SPOTIFY DATA ANALYSIS

A LINEAR REGRESSION MODEL IN R PROGRAMMING LANGUAGE.

INDIRA GANDHI DELHI TECHNICAL UNIVERSITY FOR WOMEN

EDUCATION-ENLIGHTENMENT-EMPOWERMENT

IGDTU

आपो दीपो भव:

Submitted to:
Mr. Santanoo Patnaikk

AI-ML
1. Shreya Sisodia - 06301192022
2. Siddhi Shrivastwa - 06401192022
3. Sneha Goyal -06501192022

CONTENT

S.No	Topic	Page No
1	Abstract	1
2	Introduction	2
3	Multiple Linear Regression	3
4	Data Description	7
5	Methodology	9
6	Code	12
7	Data Visualisation	18
8	Result	26
9	Evaluation & Validation	28
10	Conclusion	31
11	References	32

01

ABSTRACT:

This study aimed to develop a linear regression model in R programming to predict the energy level of songs based on various audio features.

The dataset included songs Spotify Top 100 playlist of 2018, and predictors such as danceability, valence, time signature and other relevant audio features were examined.

The analysis revealed that loudness, danceability and acousticness were significant predictors of energy, indicating that songs with higher in these audio features tend to have higher energy levels.

The model demonstrated a good fit to the data, with a significant R-squared indicating that a substantial portion of the variation in energy could be explained by the selected audio features.

These findings have practical implications for music recommendation systems and playlist curation, allowing for personalized recommendations and the creation of playlists with desired energy levels.

0 2

INTRODUCTION

The emergence of digital music streaming platforms has revolutionized the way we consume music, providing access to an extensive library of songs at our fingertips. Among these platforms, Spotify has gained widespread popularity, offering millions of tracks across various genres and catering to diverse musical tastes. With its vast user base and rich dataset, Spotify presents a valuable opportunity for data analysis to uncover insights into music preferences, user behavior, and industry trends.

Understanding the factors that contribute to the perceived energy of a song is crucial for music recommendation systems, playlist curation, and understanding the emotional impact of music. In this study, we aim to develop a **linear regression model using R programming to predict the energy level of songs based on various audio features.** Linear regression a powerful statistical tool is that it allows to quantify by what quantity the dependent variable varies when the independent variable increases by one unit. These features include danceability, valence, time signature, and other relevant audio characteristics. By analyzing the relationship between these features and the energy levels of songs, we can gain insights into the key factors that influence the energetic qualities of music. This research can provide valuable knowledge for music platforms to create personalized recommendations, curate playlists with desired energy levels, and enhance users' music listening experiences.

03

MULTIPLE LINEAR REGRESSION

Multiple linear regression is a type of linear regression which we have employed to explain the relation between certain audio features.
This approach makes it possible to relate one variable with several variables through a linear function in its parameters.

PRINCIPLE

- Multiple linear regression is used to assess the relationship between two variables while taking into account the effect of other variables.
- By taking into account the effect of other variables, we cancel out the effect of these other variables in order to isolate and measure the relationship between the two variables of interest (free of the linear effects of the other explanatory variables).

EQUATION

- Multiple linear regression models are defined by the equation

$$Y=\beta o+\beta 1X1+\beta 2X2+...+\beta pXp+\epsilon$$

- It is no longer a question of finding the best line (the one which passes closest to the pairs of points (yi,xi)), but finding the p-dimensional plane which passes closest to the coordinate points (yi,xi1,…,xip).

04

MULTIPLE LINEAR REGRESSION

INTERPRETATIONS OF COEFFICIENTS

- The hypotheses are the same as for simple linear regression, that is:

  Ho:βj=0
  H1:βj≠0

- The test of βj=0 is equivalent to testing the hypothesis: is the dependent variable associated with the independent variable studied, all other things being equal, that is to say, at constant level of the other independent variables.

CONDITIONS OF APPLICATION

As for simple linear regression, multiple linear regression requires some conditions of application for the model to be usable and the results to be interpretable. Conditions for simple linear regression also apply to multiple linear regression, that is:

1. **Linearity** of the relationships between the dependent and independent variables
2. **Independence** of the observations
3. **Normality** of the residuals
4. **Homoscedasticity** of the residuals
5. **No influential points** (outliers)

But there is one more condition for multiple linear regression:
6. **No multicollinearity**: Multicollinearity arises when there is a strong linear correlation between the independent variables, conditional on the other variables in the model.

0 5

MULTIPLE LINEAR REGRESSION

CHOOSING A GOOD MODEL

The three most common tools to select a good linear model are according to:

1. the p-value associated to the model,
2. the coefficient of determination R2 and
3. the Akaike Information Criterion (AIC)

P-value associated to the model

Before interpreting the estimates of a model, it is a good practice to first check the p-value associated to the model. This p-value indicates if the model is better than a model with only the intercept.

The hypotheses of the test (called F-test) are:

- Ho : $\beta1=\beta2=\cdots=\beta p=0$
- H1 : at least one coefficient $\beta \neq 0$

This p-value can be found at the bottom of the summary() output.

The p-value≠0. The null hypothesis is rejected, so we conclude that our model is better than a model with only the intercept because at least one coefficient β is significantly different from 0.

If this p-value > 0.05 for one of your model, it means that none of the variables you selected help in explaining the dependent variable. In other words, you should completely forget about this model because it cannot do better than simply taking the mean of the dependent variable.

Coefficient of determination R2

The coefficient of determination, R2, is a measure of the goodness of fit of the model. It measures the proportion of the total variability that is explained by the model, or how well the model fits the data.

06

MULTIPLE LINEAR REGRESSION

R2 varies between 0 and 1:

- R2=0: the model explains nothing
- R2=1: the model explains everything
- 0<R2<1: the model explains part of the variability
- the higher the R2, the better the model explains the dependent variable. As a rule of thumb, a R2>0.7 indicates a good fit of the model

Parsimony

A parsimonious model (few variables) is usually preferred over a complex model (many variables). There are two ways to obtain a parsimonious model from a model with many independent variables:

1. We can iteratively remove the independent variable least significantly related to the dependent variable (i.e., the one with the highest p-value in an ANOV table) until all of them are significantly associated to the response variable, or
2. We can select the model based on the Akaike Information Criterion (AIC). AIC expresses a desire to fit the model with the smallest number of coefficients possible and allows to compare models. According to this criterion, the best model is the one with the lowest AIC. This criterion is based on a compromise between the quality of the fit and its complexity. We usually start from a global model with many independent variables, and the procedure (referred as stepwise algorithm) automatically compares models then selects the best one according to the AIC.

07

DATA DESCRIPTION

The dataset was acquired from www.kaggle.com. It comprises of 100 rows and 16 columns. It includes the names, artists, Spotify URIs and audio features of tracks from the Top Tracks of 2018 playlist. Following descriptions of audio features are from the Spotify Web API.

1. id: Spotify URI of the song
2. name: Name of the song
3. artists: Artist(s) of the song
4. danceability: Danceability describes how suitable a track is for dancing based on a combination of musical elements including tempo, rhythm stability, beat strength, and overall regularity. A value of 0.0 is least danceable and 1.0 is most danceable.
5. energy: Energy represents a perceptual measure of intensity and activity. Typically, energetic tracks feel fast, loud, and noisy.
6. key : The key the track is in. Integers map to pitches using standard Pitch Class notation.
7. loudness: Loudness is the quality of a sound that is the primary psychological correlate of physical strength (amplitude).
8. mode: Mode indicates the modality (major or minor) of a track, the type of scale from which its melodic content is derived. Major is represented by 1 and minor is 0.
9. speechiness: Speechiness detects the presence of spoken words in a track.

DATA DESCRIPTION

10. acousticness: A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence the track is acoustic.
11. instrumentalness: Predicts whether a track contains no vocals. "Ooh" and "aah" sounds are treated as instrumental in this context. Rap or spoken word tracks are clearly "vocal".
12. liveness: Detects the presence of an audience in the recording. Higher liveness values represent an increased probability that the track was performed live.
13. valence: A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track.
14. tempo: Tempo is the speed or pace of a given piece and derives directly from the average beat duration.
15. duration_ms: The duration of the track in milliseconds.
16.time_signature: The time signature (meter) is a notational convention to specify how many beats are in each bar (or measure).

METHODOLOGY

DATA ACQUISITION

The dataset of Top 100 Songs of 2018 from Spotify was obtained and used as the primary source of data for this project. The dataset consists of 100 songs and their corresponding audio features, providing a comprehensive pool of user preferences and song information.

DATA PREPROCESSING

- **Data Cleaning**

The dataset was checked and corrected, if required) for errors or inconsistencies in the data, such as missing values, outliers, and duplicates.

- **Data Transformation**

Many different steps were taken to convert the data into a suitable format for analysis. Audio features like key, valence, mode and time signature were converted to type factor, numerical keys were converted to the actual musical, the duration milliseonds was converted to mins and the values of valence were categorized into 3 categories.

METHODOLOGY

- **Data Reduction**

The data acquired did not require any reduction as all the information it contained deemed important.

- **Splitting the Dataset**

The data is split into 2 parts in a ratio of 8:2. The data set is randomly distributed and the distribution is done with replacement.

DATA FITTING

- **Choosing an Algorithm**

Supervised machine learning is employed as the dataset obtained had labeled data. his means that the models train based on the data that has been processed (cleaned, randomized, and structured) and annotated. Regression helps to look for this correlation and predict an output. Thus, here, energy will be evaluated based on other audio features.

METHODOLOGY

- **Evaluation**

Cross-validation technique is used to evaluate the performance of a machine learning model. The sample() function was used to randomly assign data points to different folds, ensuring that each fold represents a random and diverse subset of the data. This helps in obtaining reliable performance estimates for the model across different subsets of the data.

- **Feature Selection**

This model uses backward elimination and both elimination to achieve best set of features. Combining backward elimination and both elimination provides a more thorough exploration of the predictor variables, considering the removal of non-significant predictors and the potential addition of variables that contribute to the model's fit.

- **Recreate Model**

The final model was made after performing feature selection techniques and using the suggested best set of audio features to predict energy of a song.

12

CODE

```r
#importing libraries
library(tidyverse)
library(ggcorrplot)
library(dplyr)
library(MASS)
library(gvlma)
library(readxl)

#library: readxl
my_list <- read_excel("D:/Siddhi/R/top2018.xlsx")

#dimensions of dataset
dim(my_list)
#top n rows of of the dataset (by default n=6)
head(my_list)
#column names in the dataset
names(my_list)
#checking for duplicate values
length(unique(my_list$name))
length(unique(my_list$id))
#checking for null values in each column
colSums(is.na(my_list))
#summarizing the dataset
summary(my_list)
#plotting a correlation matrix between variables
#library: ggcorrplot
corr <- round(cor(my_list[,4:16]),8)
ggcorrplot(corr)

#applying class function on the dataset
sapply(my_list, class)
#displaying unique key,mode, time_signature values
unique(my_list$key)
unique(my_list$mode)
unique(my_list$time_signature)

#converting key,mode, time_signature into factor class
my_list$key <- as.factor(my_list$key)
my_list$mode <- as.factor(my_list$mode)
my_list$time_signature <- as.factor(my_list$time_signature)
```

```
#applying class function on the dataset
sapply(my_list, class)
#converting the numerical keys to the actual musical keys
levels(my_list$key)[1] <-"C"
levels(my_list$key)[2] <-"C#"
levels(my_list$key)[3] <-"D"
levels(my_list$key)[4] <-"D#"
levels(my_list$key)[5] <-"E"
levels(my_list$key)[6] <-"F"
levels(my_list$key)[7] <-"F#"
levels(my_list$key)[8] <-"G"
levels(my_list$key)[9] <-"G#"
levels(my_list$key)[10] <-"A"
levels(my_list$key)[11] <-"A#"
levels(my_list$key)[12] <-"B"

#convert the duration milliseonds to mins
my_list$duration_ms <- my_list$duration_ms/60000

#adding popularity column to the my_list
popularity <- c(1:100)
#binding popularity with the my_list
my_list <- cbind(my_list,popularity)

#categorizing valence values
my_list$valence[my_list$valence > 0.000 & my_list$valence <= 0.350 ] <-
"sad"
my_list$valence[my_list$valence >= 0.351 & my_list$valence <= 0.700 ] <-
"happy"
my_list$valence[my_list$valence >= 0.701 & my_list$valence <= 1.000 ] <-
"Euphoric"

#converting valence into factor class
my_list$valence <- as.factor(my_list$valence)

#using library ggplot2 from tidyverse to plot graphs

#plotting density graph for variable energy
ggplot(my_list) + geom_density(aes(energy),fill="#4d5382")
```

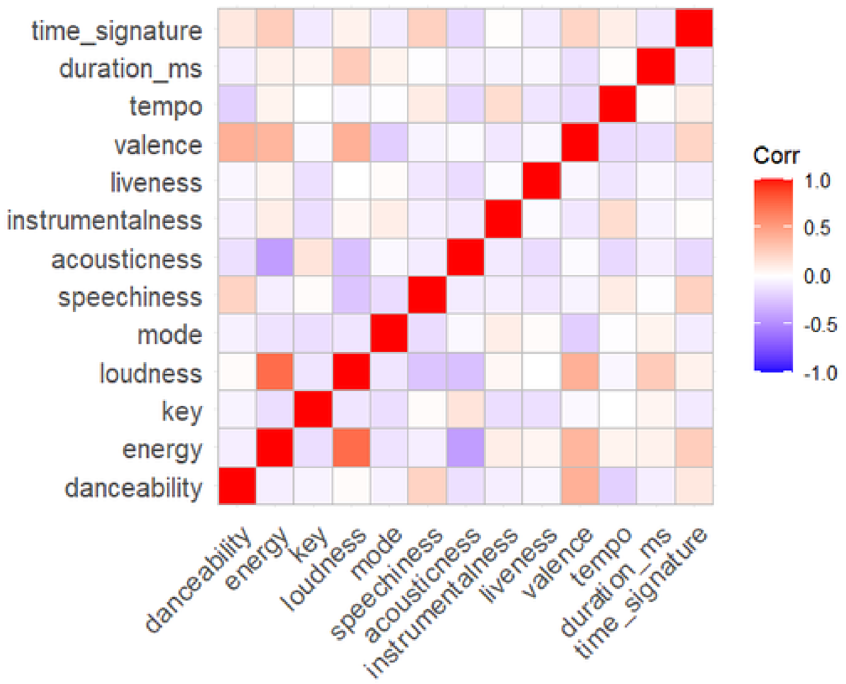
```
#plotting bar graph for variable valence
ggplot(my_list) + geom_bar(aes(valence),fill="#d77a61")

#plotting bar graph for variable time_signature
ggplot(my_list) + geom_bar(aes(time_signature),fill="#a5b2a5")

#plotting density graph for variable duration_ms
ggplot(my_list) + geom_density(aes(duration_ms),fill="#c4adac")

#plotting density graph for variable danceablity
ggplot(my_list) + geom_density(aes(danceability),fill="#43938a")

#plotting bar graph for variable mode
ggplot(my_list) + geom_bar(aes(mode),width = 0.4,fill="#c3d59f")

#plotting density graph for variable speechiness
ggplot(my_list) + geom_density(aes(speechiness),fill="#e1adad")

#plotting density graph for variable acousticness
ggplot(my_list) + geom_density(aes(acousticness),fill="#4b405f")

#plotting bar graph for variable key
ggplot(my_list) + geom_bar(aes(key),width = 0.4,fill="#d1b8a3")

#new dataframe grouped by frequency of artist
#library: dplyr
my_frequency <- data.frame(my_list %>%
                    group_by(my_list$artists) %>%
                    summarise(no_rows = n()) %>%
                    arrange(desc(no_rows)))

#new dataframe with top 10 most occurring artist
my_frequency_10 <- my_frequency[1:10,]

#plotting a bar graph for frequency of top 10 artists
ggplot(my_frequency_10,aes(my_list.artists,no_rows))+
geom_bar(stat = "identity",width=0.4,fill="#982020")+
labs(x="Top 10 Artists")+
labs(y="Count of Songs in top 100 List")+
labs(title = "Top 10 Artist Counts")
```

```r
#plotting a polar scatter graph to compare acousticness with popularity
ggplot(my_list,aes(popularity,acousticness)) +
geom_point(stat="identity")+
geom_abline(intercept = 0.65,slope=0)+
labs(x="Popularity")+
labs(y="acousticness")+
labs(title = "Popularity vs acousticness")+
coord_polar()

#plotting a scatter graph to compare instrumentalness with popularity
ggplot(my_list,aes(popularity,instrumentalness)) +
geom_point(stat="identity")+
labs(x="Popularity")+
labs(y="instrumentalness")+
labs(title = "Popularity vs instrumentalness")

my_list_1 <- my_list

#Splitting data into test and train.
sample_data <- sample(2,nrow(my_list_1),replace=TRUE,prob = c(0.8,0.2))
train_data <- my_list[sample_data == 1,]
test_data <- my_list[sample_data == 2,]

#Model creation
fit_linear <- lm(energy ~
loudness+danceability+valence+speechiness+acousticness+instrumentalness+liveness , data=train_data)

#checking assumptions
#library: gvlma
gvlma(fit_linear)

#feature selection - Backward elimination
step_1 <- stepAIC(fit_linear,direction = "backward")
step_1$anova
#Feature Selection - Both
step_2 <- stepAIC(fit_linear,direction = "both")
step_2$anova
```

```
#Final Model:
fit_final <- lm(energy ~ loudness + danceability +acousticness, data =
train_data)

#library: gvlma
gvlma(fit_final)

#summarizing the final model
summary(fit_final)

#comparing predicted and observed values
predicted <- predict(fit_final,newdata = test_data)
observed <- test_data$energy
predicted

#calculating value of R-squared & average prediction error
SSE <- sum((observed - predicted) ^ 2)
SST <- sum((observed - mean(observed)) ^ 2)
r2 <- 1 - SSE/SST
rmse <- sqrt(mean((predicted - observed)^2))

rmse
r2

library(performance)
check_model(fit_final)
```

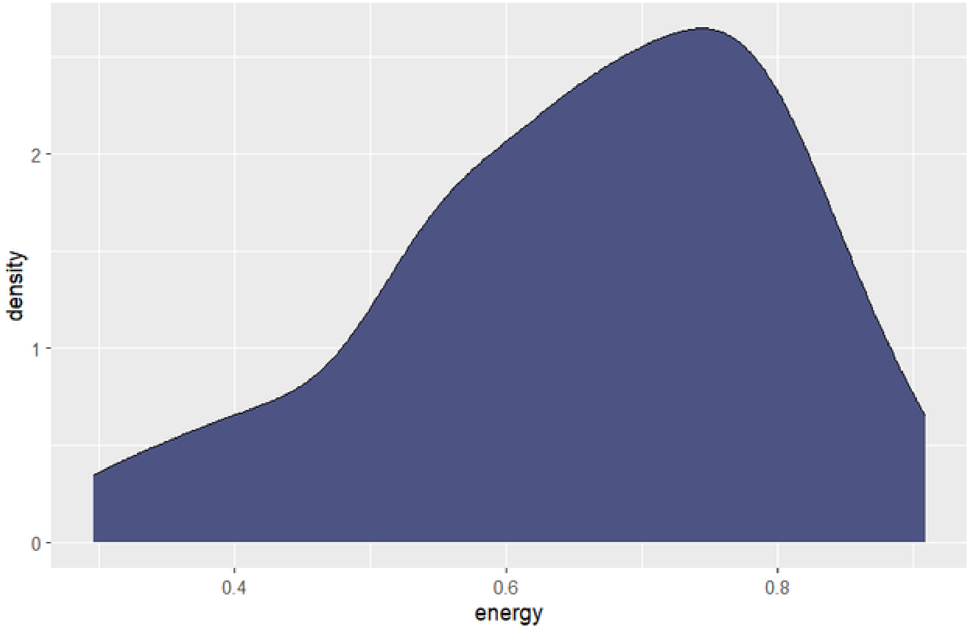
DATA VISUALISATION

CORRELATION GRAPH



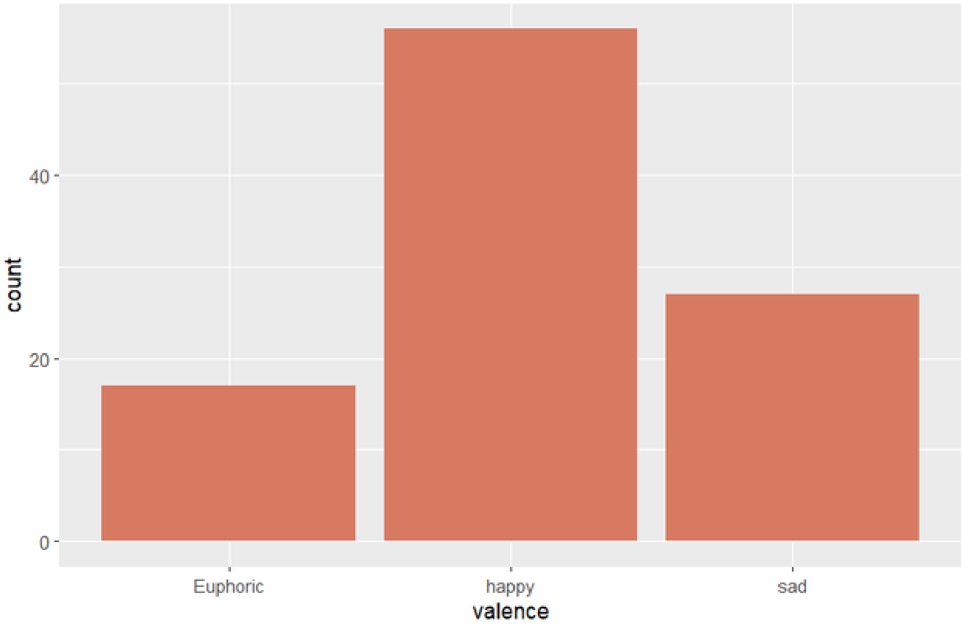
The variables loudness and energy are correlated to some extent compared to the other variables

ENERGY ANALYSIS



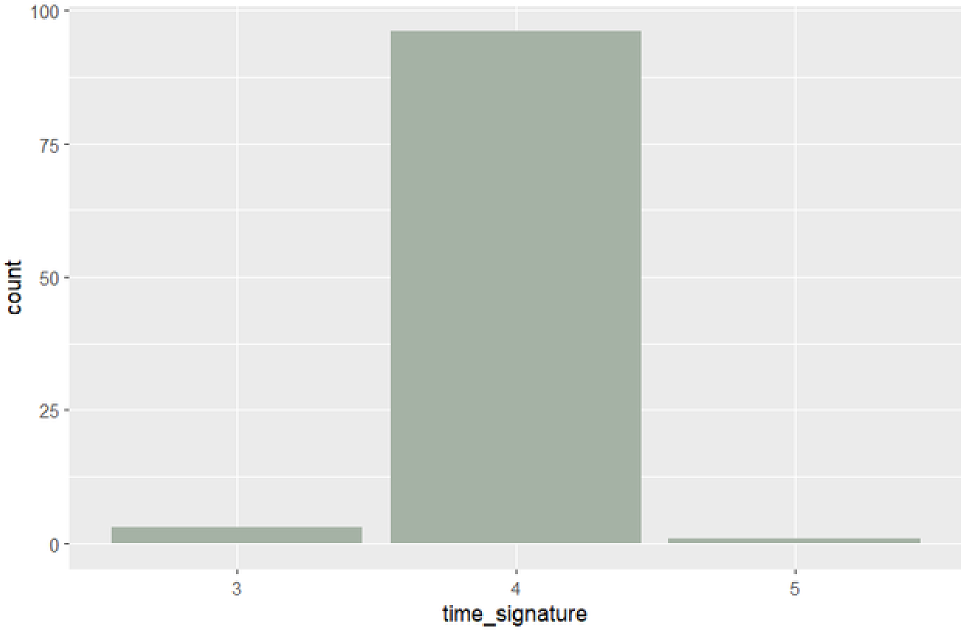
The highest intensity of energy level being greater than 0.6 i.e the measure of intensity is quite high for these songs.

VALENCE ANALYSIS



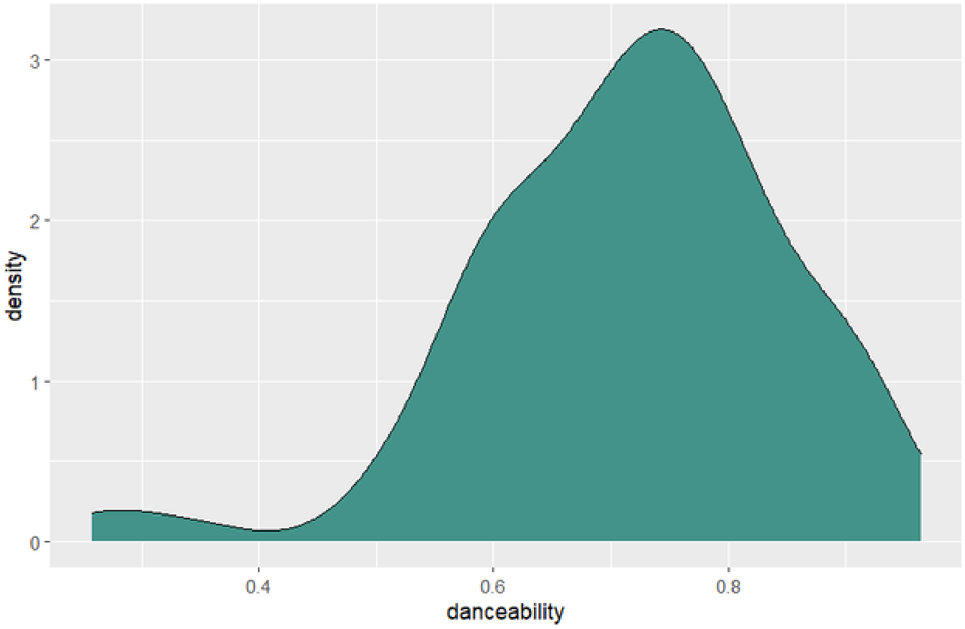
Most of the songs can be considered happy.
[Valence value < 0.350 - sad, 0.351 < Valence value < 0.701 - happy, Valence value > 0.700 - Euphoric]

TIME_SIGNATURE ANALYSIS



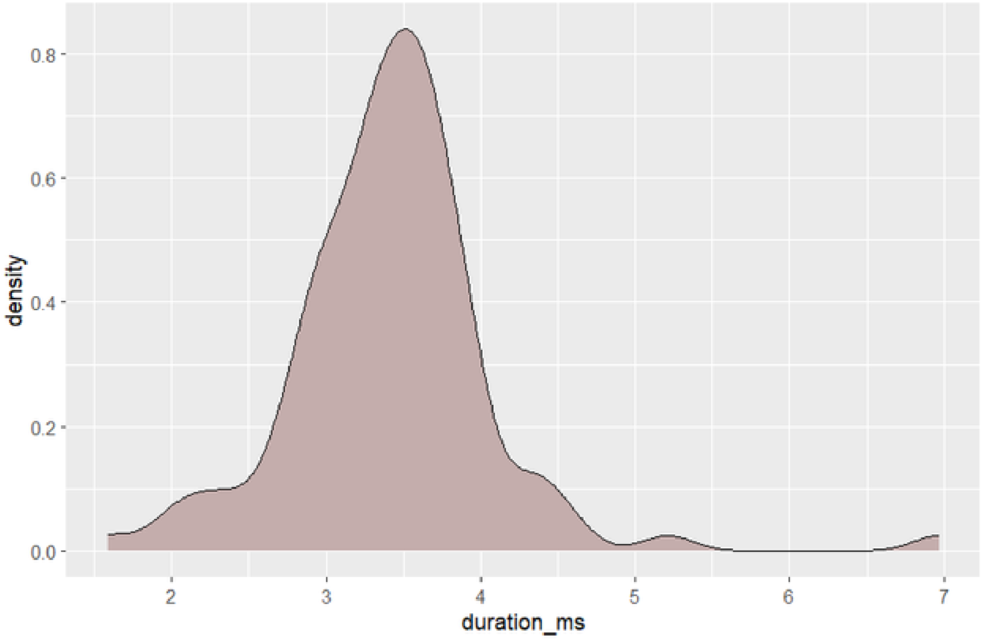
Majority the top 100 songs have a time signature of 4.

DANCEABILITY ANALYSIS



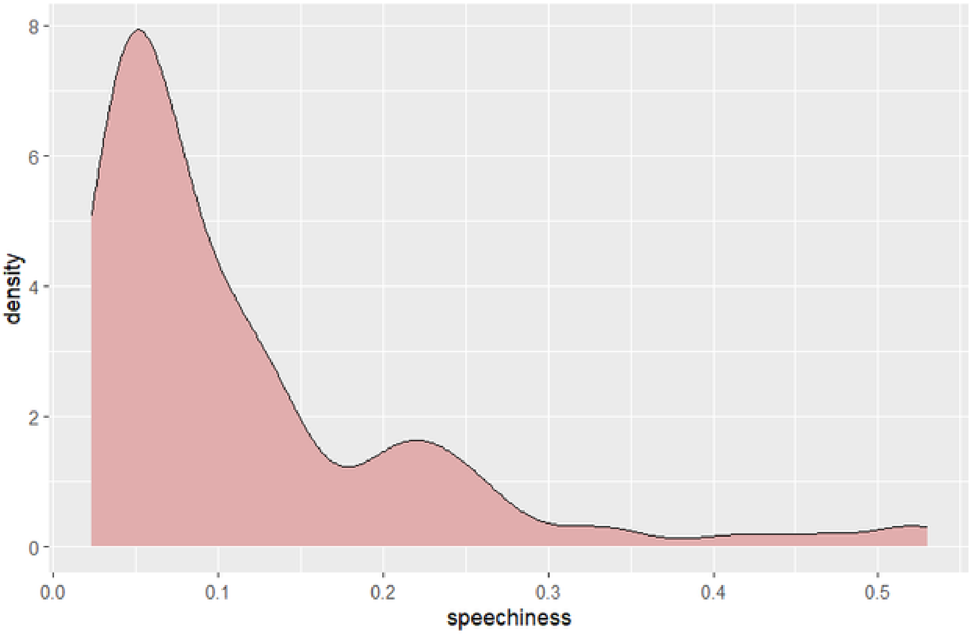
The intensity of danceability of songs is greater than 0.7 i.e danceability of these songs is high.

DURATION ANALYSIS



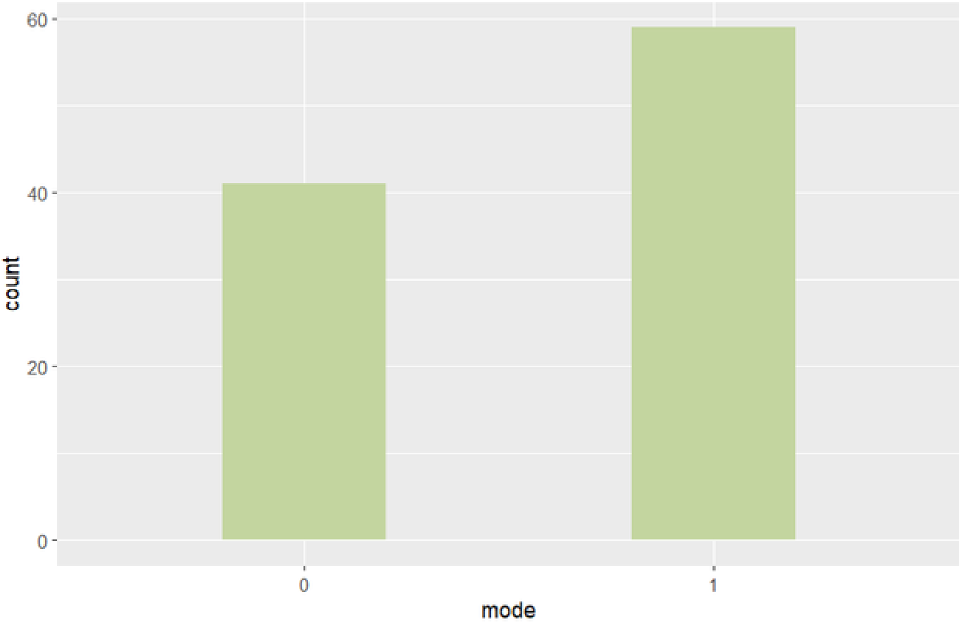
The maximum density of song duration is observed in between 3 to 4 mins.

SPEECHINESS ANALYSIS



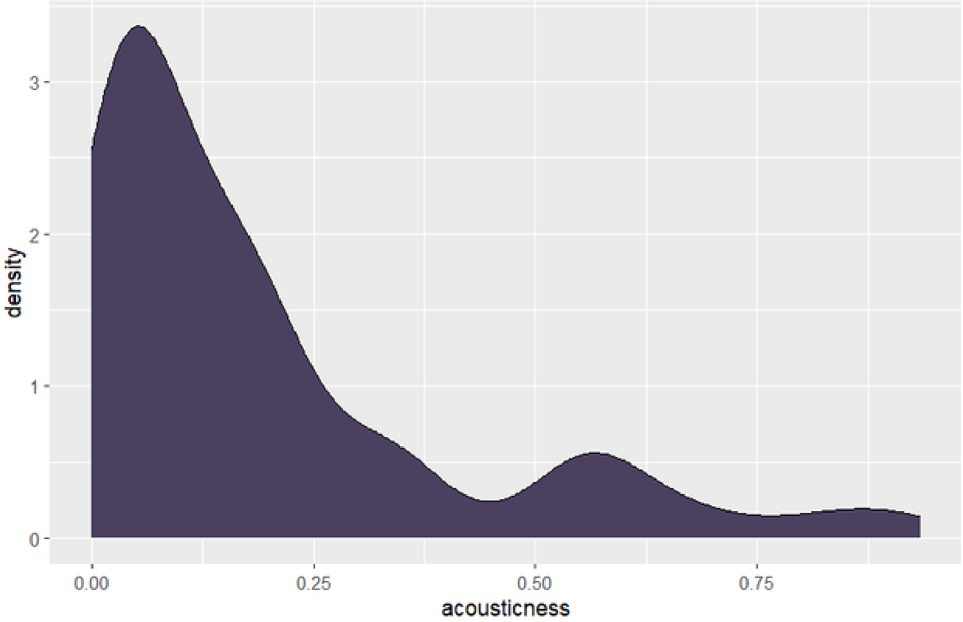
The maximum number speechiness observed here is less than 0.33 thus in the top 100 songs, most does not have speech.
[Speechiness value > 0.66 - made of spoken words, 0.33 < speechiness value < 0.66 - contains both music and words, speechiness value < 0.33 - no speech.]

MODE ANALYSIS



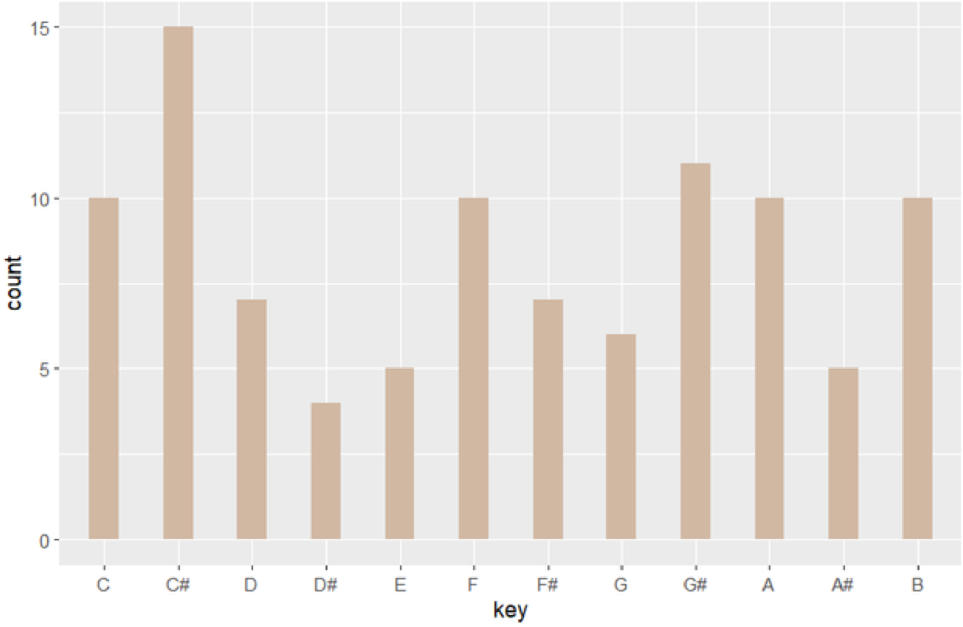
More songs have modality as 1.
[0 has around 40 counts where 1 has around 60 counts.]

ACOUSTICNESS ANALYSIS



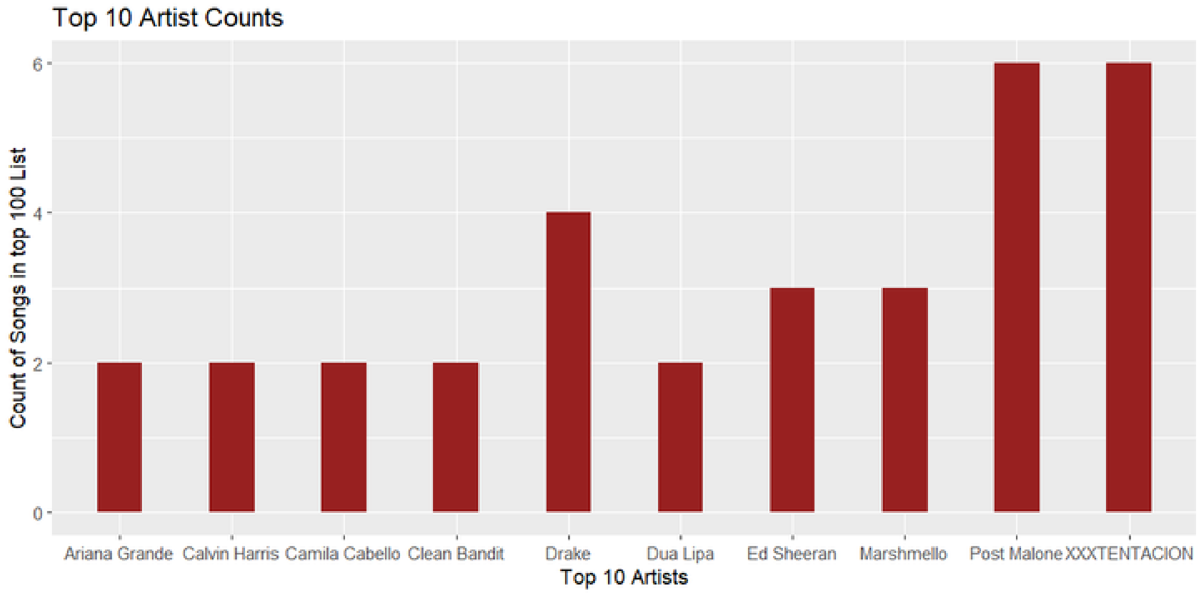
More songs have acousticness below 0.25

KEY ANALYSIS



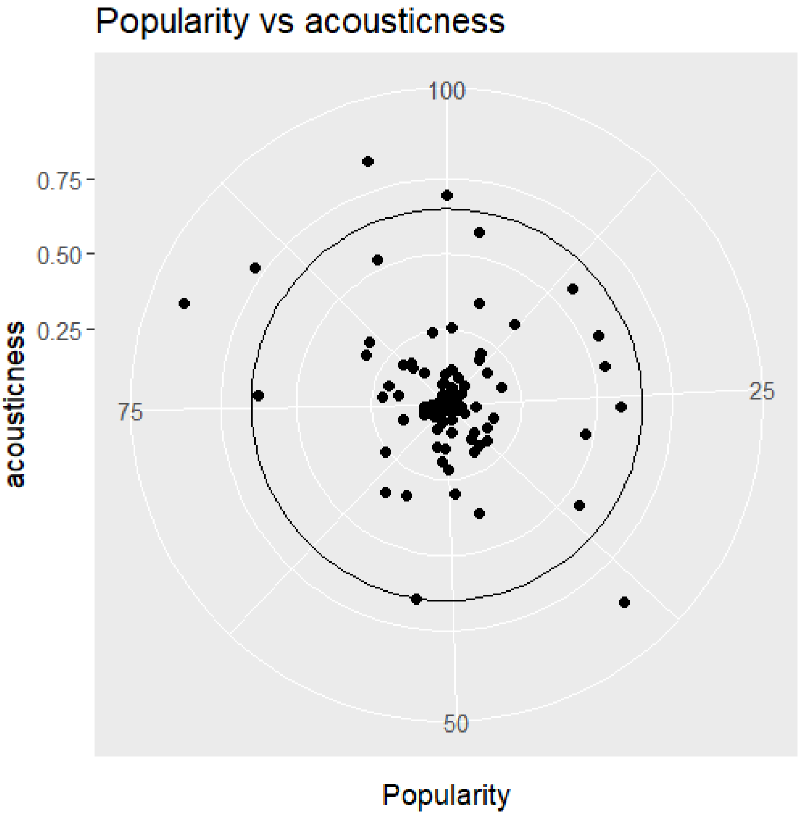
The key C# has most number of occurences.

TOP 10 ARTIST ANALYSIS



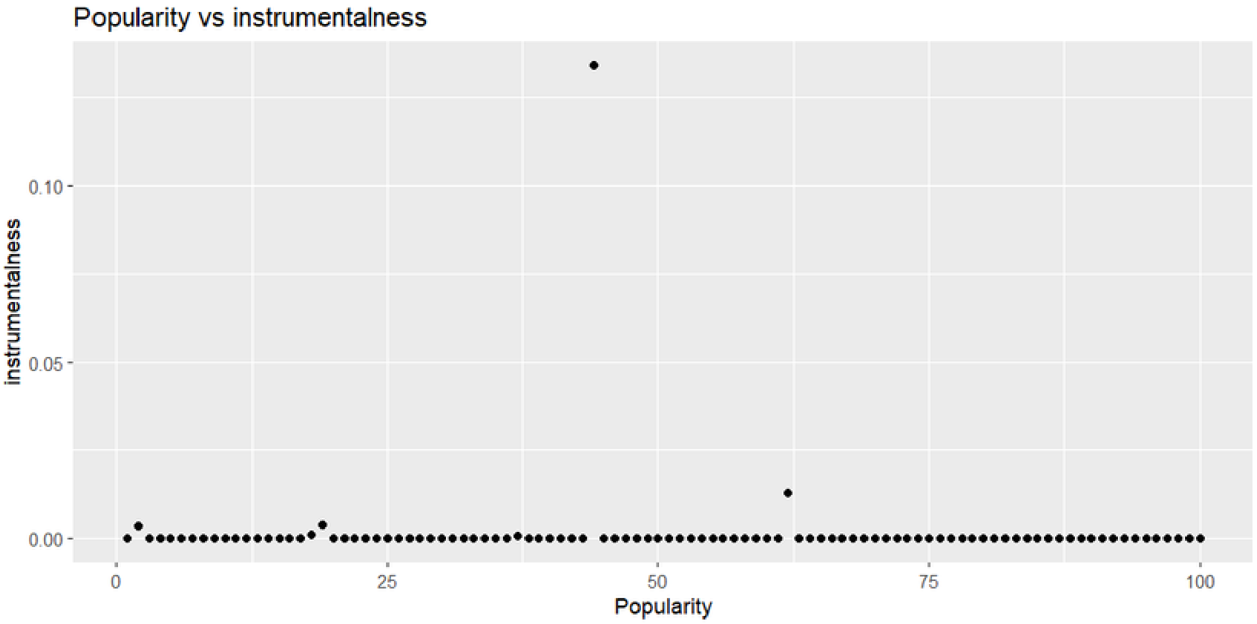
This is the frequencies for the top 10 artists in the given top 100.

POPULARITY VS ACOUSTICNESS ANALYSIS



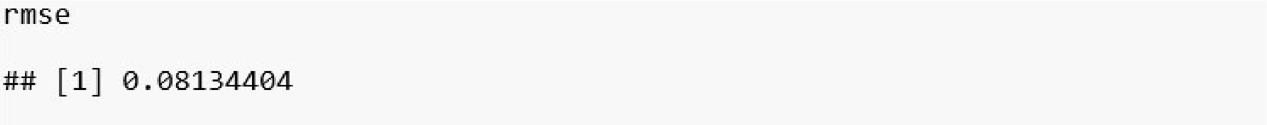
The concentration of the values is towards centre i.e approx 0.35 thus we can say that the maximum songs have inclusion of the electric sounds.

INSTRUMENTALNESS VS POPULARITY ANALYSIS



Majority songs in the top 100 have 0 or very less Instrumentalness.

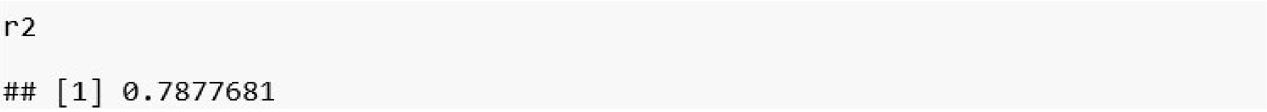
RESULT



```
rmse

## [1] 0.08134404
```

The Root Mean Squared Error (RMSE) is one of the two main performance indicators for a regression model. It provides an estimation of how well the model is able to predict the target value (accuracy).

For our model, a RMSE value of 0.08134404 is obtained indicating a very small average difference between values predicted by a model and the actual values.



```
r2

## [1] 0.7877681
```

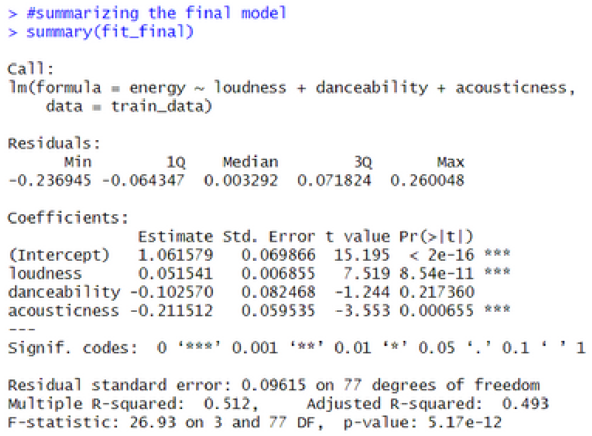
R-squared is a goodness-of-fit measure for linear regression models. This statistic indicates the percentage of the variance in the dependent variable that the independent variables explain collectively.

For our model, a R2 value of 0.7877681 is obtained indicating that the independent variables greatly explain all the variance in the dependent variable.

EVALUATION & VALIDATION

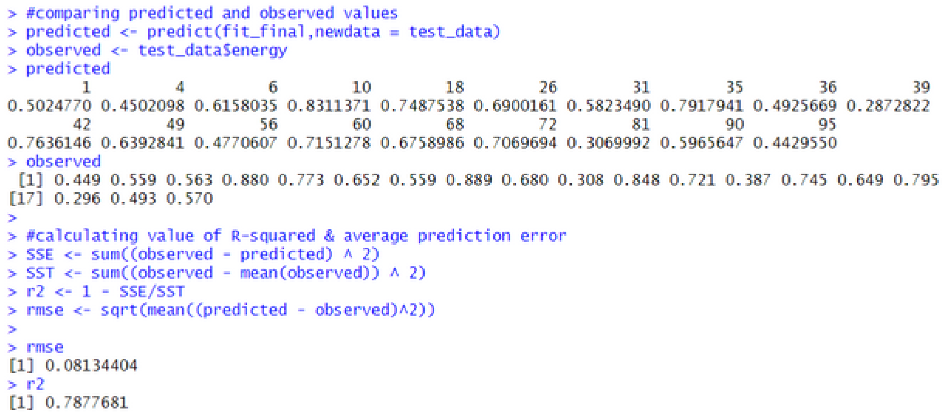
EVALUATION

1. The p-value = 5.17 e-12. The null hypothesis is rejected, so we conclude that our model is better than a model with only the intercept because at least one coefficient β is significantly different from 0.



```
> #summarizing the final model
> summary(fit_final)

Call:
lm(formula = energy ~ loudness + danceability + acousticness, 
    data = train_data)

Residuals:
      Min        1Q    Median        3Q       Max 
-0.236945 -0.064347  0.003292  0.071824  0.260048 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept)   1.061579   0.069866  15.195  < 2e-16 ***
loudness      0.051541   0.006855   7.519 8.54e-11 ***
danceability -0.102570   0.082468  -1.244 0.217360    
acousticness -0.211512   0.059535  -3.553 0.000655 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 0.09615 on 77 degrees of freedom
Multiple R-squared:  0.512,	Adjusted R-squared:  0.493 
F-statistic: 26.93 on 3 and 77 DF,  p-value: 5.17e-12
```

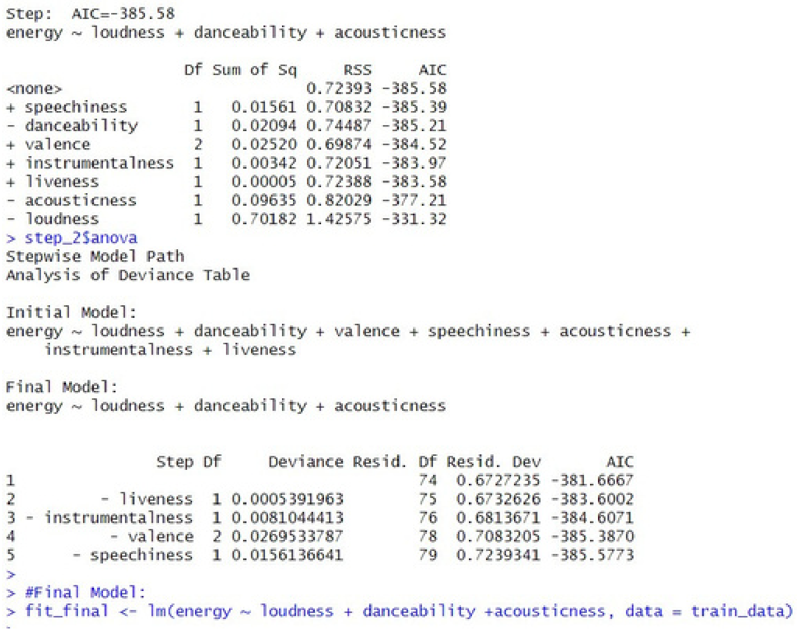
2. The higher the R2, the better the model explains the dependent variable. As a rule of thumb, a R2>0.7 indicates a good fit of the model.
The value of R2 we calculated=0.7877681



```
> #comparing predicted and observed values
> predicted <- predict(fit_final,newdata = test_data)
> observed <- test_data$energy
> predicted
        1         4         6        10        18        26        31        35        36        39 
0.5024770 0.4502098 0.6158035 0.8311371 0.7487538 0.6900161 0.5823490 0.7917941 0.4925669 0.2872822 
       42        49        56        60        68        72        81        90        95 
0.7636146 0.6392841 0.4770607 0.7151278 0.6758986 0.7069694 0.3069992 0.5965647 0.4429550 
> observed
 [1] 0.449 0.559 0.563 0.880 0.773 0.652 0.559 0.889 0.680 0.308 0.848 0.721 0.387 0.745 0.649 0.795
[17] 0.296 0.493 0.570
> 
> #calculating value of R-squared & average prediction error
> SSE <- sum((observed - predicted) ^ 2)
> SST <- sum((observed - mean(observed)) ^ 2)
> r2 <- 1 - SSE/SST
> rmse <- sqrt(mean((predicted - observed)^2))
> 
> rmse
[1] 0.08134404
> r2
[1] 0.7877681
```

EVALUATION & VALIDATION

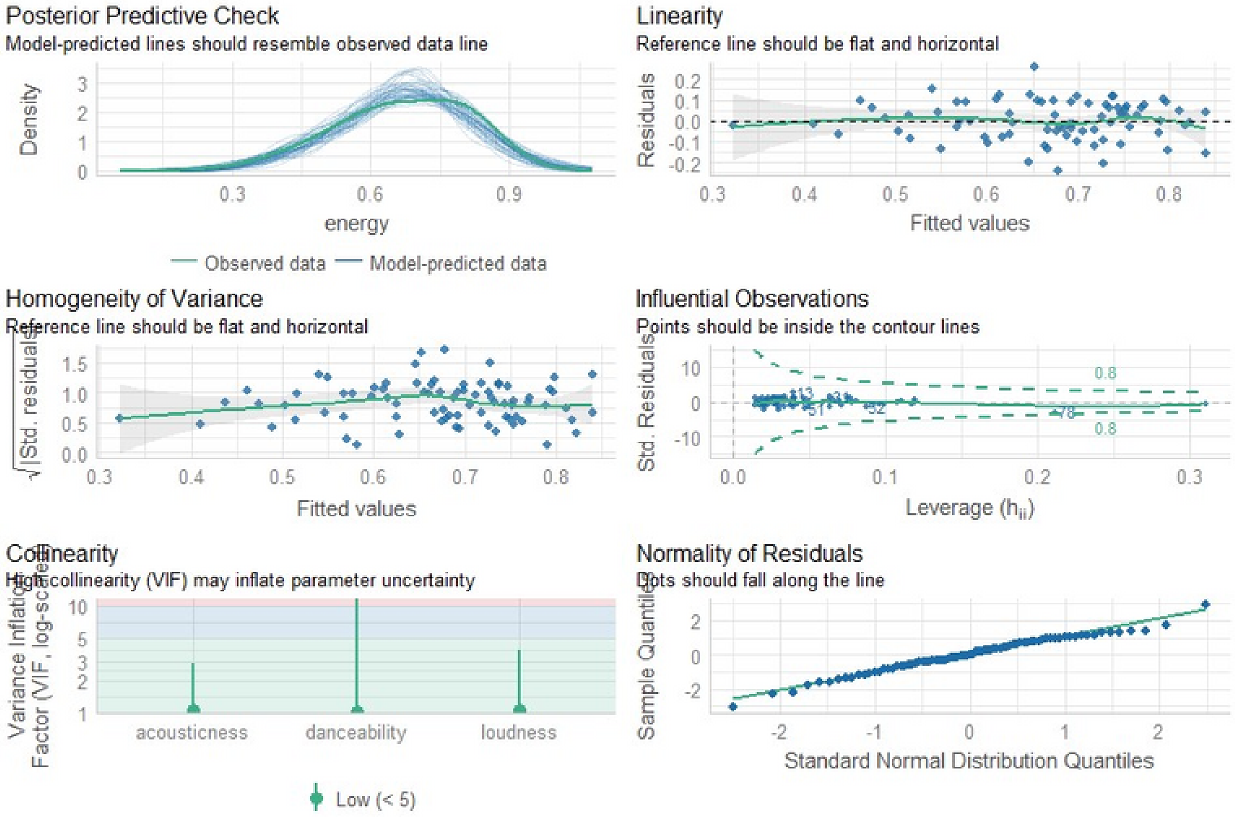
3. Out of the two ways to obtain a parsimonious model from a model with many independent variables, one is based on the Akaike Information Criterion (AIC). According to this criterion, the best model is the one with the lowest AIC. After using the stepAIC() to compare different variable sets, the set of variables with AIC = -385.58 was chosen.



```
Step:  AIC=-385.58
energy ~ loudness + danceability + acousticness

                    Df Sum of Sq     RSS     AIC
<none>                           0.72393 -385.58
+ speechiness        1   0.01561 0.70832 -385.39
- danceability       1   0.02094 0.74487 -385.21
+ valence            2   0.02520 0.69874 -384.52
+ instrumentalness   1   0.00342 0.72051 -383.97
+ liveness           1   0.00005 0.72388 -383.58
- acousticness       1   0.09635 0.82029 -377.21
- loudness           1   0.70182 1.42575 -331.32
> step_2$anova
Stepwise Model Path 
Analysis of Deviance Table

Initial Model:
energy ~ loudness + danceability + valence + speechiness + acousticness + 
    instrumentalness + liveness

Final Model:
energy ~ loudness + danceability + acousticness


                 Step Df     Deviance Resid. Df Resid. Dev       AIC
1                                           74  0.6727235 -381.6667
2          - liveness  1 0.0005391963        75  0.6732626 -383.6002
3 - instrumentalness  1 0.0081044413        76  0.6813671 -384.6071
4           - valence  2 0.0269533787        78  0.7083205 -385.3870
5       - speechiness  1 0.0156136641        79  0.7239341 -385.5773
> 
> #Final Model:
> fit_final <- lm(energy ~ loudness + danceability +acousticness, data = train_data)
```

EVALUATION & VALIDATION

VALIDATION



Based on these diagnostic plots, we see that:

- **Homogeneity** of variance (middle left plot) is respected.
- **Multicollinearity** (bottom left plot) is not an issue (I tend to use the threshold of 10 for VIF, and all of them are below 10).
- There is no **influential** points (middle right plot).
- **Normality** of the residuals (bottom right plot) is not perfect due to very few points deviating from the reference line but it still seems acceptable. In any case, the number of observations is large enough given the number of parameters and given the small deviation from normality so tests on the coefficients are (approximately) valid.
- **Linearity** (top right plot) is almost a straight line.

CONCLUSION

Look for patterns in the audio features of the songs. Why do people stream these songs the most?

1.Energetic songs
2.Time duration of the songs between 3-4 mins.
3.High dancebility.
4.Low Speechiness and low instrumentalness (very less speech in the songs)
5.Low accousticness(more inclusion of electric sounds).

See which features correlate the most

Energy and loudness correlate the most.

Try to predict one audio feature based on the others

With the above model we obtained a considerable r2 value while predicting the energy of a song using the variables loudness, danceability, acousticness.

REFERENCES

- https://www.kaggle.com/datasets/nadintamer/top-spotify-tracks-of-2018
- https://statsandr.com/blog/multiple-linear-regression-made-simple/#multiple-linear-regression
- https://www.simplilearn.com/tutorials/machine-learning-tutorial/machine-learning-steps
- https://www.geeksforgeeks.org/data-preprocessing-in-data-mining/