Final Presentation for project

Aarnav Putta and Apoorv Thite

04/23/24

```
library(dplyr)
library(tidyverse)
library(ggplot2)
```

Research Question

The primary question guiding our analysis is: "How do the characteristics of music (like tempo, valence, and energy) vary and influence a track's popularity by genre on Spotify?"

The guiding research question for the Spotify music data analysis project is: "How do the characteristics of music (like tempo, valence, and energy) vary by genre, and how do they influence a track's popularity on Spotify?" This question aims to uncover the relationship between specific musical attributes and how they correlate with the popularity of tracks across different genres on Spotify.

The purpose of this analysis is to understand the trends and preferences in music consumption over specific periods and across various musical genres. By analyzing these relationships, the project soughts to provide insights that could inform artists, producers, and marketers about the key characteristics that potentially make a track more appealing to listeners. Identifying these patterns is not only relevant for academic curiosity but also practical for enhancing music production and marketing strategies tailored to listener preferences. This understanding could help in producing music that aligns better with popular trends and possibly increase a song's success on the platform.

```
spotify_data <- read.csv(file = "SpotifyData.csv", header = TRUE, sep = ",", stringsAsFactors = FALSE)
summary(spotify_data)
```

```
##        X            track_id           artists           album_name
##  Min.   :     0   Length:114000      Length:114000      Length:114000
##  1st Qu.: 28500   Class :character   Class :character   Class :character
##  Median : 57000   Mode  :character   Mode  :character   Mode  :character
##  Mean   : 57000
##  3rd Qu.: 85499
##  Max.   :113999
##   track_name          popularity      duration_ms        explicit
##  Length:114000      Min.   :  0.00   Min.   :      0   Length:114000
##  Class :character   1st Qu.: 17.00   1st Qu.: 174066   Class :character
##  Mode  :character   Median : 35.00   Median : 212906   Mode  :character
##                     Mean   : 33.24   Mean   : 228029
##                     3rd Qu.: 50.00   3rd Qu.: 261506
##                     Max.   :100.00   Max.   :5237295
##   danceability        energy            key            loudness
##  Min.   :0.0000   Min.   :0.0000   Min.   : 0.000   Min.   :-49.531
##  1st Qu.:0.4560   1st Qu.:0.4720   1st Qu.: 2.000   1st Qu.:-10.013
##  Median :0.5800   Median :0.6850   Median : 5.000   Median : -7.004
##  Mean   :0.5668   Mean   :0.6414   Mean   : 5.309   Mean   : -8.259
##  3rd Qu.:0.6950   3rd Qu.:0.8540   3rd Qu.: 8.000   3rd Qu.: -5.003
##  Max.   :0.9850   Max.   :1.0000   Max.   :11.000   Max.   :  4.532
##       mode         speechiness       acousticness    instrumentalness
##  Min.   :0.0000   Min.   :0.00000   Min.   :0.0000   Min.   :0.00e+00
```

```
##  1st Qu.:0.0000   1st Qu.:0.03590   1st Qu.:0.0169   1st Qu.:0.00e+00
##  Median :1.0000   Median :0.04890   Median :0.1690   Median :4.16e-05
##  Mean   :0.6376   Mean   :0.08465   Mean   :0.3149   Mean   :1.56e-01
##  3rd Qu.:1.0000   3rd Qu.:0.08450   3rd Qu.:0.5980   3rd Qu.:4.90e-02
##  Max.   :1.0000   Max.   :0.96500   Max.   :0.9960   Max.   :1.00e+00
##     liveness         valence           tempo        time_signature
##  Min.   :0.0000   Min.   :0.0000   Min.   :  0.00   Min.   :0.000
##  1st Qu.:0.0980   1st Qu.:0.2600   1st Qu.: 99.22   1st Qu.:4.000
##  Median :0.1320   Median :0.4640   Median :122.02   Median :4.000
##  Mean   :0.2136   Mean   :0.4741   Mean   :122.15   Mean   :3.904
##  3rd Qu.:0.2730   3rd Qu.:0.6830   3rd Qu.:140.07   3rd Qu.:4.000
##  Max.   :1.0000   Max.   :0.9950   Max.   :243.37   Max.   :5.000
##  track_genre
##  Length:114000
##  Class :character
##  Mode  :character
##
##
##
```

Key Plots

Here are some key plots as asked:

How tempo affects popularity ?

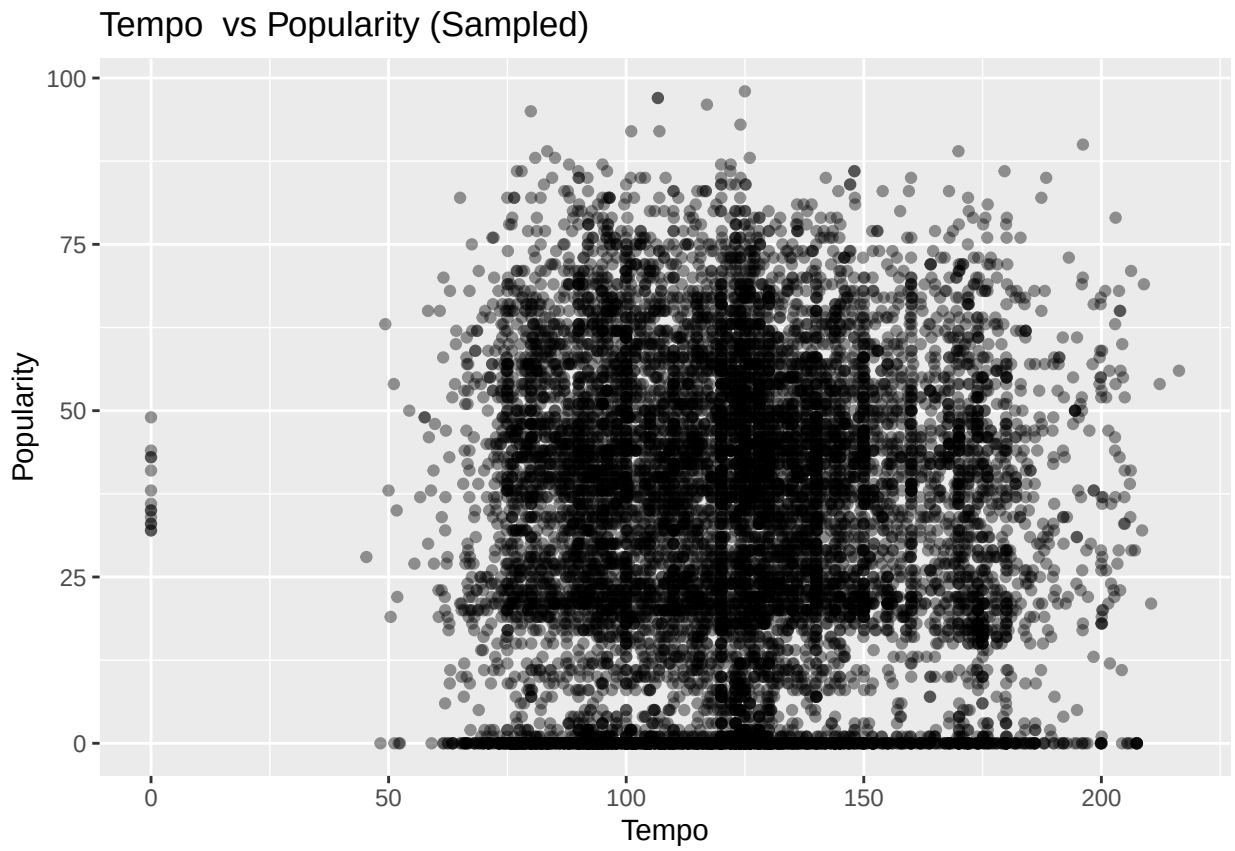
Since our topic tries to explore how these various charateristics of music affect the popularity , we chose a couple of those most renowned characteristics among the given ones in the data sets. What made these characteristics stand out so much was the standard deviation among their values. Their values vastly varied among different genres making us want to understand the correlation they share with popularity better through visual plots.

```
spotify_data <- read.csv(file = "SpotifyData.csv", header = TRUE, sep = ",", stringsAsFactors = FALSE)
nrow(spotify_data)
```

```
## [1] 114000
```

```
# sample 10% of the data as an approximate
sampled_data <- sample_frac(spotify_data, 0.1)

ggplot(sampled_data, aes(x = tempo, y = popularity)) +
  geom_point(alpha=0.4) +
  labs(title = "Tempo  vs Popularity (Sampled)", x = "Tempo", y = "Popularity")
```



Tempo also known as beats per minute is a key characteristic of these songs , since it decides the flow of these songs. Our plot helps us understand that the highest density of popularity lies around the middle. Now, from this we were able to takeaway and interesting hypothesis that suggested that , the general populace seems to prefer tempo that is not too fast paced , nor too slow paced. This suggest around 100-150 bpm would be the optimal tempo to achieve high popularity on spotify.We would not be able to reach this conclusion without the plot.

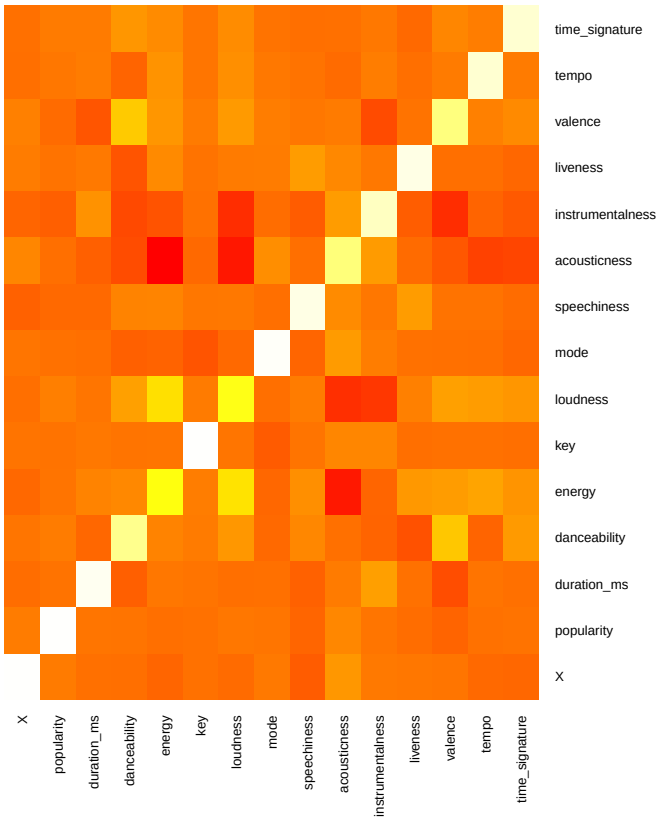
Another keyplot

Now in this plot we have a heat map that incorporates the correlation matrix which is going to show the correlation between the different variables.The lighter squares indicate weaker correlations, while darker squares represent stronger correlations. White or the lightest color squares indicate very weak or no correlation.Features with darker squares along the same row or column may have stronger correlations with each other. For instance, 'danceability' might have a stronger correlation with 'energy' (if we assume darker colors represent stronger positive correlations). The diagonal is typically the brightest because it represents the correlation of a variable with itself, which is always perfect .Clusters of darker or lighter colors can indicate groups of features that are more or less related to each other. This can be useful for feature selection in machine learning models or for understanding which features influence each other. If there are any squares that stand out from the surrounding colors , this could indicate an outlier in terms of correlation that might be interesting to investigate further.

```
spotify_data <- read.csv("SpotifyData.csv", stringsAsFactors = FALSE)

correlation_matrix <- cor(spotify_data[sapply(spotify_data, is.numeric)])


heatmap(correlation_matrix, Rowv = NA, Colv = NA, col = heat.colors(256), scale="column",
        margins = c(5,10), cexRow = 0.5, cexCol = 0.5)
```



Key insight/takeaway about research question - Summarize the key insight, takeaway, conclusion to the research question that motivated your analysis

`The key insight from the exploratory data analysis (EDA) project on Spotify music data focuses on understandin`

The analysis revealed correlations between various musical attributes and track popularity. For instance, danceability and loudness were identified as significant factors in a song's popularity. We observed that tracks with higher danceability scores tended to be more popular, suggesting that more danceable music is generally preferred by listeners on Spotify.

The analysis also touched upon the variation in music characteristics across genres, with preliminary findings indicating distinct preferences in energy levels and tempo across different styles of music. The use of plots, such as those comparing tempo and popularity, helped in visualizing these trends, offering a clearer understanding of what makes a track popular in different musical contexts.

The conclusion drawn from this analysis is that certain musical characteristics significantly influence the popularity of tracks on Spotify, which can inform music producers and marketers about the prevailing music preferences and trends. This insight is particularly useful for tailoring music production to align with listener preferences, potentially increasing a track's success on the platform.

Challenge Encountered - Describe the biggest challenge that you encountered and how you overcome it in the project.

The biggest challenge we encountered was dealing with the large volume of data—114,000 cases, each representing a unique song with multiple attributes. This high volume posed significant issues in terms of data handling and processing speed, which could potentially lead to inefficient analysis and longer processing times. To overcome this challenge, we employed a sampling strategy. Specifically, we sampled 10% of the data to create a manageable subset that could approximate the characteristics of the entire dataset. This approach allowed us to efficiently test their hypotheses and perform exploratory data analysis (EDA) without the computational overhead of handling the entire dataset at once.

Code Appendix

```
# This template file is based off of a template created by Alex Hayes
# https://github.com/alexpghayes/rmarkdown_homework_template

# Setting Document Options
knitr::opts_chunk$set(
  echo = TRUE,
  warning = FALSE,
  message = FALSE,
  fig.align = "center"
)
library(dplyr)
library(tidyverse)
library(ggplot2)

spotify_data <- read.csv(file = "SpotifyData.csv", header = TRUE, sep = ",", stringsAsFactors = FALSE)
summary(spotify_data)
spotify_data <- read.csv(file = "SpotifyData.csv", header = TRUE, sep = ",", stringsAsFactors = FALSE)
nrow(spotify_data)

# sample 10% of the data as an approximate
sampled_data <- sample_frac(spotify_data, 0.1)

ggplot(sampled_data, aes(x = tempo, y = popularity)) +
  geom_point(alpha=0.4) +
  labs(title = "Tempo  vs Popularity (Sampled)", x = "Tempo", y = "Popularity")
spotify_data <- read.csv("SpotifyData.csv", stringsAsFactors = FALSE)

correlation_matrix <- cor(spotify_data[sapply(spotify_data, is.numeric)])


heatmap(correlation_matrix, Rowv = NA, Colv = NA, col = heat.colors(256), scale="column",
        margins = c(5,10), cexRow = 0.5, cexCol = 0.5)
```