# Final Project: Top Spotify Playlists of 2018 Analysis

*Jacqueline Deprey and Julie Stone*

*March 10, 2019*

```r
knitr::opts_chunk$set(echo = TRUE)
library(rvest)
library(tidyr)
library(dplyr)
library(stringr)
library(readr)
library(magrittr)
library(tidyverse)
library(purrr)
library(lubridate)
```

## Introduction and Motivation

According to Forbes, the global recorded music industry is now worth over $17.2 billion and only continues to grow. In 2018, Drake released his single "God's Plan" which was played over 1.28 billion times since its release and made Drake over $300,000. But what makes a song a hit? With the increasing prevalence of data that is being generated about consumer preferences, one might expect that a formula could be created to come up with the "perfect" song destined to hit the top charts. Because of the profit in formulas such as these, many producers spend lots of time and energy looking into how to do just that.

For this project, we have decided to examine Spotify's data on the Top 100 Hits from 2018. Because more and more people are getting their music content from audio streaming services, we believe looking at Spotify's data would be a good way to retroactively examine this problem. Spotify, a streaming service based on the freemium model, has over 83 million subscribers and has captured roughly 36% of the market share. Since most major artist now have their music on its platform and the top charts on Spotify almost always match those of industry experts looking at all platforms, we believe analyzing Spotify's data would be representative of the industry trends as a whole.

## Description of dataset

"Top Spotify Tracks of 2018" is a dataset depicting the audio features of the top songs of the streaming platform. There are 100 entities, each representing a song on the platform. There are 16 different attributes, including: - ID: The primary key of the dataset - the Spotify URL of the song - Name: Name of the song - Artist(s): Artist of the song - Danceability: Danceability describes how suitable a track is for dancing. This is based on a combination of different musical elements such as tempo, rhythm, stability, beat strength, and overall regularity. A value of 0.0 is least danceable and a value of 1.0 is most danceable. - Energy: Energy is a measure from 0.0 to 1.0 representing a perceptual measure of intensity and activity. Energetic tracks feel fast, loud, and noisy. Perceptual features that contribute to this measure are dynamic range, percieved loudness, timbre, onset rate, and general entropy. - Key: The key the track is in. Integers map to pitches using the standard pitch class notation. For example, C=0, C#=1, D=2, and so on. - Loudness: The overall loudness of a track in decibels. These values are averaged over the course of the song. - Mode: Modality of a song (major vs. minor). Major is represented by 1 and minor is represented by 0. - Speechiness: Speechiness detects the presence of spoken words in the track. The more exclusively speech-like the song is, the closer to 1.0 the attribute value. Values above 0.66 describe tracks that are probably made entirely of spoken words. Values between 0.33 and 0.66 describe tracks that may contain both music and speech, including such cases as rap music. Values below 0.33 most likely represent music and other non-speech-like tracks. -

Acousticness: A confidence measure from 0.0 to 1.0 of whether the track is acoustic. 1.0 represents high confidence. - Instrumentalness: Predicts whether a track contains no vocals. Rap or spoken word tracks are clearly "vocal". The closer the instrumentalness value is to 1.0, the greater likelihood the track contains no vocal content. Values above 0.5 are intended to represent instrumental tracks, but confidence is higher as the value approaches 1.0. - Liveness: Detects the presence of an audience in the recording. Higher liveness values represent a greater chance that the track was performed live. - Valence: A measure from 0.0 to 1.0 describing the musical positiveness conveyed by a track. Tracks with high valence sound more positive, while tracks with low valence sound more negative. - Tempo: The overall estimated tempo of a track in beats per minute (BPM). - Duration: The duration of the track in milliseconds. - Time Signature: An estimated overall time signature of a track. The time signature is a notational convention to specify how many beats are in each measure. The dataset was obtained as a CSV file from Kaggle.com. Kaggle is an online community of data scientists, owned by Google LLC. Users find and publish data sets, explore and build models in a data-science environment, work with other data scientists, and enter competitions to solve data science challenges. This specific dataset was uploaded 3 months ago by Nadin Tamer.

```r
csv_file <- "C:/Users/jacq_/Documents/Programming/CMSC320/top2018.csv"

spotify_data <- read_csv(csv_file) %>%
  set_colnames(c("id", "name", "artists", "danceability", "energy", "key", "loudness", "mode", "speechi
  as_data_frame()
```

```
## Parsed with column specification:
## cols(
##   id = col_character(),
##   name = col_character(),
##   artists = col_character(),
##   danceability = col_double(),
##   energy = col_double(),
##   key = col_double(),
##   loudness = col_double(),
##   mode = col_double(),
##   speechiness = col_double(),
##   acousticness = col_double(),
##   instrumentalness = col_double(),
##   liveness = col_double(),
##   valence = col_double(),
##   tempo = col_double(),
##   duration_ms = col_double(),
##   time_signature = col_double()
## )
```

```r
head(spotify_data)
```

```
## # A tibble: 6 x 16
##   id    name  artists danceability energy   key loudness  mode speechiness
##   <chr> <chr> <chr>          <dbl>  <dbl> <dbl>    <dbl> <dbl>       <dbl>
## 1 6DCZ~ God'~ Drake          0.754  0.449     7    -9.21     1      0.109
## 2 3ee8~ SAD!  XXXTEN~        0.74   0.613     8    -4.88     1      0.145
## 3 0e7i~ rock~ Post M~        0.587  0.535     5    -6.09     0      0.0898
## 4 3swc~ Psyc~ Post M~        0.739  0.559     8    -8.01     1      0.117
## 5 2G7V~ In M~ Drake          0.835  0.626     1    -5.83     1      0.125
## 6 7dt6~ Bett~ Post M~        0.68   0.563    10    -5.84     1      0.0454
## # ... with 7 more variables: acousticness <dbl>, instrumentalness <dbl>,
## #   liveness <dbl>, valence <dbl>, tempo <dbl>, duration_ms <dbl>,
## #   time_signature <dbl>
```

# Tidying the Dataset

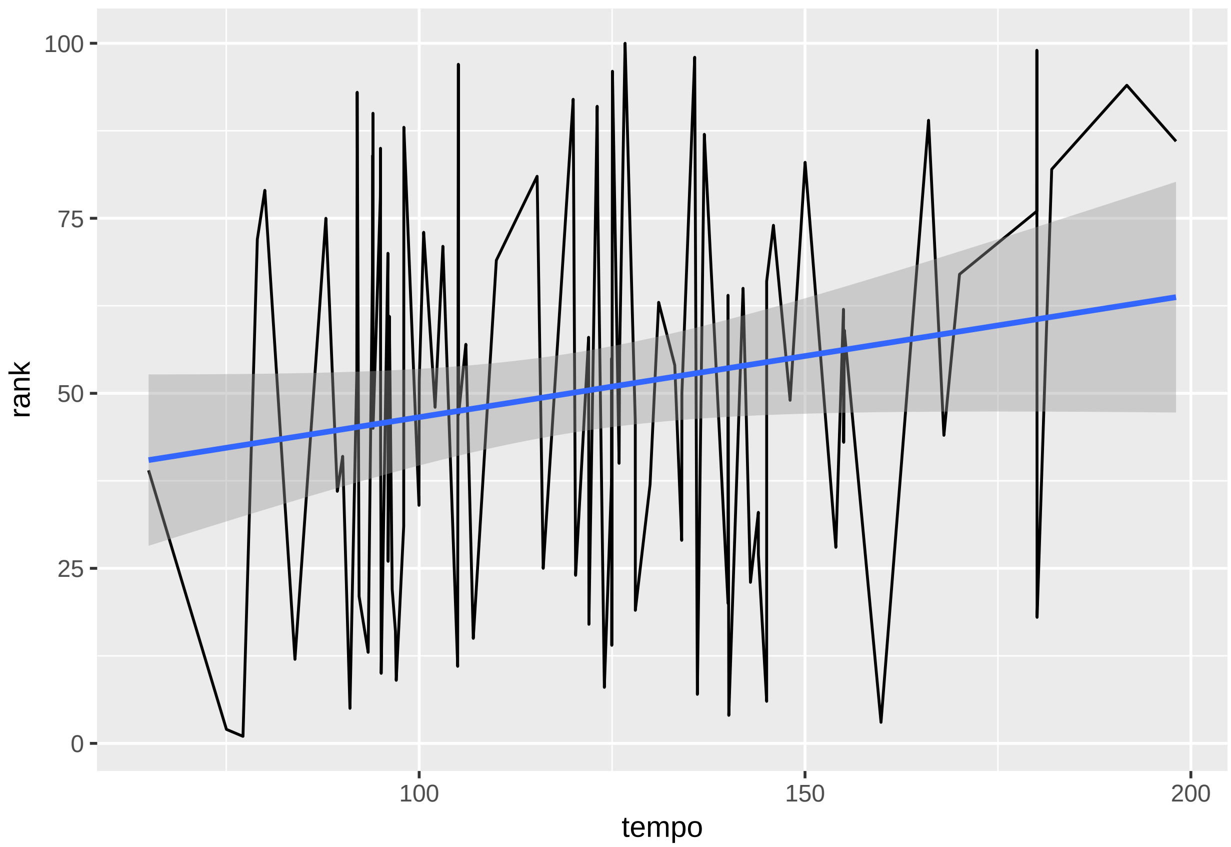
```r
spotify_data <- spotify_data %>%
  mutate(rank = seq(1, 100)) %>%
  mutate(duration_min = as.integer(duration_ms / 60000)) %>%
  mutate(duration_sec = as.integer((duration_ms - (duration_ms / 60000)) / 1000)) %>%
  unite("duration_time", duration_min, duration_sec, sep = ":") %>%
  type_convert(col_types = cols(end_datetime = col_datetime(format = "%M:%S"))) %>%
  drop_na()


head(spotify_data)
```

```
## # A tibble: 6 x 18
##   id    name  artists danceability energy   key loudness  mode speechiness
##   <chr> <chr> <chr>          <dbl>  <dbl> <dbl>    <dbl> <dbl>       <dbl>
## 1 6DCZ~ God'~ Drake          0.754  0.449     7    -9.21     1      0.109 
## 2 3ee8~ SAD!  XXXTEN~        0.74   0.613     8    -4.88     1      0.145 
## 3 0e7i~ rock~ Post M~        0.587  0.535     5    -6.09     0      0.0898
## 4 3swc~ Psyc~ Post M~        0.739  0.559     8    -8.01     1      0.117 
## 5 2G7V~ In M~ Drake          0.835  0.626     1    -5.83     1      0.125 
## 6 7dt6~ Bett~ Post M~        0.68   0.563    10    -5.84     1      0.0454
## # ... with 9 more variables: acousticness <dbl>, instrumentalness <dbl>,
## #   liveness <dbl>, valence <dbl>, tempo <dbl>, duration_ms <dbl>,
## #   time_signature <dbl>, rank <int>, duration_time <time>
```

# Statistical analysis and methods

## Is there a relationship between a song's tempo and its chart rank?

```r
spotify_data %>% ggplot(aes(x = tempo, y = rank)) + geom_line() + geom_smooth(method=lm)
```



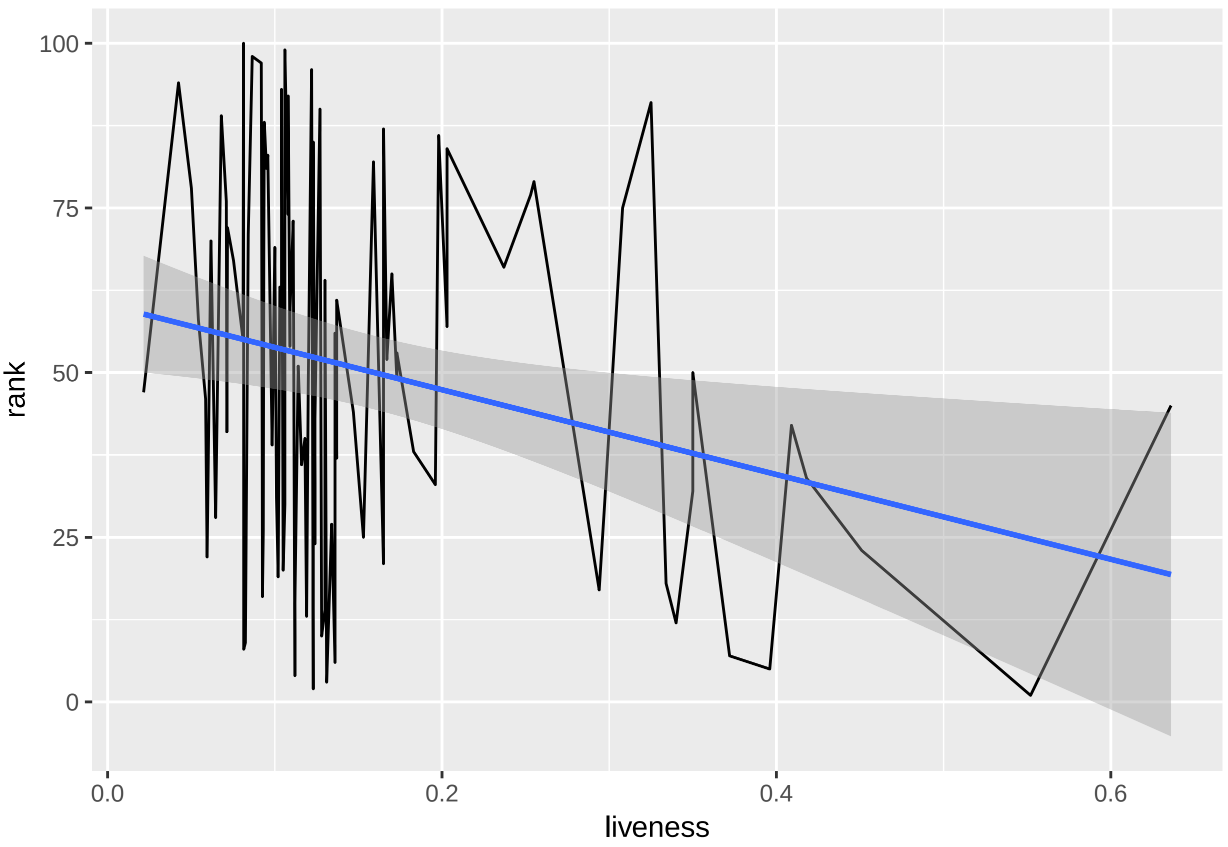
```
linear_regression <- lm(rank ~ tempo, data = spotify_data)

linear_regression %>%
  broom::tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic p.value
##   <chr>          <dbl>     <dbl>     <dbl>   <dbl>
## 1 (Intercept)   29.1      12.3         2.37  0.0196
## 2 tempo          0.175     0.0996      1.76  0.0822
```

We asked this question to help answer the overarching question of is there a perfect formula for what makes the ideal song? After noticing a slight increase in the trendline, we decided to perform a linear regression to determine if there was a relationship between the two. The linear regression analysis indicated that while the p value for the influence of tempo on rank was small at 0.0822, it was not smaller than our alpha value of 0.05 so we can not conclude that a statistically significant relationship exists between the two variables.

### Is there a relationship between a song's liveness and its chart rank?

```
spotify_data %>% ggplot(aes(x = liveness, y = rank)) + geom_line() + geom_smooth(method=lm)
```



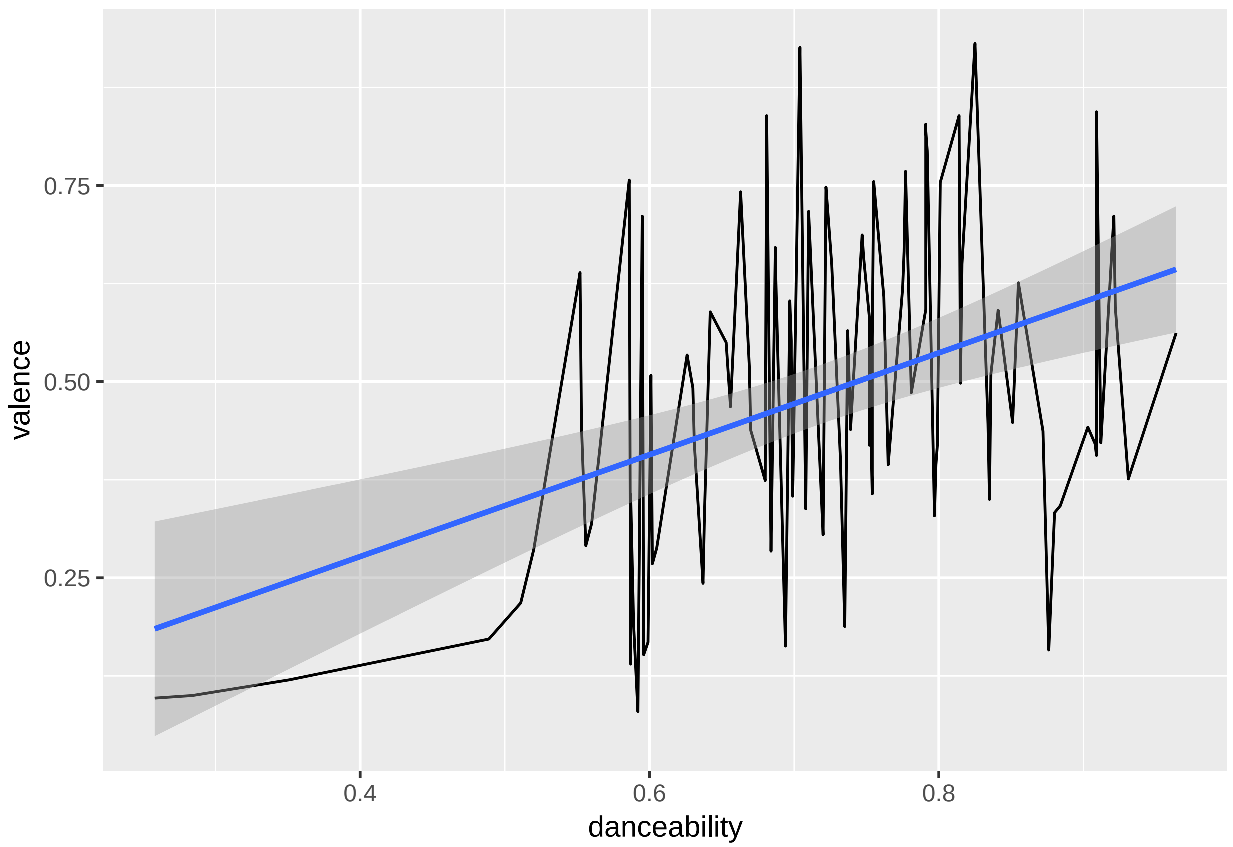
```
regression <- lm(rank ~ liveness, data = spotify_data)

regression %>%
  broom::tidy()
```

```
## # A tibble: 2 x 5
##   term        estimate std.error statistic  p.value
##   <chr>          <dbl>     <dbl>     <dbl>    <dbl>
## 1 (Intercept)     60.3      4.90     12.3  1.65e-21
## 2 liveness       -64.3     25.3      -2.55 1.24e- 2
```

After getting an inconclusive p-value for the relationship between a song's tempo and its chart rank, we began to run regressions on every value in the table to determine what factors did influence a song's rank. While most of the factors did not have an affect, we did find that there was a negative relationship between how "live" a song was and its rank. This can be seen by not only the small p-value of 0.0124 which is smaller than our alpha value of 0.05 proving that this finding is statistically significant, but also by the negative coefficient of the estimate which shows the negative relationship between these two variables. As a result of this we can conclude that songs that sound like they were performed in front of a live audience, such as in concert, do not as well as those produced in a studio.

### Is there a relationship between danceability and valence?

```
spotify_data %>% ggplot(aes(x = danceability, y = valence)) + geom_line() + geom_smooth(method=lm)
```



```
regression <- lm(valence ~ danceability, data = spotify_data)


regression %>%
  broom::tidy()
```

```
## # A tibble: 2 x 5
##   term         estimate std.error statistic   p.value
##   <chr>           <dbl>     <dbl>     <dbl>     <dbl>
## 1 (Intercept)    0.0176     0.105     0.167 0.868
## 2 danceability   0.649      0.145     4.49  0.0000198
```

With this project, we wanted to not only look for relationships between different factors and a song's performance, but we were also interested in the relationship between different variables. One hypothesis we created based on personal experiences was that more danceable songs typically came off to us as happier, thereby suggesting they might have more valence. To test this hypothesis, we decided to perform a regression analysis between these two variables. The small p-value of 0.0198 indicated that our hypothesis was true and that there is a relationship between these variables since this p-value is smaller than our alpha value of 0.05.

## Conclusion

In conclusion, while there was not a statistically significant linear relationship between most of the variables and the rank of a song, it can be concluded that the lifeness of a song negatively impacts its performance on Spotify. Although we were not able to detect many statistically significant linear relationships, because of how profitable the music industry is, we suggest that artists do more analysis to increase the number of listeners to their music. One way to improve upon our analysis would be to look into other types of relationships between variables and to use a larger dataset. Because our trendlines looked relatively linear to begin with, we decided not to go forward with any other relationship type since we did not believe it would have a more statistically significant outcome than the linear model. However, if a larger dataset was used than just the top 100 songs, it might be easier to detect if other relationships exist. In addition, because of the larger sample size, it would be easier to prove that relationships found are statistically significant because a larger sample size would decrease the standard error.

For this project, Jacqueline worked on the introduction and industry analysis. Julie researched what each of the industry terms meant to understand what each of the attributes represented. Jacqueline then tidied the data which Julie then analyzed. Both Jacqueline and Julie analyzed the results of their findings and worked to make meaningful conclusions from the results. Jacqueline then uploaded the project to Github to be submitted for the team.