Comprehension Check: Recommendation Systems

The following exercises all work with the `movielens` data, which can be loaded using the following code:

```
library(dslabs)
data("movielens")
```

Q1

1/1 point (graded)

Compute the number of ratings for each movie and then plot it against the year the movie came out. Use the square root transformation on the counts.

What year has the highest median number of ratings?

1995 ✔ **Answer:** 1995

1995

Explanation

The following code will generate the plot:

```
movielens %>% group_by(movieId) %>%
        summarize(n = n(), year = as.character(first(year))) %>%
        qplot(year, n, data = ., geom = "boxplot") +
        coord_trans(y = "sqrt") +
        theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

From the plot, you can see that the year with the highest median number of ratings is 1995.

Submit You have used 1 of 10 attempts

Answers are displayed within the problem

Q2

1/1 point (graded)
We see that, on average, movies that came out after 1993 get more ratings. We also see that with newer movies, starting in 1993, the number of ratings decreases with year: the more recent a movie is, the less time users have had to rate it.

Among movies that came out in 1993 or later, select the top 25 movies with the highest average number of ratings per year (n/year), and caculate the average rating of each of them. To calculate number of ratings per year, use 2018 as the end year.

What is the average rating for the movie The Shawshank Redemption?

4.49 ✔ **Answer:** 4.49

4.49

What is the average number of ratings per year for the movie Forrest Gump?

14.2 ✔ **Answer:** 14.2

14.2

Explanation
The top 25 movies with the most ratings per year, along with their average ratings, can be found using the following code:

```
movielens %>%
        filter(year >= 1993) %>%
        group_by(movieId) %>%
        summarize(n = n(), years = 2018 - first(year),
                                title = title[1],
                                rating = mean(rating)) %>%
        mutate(rate = n/years) %>%
        top_n(25, rate) %>%
        arrange(desc(rate))
```

Submit You have used 1 of 10 attempts

Answers are displayed within the problem

Q3

1/1 point (graded)
From the table constructed in Q2, we can see that the most frequently rated movies tend to have above average ratings. This is not surprising: more people watch popular movies. To confirm this, stratify the post-1993 movies by ratings per year and compute their average ratings. To calculate number of ratings

per year, use 2018 as the end year. Make a plot of average rating versus ratings per year and show an estimate of the trend.

What type of trend do you observe?

- ○ There is no relationship between how often a movie is rated and its average rating.
- ○ Movies with very few and very many ratings have the highest average ratings.
- ● The more often a movie is rated, the higher its average rating.
- ○ The more often a movie is rated, the lower its average rating.

✔

Explanation
The plot can be generated using the following code:

```
movielens %>%
        filter(year >= 1993) %>%
        group_by(movieId) %>%
        summarize(n = n(), years = 2018 - first(year),
                                title = title[1],
                                rating = mean(rating)) %>%
        mutate(rate = n/years) %>%
        ggplot(aes(rate, rating)) +
        geom_point() +
        geom_smooth()
```

We see that the trend is that the more often a movie is rated, the higher its average rating.

Submit You have used 1 of 2 attempts

ℹ Answers are displayed within the problem

Q4

1/1 point (graded)
Suppose you are doing a predictive analysis in which you need to fill in the missing ratings with some value.

Given your observations in the exercise in Q3, which of the following strategies would be most appropriate?

- ○ Fill in the missing values with the average rating across all movies.
- ○ Fill in the missing values with 0.
- ◉ Fill in the missing values with a lower value than the average rating across all movies.
- ○ Fill in the value with a higher value than the average rating across all movies.
- ○ None of the above.

✔

Explanation
Because a lack of ratings is associated with lower ratings, it would be most appropriate to fill in the missing value with a lower value than the average. You should try out different values to fill in the missing value and evaluate prediction in a test set.

Submit You have used 1 of 2 attempts

ℹ Answers are displayed within the problem

Q5

1/1 point (graded)
The `movielens` dataset also includes a time stamp. This variable represents the time and data in which the rating was provided. The units are seconds since January 1, 1970. Create a new column `date` with the date.

Which code correctly creates this new column?

- ○ `movielens <- mutate(movielens, date = as.date(timestamp))`
- ◉ `movielens <- mutate(movielens, date = as_datetime(timestamp))`
- ○ `movielens <- mutate(movielens, date = as.data(timestamp))`
- ○ `movielens <- mutate(movielens, date = timestamp)`

✔

Explanation
The `as_datetime` function in the lubridate package is particularly useful here.

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

Q6

1/1 point (graded)
Compute the average rating for each week and plot this average against day. Hint: use the `round_date` function before you `group_by` .

What type of trend do you observe?

- There is strong evidence of a time effect on average rating.
- **There is some evidence of a time effect on average rating.**
- There is no evidence of a time effect on average rating.

✔

Explanation
The following code can be used to generate the plot:

```
movielens %>% mutate(date = round_date(date, unit = "week")) %>%
        group_by(date) %>%
        summarize(rating = mean(rating)) %>%
        ggplot(aes(date, rating)) +
        geom_point() +
        geom_smooth()
```

We can see that there is some evidence of a time effect in the plot, but there is not a strong effect of time.

Submit You have used 1 of 1 attempt

Answers are displayed within the problem

Q7

1/1 point (graded)

Consider again the plot you generated in Q6.

If we define $d_{u,i}$ as the day for user's u rating of movie i, which of the following models is most appropriate?

- $Y_{u,i} = \mu + b_i + b_u + d_{u,i} + \varepsilon_{u,i}$
- $Y_{u,i} = \mu + b_i + b_u + d_{u,i}\beta + \varepsilon_{u,i}$
- $Y_{u,i} = \mu + b_i + b_u + d_{u,i}\beta_i + \varepsilon_{u,i}$
- $Y_{u,i} = \mu + b_i + b_u + f(d_{u,i}) + \varepsilon_{u,i}$, with f a smooth function of $d_{u,i}$

✔

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

Q8

1/1 point (graded)

The `movielens` data also has a `genres` column. This column includes every genre that applies to the movie. Some movies fall under several genres. Define a category as whatever combination appears in this column. Keep only categories with more than 1,000 ratings. Then compute the average and standard error for each category. Plot these as error bar plots.

Which genre has the lowest average rating?

Enter the name of the genre exactly as reported in the plot, including capitalization and punctuation.

Comedy ✔ **Answer:** Comedy

Explanation

The following code will generate the plot:

```
movielens %>% group_by(genres) %>%
        summarize(n = n(), avg = mean(rating), se = sd(rating)/sqrt(n())) %>%
        filter(n >= 1000) %>%
        mutate(genres = reorder(genres, avg)) %>%
        ggplot(aes(x = genres, y = avg, ymin = avg - 2*se, ymax = avg + 2*se)) +
        geom_point() +
        geom_errorbar() +
        theme(axis.text.x = element_text(angle = 90, hjust = 1))
```

Submit You have used 1 of 10 attempts

Answers are displayed within the problem

Q9

1/1 point (graded)

The plot you generated in Q8 shows strong evidence of a genre effect. Consider this plot as you answer the following question.

If we define $g_{u,i}$ as the genre for user u's rating of movie i, which of the following models is most appropriate?

- $Y_{u,i} = \mu + b_i + b_u + g_{u,i} + \varepsilon_{u,i}$
- $Y_{u,i} = \mu + b_i + b_u + g_{u,i}\beta + \varepsilon_{u,i}$
- $Y_{u,i} = \mu + b_i + b_u + \sum_{k=1}^{K} x_{u,i}\beta_k + \varepsilon_{u,i}$, with $x_{u,i}^k = 1$ if $g_{u,i}$ is genre k
- $Y_{u,i} = \mu + b_i + b_u + f(g_{u,i}) + \varepsilon_{u,i}$, with f a smooth function of $g_{u,i}$

Submit You have used 1 of 2 attempts

Answers are displayed within the problem

Ask your questions or make your comments about Recommendation Systems here! **Remember, one of the best ways to reinforce your own learning is by explaining something to someone else, so we encourage you to answer each other's questions (without giving away the answers, of course).**

Some reminders:

- Search the discussion board before posting to see if someone else has asked the same thing before asking a new question.
- Please be specific in the title and body of your post regarding which question you're asking about to facilitate answering your question.
- Posting snippets of code is okay, but posting full code solutions is not.
- If you do post snippets of code, please format it as code for readability. If you're not sure how to do this, there are instructions in a pinned post in the "general" discussion forum.

Discussion: Recommendation Systems

Show Discussion

Topic: Section 6: Model fitting and recommendation systems / 6.2: Recommendation Systems