Zeru-Zhou-project06

October 14, 2021

1 Project 6 – Zeru Zhou

TA Help: NA

Collaboration: NA

- Get help from piazza
- Get help from Dr. Ward's video

1.1 Question 1

[2]:
```
tracks <- read.csv("/depot/datamine/data/amazon/tracks.csv")
```

[3]:
```
dim(tracks)
```

1. 1000000 2. 1

[4]:
```
str(tracks)
```

```
'data.frame':   1000000 obs. of  1 variable:
 $ track_id.title.song_id.release.artist_id.artist_mbid.artist_name.duration.art
ist_familiarity.artist_hotttnesss.year.track_7digitalid.shs_perf.shs_work: chr
"TRMMMYQ128F932D901|Silent Night|SOQMMHC12AB0180CB8|Monster Ballads
X-Mas|ARYZTJS1187B98C555|357ff05d-848a-44cf-"| __truncated__
"TRMMMKD128F425225D|Tanssi vaan|SOVFVAK12A8C1350D9|Karkuteillä|ARMVN3U1187FB3A1E
B|8d7ef530-a6fd-4f8f-b2e2-74aec7"| __truncated__ "TRMMMRX128F93187D9|No One
Could Ever|SOGTUKN12AB017F4F1|Butter|ARGEKB01187FB50750|3d403d44-36ce-465c-ad43-
ae877"| __truncated__ "TRMMMCH128F425532C|Si Vos Querés|SOBNYVR12A8C13558C|De
Culo|ARNWYLR1187B9B2F9C|12be7648-7094-495f-90e6-df4189d6"| __truncated__ …
```

[1]:
```
tracks <- read.csv("/depot/datamine/data/amazon/tracks.csv",sep="|")
```

[6]:
```
dim(tracks)
```

1. 1000000 2. 14

[7]:
```
str(tracks)
```

```
'data.frame':   1000000 obs. of  14 variables:
 $ track_id          : chr  "TRMMMYQ128F932D901" "TRMMMKD128F425225D"
```

```
"TRMMMRX128F93187D9" "TRMMMCH128F425532C" …
 $ title             : chr  "Silent Night" "Tanssi vaan" "No One Could Ever" "Si
Vos Querés" …
 $ song_id           : chr  "SOQMMHC12AB0180CB8" "SOVFVAK12A8C1350D9"
"SOGTUKN12AB017F4F1" "SOBNYVR12A8C13558C" …
 $ release           : chr  "Monster Ballads X-Mas" "Karkuteillä" "Butter" "De
Culo" …
 $ artist_id         : chr  "ARYZTJS1187B98C555" "ARMVN3U1187FB3A1EB"
"ARGEKB01187FB50750" "ARNWYLR1187B9B2F9C" …
 $ artist_mbid       : chr  "357ff05d-848a-44cf-b608-cb34b5701ae5"
"8d7ef530-a6fd-4f8f-b2e2-74aec765e0f9" "3d403d44-36ce-465c-ad43-ae877e65adc4"
"12be7648-7094-495f-90e6-df4189d68615" …
 $ artist_name       : chr  "Faster Pussy cat" "Karkkiautomaatti" "Hudson
Mohawke" "Yerba Brava" …
 $ duration          : num  252 157 139 145 514 …
 $ artist_familiarity: num  0.65 0.44 0.644 0.449 0 …
 $ artist_hotttnesss : num  0.394 0.357 0.438 0.372 0 …
 $ year              : int  2003 1995 2006 2003 0 0 0 1993 0 0 …
 $ track_7digitalid  : int  7032331 1514808 6945353 2168257 2264873 3360982
552626 6435649 8376489 1043208 …
 $ shs_perf          : int  -1 -1 -1 -1 -1 -1 -1 -1 -1 -1 …
 $ shs_work          : int  0 0 0 0 0 0 0 0 0 0 …
```

[8]:
```
head(tracks)
```

A data.frame: 6 x 14

track_id <chr>	title <chr>
TRMMMYQ128F932D901	Silent Night
TRMMMKD128F425225D	Tanssi vaan
TRMMMRX128F93187D9	No One Could Ever
TRMMMCH128F425532C	Si Vos Querés
TRMMMWA128F426B589	Tangle Of Aspens
TRMMMXN128F42936A5	Symphony No. 1 G minor "Sinfonie Serieuse"/Allegro con ene

We can see that originally it has only 1 column, and there are many "|" in that column; After using "sep" when reading the data, there are 14 columns now being seperated by "|", and "str" command gives us information about each column.

1.2 Question 2

[9]:
```
library(RSQLite)

con <- dbConnect(SQLite(), dbname = "/depot/datamine/data/amazon/tracks.db")
myDF <- dbGetQuery(con, "SELECT year, AVG(duration) AS average_duration FROM␣
 ↪songs GROUP BY year;")
head(myDF)
```

A data.frame: 6 x 2

year <int>	average_duration <dbl>
0	252.3017
1922	222.2363
1924	186.1690
1925	185.5846
1926	185.9089
1927	183.8967

[11]:
```
head(tapply(tracks$duration, tracks$year, mean))
```

0 252.301709364854 **1922** 222.236281666667 **1924** 186.169016 **1925** 185.584618571429 **1926** 185.908892105263 **1927** 183.896727906977

As we can see, the results of the given code is exactly the same as the result if we use tapply function.

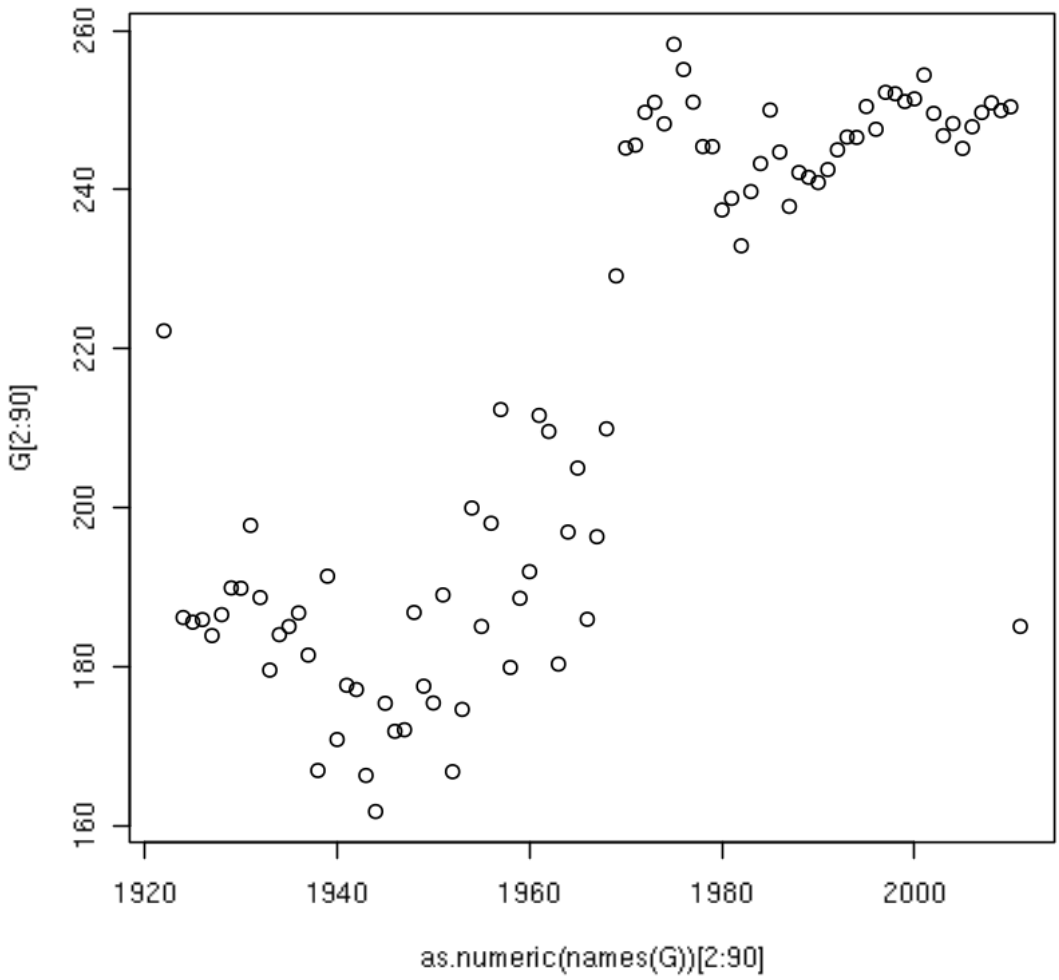
1.3 Question 3

[12]:
```
G <- tapply(tracks$duration, tracks$year, mean)
```

[14]:
```
length(G)
```

90

[19]:
```
plot(as.numeric(names(G))[2:90], G[2:90])
```



Except for several outliers, as the time goes by, the duration of musics is became longer and longer. (As the year increases, duration increases, in the general trend).

1.4 Question 4

[2]:
```
head(tapply(tracks$duration, tracks$artist_name, median))
```

->School<- 215.45751 **-123 minut** 228.93669 **-123min.** 238.96771 **-M-** 174.22322 **:Blacks On :Blondes** 291.7873 **:Metaphor:** 307.604445

[3]:
```
head(sort(tapply(tracks$duration, tracks$artist_name, median), decreasing=T))
```

Ustad Rashid Khan 3033.59955 **Galexis** 3033.44281 **Heiko Grauel** 3032.58077 **Kushal Das** 3032.5024 **Francis B** 3030.62159 **Buddhadev Dasgupta** 3030.17751

The artist_name with the highest median duration is "Ustad Rashid Khan". The 5 results sorted in decreasing order is listed above.

1.5 Question 5

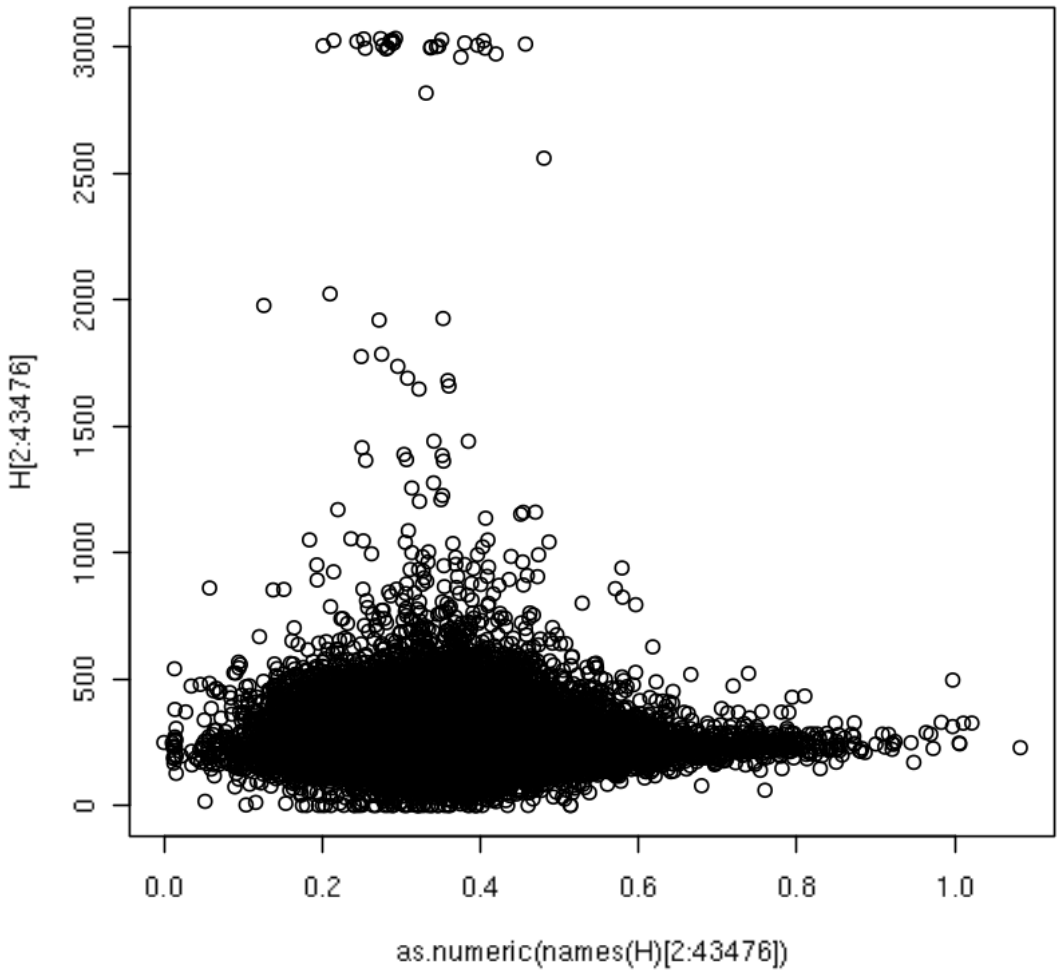
[ ]:
```
# Question: Plot the average duration with respect to artist_hotttnesss. Are␣
↪there any patterns? What is the artist_hotttnesss of the lowest average␣
↪duration?
```

[4]:
```
H <- tapply(tracks$duration, tracks$artist_hotttnesss, mean)
```

[6]:
```
length(H)
```

43476

[7]:
```
plot(as.numeric(names(H)[2:43476]),H[2:43476])
```



[9]:
```
head(sort(H))
```

0.346540022968 0.41751 **0.241032440462** 0.46975 **0.288817969432** 0.49587 **0.395026516748** 0.49587 **0.267992430844** 0.495875 **0.211040665303** 0.522

As the result, the plot shows no obvious pattern but extremely high durations have artist_hotttnesss in range between 0.2 and 0.5. The artist_hotttnesss of the lowest average duration is around 0.3465, as calculated above.

1.6 Question 6

[ ]:
```
# Average duration with respect to different artist_familiarity
```

[28]:
```
library(RSQLite)

con <- dbConnect(SQLite(), dbname = "/depot/datamine/data/amazon/tracks.db")
myDF <- dbGetQuery(con, "SELECT artist_hotttnesss, AVG(duration) AS␣
↪average_duration FROM songs GROUP BY artist_hotttnesss;")
```

[29]:
```
head(myDF)
```

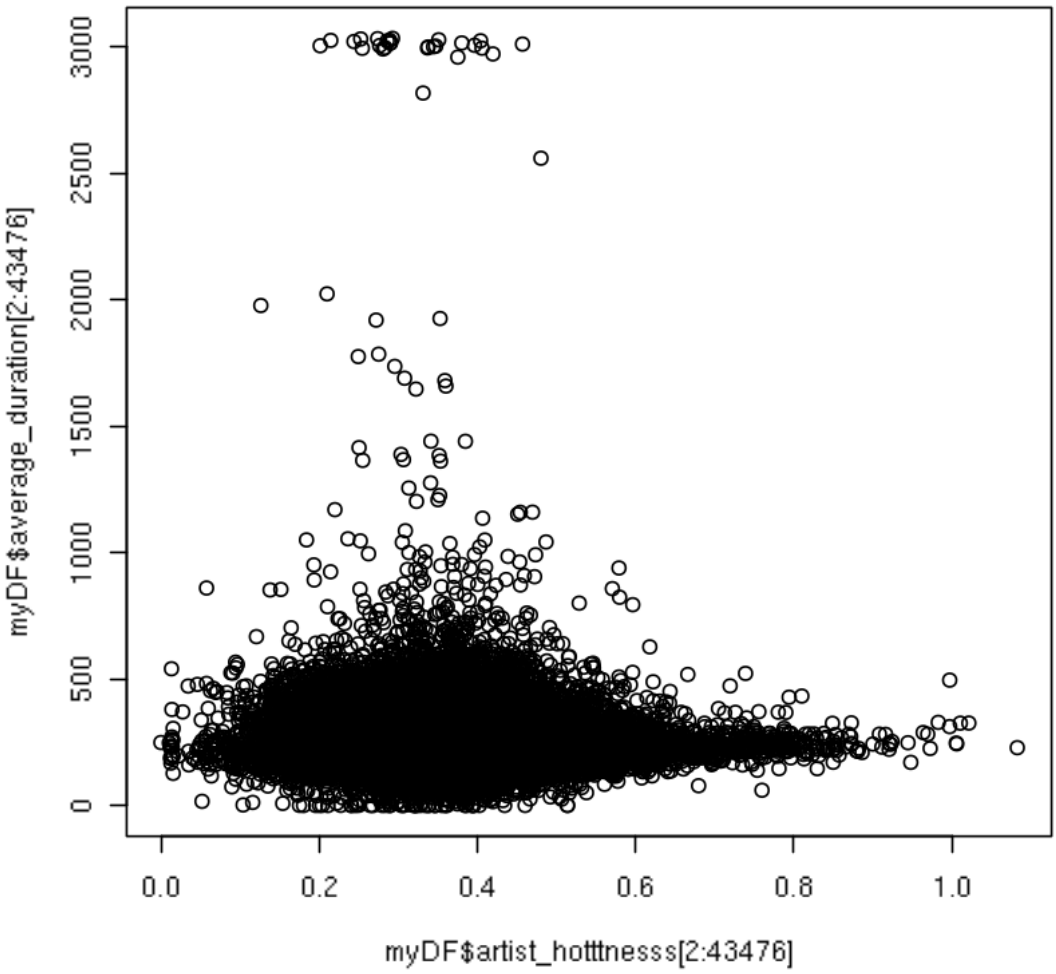
A data.frame: 6 x 2

artist_hotttnesss <dbl>	average_duration <dbl>
-1.00000000	369.2164
0.00000000	249.6095
0.01056930	247.9391
0.01156180	230.9681
0.01197357	234.9472
0.01206186	189.4424

[34]:
```
length(myDF$artist_hotttnesss)
```

43476

[35]:
```
plot(myDF$artist_hotttnesss[2:43476], myDF$average_duration[2:43476])
```



I got the same result as in question# 5, using the SQL code provided.

1.7 Pledge

By submitting this work I hereby pledge that this is my own, personal work. I've acknowledged in the designated place at the top of this file all sources that I used to complete said work, including but not limited to: online resources, books, and electronic communications. I've noted all collaboration with fellow students and/or TA's. I did not copy or plagiarize another's work.

> As a Boilermaker pursuing academic excellence, I pledge to be honest and true in all that I do. Accountable together – We are Purdue.