

# Song Data Project

Kristian Abad, Steven Truong, Nicole Magallanes

2/13/2022

```
library(readr)
library(tidyr)
library(tidyverse)
```

```
## -- Attaching packages ----- tidyverse 1.3.1 --

## v ggplot2 3.3.5    v dplyr   1.0.7
## v tibble  3.1.2    v stringr 1.4.0
## v purrr   0.3.4    v forcats 0.5.1

## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
```

## Preprocessing

```
data = read_csv("song_data.csv")
```

```
## Rows: 18835 Columns: 15
```

```
## -- Column specification -----
## Delimiter: ","
## chr  (1): song_name
## dbl (14): song_popularity, song_duration_ms, acousticness, danceability, ene...
```

```
##
## 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.
```

```
head(data)
```

```
## # A tibble: 6 x 15
##   song_name      song_popularity song_duration_ms acousticness danceability energy
##   <chr>          <dbl>          <dbl>          <dbl>          <dbl> <dbl>
## 1 Boulevard o~      73          262333      0.00552        0.496  0.682
## 2 In The End        66          216933      0.0103         0.542  0.853
```

```
## 3 Seven Natio~          76          231733      0.00817          0.737 0.463
## 4 By The Way            74          216933      0.0264          0.451 0.97
## 5 How You Rem~         56          223826      0.000954          0.447 0.766
## 6 Bring Me To~         80          235893      0.00895          0.316 0.945
## # ... with 9 more variables: instrumentalness <dbl>, key <dbl>, liveness <dbl>,
## #   loudness <dbl>, audio_mode <dbl>, speechiness <dbl>, tempo <dbl>,
## #   time_signature <dbl>, audio_valence <dbl>
```

One challenge we need to figure out is addressing the following cases in our data, if there are any:

- Remixes
- Remasters
- Single Versions
- Same name but different artists?
- Radio edits

I think maybe we can leave remixes possibly treating them as reimaginings of songs or somewhat to the same vein that songs have samples from other tracks are in themselves a separate track. Maybe the more difficult is dealing with the other cases. An example that comes to mind is “Smooth Operator” by Sade (seems like only one of the 3 versions is in the data). There’s a single version, a remastered version, and I believe an album version where there’s an immediate difference between the remastered and album version.

I think the duplicated() function finds exact duplicates of rows.

```
duplicates <- data[duplicated(data),]
duplicates
```

```
## # A tibble: 3,909 x 15
##   song_name song_popularity song_duration_ms acousticness danceability energy
##   <chr>          <dbl>          <dbl>          <dbl>          <dbl> <dbl>
## 1 Sex on Fire            81          203346          0.00172          0.542 0.905
## 2 Use Somebo~           79          230760          0.00552          0.276 0.715
## 3 Hips Don't~           84          218093          0.284          0.778 0.824
## 4 Hotel Cali~           83          391376          0.00574          0.579 0.508
## 5 Me and Bob~           69          271333          0.302          0.453 0.464
## 6 Imagine - ~           77          187866          0.907          0.547 0.257
## 7 Let It Be ~           78          243026          0.631          0.443 0.403
## 8 Rocket Man~           80          281613          0.386          0.602 0.522
## 9 My Sweet L~           78          281226          0.0794          0.538 0.704
## 10 Tangled up~          63          341626          0.414          0.421 0.661
## # ... with 3,899 more rows, and 9 more variables: instrumentalness <dbl>,
## #   key <dbl>, liveness <dbl>, loudness <dbl>, audio_mode <dbl>,
## #   speechiness <dbl>, tempo <dbl>, time_signature <dbl>, audio_valence <dbl>
```

Just testing some cases here... while scrolling through on kaggle, I just picked a random duplicate song to test

```
duplicates %>%
  filter(song_name == 'Zombie')
```

```
## # A tibble: 1 x 15
##   song_name song_popularity song_duration_ms acousticness danceability energy
```

```
##   <chr>           <dbl>           <dbl>           <dbl>           <dbl> <dbl>
## 1 Zombie              82            306410           0.0163           0.299 0.613
## # ... with 9 more variables: instrumentalness <dbl>, key <dbl>, liveness <dbl>,
## #   loudness <dbl>, audio_mode <dbl>, speechiness <dbl>, tempo <dbl>,
## #   time_signature <dbl>, audio_valence <dbl>
```

Here's an interesting case where we have 2 of the same rows and 1 with a remix with a track called "8 Letters"

```
duplicates %>%
  filter(song_name == '8 Letters')
```

```
## # A tibble: 1 x 15
##   song_name song_popularity song_duration_ms acousticness danceability energy
##   <chr>           <dbl>           <dbl>           <dbl>           <dbl> <dbl>
## 1 8 Letters              72            190026           0.649           0.607 0.478
## # ... with 9 more variables: instrumentalness <dbl>, key <dbl>, liveness <dbl>,
## #   loudness <dbl>, audio_mode <dbl>, speechiness <dbl>, tempo <dbl>,
## #   time_signature <dbl>, audio_valence <dbl>
```

```
duplicates %>%
  filter(song_name == '8 Letters - R3HAB Remix')
```

```
## # A tibble: 0 x 15
## # ... with 15 variables: song_name <chr>, song_popularity <dbl>,
## #   song_duration_ms <dbl>, acousticness <dbl>, danceability <dbl>,
## #   energy <dbl>, instrumentalness <dbl>, key <dbl>, liveness <dbl>,
## #   loudness <dbl>, audio_mode <dbl>, speechiness <dbl>, tempo <dbl>,
## #   time_signature <dbl>, audio_valence <dbl>
```

So it looks like it just picks up exact duplicates and we'll need to figure out what we're going to do with other cases.

```
data_2 <- data[!duplicated(data),]
nrow(data)
```

```
## [1] 18835
```

```
nrow(data_2)
```

```
## [1] 14926
```

```
nrow(data) - nrow(data_2)
```

```
## [1] 3909
```

Using the grepl function to find any instance of single, remastered, and radio edit/mix versions of tracks.

```

#Function was found via stackoverflow
#https://stackoverflow.com/questions/10128617/test-if-characters-are-in-a-string
data_2[grepl('Single',data_2$song_name,fixed=TRUE) |
       grepl('Remaster',data_2$song_name,fixed=TRUE) |
       grepl('Radio',data_2$song_name,fixed=TRUE),]

```

```

## # A tibble: 650 x 15
##   song_name    song_popularity song_duration_ms acousticness danceability energy
##   <chr>          <dbl>          <dbl>          <dbl>          <dbl> <dbl>
## 1 The Diary ~          69          200546      0.0000553      0.374 0.961
## 2 Hey Ya! - ~          76          235213      0.103          0.728 0.974
## 3 Single Lad~          72          193213      0.0383         0.426 0.584
## 4 Get Lucky ~          77          248413      0.0426         0.794 0.811
## 5 Maria Mari~          56          262160      0.0485         0.779 0.631
## 6 Hey Jude --          77          425653      0.0112         0.386 0.607
## 7 Surfin' U.~          70          149373      0.661          0.55 0.854
## 8 Born To Be~          67          212893      0.273          0.439 0.737
## 9 I Heard It~          61          193493      0.255          0.735 0.52
## 10 Crimson An~          56          208666      0.255          0.412 0.59
## # ... with 640 more rows, and 9 more variables: instrumentalness <dbl>,
## #   key <dbl>, liveness <dbl>, loudness <dbl>, audio_mode <dbl>,
## #   speechiness <dbl>, tempo <dbl>, time_signature <dbl>, audio_valence <dbl>

```