, Stream.

It seems like there exists positive relationship regarding Stream counts for Danceability, Energy, Speechiness and Duration while there seems to be negative relationship for Key, Acousticness and Liveness. While it is fit a linear line on these graphs, there most probably does not exist a linear relation as such. i.e. the optimum value to make the Stream count maximum lies somewhere in between the max and min value of the given variable. This interaction will be further examined in the analysis part of the report.

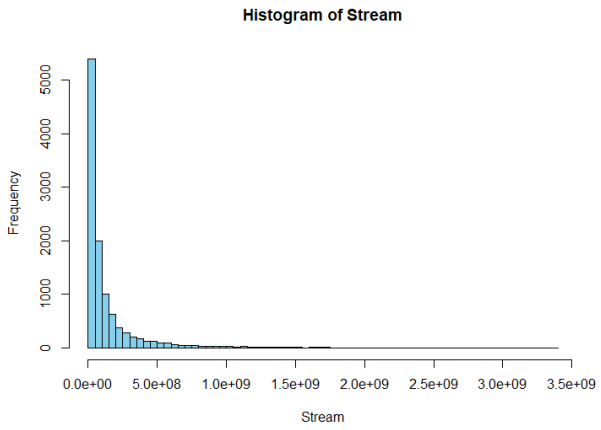

Figure 2: Distribution of Stream.

Distribution of the Stream variable (the target variable) seems to exhibit an exponentially increasing pattern where there are least number of popular songs and where the popular songs have exponentially more Stream count. Thus, a log transformation was implemented, and the resulting distribution looks more like below.

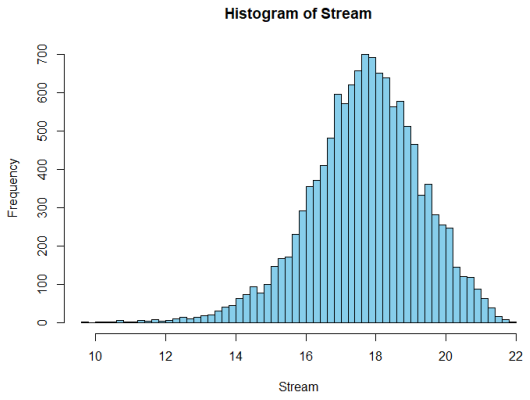

Figure 3:Distribution of Stream after the transformation

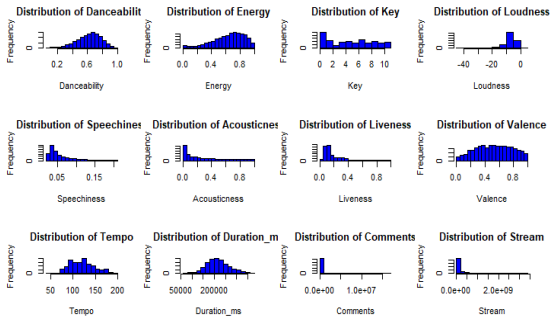

Figure 4: General information about predictors.

After looking at the distributions of the predictor variables, it is noted that while some of them exhibit a normal-like distribution, they are mostly skewed distributions, and this will be dealt with during the analysis part of the report.

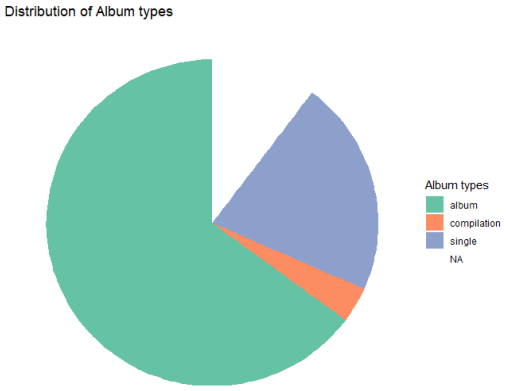

Figure 5: Distribution of Album types.

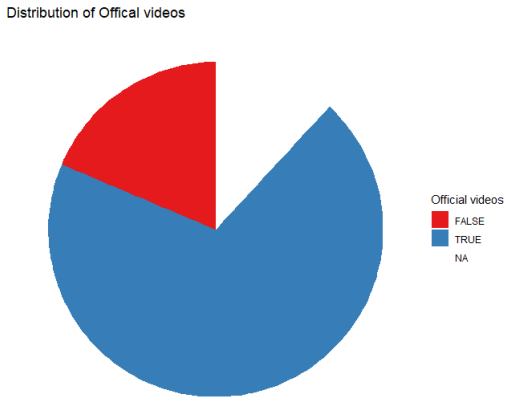

Figure 6: Distribution of Official videos.

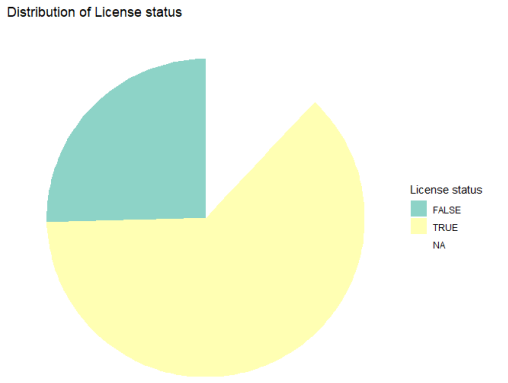

Figure 7: Distribution of License status.

While these categorical variables are not equal in terms of their categories, it is decided that this can be kept this way as the dataset in general is very large and the

inequality in these categories won't have any negative impact on the result.

1.2. Research Questions raised on EDA part of the analysis.

1.2.1. Are there any anomalies existent in our predictors?

According to the distribution plots below, most predictors exhibit a normal behavior with some of them being skewed and some being normally distributed (also key seems to have a uniform distribution). But after checking comment variable, we see that a log transformation and then an outlier detection might be appropriate, but after some consideration, it is decided to drop this variable all together as it is just a reflection of Stream count and plays no part in predicting the Stream count before the release as it won't be available.

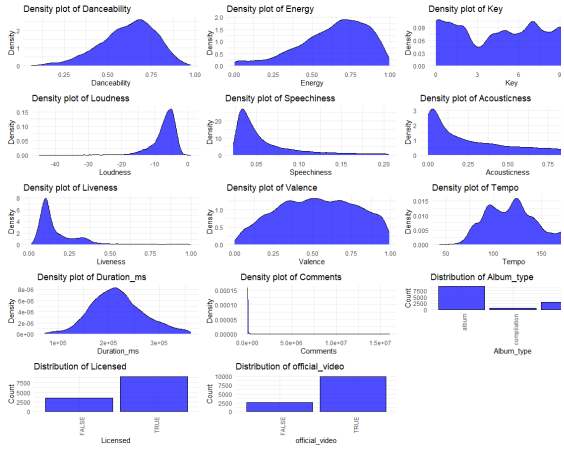

Figure 8: Density plots of the distributions of variables

1.2.2. Is a non-linear model more appropriate to show the relationship between predictors and the target variable? Where do the optimum values that maximize the target variable lie?

	Variable	Linear_AIC	Nonlinear_AIC	Better_Model
1	Danceability	-10431.7748	-10431.7803	Nonlinear
2	Energy	-2751.4277	-2751.4473	Nonlinear
3	Key	67705.7990	67704.5930	Nonlinear
4	Loudness	73405.7110	73405.7080	Nonlinear
5	Speechiness	-45059.2647	-45059.8089	Nonlinear
6	Acousticness	4362.9405	4362.9366	Nonlinear
7	Liveness	-10031.3554	-10031.4016	Nonlinear
8	Valence	579.4838	578.4024	Nonlinear
9	Tempo	119206.3060	119204.5553	Nonlinear
10	Duration_ms	309014.6983	309014.6853	Nonlinear

Figure 9: AIC scores of variables

With a lower Akaike Information Criterion (AIC) score for every feature, we see that fitting a non-linear line to them for prediction always makes more sense.

1.2.3. Is there a high correlation between predictors?

Firstly, a correlation matrix is created across all variables then the VIF value for each feature is calculated to assess any multicollinearity problem that could be encountered early on.

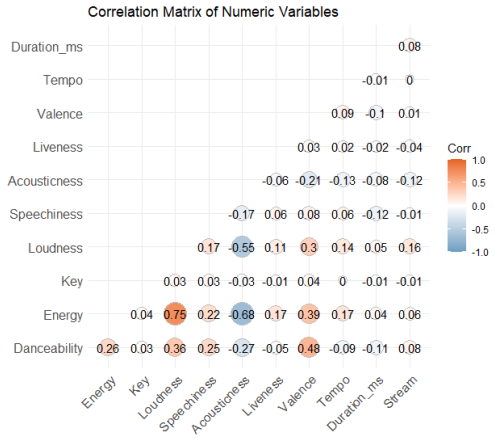

Figure 10: Correlation matrix of numeric variables

Danceability	Energy	Key	Loudness	Speechiness
1.637246	3.524002	1.003126	2.491077	1.133883
Acousticness	Liveness	Valence	Tempo	Duration_ms
1.990316	1.051424	1.546723	1.073426	1.050432

Figure 11:VIF scores of variables

As there are no VIF values that are higher than 3.6, it is claimed that there is no multicollinearity problem in the dataset.

1.2.4. Is License status effective for the Stream count?

First, a single violin plot is constructed to visualize the relation better.

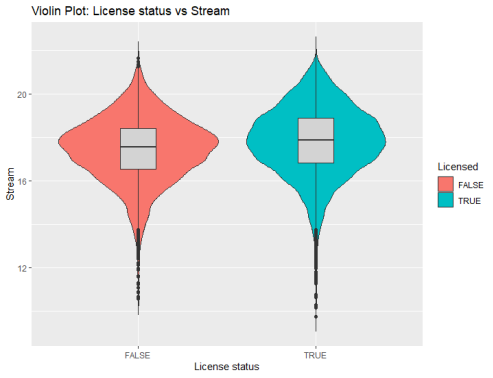

Figure 12: Violin plot of License against Stream.

The graph does not show a clear relationship although Licensed song might have a slightly more Stream count.

From a Wilcox test with H_0: $\mu_F = \mu_T$ & H_1: $\mu_F \neq \mu_T$ it is concluded with a p-value of <0.001 we reject the null hypothesis and conclude that there exists a significant relationship that indicates the Licensed songs have overall more stream count.

1.2.5. Does the Album type affect the Stream count?

A violin graph of Stream count against Album type is constructed.

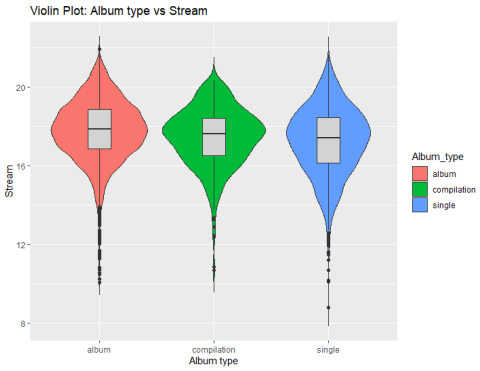

Figure 13:Violin plot of Album type against Stream

The graph shows a slight relationship, but a Kruskal test was formed to formally declare a relationship. Hypothesis are constructed as:

H_0: μa = μc = μs, H_1: at least one of the means differ.

With a p-value < 0.01 on Kruskal test, the null hypothesis is rejected, and it is concluded that the Album types are significantly effective for the Stream count. To assess which album types are significant, post hoc analysis using Dunn test was performed and it is determined that every group is significant with Album typed songs having higher Stream counts followed by compilations and then singles.

1.3. Missingness Mechanism and Imputation

To identify the pattern of missing values, gg_miss_upset function from "naniar" package was used. The following graphs are generated:

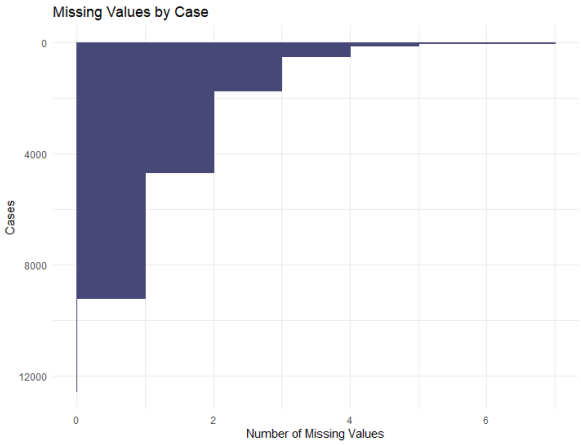

Figure 14: Missing value count

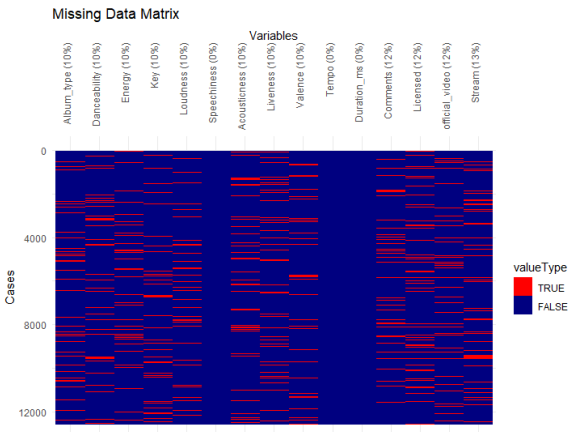

Figure 15: Missing data matrix

Upon inspection of these graphs, it is seen that the missing data is MCAR. To overcome this missing data problem, as the data was MCAR, it was decided to use an imputation method. Finally, it was decided that the best way would be to use PMM for numeric variables and k-nearest for the imputation for categorical variables. After using the PMM and k-nearest methods, every missing value was replaced with an approximate value and the remainder of the data was kept strengthening the analysis.

Then, to check whether the imputation method changed the distributions of the features, their density plots were examined

and a Kolmogorov-Smirnov (K-S) test was conducted to test the hypothesis; H_0: $F(x)=G(x)$ against H_1: $F(x)\neq G(x)$ where $F(x)$ is the cdf of a variable before the imputation and $G(x)$ is the variables cdf after the imputation. The said graphs and tests are provided below.

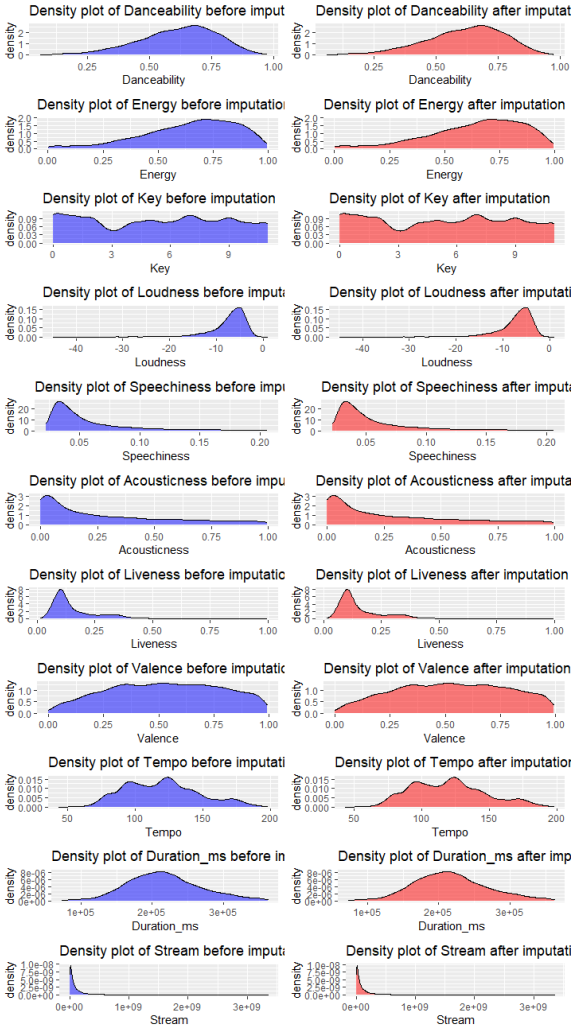

Figure 16: Comparision of distributions with imputation

	Variable	Statistic	P_Value
D	Danceability	0.0036738168	0.9999980
D1	Energy	0.0021737213	1.0000000
D2	Key	0.0014359381	1.0000000
D3	Loudness	0.0014789604	1.0000000
D4	Speechiness	0.0040534096	0.9999489
D5	Acousticness	0.0029127874	1.0000000
D6	Liveness	0.0100936584	0.5783219
D7	Valence	0.0028932799	1.0000000
D8	Tempo	0.0001589572	1.0000000
D9	Duration_ms	0.0002384359	1.0000000
D10	Stream	0.0014308466	1.0000000

Figure 17: Test results for whether the distributions changed

As seen from the graphs, the distributions of the features are kept the same and as the p-values are less than 0.01, it is concluded that the null hypothesis is rejected and there is no significant difference between the distributions of the features.

1.4.Aim of the study

The aim of the study is to predict a songs Stream count from the indexes provided by Spotify and YouTube's API. The described variables are analyzed to find the said impact. A statistical model, Random Forest model, Neural Network model, Support Vector Machines model and a XGboost model were used to model the predictions and the best model was to be picked.

2. Analysis

*The variables were standardized before building the models.

2.1. Statistical Model with k-fold Cross Validation

Firstly, a comparison was made between linear and polynomial models to see the effectiveness of each model.

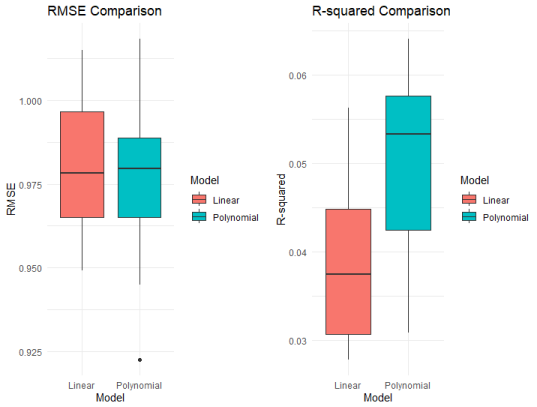

Figure 18: Comparision of Linear & Polynomial models

From this graph, it was seen that the overall performance of a polynomial model was more suitable for the dataset at hand. Thus, a polynomial model was selected. Following table provides the coefficients for the linear and quadratic values and their interpretations:

Variable	Linear	Quadratic
Danceability	Increase (3.450865e+00)	Accelerates Increase (5.533218e-01)
Key	Decrease (-2.209848e+00)	Accelerates Increase (4.482916e-01)
Speechiness	Decrease (-1.766478e+00)	Accelerates Decrease (-2.374595e+00)
Acousticness	Decrease (-1.115800e+01)	Accelerates Decrease (-7.029793e-02)
Energy	Decrease (-2.053634e+01)	Accelerates Increase (3.943046e-01)
Valence	Decrease (-3.087851e+00)	Accelerates Decrease (-2.399602e+00)
Duration_ms	Decrease (-1.104106e+01)	Accelerates Increase (8.308676e+00)
Loudness	Decrease (-2.180190e+01)	Accelerates Increase (4.900162e+00)
Liveness	Decrease (-4.261780e+00)	Accelerates Decrease (-1.033967e+00)
Tempo	Decrease (-3.502434e-02)	Accelerates Increase (2.017610e+00)

Figure 19: Model formula interpretation

RMSE	Rsquared	MAE
0.9750434	0.04977421	0.7580468

Figure 20: Polynomial models scores

This table shows that, i.e. an increase in Key results in a decrease in Stream while a more accelerated increase in Key results in a decrease in the slope. Also, an MSE score of 0.95070963188356 was achieved. The following graph shows the feature importances.

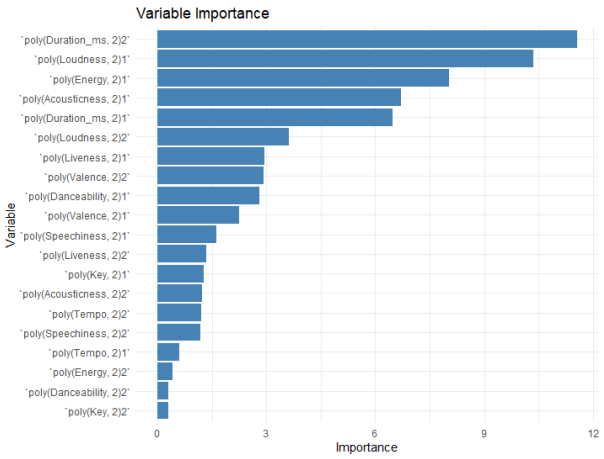

Figure 21: Variable importance of Polynomial model

This graph shows that accelerated increase (change of slope) of duration and increase in loudness were the most important variables in the model while Key(2,2) and Danceability(2,2) were the least important.

2.2. Random Forest Algorithm with k-fold Cross Validation

randomForest and caret packages are used to construct the Random Forest model. Predictions were saved and 500 trees were created. A grid search for the best mtry hyperparameter was conducted as below.

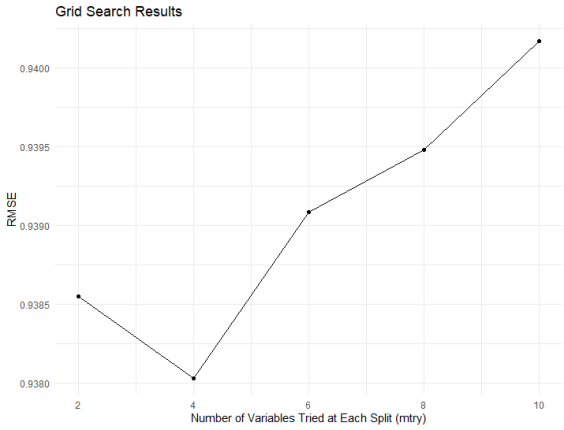

Figure 22: Grid search results of Random Forest model

As the mtry hyperparameter minimized at 4, it was picked to be the best hyperparameter. The following results were then obtained for the best model.

The following results were obtained:

Cross-validated Mean Squared Error: 0.8775752

Cross-validated R-squared: 0. 1224

A MSE as low as 0.878 was obtained. This value indicates that the square of prediction errors is approximately 0.878, lower values of MSE is usually better but within the context of the dataset it is fine to assume that a good MSE value was obtained due to Stream variable being in the range [1,22].

Also, the R-squared value indicates that the constructed model explains 12% of the variance in the Stream variable. This means that the constructed model is good for 12% of the variability while the remainder of the 88% variability is due to some other variables outside of the starting dataset.

Finally, a Variable Importance graph was generated with the values of each predictor for the effect they have on the target variable. The outputs are as follows:

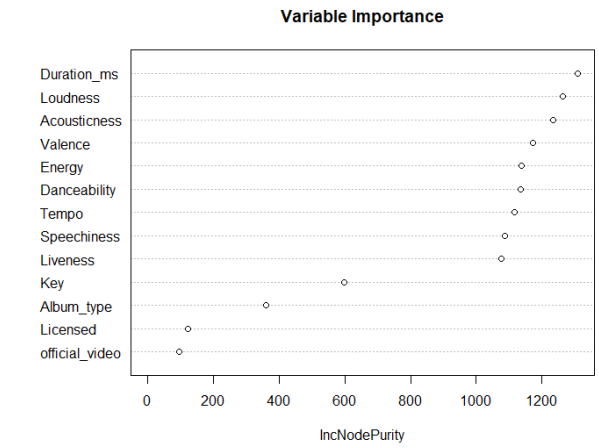

Figure 23: Variable importance of Random Forest model

These values indicate the importance of the variables on the target variable, Stream. From these graphs, we see that the worst predictors in the model are license status, compilation album type, official video, single album type and key. Every other variable fall in the same range and approximately has the same effect on the target variable.

Also, the 4 most effective variables' Partial Dependence Plots were constructed to visualize the optimum values for the predictors to maximize the target variable.

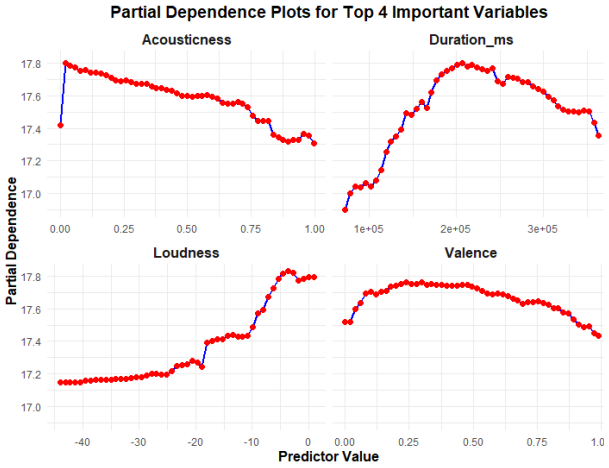

Figure 24: PDP for the 4 best variables

As noted, before in the report, all four of these variables seem to have a polynomial relation with the target variable.

2.3.Neural Network Algorithm with k-fold Cross Validation

nnet and caret functions are used to construct a Neural Network model. Grid search to find the best hyperparameters for the model (decay & size) was conducted and the following results are obtained.

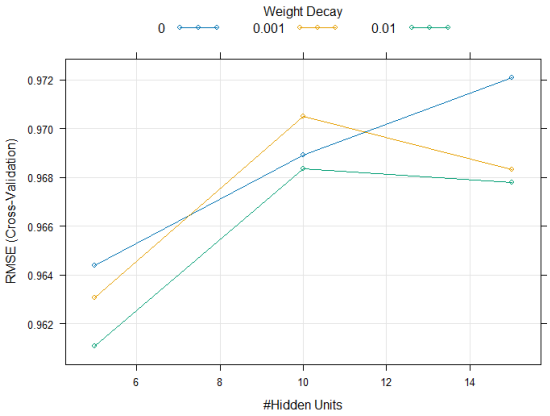

Figure 25: Grid search results of Neural Network model

The graph suggests that the best value for decay is 0.01 while the best value for size is 5. The MSE for the best model is 0.9054968 and the R-squared is 0.094431. Then, a variable importance graph was constructed to see which variables are the most important.

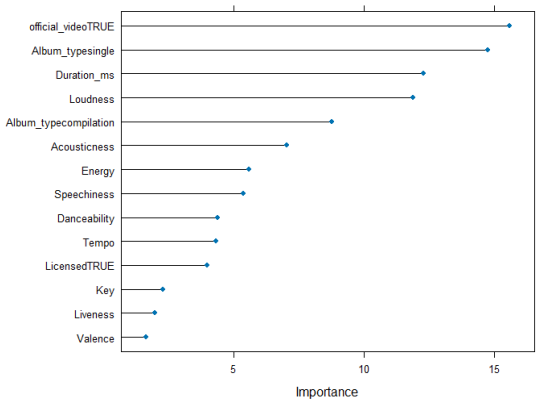

Figure 26: Variable importance of Neural Network model

The neural network model suggests that the most important variable is official video = True, contrary to the random forest model. Also, Liveness and Valence variables seem to be less effective in this model.

2.4. Support Vector Machines model with k-fold Cross Validation

e1071, mlbench and caret packages are used to construct a SVM model. Best hyperparameters are selected using grid search which had a resulting graph as,

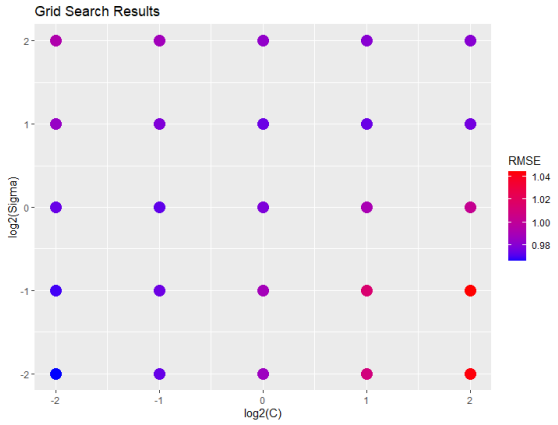

Figure 27: Grid search results of Support Vector Machines model

A log2 transformation to the c and sigma hyperparameters is implemented to better visualize the ranges. The grid search determined the best values for the hyperparameters to be 0.25 for both. The best model gave an MSE score of 0.9329445 while the R-squared for the said model was 0.07250329808.

2.5. XGboost model with k-fold Cross Validation

Finally, to obtain the best possible XGboost model for the dataset, a grid search was conducted which resulted in the best hyperparameters as: nrounds=300, max depth=9, eta=0.01, gamma=0.2, colsample_bytree=0.7, min_child_weight=1, subsample=0.9. The model had MSE= 0.968140980022566 and R-squared= 0.0639195407265462. The following graph for the variable importance was acquired.

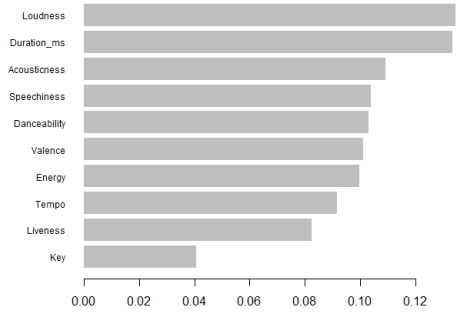

Figure 28: Variable importance of XGboost model

This graph showed that the most important variables in the model were Duration (ms) and Loudness while the worst variable in terms of importance was Key.

3. Discussion/Conclusion

To conclude, firstly, the effects of License status, Album type, Energy and Acousticness on Stream variable were investigated, and it was concluded that there was some significant importance. After doing necessary EDA and data tidying steps, the imputation for the MCAR data points were replaced with PMM values regarding the imputation step. Finally, Polynomial model, Random Forest model, Neural Network model, Support Vector Machines model and XGboost model was constructed with grid search to achieve the best models and the following results were obtained.

Model	MSE	R-Squared
Polynomial model	0.9507	0.0498
Random Forest model	0.8776	0.1224
Neural Network model	0.9055	0.0944
Support Vector Machines	0.9329	0.0725
XGboost model	0.9681	0.0639

Figure 29: Comparison of MSE and R-squared values of models

Finally, the best model was determined to be the one with the smallest MSE and the highest R-squared which was the Random Forest model.