Data 607 - Homework 2

Cameron Smith

9/4/2020

Overview

For this assignment we were asked to do the following:

Choose six recent popular movies. Ask at least five people that you know (friends, family, classmates, imaginary friends if necessary) to rate each of these movies that they have seen on a scale of 1 to 5. Take the results (observations) and store them in a SQL database of your choosing. Load the information from the SQL database into an R dataframe.

A combination of PostgreSQL, Google Forms, and R was used for this solution. Details below.

Data collection and database creation

Google Survey

This data was collected via a Google Form accessible at the following link, the results of which were exported into a .CSV file and imported into the 'responses' table of the database.

Link to Form: https://docs.google.com/forms/d/e/1FAIpQLSeUDF4h8RW-FSAWgZcM2p1OVLhsQ6Nl61ExdHVZ28dWFu4bow/viewform

Link to CSV File: https://github.com/cwestsmith/cuny-msds/blob/master/data607/homework/homework%202/response_data.csv

Database and table creation

A PostgreSQL database named 'movieratings' was created for this homework, with two separate tables: (1) movies; and (2) responses. Movies was populated with movie titles, release dates, and genre categories. Responses was populated with information from the above-mentioned survey results.

The SQL code is included in Github as a separate file at the following link:

https://github.com/cwestsmith/cuny-msds/blob/master/data607/homework/homework%202/sql_code.sql

R Code to load and prepare data

Load required packages

```r
library("tibble")
library("DBI")
library("ggplot2")
library("formattable")
```

Connect to PostgreSQL database

For security a temporary user with limited privileges was created

```
db <- "movieratings"
host_db <- "localhost"
db_port <- "5432"
db_user <- "data607"
db_password <- "data607"
con <- dbConnect(RPostgres::Postgres(), dbname = db, host=host_db, port=db_port, user=db_user, password=
# Confirm database is connected
dbListTables(con)
```

```
## [1] "movies"  "ratings"
```

Query the two normalized tables for the required data and save the results to a dataframe

The query includes a WHERE clause to exclude the missing data. Another option could be to pull in all the data and then use subset.

```
sql_query <- dbSendQuery(con, "SELECT ratings.movie_name, respondent, date, rating, release_date, categ
df <- dbFetch(sql_query)
head(df)
```

```
##                                           movie_name
## 1 The Gentlemen
## 2 Ordinary Love
## 3 The Invisible Man
## 4 Bad Boys for Life
## 5 The Gentlemen
## 6 Ordinary Love
##                                           respondent       date rating
## 1 User 1                                             2020-09-03      2
## 2 User 1                                             2020-09-03      3
## 3 User 1                                             2020-09-03      2
## 4 User 1                                             2020-09-03      3
## 5 User 2                                             2020-09-03      2
## 6 User 2                                             2020-09-03      1
##   release_date                                           category
## 1   2020-03-24 Action
## 2   2019-09-09 Drama
## 3   2020-02-24 Horror
## 4   2020-01-17 Action
## 5   2020-03-24 Action
## 6   2019-09-09 Drama
```

```
dbClearResult(sql_query)
dbDisconnect(con)
```

Verify data is prepared correctly and ready for analysis

```
glimpse(df)
```

```
## Rows: 26
## Columns: 6
## $ movie_name   <chr> "The Gentlemen                                     ", ...
## $ respondent   <chr> "User 1                                            ", ...
## $ date         <date> 2020-09-03, 2020-09-03, 2020-09-03, 2020-09-03, 2020-...
## $ rating       <int> 2, 3, 2, 3, 2, 1, 1, 3, 4, 1, 2, 5, 4, 4, 2, 5, 2, 3, ...
## $ release_date <date> 2020-03-24, 2019-09-09, 2020-02-24, 2020-01-17, 2020-...
## $ category     <chr> "Action                                            ", ...
```

Narrative questions

Answer to analysis question in rubric document

Is there a movie that you would recommend or not recommend to one of the participants? Explain your reasoning.

I would recommend movies to the participants based on movies with similar attributes to ones that were ranked high (3 or above on the 1 through 5 scale).

Answer to additional questions in assignment

Use survey software to gather the information

Done, via Google Forms

Are you able to use a password without having to share the password with people who are viewing your code? There are a lot of interesting approaches that you can uncover with a little bit of research.

Temporary user with read only (SELECT) access created for increased security.

While it's acceptable to create a single SQL table, can you create a normalized set of tables that corresponds to the relationship between your movie viewing friends and the movies being rated?

Done, please see above for details.

Is there any benefit in standardizing ratings? How might you approach this?

Definitely. I would approach this by including clear definitions with examples for ranking options so that users are 'singing from the same song book'.

END OF DOCUMENT