# Spotify Dataset 2023

NK

2024-03-11

```
library(tidyverse)
library(stringr)
library(patchwork)
```

### Introduction

**The Spotify Dataset 2023 consists of data from Kaggle database. We will be doing analysis on music trends in relation to artist, album and track data, and how each variable affects one another.**

```
# Read the CSV files
spotify_artist <- read.csv("../Data/Spotify Dataset 2023/spotify_artist_data_2023.csv")
spotify_data <- read.csv("../Data/Spotify Dataset 2023/spotify_data_12_20_2023.csv")
```

```
#Removing columns unlikely needed
spotify_data1 = spotify_data %>%
  select(-artist_0, -artist_1, -artist_2, -artist_3, -artist_4, -label, -release_date, -total_tracks, -
```

## Artist Insights:

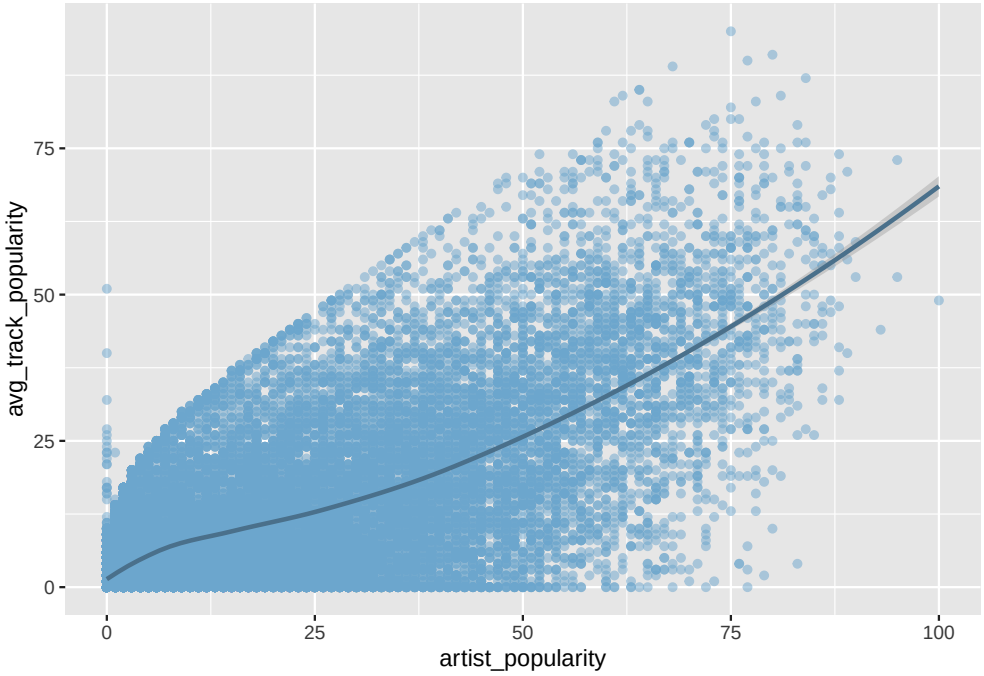
### Artist Popularity vs Track Popularity

Analysis of whether there is a correlation between the main artist's popularity and the average popularity of their songs

```
popularity = spotify_data1 %>%
  select(track_id,track_popularity, name, artist_id, artist_popularity) %>%
  group_by(artist_id) %>%
  mutate(avg_track_popularity = as.integer(mean(track_popularity))) %>%
  arrange(desc(avg_track_popularity)) %>%
  select(artist_id, name, artist_popularity, avg_track_popularity) %>%
  distinct()

head(popularity)
```

```
## # A tibble: 6 x 4
## # Groups:   artist_id [6]
##   artist_id              name           artist_popularity avg_track_popularity
##   <chr>                  <chr>                      <int>                 <int>
## 1 6dLuQ5qXxIuWc5urxfIiZR Calle 24                      75                    95
## 2 5Pwc4xIPtQLFEnJriah9YJ OneRepublic                   80                    91
## 3 46pWGuE3dSwY3bMMXGBvVS Rema                          77                    90
## 4 027TpXKGwdXP7iwbjUSpV8 The Walters                   68                    89
## 5 6XkjpgcEsYab502Vr1bBeW Grupo Frontera                84                    87
## 6 1JvbNeV9zG9Sew1JyaWsyx Anggi Marito                  64                    85
```

```
ggplot(popularity, aes(x = artist_popularity, y = avg_track_popularity)) +
  geom_point(color = "skyblue3", alpha = 0.5) +
  geom_smooth(method = "loess", color = "skyblue4")
```




Conclusion: The more popular the artist is, the higher the average popularity of their songs, which is logical

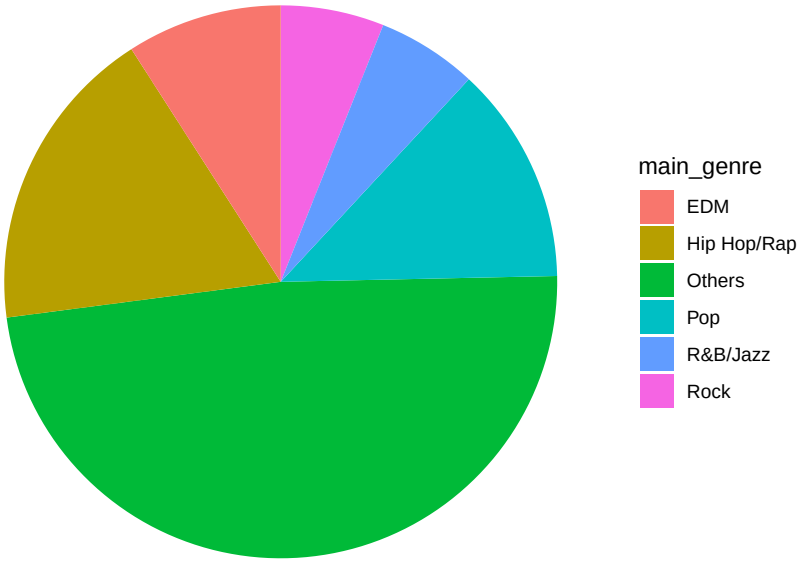
## Distribution of main genres for all artists

Focusing on the most popular genres, Pop, Rock, R&B/Jazz, Hip Hop/Rap and EDM

Cases in order of specificity of the genre and keywords as some keywords do overlap due to blended genres: 1. EDM (KW: edm, trap, dubstep, electro, trance, techno, house) 2. Hip Hop/Rap (KW: hop, rap, phonk, drill, lo-fi) 3. R&B/Jazz (KW: r&b, soul, funk, jazz) 4. Rock (KW: rock, metal) 5. Pop (KW: pop) 6. Others

```
genre_count = spotify_artist %>%
  select(id, name, genre_0) %>%
  filter(str_length(trimws(genre_0)) > 0) %>%
  group_by(genre_0) %>%
```

```
  mutate(count = n_distinct(id)) %>%
  ungroup() %>%
  select(genre_0, count) %>%
  distinct() %>%
  arrange(desc(count))

genre_sorting = genre_count %>%
  mutate(main_genre = case_when (
    str_detect(genre_0, "edm|trap|dubstep|electro|trance|techno|house") ~ "EDM",
    str_detect(genre_0, "hop|rap|phonk|drill|lo-fi") ~ "Hip Hop/Rap",
    str_detect(genre_0, "r&b|soul|funk|jazz") ~ "R&B/Jazz",
    str_detect(genre_0, "rock|metal") ~ "Rock",
    str_detect(genre_0, "pop") ~ "Pop",
    TRUE ~ "Others"  # For other cases not matching, assign "Others"
  )) %>%
  group_by(main_genre) %>%
  mutate(total_count = sum(count)) %>%
  select(main_genre, total_count) %>%
  distinct()

ggplot(genre_sorting, aes(x="", y= total_count, fill= main_genre)) +
  geom_bar(stat="identity", width=1) +
  coord_polar("y", start=0) +
  theme_void()
```



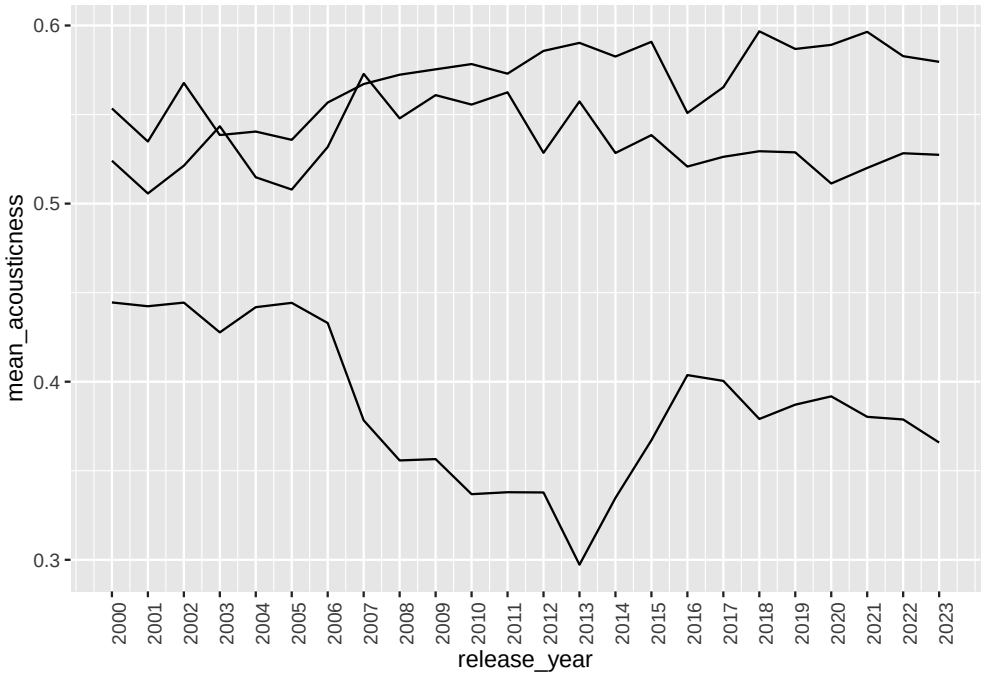

```
#https://r-graph-gallery.com/piechart-ggplot2.html
```

Largest genre is Hip Hop/Rap, followed by Pop. Significant number of genres that don't classify under the top 5 genres, which includes tracks such as instrumentals, karaokes, lullabies, ASMR, and even nature environmental sounds.

## Track Characteristics:

Audio Feature Trends: Track how overall danceability, energy, acousticness, etc., have evolved over the years from 2000 to 2023

```
overall_track_trends = spotify_data1 %>%
  select(track_name, acousticness, danceability, energy, release_year) %>%
  filter(release_year %in% c(2000:2023)) %>%
  group_by(release_year) %>%
  mutate(mean_acousticness = mean(acousticness, na.rm = TRUE),
         mean_danceability = mean(danceability, na.rm = TRUE),
         mean_energy = mean(energy, na.rm = TRUE)) %>%
  ungroup() %>%
  select(release_year, mean_acousticness, mean_danceability, mean_energy) %>%
  distinct() %>%
  arrange(desc(release_year))

ggplot(data = overall_track_trends, aes(x = release_year)) +
  geom_line(aes(y = mean_acousticness)) +
  geom_line(aes(y = mean_danceability)) +
  geom_line(aes(y = mean_energy)) +
  scale_x_continuous(breaks = seq(2000,2023,1), limits = c(2000, 2023)) +
  #scale_y_continuous(breaks = seq(0.0, 1.0, 0.2), limits = c(0.0,1.0))
  theme(axis.text.x = element_text(angle = 90, hjust = 1))
```




Track how danceability, energy, acousticness, etc., have evolved across the different genres Pop, Rock, R&B/Jazz, Hip Hop/Rap and EDM

```
overall_track_trends_bygenre = spotify_data1 %>%
  select(track_name, acousticness, danceability, energy, release_year, genre_0) %>%
```

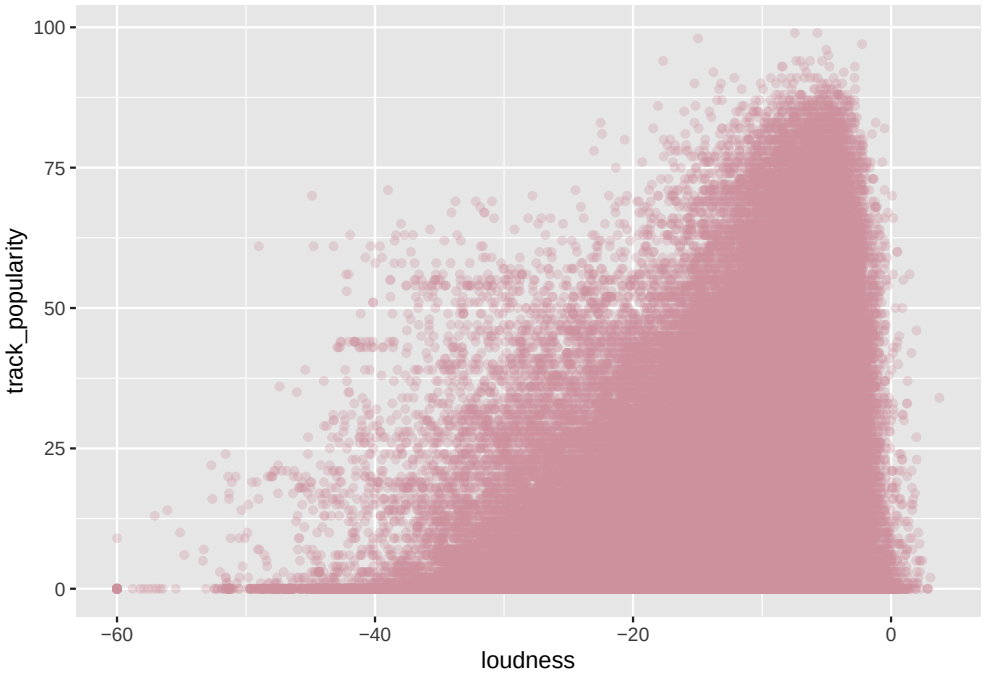
```
  filter(str_length(trimws(genre_0)) > 0) %>%
  mutate(main_genre = case_when (
    str_detect(genre_0, "edm|trap|dubstep|electro|trance|techno|house") ~ "EDM",
    str_detect(genre_0, "hop|rap|phonk|drill|lo-fi") ~ "Hip Hop/Rap",
    str_detect(genre_0, "r&b|soul|funk|jazz") ~ "R&B/Jazz",
    str_detect(genre_0, "rock|metal") ~ "Rock",
    str_detect(genre_0, "pop") ~ "Pop",
    TRUE ~ "Others"  # For other cases not matching, assign "Others"
  )) %>%
  group_by(release_year, main_genre) %>%
  mutate(mean_acousticness = mean(acousticness, na.rm = TRUE),
         mean_danceability = mean(danceability, na.rm = TRUE),
         mean_energy = mean(energy, na.rm = TRUE)) %>%
  select(mean_acousticness, mean_danceability, mean_energy, release_year, main_genre) %>%
  ungroup() %>%
  distinct() %>%
  filter(release_year %in% c(2000:2023), main_genre != "Others") %>%
  arrange(desc(release_year))

ggplot(data = overall_track_trends_bygenre, aes(x = release_year, group = 1)) +
  geom_line(aes(y = mean_acousticness), color = "red") +
  geom_line(aes(y = mean_danceability), color = "blue") +
  geom_line(aes(y = mean_energy), color = "green") +
  facet_wrap(~main_genre, scales = "free", strip.position = "bottom") +
  scale_x_continuous(breaks = seq(2000,2023,5), limits = c(2000, 2023)) +
  scale_y_continuous(breaks = seq(0.0, 1.0, 0.2), limits = c(0.0,1.0))
```


```
  #theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

## Correlation Analysis: Investigate relationships between acoustic features (loudness, tempo, etc.) and song popularity.

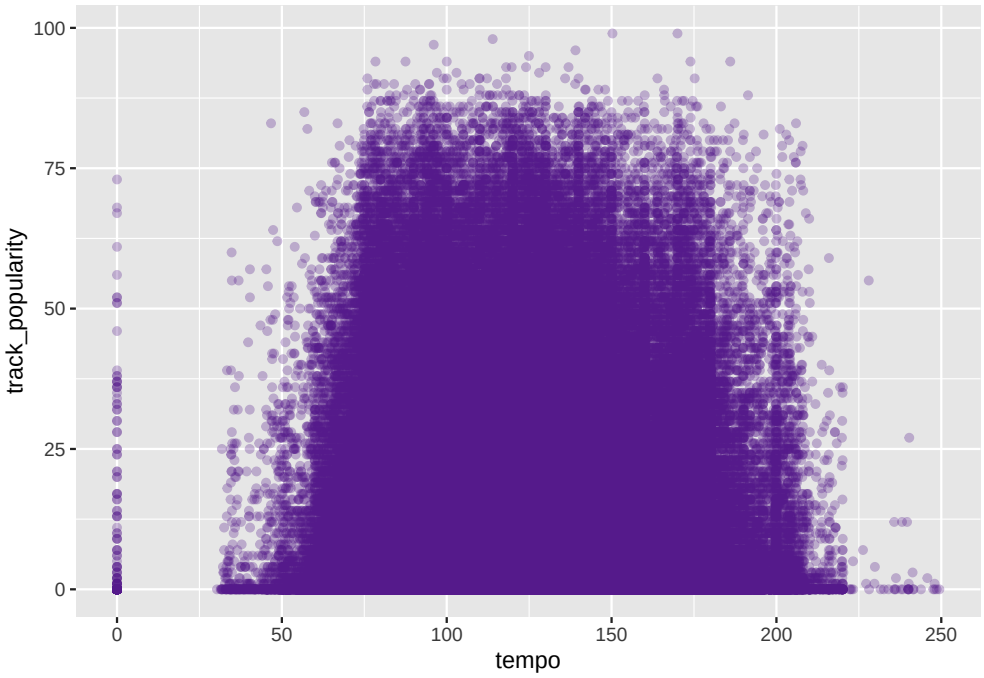
loudness vs song popularity

```
spotify_data1 %>%
  select(track_id, track_name, loudness, track_popularity) %>%
  filter(!is.na(loudness)) %>%
  ggplot(aes(x = loudness, y = track_popularity)) +
  geom_point(color = "pink3", alpha = 0.3)
```




tempo vs song popularity

```
spotify_data1 %>%
  select(track_id, track_name, tempo, track_popularity) %>%
  filter(!is.na(tempo)) %>%
  ggplot(aes(x = tempo, y = track_popularity)) +
  geom_point(color = "purple4", alpha = 0.3)
```

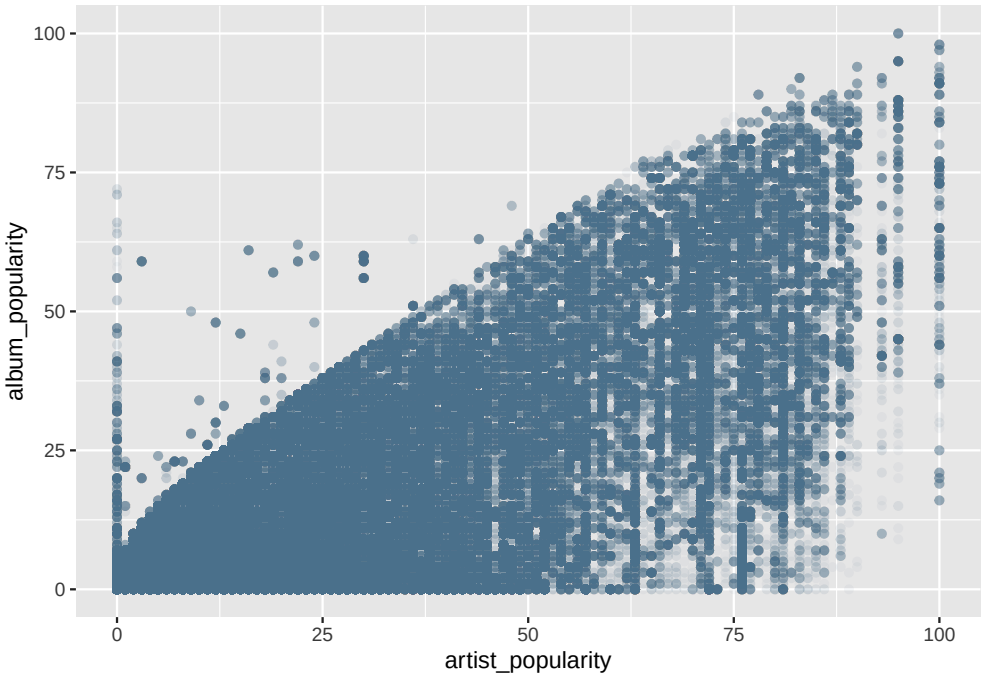



#Album Analysis:

Album Popularity vs. Artist Popularity: Check if there's a correlation between an album's popularity and the popularity of the artist associated with it.

album popularity vs artist popularity

```
spotify_data1 %>%
  select(album_id, album_popularity, artist_popularity) %>%
  ggplot(aes(x = artist_popularity, album_popularity)) +
  geom_point(color = "skyblue4", alpha = 0.05)
```




## Taylor Swift Case Study

What is the distribution of genres of her songs?

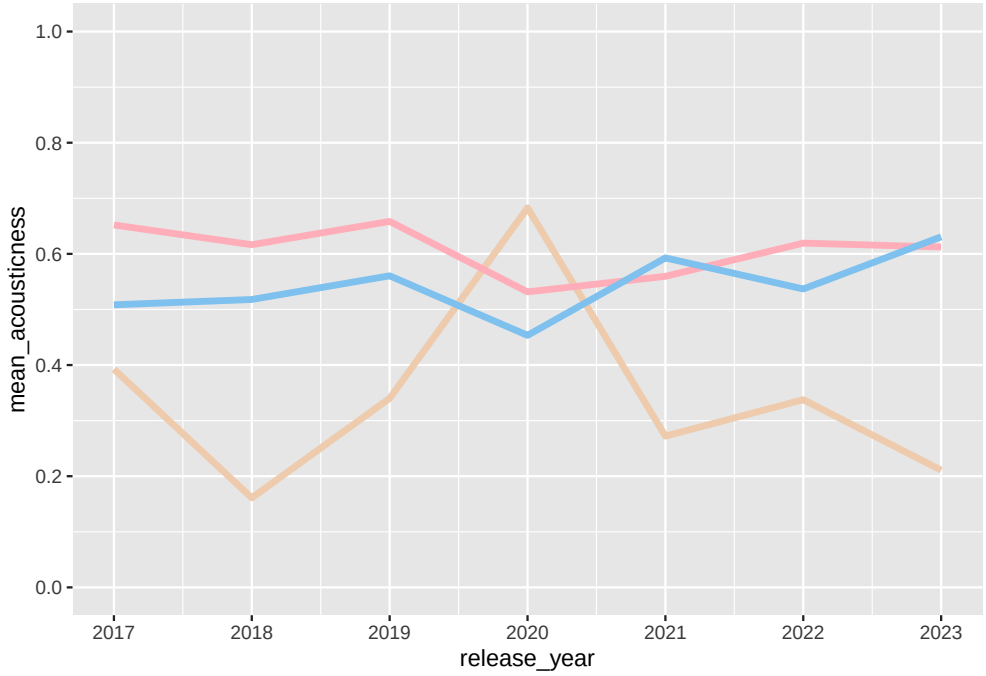
How the distribution of her song acousticness, danceability, and energy changes across the release years.

```
ts_mean_data = spotify_data1 %>%
  filter(name == "Taylor Swift") %>%
  select(track_id, track_name, acousticness, danceability, energy, release_year, album_type) %>%
  # inaccuracy due to repeat songs in both normal album and deluxe album, would cause duplicated values
  # distinct track_name only, note: identical songs have slightly differing acousticness, danceability
  # hence take the avg of the 3 cols
  group_by(track_name) %>%
  mutate(acousticness2 = mean(acousticness), danceability2 = mean(danceability), energy2 = mean(energy)
  ungroup() %>%
  distinct(track_name, .keep_all = TRUE) %>%
  group_by(release_year) %>%
  mutate(mean_acousticness = mean(acousticness2), mean_danceability = mean(danceability2), mean_energy =
  select(release_year, mean_acousticness, mean_danceability, mean_energy) %>%
  distinct() %>%
  arrange(desc(release_year)) %>%
  #since there's missing release years such as 2016, years where she did not release any songs
  #do 2017 to 2023
  filter(release_year %in% c(2017,2018,2019,2020,2021,2022,2023)) #%>%
  #mutate(release_year = as.factor(release_year))
```

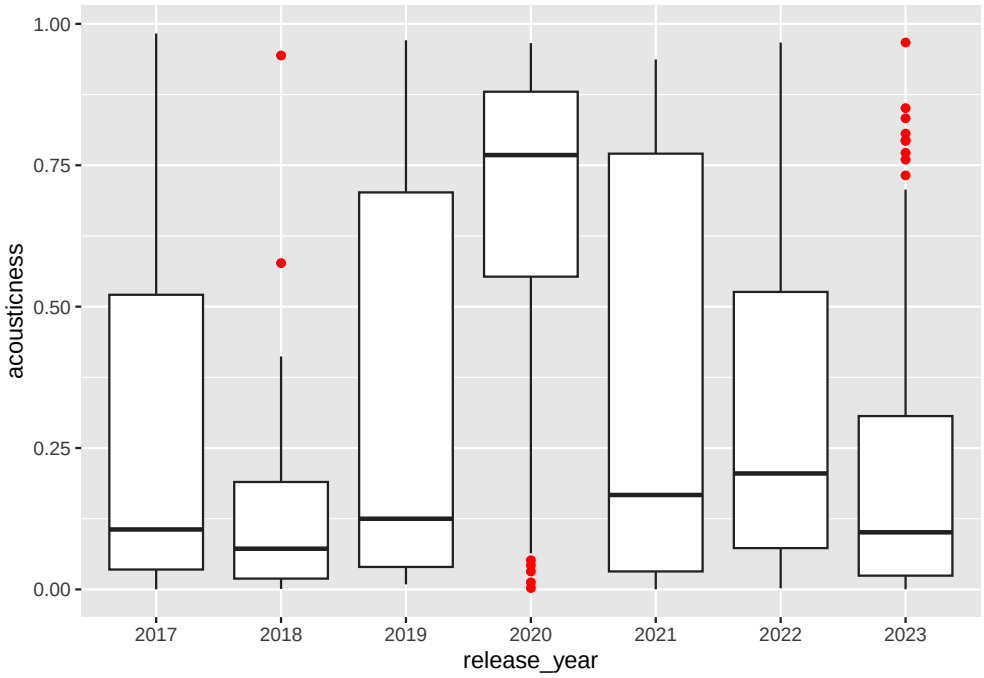
```
#Taylor Swift's separation from Big Machine Label Group occurred in November 2018.
ts_raw_data = spotify_data1 %>%
  filter(name == "Taylor Swift") %>%
  select(track_id, acousticness, danceability, energy, release_year) %>%
```

```
  filter(release_year %in% c(2017,2018,2019,2020,2021,2022,2023)) %>%
  mutate(release_year = as.factor(release_year))

 # ggplot(aes(x = release_year, y = energy)) +
 # geom_boxplot(outlier.color = "red")
```

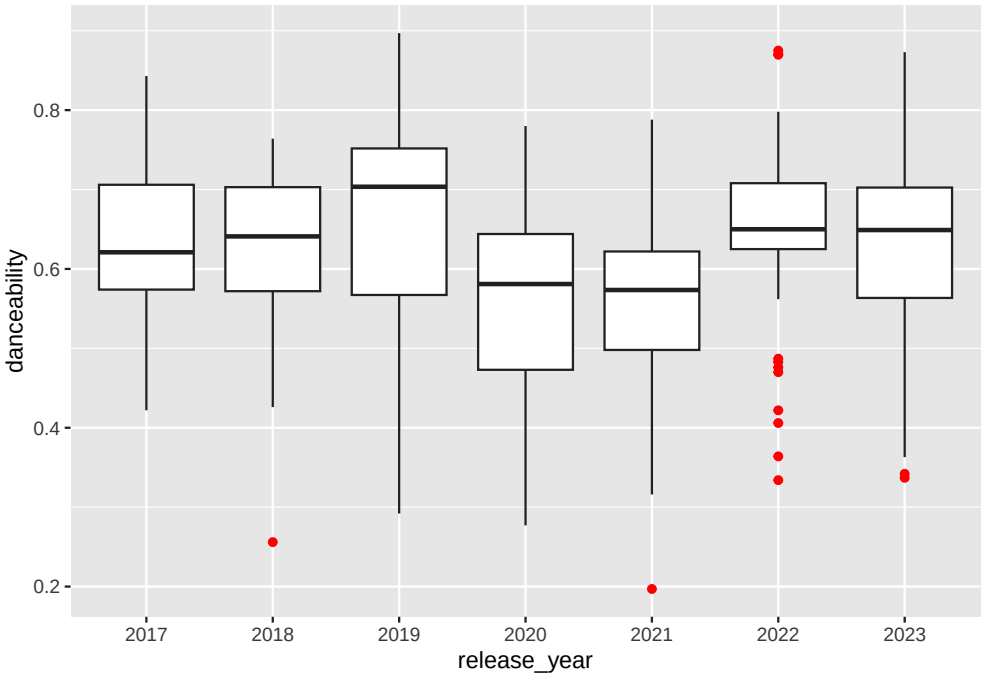
```
p1 = ggplot(data = ts_mean_data, aes(x = release_year)) +
  geom_line(aes(y = mean_acousticness), color = "peachpuff2", linewidth = 1.5) +
  geom_line(aes(y = mean_danceability), color = "lightpink1", linewidth = 1.5) +
  geom_line(aes(y = mean_energy), color = "skyblue2", linewidth = 1.5) +
  scale_x_continuous(breaks = seq(2017,2023,1), limits = c(2017, 2023)) +
  scale_y_continuous(breaks = seq(0.0, 1.0, 0.2), limits = c(0.0,1.0))


p2 = ggplot(data = ts_raw_data, aes(x = release_year)) +
  geom_boxplot(aes(x = as.factor(release_year), y = acousticness), outlier.color = "red")

p3 = ggplot(data = ts_raw_data, aes(x = release_year)) +
  geom_boxplot(aes(x = as.factor(release_year), y = danceability), outlier.color = "red")

p4 = ggplot(data = ts_raw_data, aes(x = release_year)) +
  geom_boxplot(aes(x = as.factor(release_year), y = energy), outlier.color = "red")

p1
```



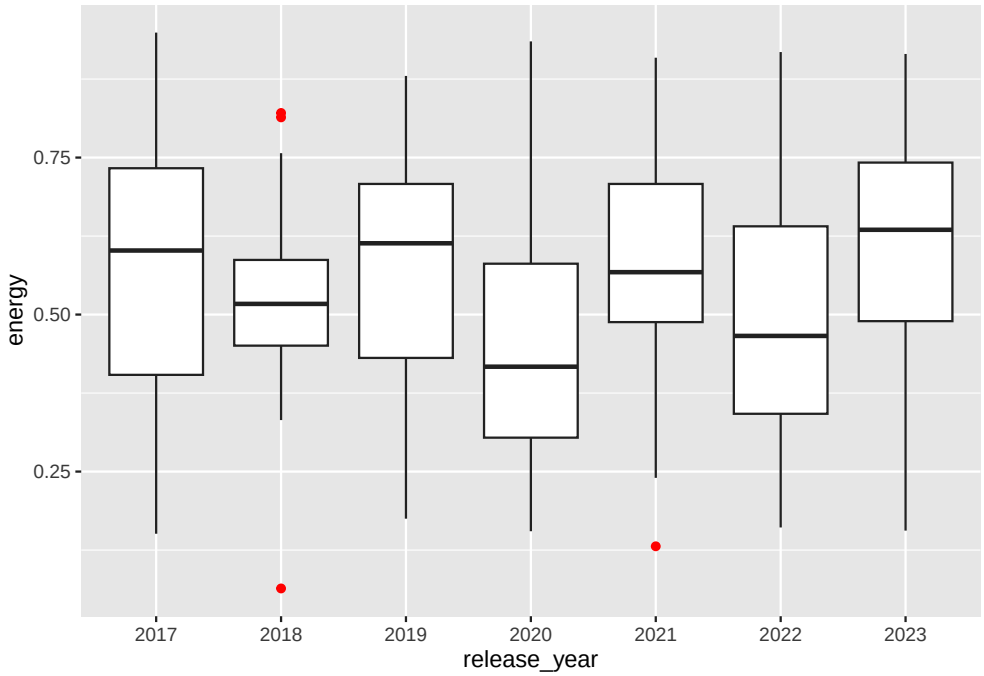

```
p2
```


1.00
0.75
0.50
0.25
0.00
acousticness
2017
2018
2019
2020
2021
2022
2023
release_year

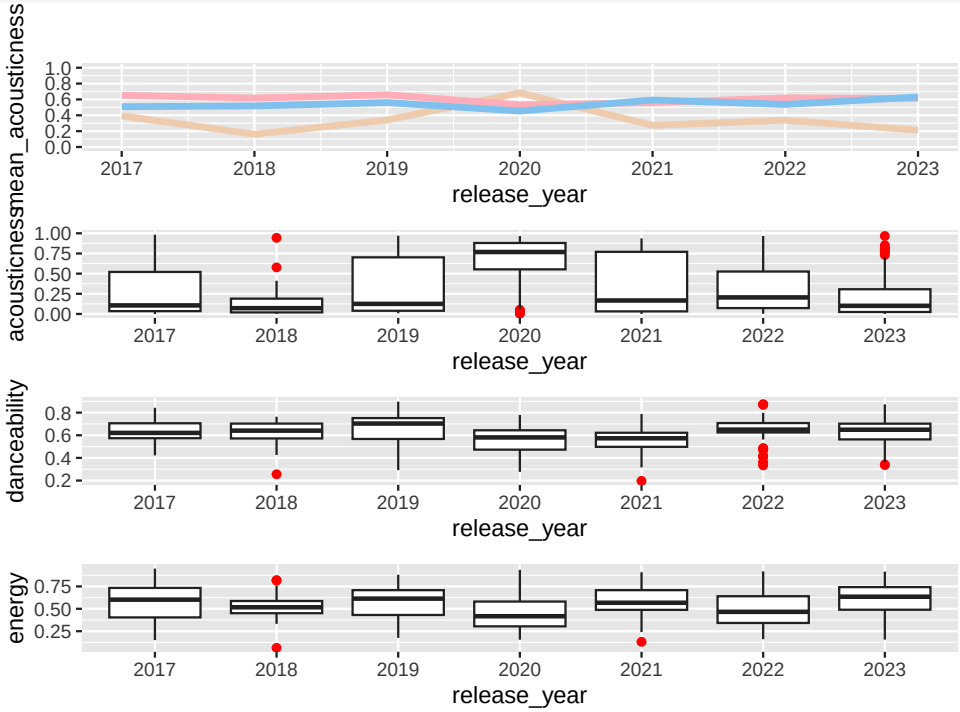
```
p3
```


0.8
0.6
0.4
0.2
danceability
2017
2018
2019
2020
2021
2022
2023
release_year

```
p4
```



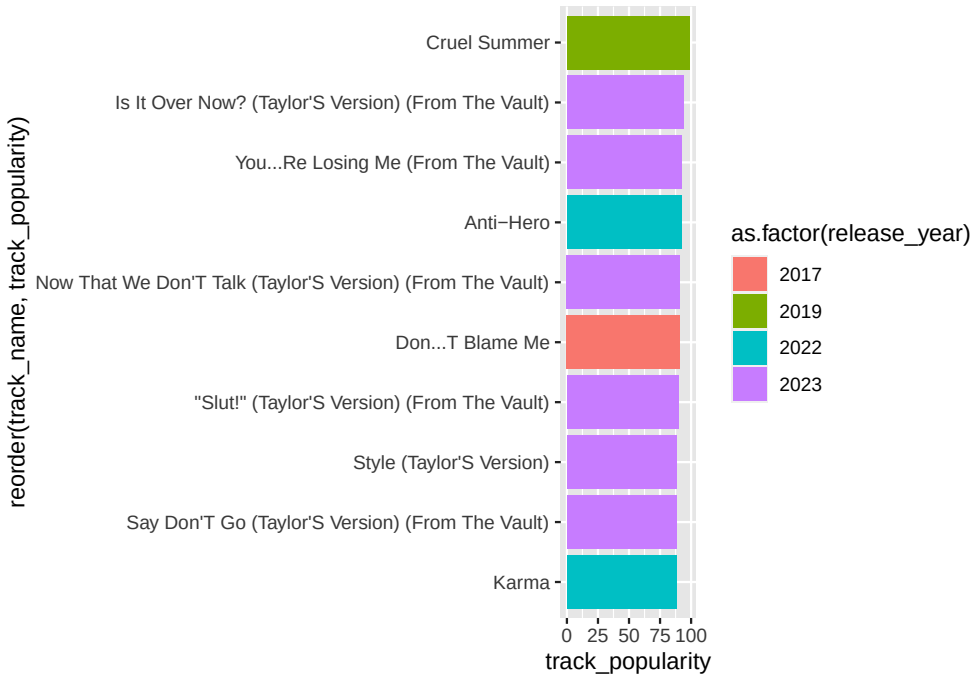

```
p1/p2/p3/p4
```




Taylor Swift's most top 5 most popular tracks and albums in 2023

Top 5 songs in 2023: Does the release year matter? Does the same exact song being in album/single/deluxe album affect its popularity? if so, why? isit because one released earlier?

```r
# Rating duplicate tracks independently (Same track from a single and an album)
top_10_songs = spotify_data1 %>%
  filter(name == "Taylor Swift") %>%
  arrange(desc(track_popularity)) %>%
  select(album_name, track_name,track_popularity, release_year) %>%
  head(10)

ggplot(top_10_songs, aes(x = reorder(track_name, track_popularity), y = track_popularity, fill = as.fac
  geom_col() +
  coord_flip()
```




#Temporal Trends:

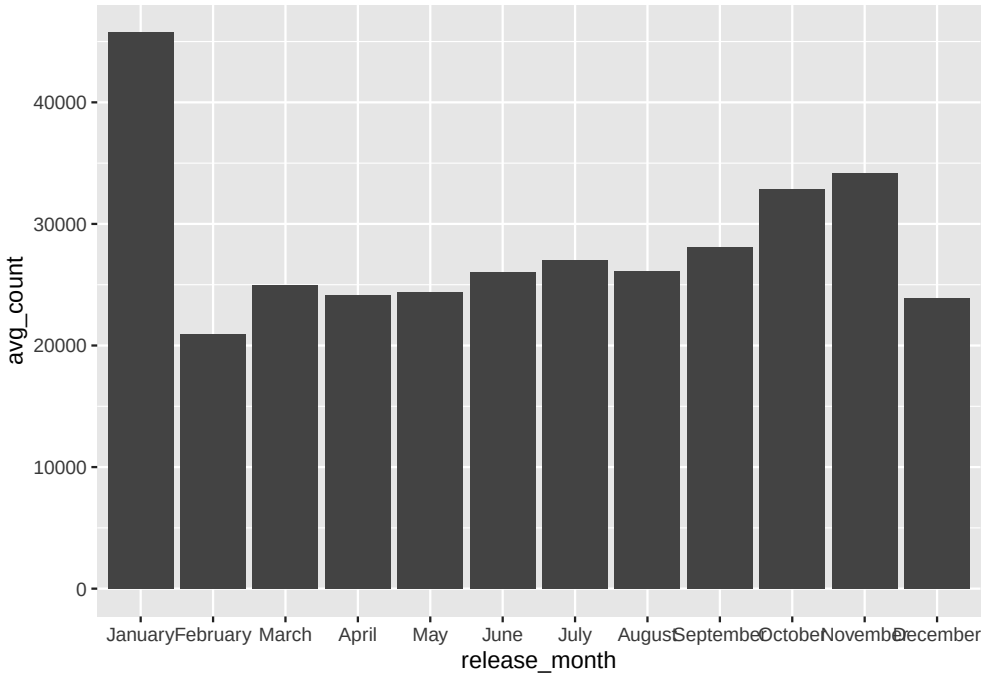
Release Trends Over Time: Visualize the number of releases per year/month to identify trends or seasonality in music releases.

Number of releases by year from 2000 to 2023 by month x = density of releases y = year

Average number of releases from 2000 to 2023 split by month

```r
spotify_data1 %>%
  select(release_year,release_month, track_id) %>%
  filter(release_year %in% c(2000:2023)) %>%
  group_by(release_year, release_month) %>%
  mutate(count = n()) %>%
  ungroup() %>%
  select(-track_id) %>%
  distinct() %>%
  group_by(release_month) %>%
  mutate(avg_count = round(mean(count), 0),
         release_month = factor(release_month,
```

```
                                levels = c("January", "February", "March", "April",
                                           "May", "June", "July", "August",
                                           "September", "October", "November", "December"))) %>%
  select(-count, -release_year) %>%
  ggplot(aes(x = release_month, y = avg_count)) +
  geom_col()
```



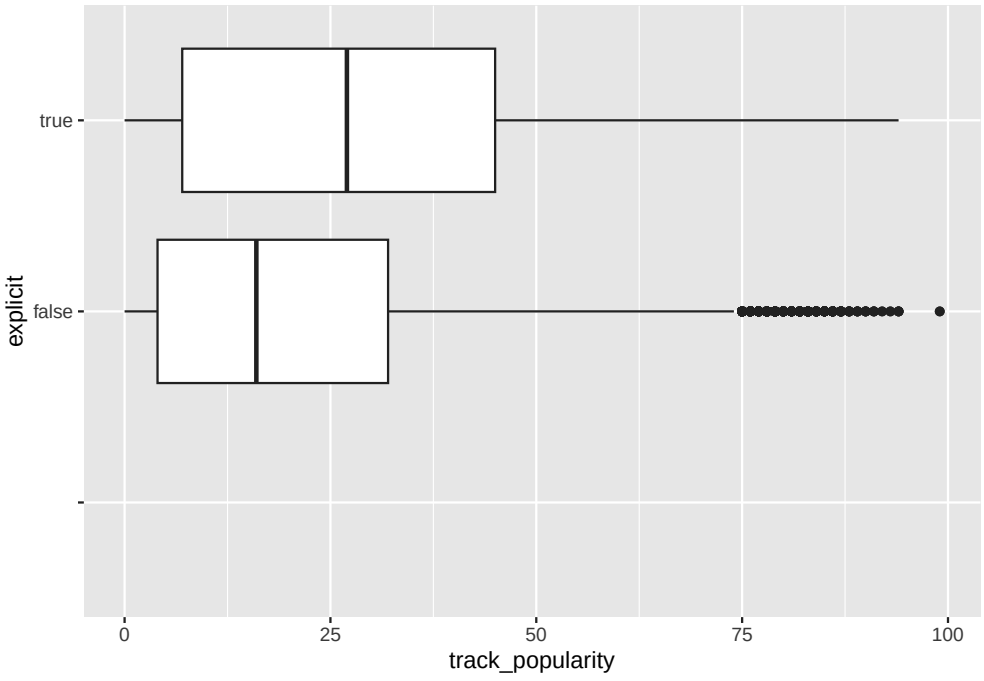

## Evolution of Genres: See how different genres have gained or lost popularity over the years.

```
# spotify_data1 %>%
#   filter(release_year %in% c(2000:2023), str_length(trimws(genre_0)) > 0) %>%
#   mutate(main_genre = case_when (
#     str_detect(genre_0, "edm|trap|dubstep|electro|trance|techno|house") ~ "EDM",
#     str_detect(genre_0, "hop|rap|phonk|drill|lo-fi") ~ "Hip Hop/Rap",
#     str_detect(genre_0, "r&b|soul|funk|jazz") ~ "R&B/Jazz",
#     str_detect(genre_0, "rock|metal") ~ "Rock",
#     str_detect(genre_0, "pop") ~ "Pop",
#     TRUE ~ "Others"  # For other cases not matching, assign "Others"
#   )) %>%
#   mutate(release_year = factor(release_year,
#                                levels = c("2000", "2001","2002","2003","2004",
#                                           "2005","2006", "2007","2008","2009"
#                                           )))

# still editing
```

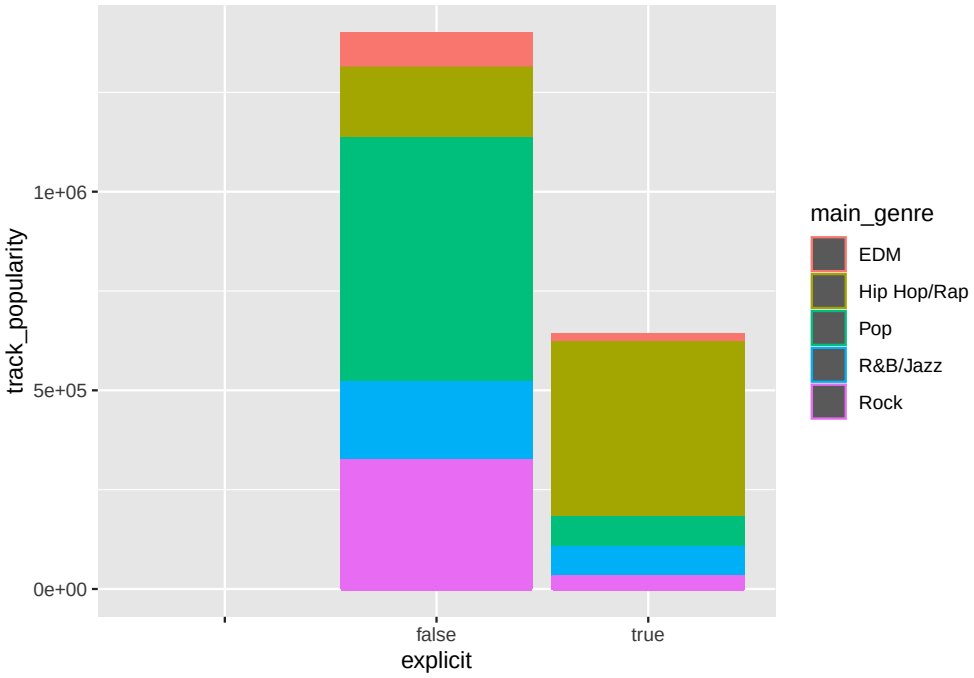
#Exploring Explicit Content:

Explicit Content Analysis: Examine if explicit songs tend to be more popular or if they belong to certain genres.

```
#removing "Others" because it consists of a lot of non-musical tracks

explicit1 = spotify_data1 %>%
  filter(str_length(trimws(genre_0)) > 0) %>%
  mutate(main_genre = case_when (
    str_detect(genre_0, "edm|trap|dubstep|electro|trance|techno|house") ~ "EDM",
    str_detect(genre_0, "hop|rap|phonk|drill|lo-fi") ~ "Hip Hop/Rap",
    str_detect(genre_0, "r&b|soul|funk|jazz") ~ "R&B/Jazz",
    str_detect(genre_0, "rock|metal") ~ "Rock",
    str_detect(genre_0, "pop") ~ "Pop",
    TRUE ~ "Others"  # For other cases not matching, assign "Others"
  )) %>%
  select(track_id, track_popularity, explicit, main_genre)

# explicit1 %>%
#   ggplot(aes(x = explicit, y = track_popularity)) +
#   geom_boxplot() +
#   coord_flip()

# explicit1 %>%
#   ggplot(aes(x=explicit, y = track_popularity, col = main_genre)) +
#   geom_bar(position="stack", stat="identity")

explicit2 = explicit1 %>%
  filter(main_genre != "Others")

explicit2 %>%
  ggplot(aes(x = explicit, y = track_popularity)) +
  geom_boxplot() +
  coord_flip()
```




```
explicit2 %>%
  ggplot(aes(x=explicit, y = track_popularity, col = main_genre)) +
  geom_bar(position="stack", stat="identity")
```




#Collaborations:

#Impact of Co-Artists: Determine if songs featuring co-artists tend to be more popular or belong to specific genres.