DATA 180 - Introduction to Data Science

Final report

Spring 2022

Professor Xiexin Liu

Tai Nguyen

I. REPORT OBJECTIVE

Music plays an essential role in people's everyday lives. Listening to music contributes to enhancing life quality as one of the most common pastimes. In fact, the last decade has witnessed a huge rise of music songs as well as streaming platforms thanks to the advent of technology. Speaking from a data science perspective, every song created is within itself a unique source of data: singer, composer, language, genre, duration, release time and date, etc. Such piece of information serves as a huge potential to help us, especially music lovers, understand the mechanism behind some well-known music charts, such as Billboard or Youtube trending. In this report, using the tools and techniques from the DATA180 course, we are trying to get a deeper understanding of an available music dataset.

II. DATASET OF CHOICE

Spotify's Top 100 Songs Of 2010-2019

Source: https://www.kaggle.com/datasets/muhmores/spotify-top-100-songs-of-20152019

III. DESCRIPTION OF DATASET

The dataset compiles top 100 songs that were chosen by Spotify – the most popular music streaming platform – each year in a period of ten years from 2010 to 2019.

Total number of observations: 1003

Total number of variables: 17

IV. VARIABLES IN THE DATASET & DESCRIPTORS

title	Song's Title
artist	Song's Artist
genre	Genre of Song
year released	Year in which the song was released

added	Date the song was added to Spotify's playlist
bpm	Beats per minute
nrgy	Energy (Indicative score of song's energy level)
dnce	Danceability (Indicative score of song's danceability level)
dB	Decibel (How loud the song is)
live	Indicative score of how likely the song is a live recording instead of professional (e.g studio) recording
val	Indicative score of positive level of the song's mood
dur	Duration of the song (in seconds)
acous	Indicative score of acoustic level of the song is
spch	Indicative score of how strong the song is concentrated on spoken words.
pop	Indicative score of song's popularity prior to being listed on Spotify's Top List.
top year	Year in which the song was a hit
artist type	Type of artist (solo, duo, trio, or band) who performs

V. <u>BASIC VISUALIZATIONS</u>

- **Histogram**

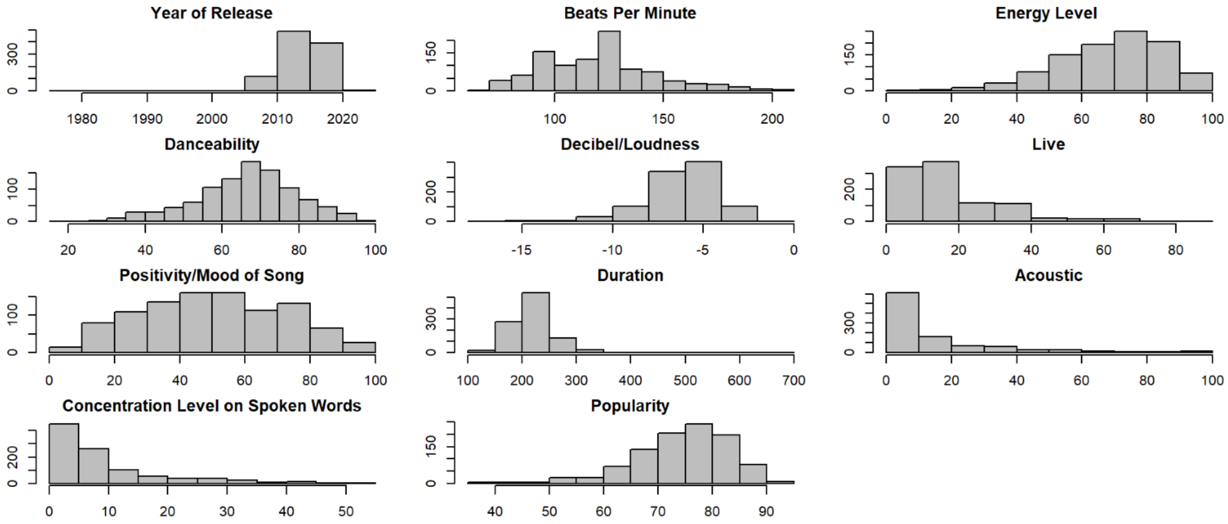


Figure 1: Overview of Histogram Plot Type. Note: y-axis is frequency, x-axis is same as plot's title

The purpose of visualizing by histogram graphs is to be able to summarize the distribution of a specific variable in the dataset. For example, looking at the visualization above, specifically by the height of the bars, we can make an overall conclusion that the majority of songs included in Spotify's list were released in the years between 2010 – 2015, whereas the period between 2005 – 2010 witnessed the least number of songs to be in the list. Or, the histogram representing variable 'Duration' tells the readers that the 200 – 250 seconds (about 3 to 4 minutes) is the most common length among all the top songs in the dataset.

- **Boxplot**

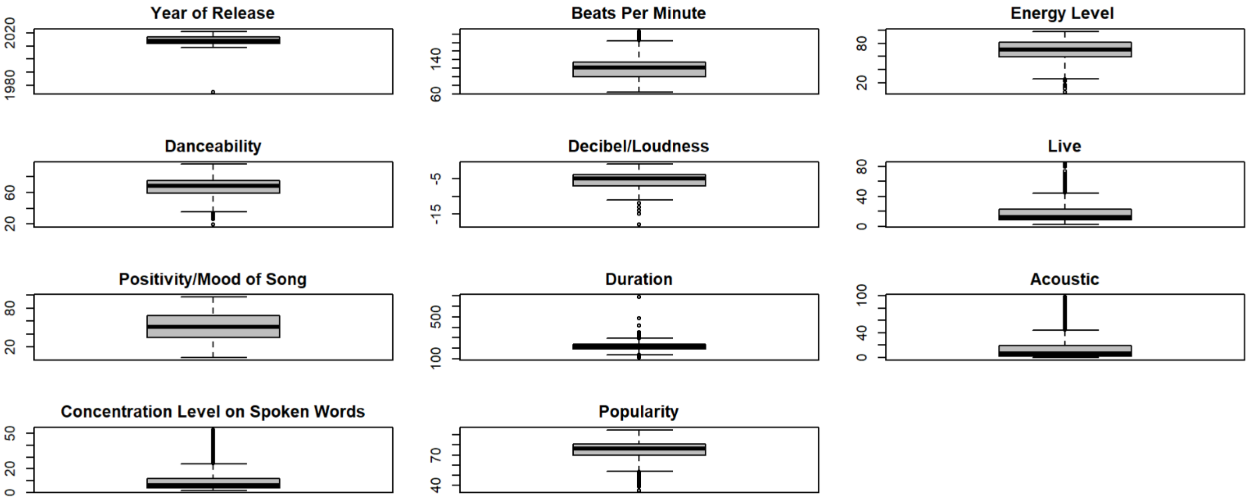


Figure 2: Overview of Box Plots across all numerical variables

Box plots, on the other hand, deliver a more mathematical communication of data compared to histograms. Looking at box plot visualizations, readers are able to tell information that can be used for computation. For instance, the box plot representing **bpm**) 'Beats per Minute' variable shows that across all the observations (songs) in this dataset:

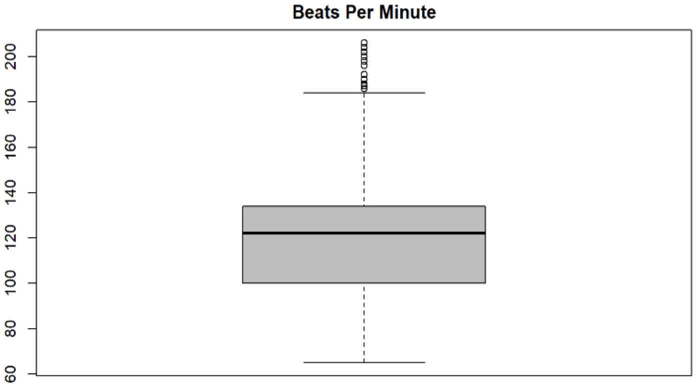


Figure 3: Box plot of 'bpm'

The highest bpm value is roughly 180.

The minimum bpm value is about 65.

The lowest bpm value is 120.

The bpm range at which most songs are is between 120 and 135.

There are also some outliers (those whose values are far greater or lesser than the rest) with bpm value over 180 to 200.

VI. CLUSTERING

1. Objective

We use cluster analysis to identify groups that have similar patterns within a single or multiple variables. For this specific dataset, we can look into clusters with different levels of popularity, or clusters with a combination of top music genre and artist type.

2. Overview of clustering

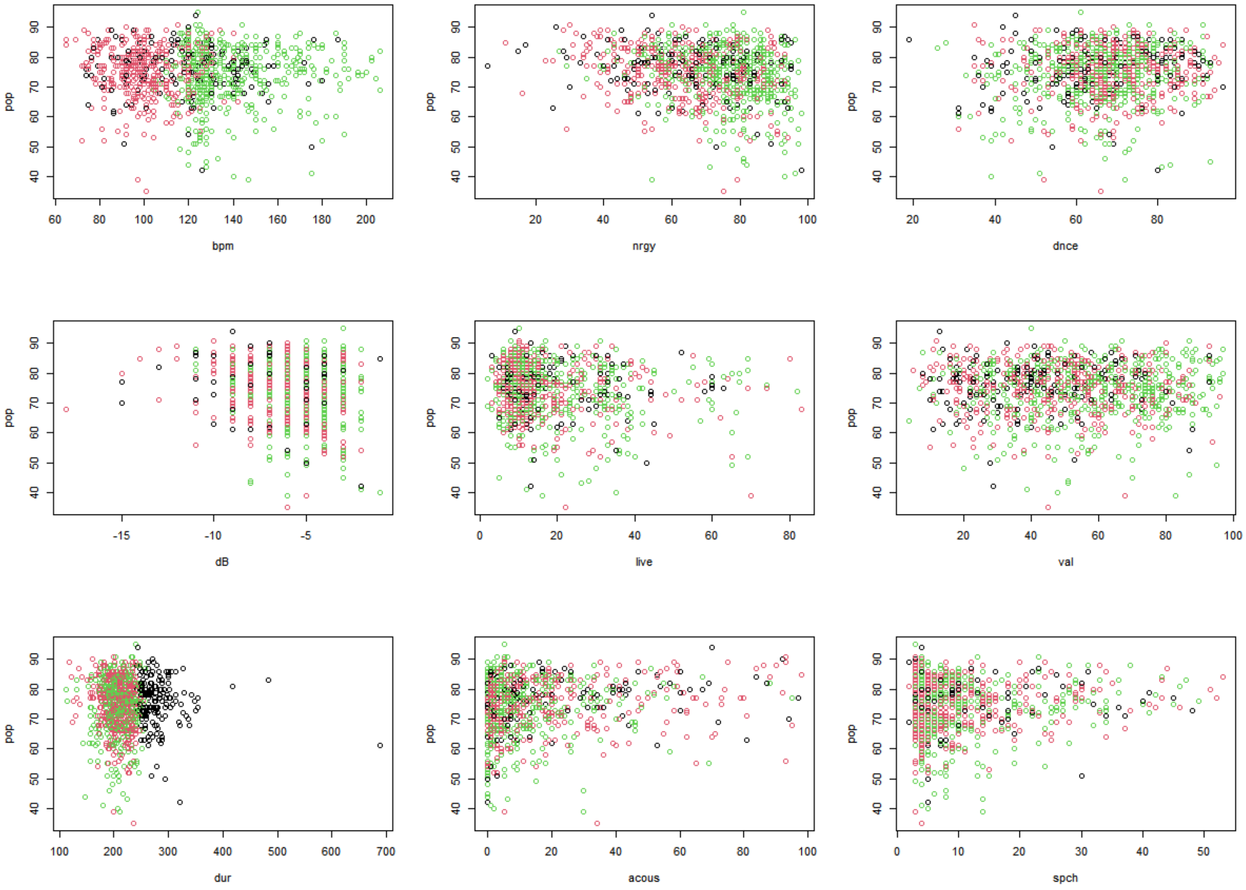


Figure 4: Overview of Clustering

3. Detailed interpretation

Color notations:

Group 1 – red

Group 2 – green

Group 3 - black

- **bpm**

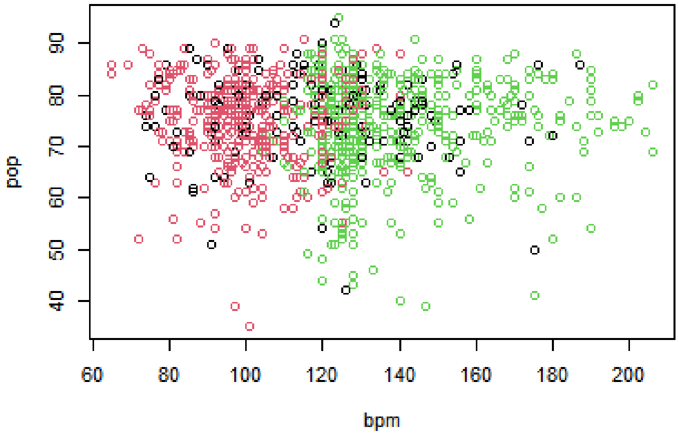


Figure 5: Cluster of 'pop' and 'bpm'

From clustering, we can identify three groups of similar characteristics. The first group (red) mainly has songs with high popularity rates (>60) and low beats per minute (80-120). The second group (green) has song with also high popularity rates (>60) and high beats per minute (>120). The third group (black) - the least dense one – has songs with high popularity rates (>60) and varied beats per minute.

- **nrgy**

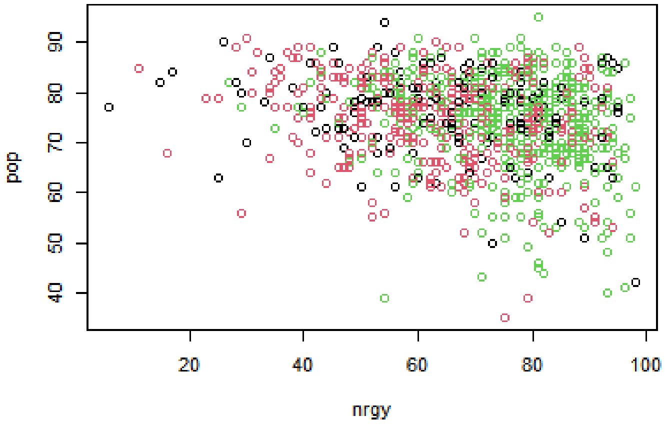


Figure 6: Cluster of 'pop' and 'nrgy'

All three groups mostly have popularity rates ranging between 60 to 90, with group 1 and 3 representing varied levels of energy rates (30-90) and group 2 representing energy rate in the range between 60-100.

- **dnce**

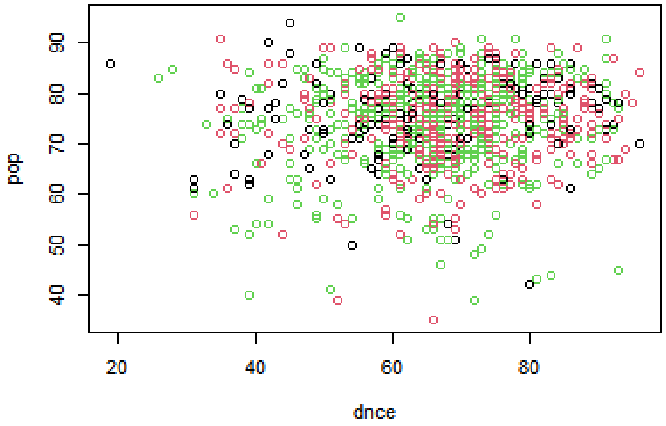


Figure 7: Cluster of 'pop' and 'dnce'

All three groups indicate a similar characteristic: Songs with popularity rates ranging from 60-90 mostly have medium to high danceability rates (between 50
and 90).

- **live**

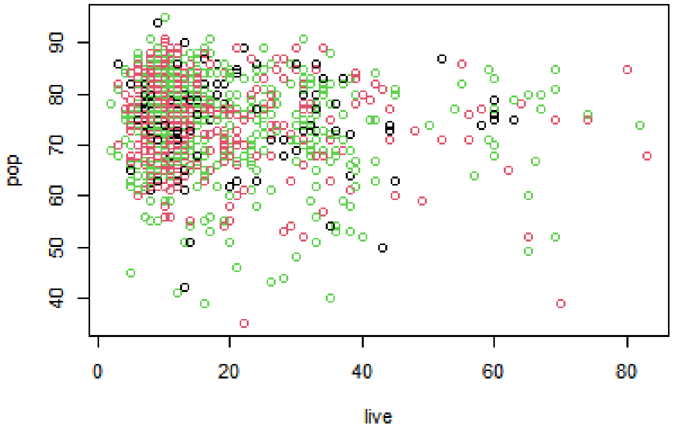


Figure 8: Cluster of 'pop' and 'live'

Group 1 and 2 cluster mainly around high popularity rates (60-90) with low likeliness (0-30) of being recorded live. In other words, a large number of songs in Spotify's list with high popularity have low likelihood of being recorded live.

- **dur**

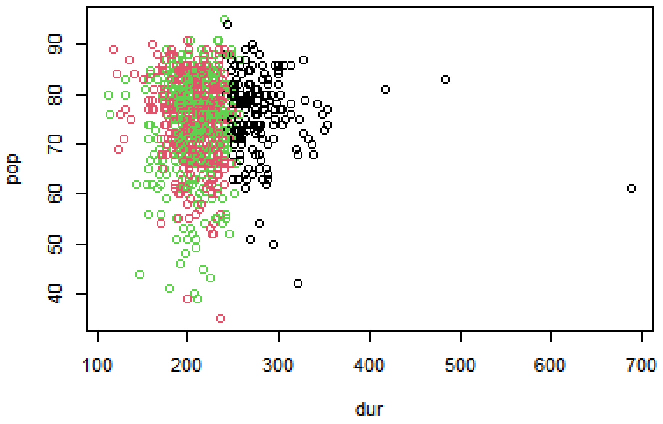


Figure 9: Cluster of 'pop' and 'dur'

Group 1 includes songs with duration from roughly 180 to 250 seconds and with popularity rates ranging from 60-90. Group 2 includes songs with duration from 100 to 250 seconds and with a more diverse range of popularity rates (40-90). Group 3, distinctly, has popular songs with over 250 seconds in duration.

- **acous**

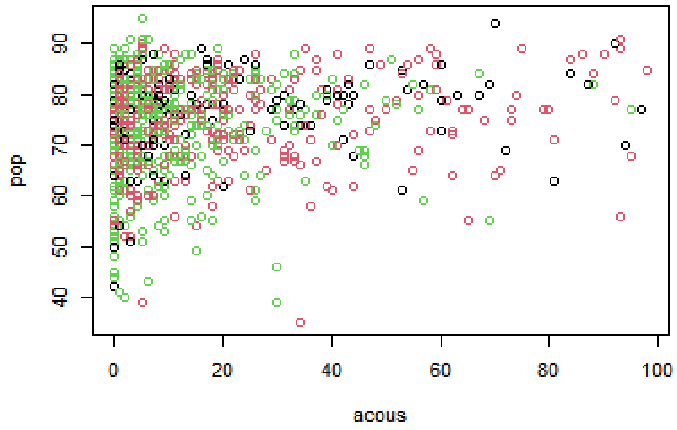


Figure 10: Cluster of 'pop' and 'acous'

All three groups share a similar characteristic: most dense at 0-20 and spread out as acoustic level increases from 0 to 100. We can say that most popular songs do not have high acoustic rates.

- **spch**

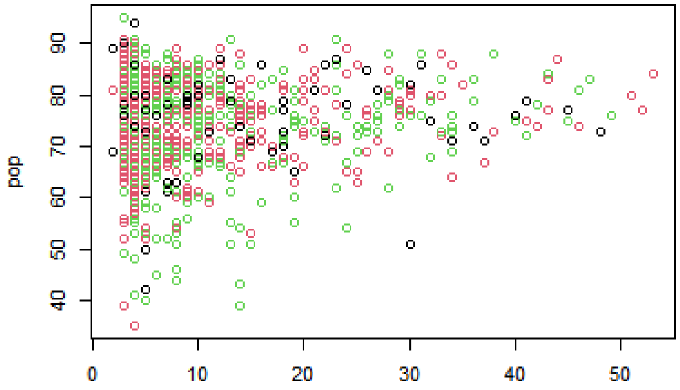


Figure 11: Cluster of 'pop' and 'spch'

All three groups share a similar characteristic: most dense at 0-10 and spread out as speech level increases from 0 to 100. We can say that most popular songs do not have or have low speech rates.

VII. REGRESSION

1. Objective

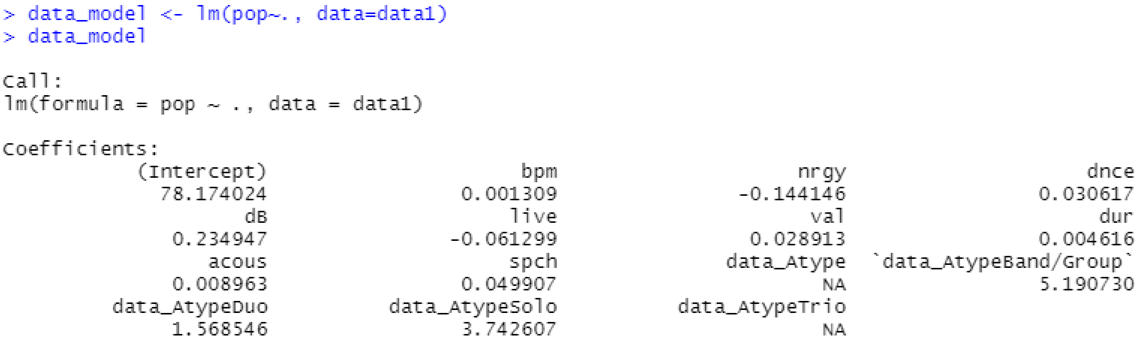
We use regression model to explain the change in a response variable for every unit increase in the predictors. In other words, we measure the volume of effect of one or more variables on one specific variable of interest in order to make practical inference. For example, would a song performed by a group be rated higher than a song by a solo singer? Or, an USUK-pop fan may ask if a song by Ariana Grande would always be more popular than Sia's on Spotify's data?

In this report, a regression model is used to identify relationships between 'pop' as the response variable and other variables as the predictors.

2. Pre-processing task

Converting categorial variable 'artist.type' into a binary matrix since regression cannot take on non-numeric values.

3. Results



```
> data_model <- lm(pop~., data=data1)
> data_model

Call:
lm(formula = pop ~ ., data = data1)

Coefficients:
          (Intercept)                     bpm                    nrgy                    dnce  
            78.174024                0.001309               -0.144146                0.030617  
                   dB                    live                     val                     dur  
             0.234947               -0.061299                0.028913                0.004616  
                acous                    spch              data_Atype  `data_AtypeBand/Group`  
             0.008963                0.049907                      NA                5.190730  
        data_AtypeDuo          data_AtypeSolo          data_AtypeTrio  
             1.568546                3.742607                      NA  
```

Figure 12: Regression model

- Intercept value of 78.17 (roughly) indicates the base popularity rate, prior to taking into consideration other independent variables/predictors, or when all predictors equal to 0.
- Predictors that have **positive associations** with the response variable.

Variable	Interpretation of coefficient value
bpm	every increase of a beat per minute associates with a popularity rate increase roughly by 0.001

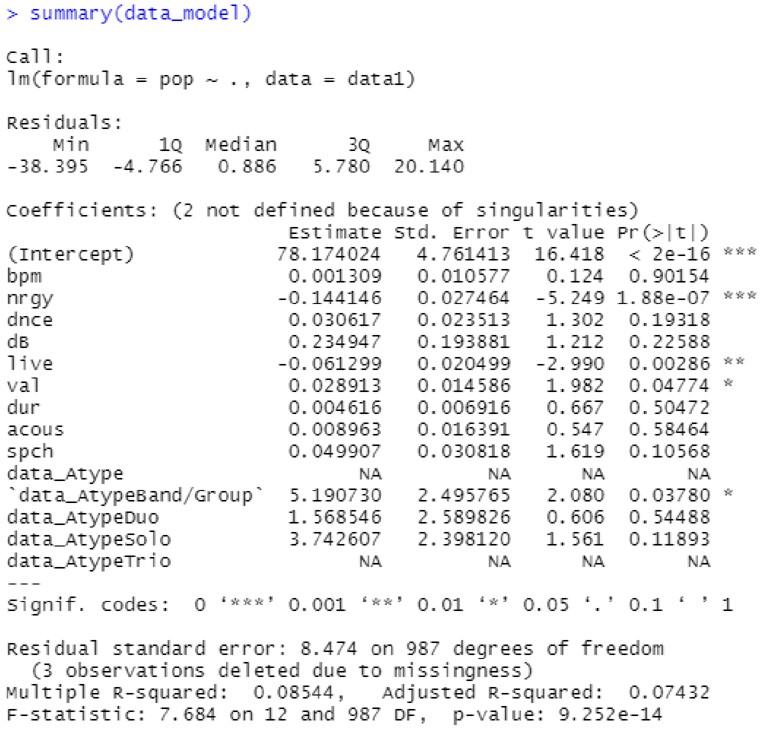
dnce	every increase of a danceability level associates with a popularity rate increase roughly by 0.03
dB	every increase of a decibel associates with a popularity rate increase roughly by 0.23
val	every increase of a positivity level associates with a popularity rate increase roughly by 0.03
dur	every increase of a second in duration associates with a popularity rate increase roughly by 0.005
artist type-group	a song identified as a group performance associates with a popularity increase roughly by 5.19
artist type-duo	a song identified as a duo performance associates with a popularity increase roughly by 1.57
artist type-solo	a song identified as a solo performance associates with a popularity increase roughly by 3.74

- Predictors that have **negative associations** with the response variable.

Variable	**Interpretation of coefficient value**
nrgy	every increase of an energy level associates with a popularity rate decrease roughly by 0.14
live	every increase of a live associates with a popularity rate decrease roughly by 0.06

- Practical implication from regression results
  A 21 beat-per-minutes (bpm) difference has earned Bruno Mars' 'Talking to the moon' 22 popularity score (pop) more than Yolanda Be Cool's 'We no speak Americano' on Spotify's Top List. As a matter of fact, basing Youtube as an example standard, 'Talking to the moon' has 123 million views as of May 2022, whereas 'We no speak americano' gains approximately 51 million views at the same time. This example proves that beats per minute does have a positive correlation to the popularity of the two songs prior being on Spotify's Top List!

- **Model summary**



```
> summary(data_model)

Call:
lm(formula = pop ~ ., data = data1)

Residuals:
    Min      1Q  Median      3Q     Max 
-38.395  -4.766   0.886   5.780  20.140 

Coefficients: (2 not defined because of singularities)
                        Estimate Std. Error t value Pr(>|t|)    
(Intercept)            78.174024   4.761413  16.418  < 2e-16 ***
bpm                     0.001309   0.010577   0.124  0.90154    
nrgy                   -0.144146   0.027464  -5.249 1.88e-07 ***
dnce                    0.030617   0.023513   1.302  0.19318    
dB                      0.234947   0.193881   1.212  0.22588    
live                   -0.061299   0.020499  -2.990  0.00286 ** 
val                     0.028913   0.014586   1.982  0.04774 *  
dur                     0.004616   0.006916   0.667  0.50472    
acous                   0.008963   0.016391   0.547  0.58464    
spch                    0.049907   0.030818   1.619  0.10568    
data_Atype                    NA         NA      NA       NA    
`data_AtypeBand/Group`  5.190730   2.495765   2.080  0.03780 *  
data_AtypeDuo           1.568546   2.589826   0.606  0.54488    
data_AtypeSolo          3.742607   2.398120   1.561  0.11893    
data_AtypeTrio                NA         NA      NA       NA    
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 8.474 on 987 degrees of freedom
  (3 observations deleted due to missingness)
Multiple R-squared:  0.08544,	Adjusted R-squared:  0.07432 
F-statistic: 7.684 on 12 and 987 DF,  p-value: 9.252e-14
```

Figure 13: Regression model summary

Interpretations:

- R-squared value of 0.07432 indicates that roughly 7.4% of the variation in the popularity rate could be explained using this regression model.
- Pr(>|t|) values less than 0.05, denoted by one or more ‘*’, indicate that the variable is statistically significant. In this model, we have **nrgy**, **live**, **val** and **artist type ‘group’** as statistically significant variables.
- P-value of 9.252e-14 (0.000000009252) < 0.05 indicates that at least one predictor is related to the popularity rate.

VIII. CLASSIFICATION

1. Objectives:

We use KNN (K-Nearest Neighbors) as a classification technique to classify a data point on how it closest (neighbor) data points are classified. On this dataset, we choose ‘pop’ as a classification variable.

2. Pre-processing tasks

- Classify 'pop' into two categories

  pop value greater than or equal to 75: popular

  pop value less than 75: unpopular

- Use 'mutate' function from 'dplyr' package to add a new variable to existing dataset:

  Class = 1: A popular song (pop value greater than or equal to 75)

  Class = 0: An unpopular song (pop value fewer than 75)

- Create a training set and validation set
- Compute a KNN prediction to find the class prediction for every unit in the validation set. On choosing the best K, the plot on the right shows us that K = 5 and fewer seem to give higher accuracy levels when running KNN prediction.

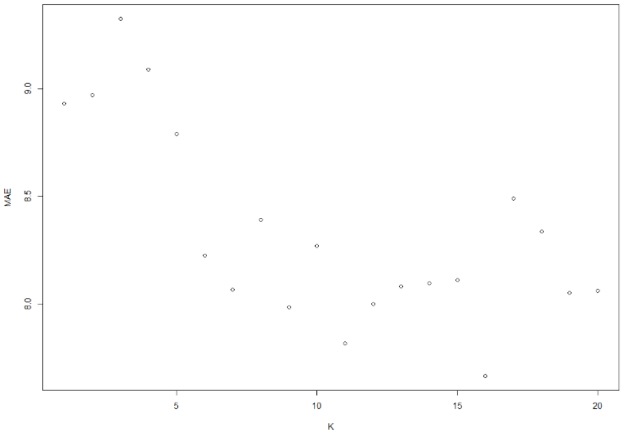
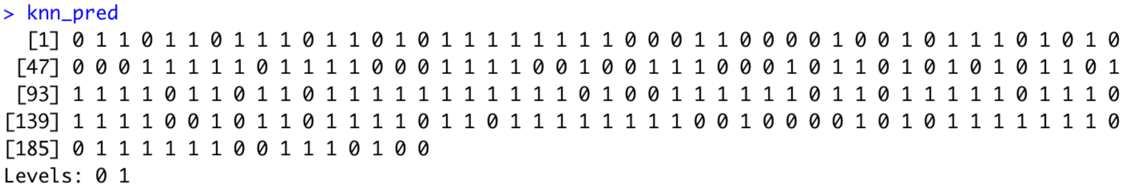


Figure 14: Best K-Values Plot

- Use a confusion matrix – the table() function – to compare the predicted class with the actual class.

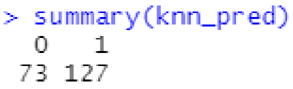
3. Results



```
> knn_pred
  [1] 0 1 1 0 1 1 0 1 1 1 0 1 1 0 1 0 1 1 1 1 1 1 1 1 0 0 0 1 1 0 0 0 0 1 0 0 1 0 1 1 1 0 1 0 1 0
 [47] 0 0 0 1 1 1 1 1 0 1 1 1 1 0 0 0 1 1 1 1 0 0 1 0 0 1 1 1 0 0 0 1 0 1 1 0 1 0 1 0 1 0 1 1 0 1
 [93] 1 1 1 1 0 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 0 1 0 0 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 0 1 1 1 0
[139] 1 1 1 1 0 0 1 0 1 1 0 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 0 0 1 0 0 0 0 1 0 1 0 1 1 1 1 1 1 1 0
[185] 0 1 1 1 1 1 1 0 0 1 1 1 0 1 0 0
Levels: 0 1
```

Figure 15: KNN prediction outcome

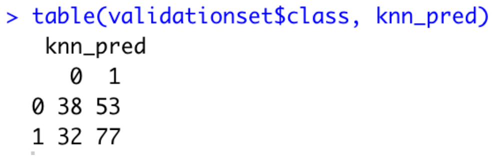
- **Knn_pred** gives the 'class' prediction for every unit in the validation set.



```
> summary(knn_pred)
  0   1 
 73 127 
```

Figure 16: KNN prediction summary

- **Summary()** function tells us that there are 73 songs in the validation set are classified as ‘unpopular songs’ and the rest 127 songs are songs are classified as ‘popular songs’.



```
> table(validationset$class, knn_pred)
   knn_pred
     0  1
  0 38 53
  1 32 77
```

Figure 17: Confusion Matrix

- The **confusion matrix** created by using table() function tells us that in the validation set, there 38+53 = 88 unpopular songs, and 32+77=103 popular songs. In particular, 38 out of 88 unpopular songs were accurately predicted by the classification model. Likewise, 32 out of 103 popular songs were accurately predicted by the model.
- Estimated **accuracy level** of the classification model: 57.5%.