MichaelY-DATA607-Week02-Movies

Michael Y.

September 9, 2018

Assignment 2: Movies Database

Already done:

Choose six recent popular movies.

Ask at least five people that you know (friends, family, classmates, imaginary friends)

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.

To be done here: Load the information into an R dataframe, and examine it.

Load up some libraries

```
library(RMySQL)
```

```
## Loading required package: DBI
```

```
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(ggplot2)
library(psych)
```

```
##
## Attaching package: 'psych'
```

```
## The following objects are masked from 'package:ggplot2':
##
##     %+%, alpha
```

Connect to the MySql database and retrieve the data:

```
# I created "stduser" as a read-only account in my database which only has "select" privilege
connstd <- dbConnect(MySQL(), user="stduser", password="password",
                     dbname="Week2_Movies", host="localhost")

# create a query which joins the 3 database tables,
# replacing the auto-generated ID codes with the movie names and the reviewers' names

query <- 'Select M.Movie_title, F.Friend_name, R.Rating
          From Movies as M, Friends as F, Ratings as R
          Where (M.Movie_id = R.Movie_ID AND F.Friend_id = R.Friend_ID);'

# execute the query
result <- dbGetQuery(connstd, query)

# close the database connection
discard <- dbDisconnect(connstd) # this function returns "TRUE", so assignment suppresses printing
```

The dimensions of the results dataframe are 30, 3 .

```
# structure of the results dataframe
str(result)
```

```
## 'data.frame':    30 obs. of  3 variables:
##  $ Movie_title: chr  "Crazy Rich Asians" "Crazy Rich Asians" "Crazy Rich Asians" "Crazy Rich Asians"
##  $ Friend_name: chr  "Alice" "Bob" "Carol" "Dave" ...
##  $ Rating     : int  4 1 3 2 1 2 1 2 5 3 ...
```

```
# summary of the results dataframe
summary(result)
```

```
##  Movie_title        Friend_name            Rating
##  Length:30          Length:30          Min.   :1.000
##  Class :character   Class :character   1st Qu.:1.000
##  Mode  :character   Mode  :character   Median :2.000
##                                        Mean   :2.517
##                                        3rd Qu.:4.000
##                                        Max.   :5.000
##                                        NA's   :1
```

List the results (there are only 30 rows):

```
result
```

```
##                    Movie_title Friend_name Rating
## 1            Crazy Rich Asians       Alice      4
## 2            Crazy Rich Asians         Bob      1
## 3            Crazy Rich Asians       Carol      3
## 4            Crazy Rich Asians        Dave      2
## 5            Crazy Rich Asians       Eddie      1
## 6   Disney's Christopher Robin       Alice      2
## 7   Disney's Christopher Robin         Bob      1
## 8   Disney's Christopher Robin       Carol      2
## 9   Disney's Christopher Robin        Dave      5
## 10  Disney's Christopher Robin       Eddie      3
## 11 Mamma Mia! Here We Go Again       Alice      3
## 12 Mamma Mia! Here We Go Again         Bob      2
## 13 Mamma Mia! Here We Go Again       Carol      2
## 14 Mamma Mia! Here We Go Again        Dave      1
## 15 Mamma Mia! Here We Go Again       Eddie     NA
## 16                  Ocean's 8       Alice      4
## 17                  Ocean's 8         Bob      5
## 18                  Ocean's 8       Carol      5
## 19                  Ocean's 8        Dave      2
## 20                  Ocean's 8       Eddie      4
## 21               Peter Rabbit       Alice      2
## 22               Peter Rabbit         Bob      1
## 23               Peter Rabbit       Carol      1
## 24               Peter Rabbit        Dave      2
## 25               Peter Rabbit       Eddie      2
## 26    Solo: A Star Wars Story       Alice      1
## 27    Solo: A Star Wars Story         Bob      5
## 28    Solo: A Star Wars Story       Carol      1
## 29    Solo: A Star Wars Story        Dave      2
## 30    Solo: A Star Wars Story       Eddie      4
```

Describe the results:

```
describe(result$Rating)
```

```
##    vars  n mean  sd median trimmed  mad min max range skew kurtosis   se
## X1    1 29 2.52 1.4      2    2.44 1.48   1   5     4 0.56     -1.1 0.26
```

(Note that there is one “NA” value, which we will have to exclude later.)

Let’s look at the results, grouped by Movie :

```
describeBy(result$Rating,group = result$Movie_title )
```

```
##
##  Descriptive statistics by group
## group: Crazy Rich Asians
##    vars n mean  sd median trimmed  mad min max range skew kurtosis   se
## X1    1 5  2.2 1.3      2     2.2 1.48   1   4     3 0.26    -1.96 0.58
## --------------------------------------------------------
## group: Disney's Christopher Robin
##    vars n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 5  2.6 1.52      2     2.6 1.48   1   5     4 0.54    -1.49 0.68
## --------------------------------------------------------
## group: Mamma Mia! Here We Go Again
##    vars n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 4    2 0.82      2       2 0.74   1   3     2    0    -1.88 0.41
## --------------------------------------------------------
## group: Ocean's 8
##    vars n mean   sd median trimmed  mad min max range  skew kurtosis   se
## X1    1 5    4 1.22      4       4 1.48   2   5     3 -0.65     -1.4 0.55
## --------------------------------------------------------
## group: Peter Rabbit
##    vars n mean   sd median trimmed mad min max range  skew kurtosis   se
## X1    1 5  1.6 0.55      2     1.6    0   1   2     1 -0.29    -2.25 0.24
## --------------------------------------------------------
## group: Solo: A Star Wars Story
##    vars n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 5  2.6 1.82      2     2.6 1.48   1   5     4 0.27    -2.08 0.81
```

We need to drop the item with the NA rating in order to obtain non-NA summary results.

Subsetting using !is.na(result$Rating) :

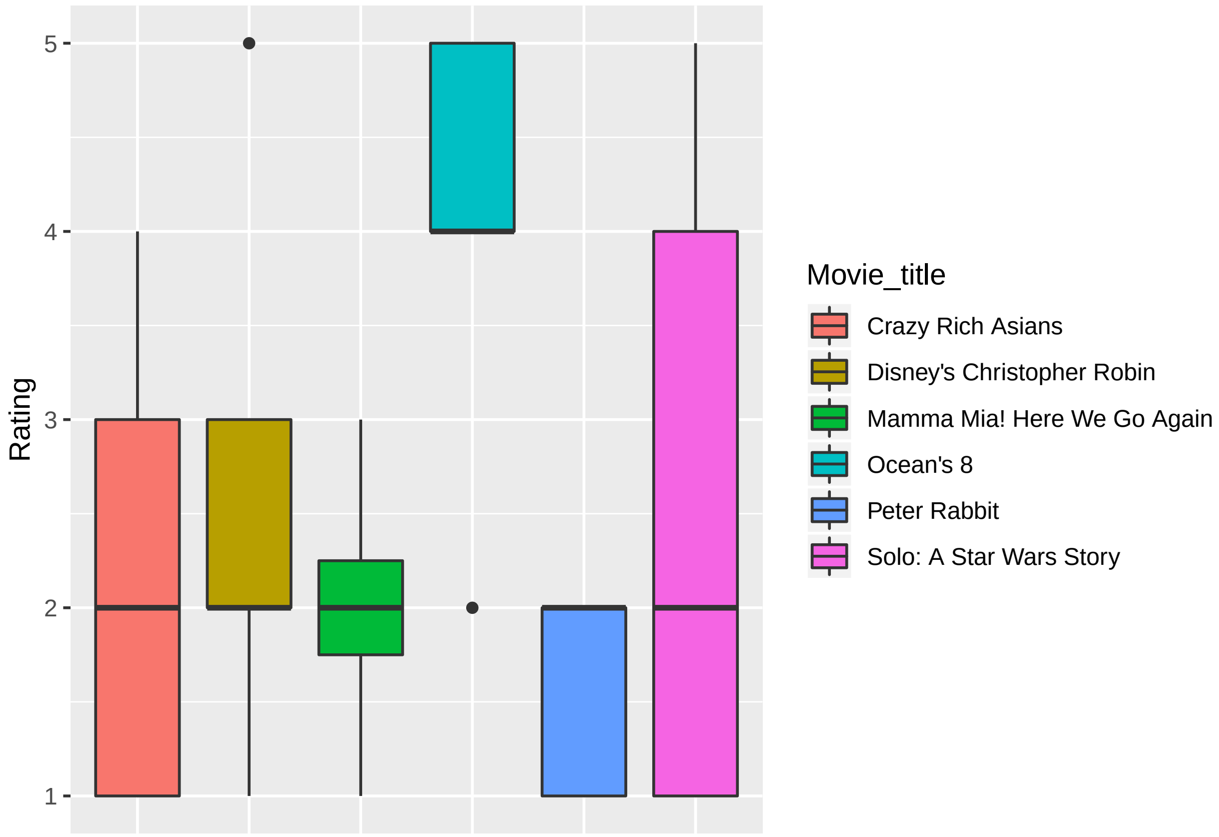
```
result[!is.na(result$Rating),] %>%
  group_by(Movie_title) %>%
  summarize(count=n(),
            min=min(Rating),
            mean=mean(Rating),
            median=median(Rating),
            max=max(Rating),
            sd=sd(Rating),
            IQR=IQR(Rating)
            )
```

```
## # A tibble: 6 x 8
##   Movie_title                 count   min  mean median   max    sd   IQR
##   <chr>                       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Crazy Rich Asians               5     1   2.2      2     4 1.30    2
## 2 Disney's Christopher Robin      5     1   2.6      2     5 1.52    1
## 3 Mamma Mia! Here We Go Again     4     1   2        2     3 0.816   0.5
## 4 Ocean's 8                       5     2   4        4     5 1.22    1
## 5 Peter Rabbit                    5     1   1.6      2     2 0.548   1
## 6 Solo: A Star Wars Story         5     1   2.6      2     5 1.82    3
```

Now, let's make a boxplot by Movie :

```
ggplot(result, aes(x=Movie_title, y=Rating, fill=Movie_title)) +
  geom_boxplot() +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```



We can see that Ocean's 8 was quite popular, with mean and median ratings of 4:

```
result[result$Movie_title=="Ocean's 8",]
```

```
##    Movie_title Friend_name Rating
## 16   Ocean's 8       Alice      4
## 17   Ocean's 8         Bob      5
## 18   Ocean's 8       Carol      5
## 19   Ocean's 8        Dave      2
## 20   Ocean's 8       Eddie      4
```

while Peter Rabbit was at the opposite end of the spectrum, receiving the lowest ratings:

```
result[result$Movie_title=="Peter Rabbit",]
```

```
##     Movie_title Friend_name Rating
## 21 Peter Rabbit       Alice      2
## 22 Peter Rabbit         Bob      1
## 23 Peter Rabbit       Carol      1
## 24 Peter Rabbit        Dave      2
## 25 Peter Rabbit       Eddie      2
```

Now, Let's look at how each friend tended to rate the films:

```
describeBy(result$Rating,group = result$Friend_name )
```

```
##
##  Descriptive statistics by group
## group: Alice
##    vars n mean   sd median trimmed  mad min max range  skew kurtosis   se
## X1    1 6 2.67 1.21    2.5    2.67 1.48   1   4     3 -0.04    -1.88 0.49
## --------------------------------------------------------
## group: Bob
##    vars n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 6  2.5 1.97    1.5     2.5 0.74   1   5     4 0.45    -1.98 0.81
## --------------------------------------------------------
## group: Carol
##    vars n mean   sd median trimmed  mad min max range skew kurtosis   se
## X1    1 6 2.33 1.51      2    2.33 1.48   1   5     4 0.71    -1.15 0.61
## --------------------------------------------------------
## group: Dave
##    vars n mean   sd median trimmed mad min max range skew kurtosis   se
## X1    1 6 2.33 1.37      2    2.33   0   1   5     4 1.07    -0.43 0.56
## --------------------------------------------------------
## group: Eddie
##    vars n mean  sd median trimmed  mad min max range  skew kurtosis   se
## X1    1 5  2.8 1.3      3     2.8 1.48   1   4     3 -0.26    -1.96 0.58
```

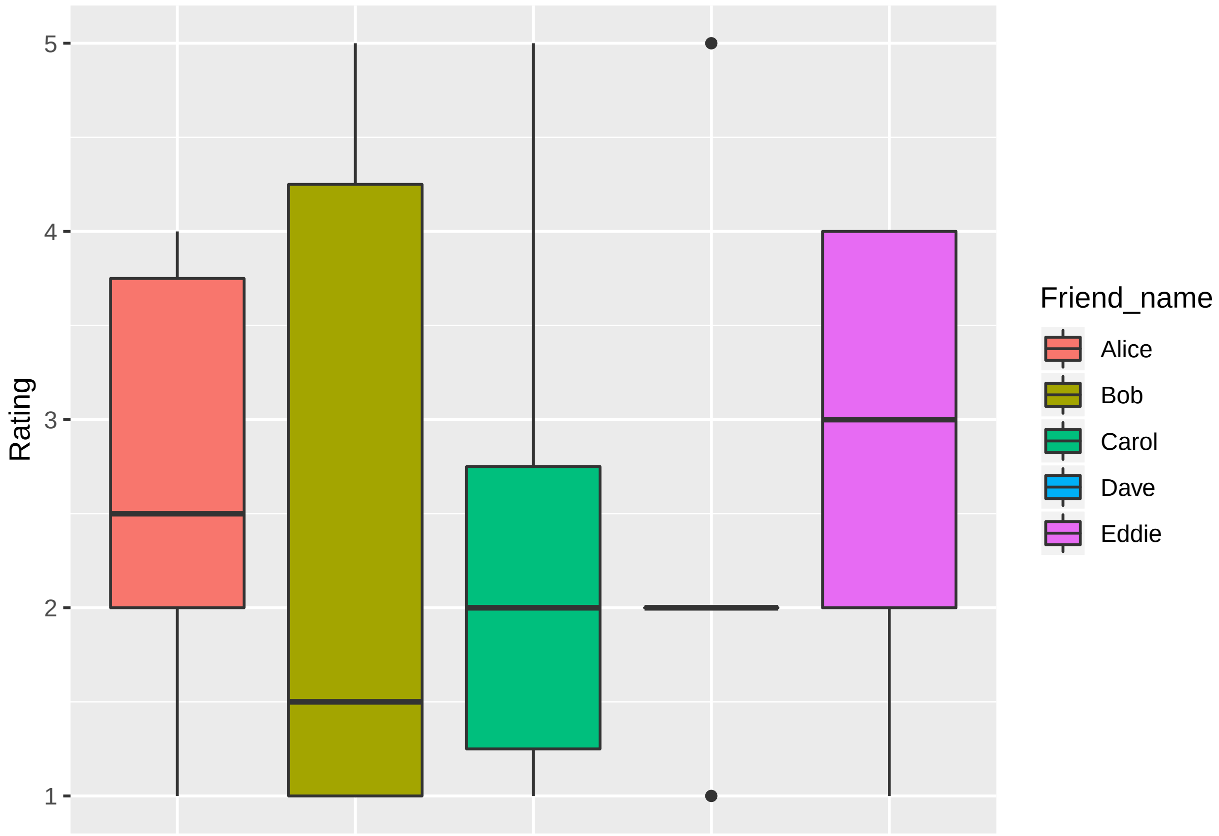
Again, we have to exclude the item with the NA:

```
result[!is.na(result$Rating),] %>%
  group_by(Friend_name) %>%
  summarize(count=n(),
            min=min(Rating),
            mean=mean(Rating),
            median=median(Rating),
            max=max(Rating),
            sd=sd(Rating),
            IQR=IQR(Rating)
            )
```

```
## # A tibble: 5 x 8
##   Friend_name count   min  mean median   max    sd   IQR
##   <chr>       <int> <dbl> <dbl>  <dbl> <dbl> <dbl> <dbl>
## 1 Alice           6     1  2.67    2.5     4  1.21  1.75
## 2 Bob             6     1  2.5     1.5     5  1.97  3.25
## 3 Carol           6     1  2.33    2       5  1.51  1.5
## 4 Dave            6     1  2.33    2       5  1.37  0
## 5 Eddie           5     1  2.8     3       4  1.30  2
```

```
ggplot(result, aes(x=Friend_name, y=Rating, fill=Friend_name)) +
  geom_boxplot() +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank())
```

```
## Warning: Removed 1 rows containing non-finite values (stat_boxplot).
```



We observe that Bob either likes a film or hates it – with Bob, there is no middle ground.

Bob gave the widest disperion among his ratings, using mostly "1"s and "5"s, which explains his large IQR and standard deviation.

His Median is the lowest, as half his ratings were "1"s:

```
result[result$Friend_name=="Bob",]
```

```
##                     Movie_title Friend_name Rating
## 2           Crazy Rich Asians         Bob      1
## 7   Disney's Christopher Robin         Bob      1
## 12 Mamma Mia! Here We Go Again         Bob      2
## 17                   Ocean's 8         Bob      5
## 22                Peter Rabbit         Bob      1
## 27      Solo: A Star Wars Story         Bob      5
```

Because Dave gave so many ratings of "2", his IQR = 0, thus his box is flat, with outliers at "1" and "5" :

```
result[result$Friend_name=="Dave",]
```

```
##                     Movie_title Friend_name Rating
## 4           Crazy Rich Asians        Dave      2
## 9   Disney's Christopher Robin        Dave      5
## 14 Mamma Mia! Here We Go Again        Dave      1
## 19                   Ocean's 8        Dave      2
## 24                Peter Rabbit        Dave      2
## 29      Solo: A Star Wars Story        Dave      2
```

Conclusion: With a small data set (6 movies and 5 reviewers) the aggregated figures display interesting results across both movie and reviewer.

It would be interesting to see the results across a larger sample, for example using the data assembled by "Rotton Tomatoes" which tabulates published movie reviews and scores films on a scale of 0%-100% based upon the percentage of reviews which are favorable vs. unfavorable.

Furthermore, it would be interesting to compare/contrast such "professional" assessments with opinions from individuals, such as those assembled by firms like Amazon.