Spotify 2023 Data Analysis

Shahrukh Rahman

2025-03-31

Introduction

This report analyzes the **Spotify 2023 dataset**, performing various data manipulation, visualization, and statistical computations in R.

1. Load and View Dataset

```
spotify_2023_Copy <- read_excel("spotify-2023 - Copy.xls")
View(spotify_2023_Copy)
```

2. List the variables in your dataset

```
str(spotify_2023_Copy)  # Check structure of dataset
```

```
## tibble [953 × 24] (S3: tbl_df/tbl/data.frame)
##  $ track_name          : chr [1:953] "Seven (feat. Latto) (Explicit Ver.)" "LALA" "vampire" "Cruel Summer" ...
##  $ artist(s)_name      : chr [1:953] "Latto, Jung Kook" "Myke Towers" "Olivia Rodrigo" "Taylor Swift" ...
##  $ artist_count        : num [1:953] 2 1 1 1 1 2 2 1 1 2 ...
##  $ released_year       : num [1:953] 2023 2023 2023 2019 2023 ...
##  $ released_month      : num [1:953] 7 3 6 8 5 6 3 7 5 3 ...
##  $ released_day        : num [1:953] 14 23 30 23 18 1 16 7 15 17 ...
##  $ in_spotify_playlists: num [1:953] 553 1474 1397 7858 3133 ...
##  $ in_spotify_charts   : num [1:953] 147 48 113 100 50 91 50 43 83 44 ...
##  $ streams             : chr [1:953] "141381703" "133716286" "140003974" "800840817" ...
##  $ in_apple_playlists  : num [1:953] 43 48 94 116 84 67 34 25 60 49 ...
##  $ in_apple_charts     : num [1:953] 263 126 207 207 133 213 222 89 210 110 ...
##  $ in_deezer_playlists : num [1:953] 45 58 91 125 87 88 43 30 48 66 ...
##  $ in_deezer_charts    : num [1:953] 10 14 14 12 15 17 13 13 11 13 ...
##  $ in_shazam_charts    : num [1:953] 826 382 949 548 425 946 418 194 953 339 ...
##  $ bpm                 : num [1:953] 125 92 138 170 144 141 148 100 130 170 ...
##  $ key                 : chr [1:953] "B" "C#" "F" "A" ...
##  $ mode                : chr [1:953] "Major" "Major" "Major" "Major" ...
##  $ danceability_%      : num [1:953] 80 71 51 55 65 92 67 67 85 81 ...
##  $ valence_%           : num [1:953] 89 61 32 58 23 66 83 26 22 56 ...
##  $ energy_%            : num [1:953] 83 74 53 72 80 58 76 71 62 48 ...
##  $ acousticness_%      : num [1:953] 31 7 17 11 14 19 48 37 12 21 ...
##  $ instrumentalness_%  : num [1:953] 0 0 0 0 63 0 0 0 0 0 ...
##  $ liveness_%          : num [1:953] 8 10 31 11 11 8 8 11 28 8 ...
##  $ speechiness_%       : num [1:953] 4 4 6 15 6 24 3 4 9 33 ...
```

```
colnames(spotify_2023_Copy)  # Display column names
```

```
##  [1] "track_name"           "artist(s)_name"       "artist_count"
##  [4] "released_year"        "released_month"       "released_day"
##  [7] "in_spotify_playlists" "in_spotify_charts"    "streams"
## [10] "in_apple_playlists"   "in_apple_charts"      "in_deezer_playlists"
## [13] "in_deezer_charts"     "in_shazam_charts"     "bpm"
## [16] "key"                  "mode"                 "danceability_%"
## [19] "valence_%"            "energy_%"             "acousticness_%"
## [22] "instrumentalness_%"   "liveness_%"           "speechiness_%"
```

3.Top 15 rows of your dataset

```
head(spotify_2023_Copy, 15)  # Display first 15 rows
```

```
## # A tibble: 15 × 24
##    track_name         `artist(s)_name` artist_count released_year released_month
##    <chr>              <chr>                   <dbl>         <dbl>          <dbl>
##  1 Seven (feat. Latt… Latto, Jung Kook            2          2023              7
##  2 LALA               Myke Towers                 1          2023              3
##  3 vampire            Olivia Rodrigo              1          2023              6
##  4 Cruel Summer       Taylor Swift                1          2019              8
##  5 WHERE SHE GOES     Bad Bunny                   1          2023              5
##  6 Sprinter           Dave, Central C…            2          2023              6
##  7 Ella Baila Sola    Eslabon Armado,…            2          2023              3
##  8 Columbia           Quevedo                     1          2023              7
##  9 fukumean           Gunna                       1          2023              5
## 10 La Bebe - Remix    Peso Pluma, Yng…            2          2023              3
## 11 un x100to          Bad Bunny, Grup…            2          2023              4
## 12 Super Shy          NewJeans                    1          2023              7
## 13 Flowers            Miley Cyrus                 1          2023              1
## 14 Daylight           David Kushner               1          2023              4
## 15 As It Was          Harry Styles                1          2022              3
## # ℹ 19 more variables: released_day <dbl>, in_spotify_playlists <dbl>,
## #   in_spotify_charts <dbl>, streams <chr>, in_apple_playlists <dbl>,
## #   in_apple_charts <dbl>, in_deezer_playlists <dbl>, in_deezer_charts <dbl>,
## #   in_shazam_charts <dbl>, bpm <dbl>, key <chr>, mode <chr>,
## #   `danceability_%` <dbl>, `valence_%` <dbl>, `energy_%` <dbl>,
## #   `acousticness_%` <dbl>, `instrumentalness_%` <dbl>, `liveness_%` <dbl>,
## #   `speechiness_%` <dbl>
```

4. User defined function

Categorize Streams

```
categorize_streams <- function(streams) {
  if (streams >= 1000000) {
    return("High Stream")
  } else {
    return("Low Stream")
  }
}
spotify_2023_Copy$Stream_Category <- sapply(spotify_2023_Copy$streams, categorize_streams)
head(spotify_2023_Copy)
```

```
## # A tibble: 6 × 25
##   track_name          `artist(s)_name` artist_count released_year released_month
##   <chr>               <chr>                   <dbl>         <dbl>          <dbl>
## 1 Seven (feat. Latto… Latto, Jung Kook            2          2023              7
## 2 LALA                Myke Towers                 1          2023              3
## 3 vampire             Olivia Rodrigo              1          2023              6
## 4 Cruel Summer        Taylor Swift                1          2019              8
## 5 WHERE SHE GOES      Bad Bunny                   1          2023              5
## 6 Sprinter            Dave, Central C…            2          2023              6
## # ℹ 20 more variables: released_day <dbl>, in_spotify_playlists <dbl>,
## #   in_spotify_charts <dbl>, streams <chr>, in_apple_playlists <dbl>,
## #   in_apple_charts <dbl>, in_deezer_playlists <dbl>, in_deezer_charts <dbl>,
## #   in_shazam_charts <dbl>, bpm <dbl>, key <chr>, mode <chr>,
## #   `danceability_%` <dbl>, `valence_%` <dbl>, `energy_%` <dbl>,
## #   `acousticness_%` <dbl>, `instrumentalness_%` <dbl>, `liveness_%` <dbl>,
## #   `speechiness_%` <dbl>, Stream_Category <chr>
```

5. Filtering Recent Songs

```
recent_songs <- spotify_2023_Copy %>% filter(released_year > 2020)
head(recent_songs)
```

```
## # A tibble: 6 × 25
##   track_name          `artist(s)_name` artist_count released_year released_month
##   <chr>               <chr>                   <dbl>         <dbl>          <dbl>
## 1 Seven (feat. Latto… Latto, Jung Kook            2          2023              7
## 2 LALA                Myke Towers                 1          2023              3
## 3 vampire             Olivia Rodrigo              1          2023              6
## 4 WHERE SHE GOES      Bad Bunny                   1          2023              5
## 5 Sprinter            Dave, Central C…            2          2023              6
## 6 Ella Baila Sola     Eslabon Armado,…            2          2023              3
## # ℹ 20 more variables: released_day <dbl>, in_spotify_playlists <dbl>,
## #   in_spotify_charts <dbl>, streams <chr>, in_apple_playlists <dbl>,
## #   in_apple_charts <dbl>, in_deezer_playlists <dbl>, in_deezer_charts <dbl>,
## #   in_shazam_charts <dbl>, bpm <dbl>, key <chr>, mode <chr>,
## #   `danceability_%` <dbl>, `valence_%` <dbl>, `energy_%` <dbl>,
## #   `acousticness_%` <dbl>, `instrumentalness_%` <dbl>, `liveness_%` <dbl>,
## #   `speechiness_%` <dbl>, Stream_Category <chr>
```

6. Reshape Data

```
spotify_2023_Copy <- spotify_2023_Copy %>%
  rename(artist_name = `artist(s)_name`) %>%  # Rename artist column
  select(track_name, artist_name, streams, bpm, `danceability_%`, `energy_%`) %>%  # Select columns
  pivot_longer(
    cols = c(bpm, `danceability_%`, `energy_%`),  # Use backticks for special names
    names_to = "Feature",
    values_to = "Value"
  )

head(spotify_2023_Copy)
```

```
## # A tibble: 6 × 5
##   track_name                          artist_name      streams   Feature   Value
##   <chr>                               <chr>            <chr>     <chr>     <dbl>
## 1 Seven (feat. Latto) (Explicit Ver.) Latto, Jung Kook 141381703 bpm         125
## 2 Seven (feat. Latto) (Explicit Ver.) Latto, Jung Kook 141381703 danceabi…    80
## 3 Seven (feat. Latto) (Explicit Ver.) Latto, Jung Kook 141381703 energy_%     83
## 4 LALA                                Myke Towers      133716286 bpm          92
## 5 LALA                                Myke Towers      133716286 danceabi…    71
## 6 LALA                                Myke Towers      133716286 energy_%     74
```

7. Data Cleaning

Remove Missing Values

```
cleaned_df <- spotify_2023_Copy %>% drop_na()
sum(is.na(cleaned_df))  # Check if missing values are removed
```

```
## [1] 0
```

8. Remove Duplicated Rows

```
duplicated_rows <- spotify_2023_Copy %>% filter(duplicated(spotify_2023_Copy))
unique_df <- spotify_2023_Copy %>% distinct()
```

9. Reorder Rows in Descending Order

```
sorted_df <- spotify_2023_Copy %>% arrange(desc(streams))
head(sorted_df)
```

```
## # A tibble: 6 × 5
##   track_name                          artist_name       streams    Feature Value
##   <chr>                               <chr>             <chr>      <chr>   <dbl>
## 1 Love Grows (Where My Rosemary Goes) Edison Lighthouse BPM110Key… bpm       110
## 2 Love Grows (Where My Rosemary Goes) Edison Lighthouse BPM110Key… dancea…    53
## 3 Love Grows (Where My Rosemary Goes) Edison Lighthouse BPM110Key… energy…    69
## 4 Anti-Hero                           Taylor Swift      999748277  bpm        97
## 5 Anti-Hero                           Taylor Swift      999748277  dancea…    64
## 6 Anti-Hero                           Taylor Swift      999748277  energy…    63
```

10. Data Transformation

Renaming Column Name

```
spotify_2023_Copy <- spotify_2023_Copy %>% rename(artis_name = `artist_name`)
colnames(spotify_2023_Copy)
```

```
## [1] "track_name" "artis_name" "streams"    "Feature"    "Value"
```

11. Creating a New Variable Using Mathematical Function

```
spotify_2023_Copy <- spotify_2023_Copy %>%
  mutate(Value = ifelse(Feature == "bpm", as.numeric(Value) * 2, Value))
```

12. Create a Training Set

```
set.seed(123)
train_index <- sample(1:nrow(spotify_2023_Copy), 0.8 * nrow(spotify_2023_Copy))
train_set <- spotify_2023_Copy[train_index, ]
test_set <- spotify_2023_Copy[-train_index, ]
```

```
nrow(train_set)  # Check training set size
```

```
## [1] 2287
```

```
nrow(test_set)   # Check test set size
```

```
## [1] 572
```

13. Summary Statistics

```
summary(spotify_2023_Copy)
```

```
##   track_name         artis_name          streams            Feature
##  Length:2859        Length:2859        Length:2859        Length:2859
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##      Value
##  Min.   :  9.0
##  1st Qu.: 62.0
##  Median : 78.0
##  Mean   :125.4
##  3rd Qu.:199.0
##  Max.   :412.0
```

14. Statistical Functions

```
spotify_2023_Copy <- spotify_2023_Copy %>%
  mutate(streams = gsub("[^0-9]", "", streams),  # Remove non-numeric characters
         streams = as.numeric(streams))          # Convert to numeric

mean(spotify_2023_Copy$streams, na.rm = TRUE)
```

```
## [1] 12112503461
```

```
median(spotify_2023_Copy$streams, na.rm = TRUE)
```

```
## [1] 290833204
```

```
get_mode <- function(v) {
  uniqv <- unique(v)
  uniqv[which.max(tabulate(match(v, uniqv)))]
}
get_mode(spotify_2023_Copy$streams)
```

```
## [1] 156338624
```

```
range(spotify_2023_Copy$streams, na.rm = TRUE)
```

```
## [1] 2.762000e+03 1.105376e+13
```

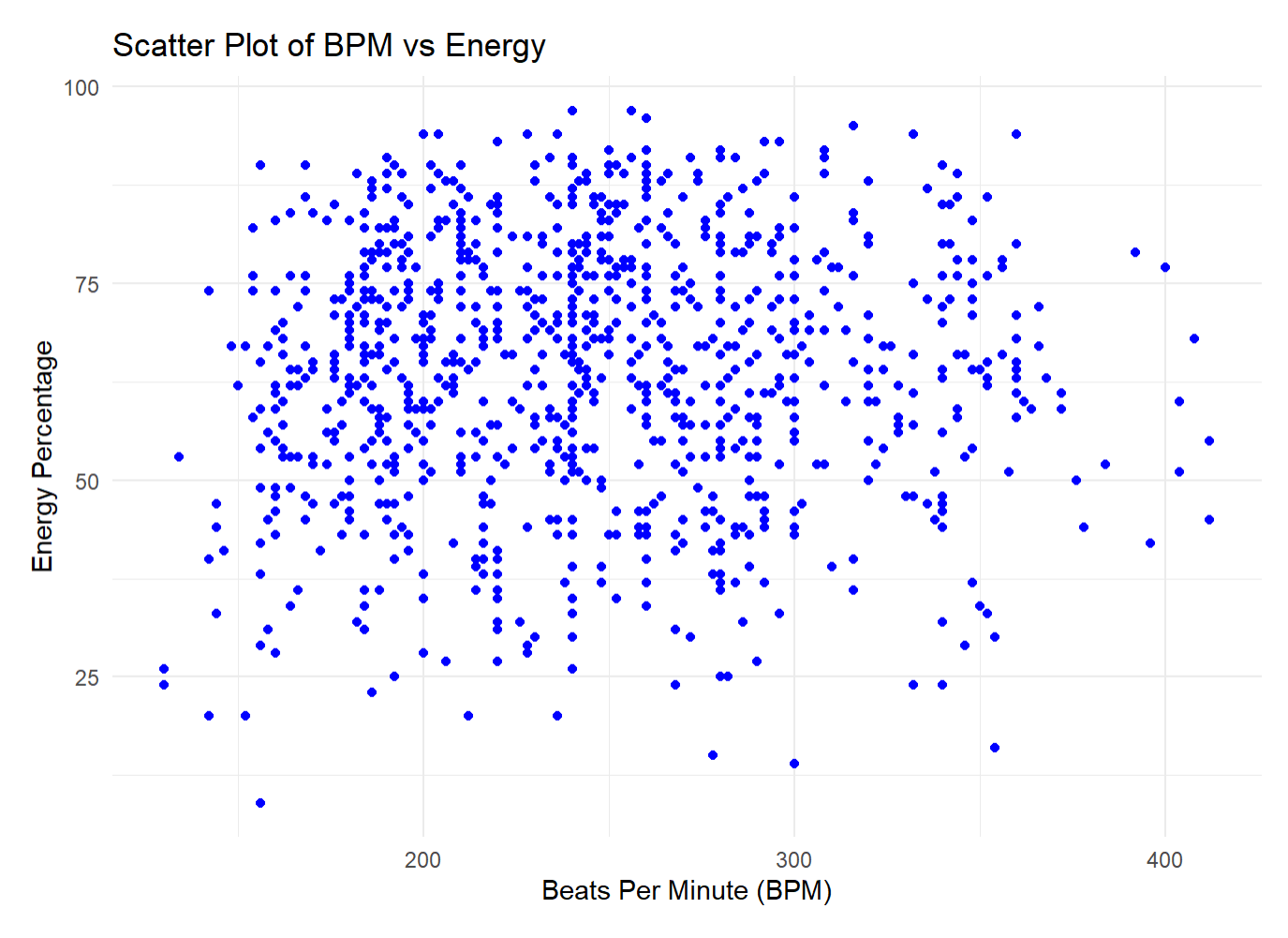
15. Data Visualization

Scatter Plot (BPM vs Energy)

```
# Summarize duplicates by averaging the Value column
spotify_summarized <- spotify_2023_Copy %>%
  group_by(track_name, artis_name, streams, Feature) %>%
  summarise(Value = mean(Value, na.rm = TRUE), .groups = "drop")

# Now pivot the data
spotify_wide <- spotify_summarized %>%
  pivot_wider(names_from = Feature, values_from = Value)

# Convert necessary columns to numeric
spotify_wide <- spotify_wide %>%
  mutate(bpm = as.numeric(bpm), `energy_%` = as.numeric(`energy_%`))

# Check the result
head(spotify_wide)
```

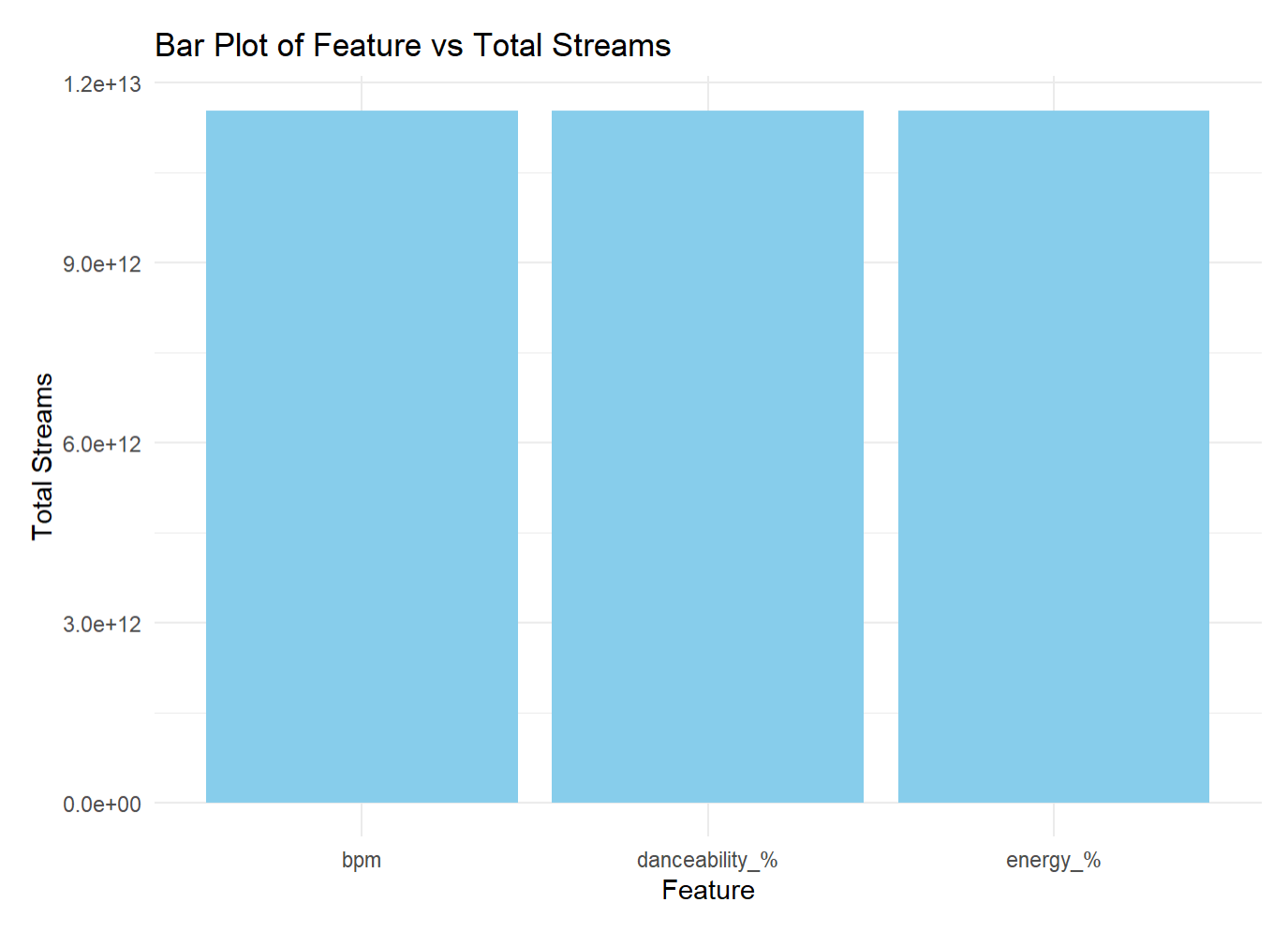
```
## # A tibble: 6 × 6
##   track_name                 artis_name streams   bpm `danceability_%` `energy_%`
##   <chr>                      <chr>        <dbl> <dbl>            <dbl>      <dbl>
## 1 'Till I Collapse           Eminem, N…  1.70e9   342               55         85
## 2 (It Goes Like) Nanana - … Peggy Gou   5.79e7   260               67         88
## 3 0.44097222222222221        Tiï¿½ï¿½s…  3.26e8   240               70         79
## 4 10 Things I Hate About Y… Leah Kate   1.86e8   308               54         79
## 5 2 Be Loved (Am I Ready)    Lizzo       2.48e8   312               72         77
## 6 2055                       Sleepy ha…  6.25e8   322               78         52
```

```
ggplot(spotify_wide, aes(x = bpm, y = `energy_%`)) +
  geom_point(color = "blue") +
  labs(title = "Scatter Plot of BPM vs Energy",
       x = "Beats Per Minute (BPM)",
       y = "Energy Percentage") +
  theme_minimal()
```



16. Bar Plot (Released Year vs Spotify Playlists)

```
spotify_summary <- spotify_2023_Copy %>%
  group_by(Feature) %>%  # Grouping by the 'Feature' column
  summarise(total_streams = sum(streams, na.rm = TRUE), .groups = "drop")

ggplot(spotify_summary, aes(x = Feature, y = total_streams)) +
  geom_bar(stat = "identity", fill = "skyblue") +
  labs(title = "Bar Plot of Feature vs Total Streams",
       x = "Feature",
       y = "Total Streams") +
  theme_minimal()
```



17. Correlation Analysis

```
library(dplyr)
library(tidyr)

# Summarize duplicates by calculating the mean for each combination
spotify_filtered <- spotify_2023_Copy %>%
  filter(Feature %in% c("bpm", "energy_%")) %>%
  group_by(track_name, artis_name, streams, Feature) %>%
  summarise(Value = mean(Value, na.rm = TRUE), .groups = "drop") %>%
  pivot_wider(names_from = Feature, values_from = Value)

# Check the data to ensure bpm and energy_% are in separate columns
head(spotify_filtered)
```

```
## # A tibble: 6 × 5
##   track_name                    artis_name             streams   bpm `energy_%`
##   <chr>                         <chr>                    <dbl> <dbl>      <dbl>
## 1 'Till I Collapse              Eminem, Nate Dogg   1695712020   342         85
## 2 (It Goes Like) Nanana - Edit Peggy Gou             57876440   260         88
## 3 0.44097222222222221           Tiï¿½ï¿½sto, Tate M  325592432   240         79
## 4 10 Things I Hate About You    Leah Kate            185550869   308         79
## 5 2 Be Loved (Am I Ready)       Lizzo                247689123   312         77
## 6 2055                          Sleepy hallow        624515457   322         52
```

```
spotify_filtered$bpm <- as.numeric(spotify_filtered$bpm)
spotify_filtered$`energy_%` <- as.numeric(spotify_filtered$`energy_%`)

# Calculate the correlation
correlation_value <- cor(spotify_filtered$bpm, spotify_filtered$`energy_%`, use = "complete.obs", method = "pears
on")
print(correlation_value)
```

```
## [1] 0.02610044
```

9. Save and Load Session

```
# Saving the session in the specified directory
save.image("C:/Users/anjal/Downloads/Spotify v2/Spotify v2/Spotify_Session.RData")
# Loading the session from the specified directory
load("C:/Users/anjal/Downloads/Spotify v2/Spotify v2/Spotify_Session.RData")
```

Github Link: https://github.com/AnjaliSagpariya/Spotify.git