Final Project (Week-9)

Elise Wong

2023-10-19

Week-9: Webpage and Diary Entry

For my data story, I have chosen to understand how data analysts categorise songs on **Spotify**.

I have obtained the **'Spotify Songs'** data-set from the **#TidyTuesday project**, hosted on GitHub.

A **sample** of the aforementioned data-set is as follows:

```
library(tidytuesdayR)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.0     ✔ readr     2.1.4
## ✔ forcats   1.0.0     ✔ stringr   1.5.0
## ✔ ggplot2   3.4.3     ✔ tibble    3.1.8
## ✔ lubridate 1.9.2     ✔ tidyr     1.3.0
## ✔ purrr     1.0.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the ]8;;http://conflicted.r-lib.org/ conflicted package]8;;  to force all conflicts to become errors
```

```
spotify_songs <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/data/2020/2020-01-21/spotify_songs.csv')
```

```
## Rows: 32833 Columns: 23
## ── Column specification ────────────────────────────────────────────────────────
## Delimiter: ","
## chr (10): track_id, track_name, track_artist, track_album_id, track_album_na...
## dbl (13): track_popularity, danceability, energy, key, loudness, mode, speec...
##
## ℹ Use `spec()` to retrieve the full column specification for this data.
## ℹ Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```
tuesdata <- tidytuesdayR::tt_load('2020-01-21')
```

```
## --- Compiling #TidyTuesday Information for 2020-01-21 ----
## --- There is 1 file available ---
## --- Starting Download ---
```

```
##
##  Downloading file 1 of 1: `spotify_songs.csv`
```

```
## --- Download complete ---
```

```
tuesdata <- tidytuesdayR::tt_load(2020, week = 4)
```

```
## --- Compiling #TidyTuesday Information for 2020-01-21 ----
## --- There is 1 file available ---
## --- Starting Download ---
```

```
##
##  Downloading file 1 of 1: `spotify_songs.csv`
```

```
## --- Download complete ---
```

```
spotify_songs <- tuesdata$spotify_songs
```

```
head(spotify_songs)
```

```
## # A tibble: 6 × 23
##   track_id               track_name track_artist track_popularity track_album_id
##   <chr>                  <chr>      <chr>                   <dbl> <chr>
## 1 6f807x0ima9a1j3VPbc7VN I Don't C… Ed Sheeran                 66 2oCs0DGTsRO98…
## 2 0r7CVbZTWZgbTCYdfa2P31 Memories … Maroon 5                   67 63rPSO264uRjW…
## 3 1z1Hg7Vb0AhHDiEmnDE79l All the T… Zara Larsson               70 1HoSmj2eLcsrR…
## 4 75FpbthrwQmzHlBJLuGdC7 Call You … The Chainsm…               60 1nqYsOef1yKKu…
## 5 1e8PAfcKUYoKkxPhrHqw4x Someone Y… Lewis Capal…               69 7m7vv9wlQ4i0L…
## 6 7fvUMiyapMsRRxr07cU8Ef Beautiful… Ed Sheeran                 67 2yiy9cd2QktrN…
## # ℹ 18 more variables: track_album_name <chr>, track_album_release_date <chr>,
## #   playlist_name <chr>, playlist_id <chr>, playlist_genre <chr>,
## #   playlist_subgenre <chr>, danceability <dbl>, energy <dbl>, key <dbl>,
## #   loudness <dbl>, mode <dbl>, speechiness <dbl>, acousticness <dbl>,
## #   instrumentalness <dbl>, liveness <dbl>, valence <dbl>, tempo <dbl>,
## #   duration_ms <dbl>
```

Thank you!