IMT 573: Module 3 Lab

Advanced Visualization

Ali Qazi

Due: July 8, 2021

Collaborators: Akeel Qazi List collaborators here.

Objectives

As we continue our data science journey, we are gaining skills in working with data. This might be reflected in more efficient ways to manipulate and summarize data, both of which can be useful for creating more advanced visualizations of that data. To accomplish many of the visualization tasks in these exercises you will need to make use of newly acquired data manipulation skills!

Instructions

Before beginning this assignment, please ensure you have access to R and RStudio; this can be on your own personal computer or on the IMT 573 R Studio Cloud.

1. Open the `03_lab_advancedviz.Rmd` and save a copy to your local directory. Supply your solutions to the assignment by editing `03_lab_advancedviz.Rmd`.
2. First, replace the "YOUR NAME HERE" text in the `author:` field with your own full name. Any collaborators must be listed on the top of your assignment.
3. Be sure to include well-documented (e.g. commented) code chucks, figures, and clearly written text chunk explanations as necessary. Any figures should be clearly labeled and appropriately referenced within the text. Be sure that each visualization adds value to your written explanation; avoid redundancy – you do no need four different visualizations of the same pattern.
4. Collaboration on problem sets is fun and useful, and I encourage it, but each student must turn in an individual write-up in their own words as well as code/work that is their own. Regardless of whether you work with others, what you turn in must be your own work; this includes code and interpretation of results. The names of all collaborators must be listed on each assignment. Do not copy-and-paste from other students' responses or code.
5. All materials and resources that you use (with the exception of lecture slides) must be appropriately referenced within your assignment.
6. When you have completed the assignment and have **checked** that your code both runs in the Console and knits correctly when you click `Knit`. When the PDF report is generated rename the knitted PDF file to `lab3_YourLastName_YourFirstName.pdf`, and submit the PDF file on Canvas.

Setup

In this lab you will need, at minimum, the following R packages.

```
# Load standard libraries
library(tidyverse)
```

The data we will use in this lab comes from the Million Song Dataset. The Million Song Dataset is a collaboration between the Echo Nest and LabROSA, a laboratory working towards intelligent machine listening. The project was also funded in part by the National Science Foundation of America (NSF) to provide a large data set to evaluate research related to algorithms and information retrieval.

http://millionsongdataset.com/

Thierry Bertin-Mahieux, Daniel P.W. Ellis, Brian Whitman, and Paul Lamere. The Million Song Dataset. In Proceedings of the 12th International Society for Music Information Retrieval Conference (ISMIR 2011), 2011.

We will use a subset of this data created by Ryan Whitcomb, `rwhit94@vt.edu`, which contains data on 10,000 songs. The data contains standard information about the songs such as artist name, title, and year released. Additionally, the data contains more advanced information; for example, the length of the song, how many musical bars long the song is, and how long the fade in to the song was.

```
# Load music data
music_data <- read_csv("data/music.csv")
```

Problem 1: Inspection

First, inspect the data. You can use functions such as glimpse, head, tail, etc. to help you get a sense of what is contained in the data.

```
glimpse(music_data)
```

```
## Rows: 10,000
## Columns: 35
## $ artist.familiarity            <dbl> 0.58179377, 0.63063004, 0.48735679, 0.6~
## $ artist.hotttnesss             <dbl> 0.4019975, 0.4174996, 0.3434284, 0.4542~
## $ artist.id                     <chr> "ARD7TVE1187B99BFB1", "ARMJAGH1187FB546~
## $ artist.latitude               <dbl> 0.00000, 35.14968, 0.00000, 0.00000, 0.~
## $ artist.location               <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ artist.longitude              <dbl> 0.00000, -90.04892, 0.00000, 0.00000, 0~
## $ artist.name                   <chr> "Casual", "The Box Tops", "Sonora Santa~
## $ artist.similar                <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ artist.terms                  <chr> "hip hop", "blue-eyed soul", "salsa", "~
## $ artist.terms_freq             <dbl> 1.0000000, 1.0000000, 1.0000000, 0.9885~
## $ release.id                    <dbl> 300848, 300822, 514953, 287650, 611336,~
## $ release.name                  <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ song.artist_mbtags            <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ song.artist_mbtags_count      <dbl> 0, 1, 0, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ song.bars_confidence          <dbl> 0.643, 0.007, 0.980, 0.017, 0.175, 0.12~
## $ song.bars_start               <dbl> 0.58521, 0.71054, 0.73152, 1.30621, 1.0~
## $ song.beats_confidence         <dbl> 0.834, 1.000, 0.980, 0.809, 0.883, 0.43~
## $ song.beats_start              <dbl> 0.58521, 0.20627, 0.73152, 0.81002, 0.1~
## $ song.duration                 <dbl> 218.9318, 148.0355, 177.4755, 233.4036,~
## $ song.end_of_fade_in           <dbl> 0.247, 0.148, 0.282, 0.000, 0.066, 2.26~
## $ song.hotttnesss               <dbl> 0.6021200, -1.0000000, -1.0000000, -1.0~
## $ song.id                       <chr> "SOMZWCG12A8C13C480", "SOCIWDW12A8C13D4~
## $ song.key                      <dbl> 1, 6, 8, 0, 2, 5, 1, 4, 4, 7, 5, 7, 9, ~
## $ song.key_confidence           <dbl> 0.736, 0.169, 0.643, 0.751, 0.092, 0.63~
## $ song.loudness                 <dbl> -11.197, -9.843, -9.689, -9.013, -4.501~
## $ song.mode                     <dbl> 0, 0, 1, 1, 1, 1, 1, 0, 1, 0, 0, 1, 1, ~
```

```
## $ song.mode_confidence           <dbl> 0.636, 0.430, 0.565, 0.749, 0.371, 0.55~
## $ song.start_of_fade_out         <dbl> 218.932, 137.915, 172.304, 217.124, 198~
## $ song.tatums_confidence         <dbl> 0.779, 0.969, 0.482, 0.601, 1.000, 0.13~
## $ song.tatums_start              <dbl> 0.28519, 0.20627, 0.42132, 0.56254, 0.1~
## $ song.tempo                     <dbl> 92.198, 121.274, 100.070, 119.293, 129.~
## $ song.time_signature            <dbl> 4, 4, 1, 4, 4, 3, 1, 3, 4, 4, 1, 4, 4, ~
## $ song.time_signature_confidence <dbl> 0.778, 0.384, 0.000, 0.000, 0.562, 0.45~
## $ song.title                     <dbl> 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, 0, ~
## $ song.year                      <dbl> 0, 1969, 0, 1982, 2007, 0, 0, 0, 1984, ~
```

```
head(music_data)
```

```
## # A tibble: 6 x 35
##   artist.familiari~ artist.hotttnes~ artist.id   artist.latitude artist.location
##               <dbl>            <dbl> <chr>                <dbl>           <dbl>
## 1             0.582            0.402 ARD7TVE118~            0                0
## 2             0.631            0.417 ARMJAGH118~           35.1              0
## 3             0.487            0.343 ARKRRTF118~            0                0
## 4             0.630            0.454 AR7G5I4118~            0                0
## 5             0.651            0.402 ARXR32B118~            0                0
## 6             0.535            0.385 ARKFYS9118~            0                0
## # ... with 30 more variables: artist.longitude <dbl>, artist.name <chr>,
## #   artist.similar <dbl>, artist.terms <chr>, artist.terms_freq <dbl>,
## #   release.id <dbl>, release.name <dbl>, song.artist_mbtags <dbl>,
## #   song.artist_mbtags_count <dbl>, song.bars_confidence <dbl>,
## #   song.bars_start <dbl>, song.beats_confidence <dbl>, song.beats_start <dbl>,
## #   song.duration <dbl>, song.end_of_fade_in <dbl>, song.hotttnesss <dbl>,
## #   song.id <chr>, song.key <dbl>, song.key_confidence <dbl>,
## #   song.loudness <dbl>, song.mode <dbl>, song.mode_confidence <dbl>,
## #   song.start_of_fade_out <dbl>, song.tatums_confidence <dbl>,
## #   song.tatums_start <dbl>, song.tempo <dbl>, song.time_signature <dbl>,
## #   song.time_signature_confidence <dbl>, song.title <dbl>, song.year <dbl>
```

```
tail(music_data)
```

```
## # A tibble: 6 x 35
##   artist.familiari~ artist.hotttnes~ artist.id   artist.latitude artist.location
##               <dbl>            <dbl> <chr>                <dbl>           <dbl>
## 1             0.607            0.401 ARDK055118~           31.3              0
## 2             0.723            0.500 AR4C6V0118~           39.6              0
## 3             0.512            0.410 AR9JLBU118~          -34.0              0
## 4             0.434            0.290 ARS1DCR118~            0                0
## 5             0.334            0.217 ARAGMIV11F~            0                0
## 6             0.609            0.509 ARYXOV8118~            0                0
## # ... with 30 more variables: artist.longitude <dbl>, artist.name <chr>,
## #   artist.similar <dbl>, artist.terms <chr>, artist.terms_freq <dbl>,
## #   release.id <dbl>, release.name <dbl>, song.artist_mbtags <dbl>,
## #   song.artist_mbtags_count <dbl>, song.bars_confidence <dbl>,
## #   song.bars_start <dbl>, song.beats_confidence <dbl>, song.beats_start <dbl>,
## #   song.duration <dbl>, song.end_of_fade_in <dbl>, song.hotttnesss <dbl>,
## #   song.id <chr>, song.key <dbl>, song.key_confidence <dbl>,
## #   song.loudness <dbl>, song.mode <dbl>, song.mode_confidence <dbl>,
## #   song.start_of_fade_out <dbl>, song.tatums_confidence <dbl>,
## #   song.tatums_start <dbl>, song.tempo <dbl>, song.time_signature <dbl>,
```

```
## #   song.time_signature_confidence <dbl>, song.title <dbl>, song.year <dbl>
```

```
summary(music_data)
```

```
##  artist.familiarity artist.hotttnesss  artist.id         artist.latitude
##  Min.   :0.0000     Min.   :0.0000    Length:10000       Min.   :-41.28
##  1st Qu.:0.4676     1st Qu.:0.3253    Class :character   1st Qu.:  0.00
##  Median :0.5636     Median :0.3807    Mode  :character   Median :  0.00
##  Mean   :0.5652     Mean   :0.3856                       Mean   : 13.90
##  3rd Qu.:0.6680     3rd Qu.:0.4539                       3rd Qu.: 34.42
##  Max.   :1.0000     Max.   :1.0825                       Max.   : 69.65
##  artist.location   artist.longitude  artist.name        artist.similar
##  Min.   :  0.000   Min.   :-162.44   Length:10000       Min.   :0
##  1st Qu.:  0.000   1st Qu.: -73.95   Class :character   1st Qu.:0
##  Median :  0.000   Median :   0.00   Mode  :character   Median :0
##  Mean   :  0.078   Mean   : -23.92                      Mean   :0
##  3rd Qu.:  0.000   3rd Qu.:   0.00                      3rd Qu.:0
##  Max.   :780.000   Max.   : 174.77                      Max.   :0
##  artist.terms       artist.terms_freq     release.id      release.name
##  Length:10000       Min.   :      0.0   Min.   :     0   Min.   :    0.0
##  Class :character   1st Qu.:      0.9   1st Qu.:172858   1st Qu.:    0.0
##  Mode  :character   Median :      1.0   Median :333103   Median :    0.0
##                     Mean   :    224.9   Mean   :371024   Mean   :   23.1
##                     3rd Qu.:      1.0   3rd Qu.:573532   3rd Qu.:    0.0
##                     Max.   :2239217.0   Max.   :823599   Max.   :85555.0
##  song.artist_mbtags song.artist_mbtags_count song.bars_confidence
##  Min.   :0.00e+00   Min.   :0.0000           Min.   :0.0000
##  1st Qu.:0.00e+00   1st Qu.:0.0000           1st Qu.:0.0350
##  Median :0.00e+00   Median :0.0000           Median :0.1200
##  Mean   :3.33e-05   Mean   :0.5247           Mean   :0.2396
##  3rd Qu.:0.00e+00   3rd Qu.:1.0000           3rd Qu.:0.3510
##  Max.   :3.33e-01   Max.   :9.0000           Max.   :8.8552
##  song.bars_start   song.beats_confidence song.beats_start   song.duration
##  Min.   : 0.0000   Min.   :0.0000        Min.   :-60.0000   Min.   :    1.044
##  1st Qu.: 0.4416   1st Qu.:0.4098        1st Qu.:  0.1947   1st Qu.:  176.032
##  Median : 0.7855   Median :0.6860        Median :  0.3326   Median :  223.059
##  Mean   : 1.0653   Mean   :0.6140        Mean   :  0.4285   Mean   :  240.622
##  3rd Qu.: 1.2241   3rd Qu.:0.8820        3rd Qu.:  0.5008   3rd Qu.:  276.375
##  Max.   :59.7435   Max.   :1.0000        Max.   : 12.2458   Max.   :22050.000
##  song.end_of_fade_in song.hotttnesss     song.id            song.key
##  Min.   : 0.0000     Min.   :-1.0000   Length:10000       Min.   :  0.000
##  1st Qu.: 0.0000     1st Qu.:-1.0000   Class :character   1st Qu.:  2.000
##  Median : 0.1990     Median : 0.0000   Mode  :character   Median :  5.000
##  Mean   : 0.7567     Mean   :-0.2415                      Mean   :  5.367
##  3rd Qu.: 0.4210     3rd Qu.: 0.4051                      3rd Qu.:  8.000
##  Max.   :43.1190     Max.   : 1.0000                      Max.   :904.803
##  song.key_confidence song.loudness       song.mode     song.mode_confidence
##  Min.   : 0.0000     Min.   :-51.643   Min.   :0.000   Min.   :0.0000
##  1st Qu.: 0.2250     1st Qu.:-13.160   1st Qu.:0.000   1st Qu.:0.3600
##  Median : 0.4690     Median : -9.380   Median :1.000   Median :0.4870
##  Mean   : 0.4515     Mean   :-10.484   Mean   :0.691   Mean   :0.4778
##  3rd Qu.: 0.6590     3rd Qu.: -6.531   3rd Qu.:1.000   3rd Qu.:0.6060
##  Max.   :19.0810     Max.   :  0.566   Max.   :1.000   Max.   :1.0000
##  song.start_of_fade_out song.tatums_confidence song.tatums_start
```

```
##  Min.   : -21.39        Min.   :0.0000         Min.   : 0.0000
##  1st Qu.: 168.86        1st Qu.:0.2370         1st Qu.: 0.1107
##  Median : 213.86        Median :0.5000         Median : 0.1915
##  Mean   : 229.88        Mean   :0.5079         Mean   : 0.2999
##  3rd Qu.: 266.27        3rd Qu.:0.7742         3rd Qu.: 0.2947
##  Max.   :1813.43        Max.   :9.2276         Max.   :12.2458
##    song.tempo     song.time_signature song.time_signature_confidence
##  Min.   :  0.00   Min.   :0.000       Min.   :  0.0000
##  1st Qu.: 96.96   1st Qu.:3.000       1st Qu.:  0.0978
##  Median :120.16   Median :4.000       Median :  0.5510
##  Mean   :122.90   Mean   :3.564       Mean   :  0.5998
##  3rd Qu.:144.01   3rd Qu.:4.000       3rd Qu.:  0.8640
##  Max.   :262.83   Max.   :7.000       Max.   :898.8910
##    song.title         song.year
##  Min.   :    0.00   Min.   :   0.0
##  1st Qu.:    0.00   1st Qu.:   0.0
##  Median :    0.00   Median :   0.0
##  Mean   :   10.01   Mean   : 934.7
##  3rd Qu.:    0.00   3rd Qu.:2000.0
##  Max.   :94496.00   Max.   :2010.0
```

```
dim(music_data)
```

```
## [1] 10000    35
```

Looking at the music dataset, I can see that there is 35 variables and 10,000 observations. The variable types are doubles and characters. There are multiple columns with all rows value of 0. This means that there is no information about the variable or the information is not useful such as a song title.

Problem 2: Pose a Question

Propose a question to guide your analysis. For example, you might ask if the average hotness scores of songs change over time? Or perhaps, what is the relationship between song duration and tempo? You can use one of these questions or develop your own. State which question you want to answer.

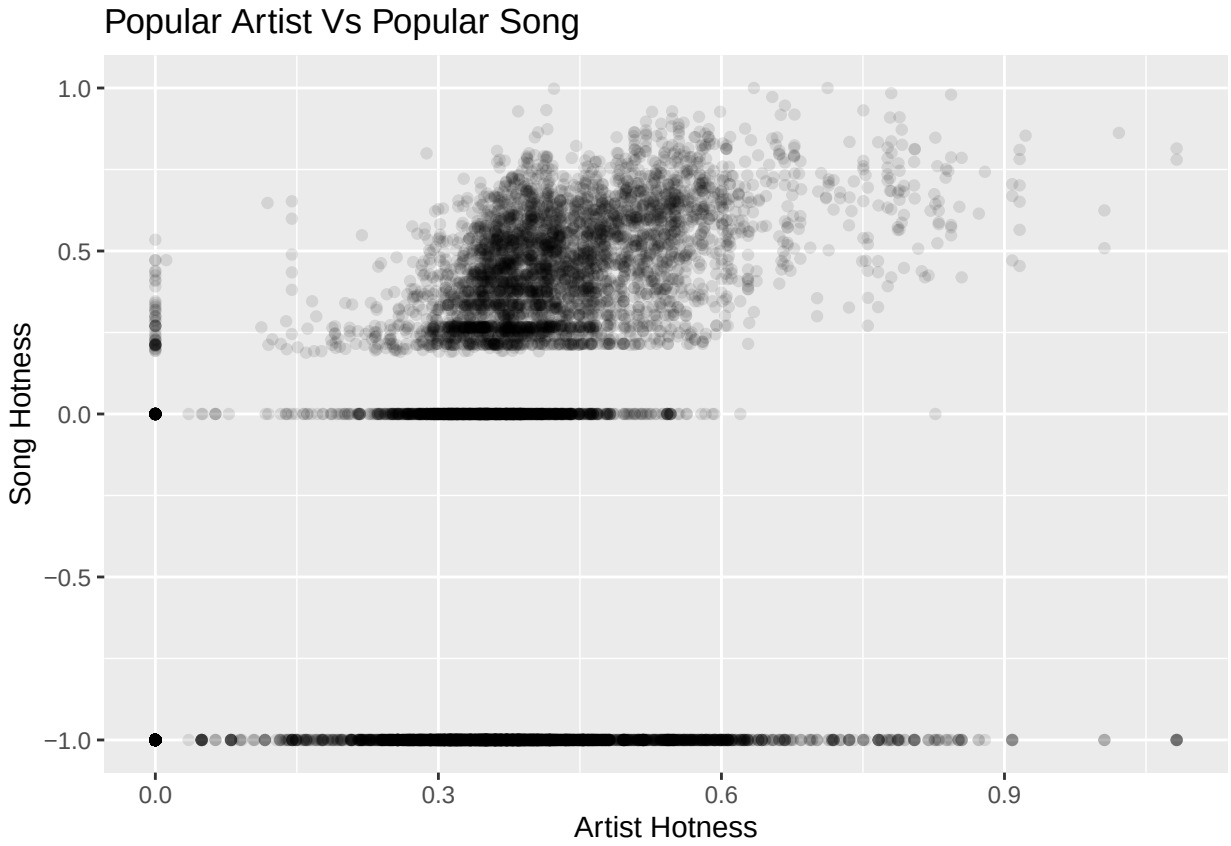
What is song hotttnesss you ask? According to the dataset description, it is a measure of the song's popularity, when downloaded (in December 2010). And measured on a scale of 0 to 1.

Question: What is the relationship between the artist hotttness and the song hotttness?

Problem 3: Visualization

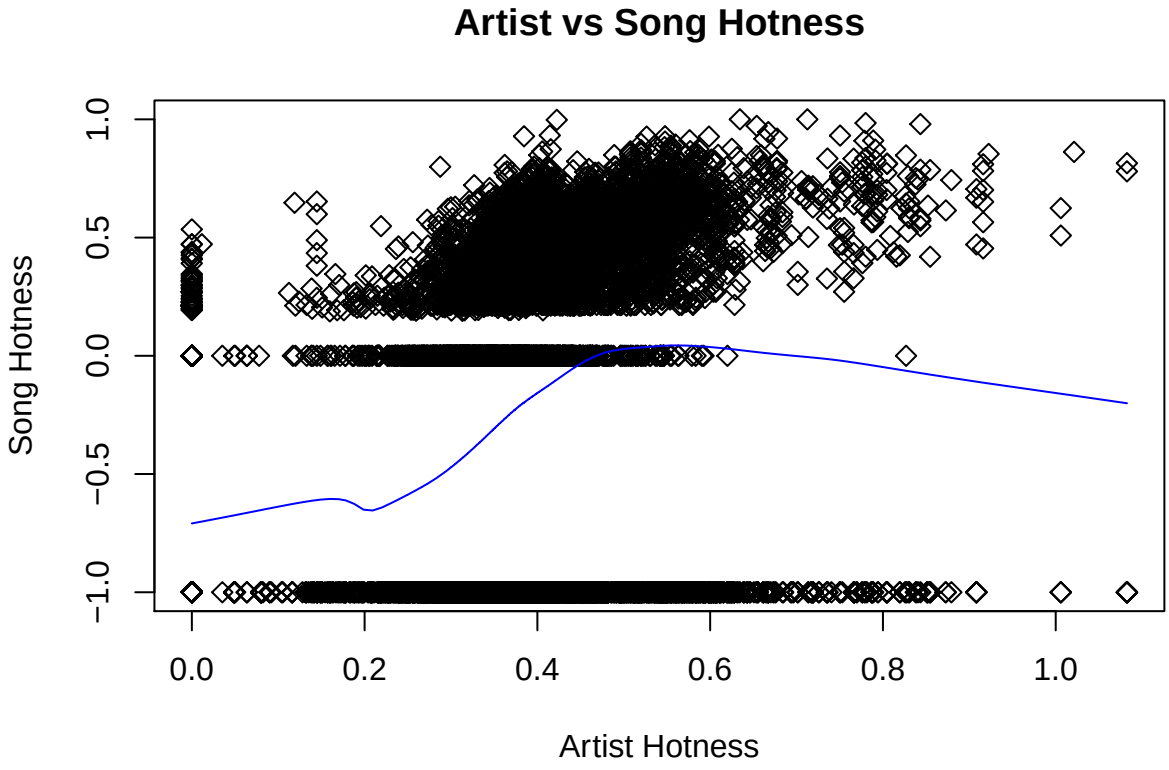
Create two visualizations to help gain insight into your question. Be sure to explain the visuals you create and what you take away from them.

```
ggplot(data = music_data,
mapping = aes(x = artist.hotttnesss, y = song.hotttnesss)) +
ggtitle("Popular Artist Vs Popular Song", subtitle = waiver()) +
xlab("Artist Hotness") +
ylab("Song Hotness") +
geom_point(alpha = 1/10) ## opacity of the geom
```



In the scatterplot above we see that there is a general relationship between artist hotness and song hotness. There are many songs that are 0 or -1 that is seen in a wide range of artist hotness. At the top of the visualization

```
artisthotness = music_data$artist.hotttnesss
songhotness = music_data$song.hotttnesss
plot(artisthotness, songhotness, main = "Artist vs Song Hotness",
xlab = "Artist Hotness", ylab = "Song Hotness", pch = 5)
lines(lowess(artisthotness, songhotness), col = "blue") ## locally weighted scatter plot smoothing
```



The line in the scatterplot is suppose to show the relationship between the two variables, artist hotness and song hotness. Lowess stands for locally weighted scatterplot smoothing. It creates a smooth line that shows the relationship. It looks that the song hotness is having a big impact with the lowess line as there are many songs that are 0 or -1.