Webscrape

Part 2

Dr. Maria Tackett

09.26.19

Click for PDF of slides

Announcements

- Writing Exercise #2 initial draft - **due TODAY at 11:59p**
  - Peer Review available tomorrow and due Sunday 9/29 at 11:59p
- HW 02 - **due TODAY at 11:59p**
- Team Feedback #1 **due TODAY at 11:59p**
  - Please provide honest and constructive feedback. This team feedback will be graded for completion.

Check in: Regrade Requests

- All regrade requests should be submitted through Gradescope. See updated course policy
- Only submit a regrade request if you still have concerns about your grade after you have attended office hours and asked a member of the teaching team to explain the feedback you received. This will ultimately help with your understanding of the course material and help the teaching team get an idea about points to clarify.
- **When you submit a regrade request, please indicate who you've talked with prior to submitting the request.**
- Professor Tackett is the only person who can update grades, so do not ask your TAs to regrade your assignment.

Check in: Lab 04

- Will get Lab 04 assignment from RStudio Cloud project.
- Fill out form with the name of the RStudio Cloud project for grading.

Web scraping

Clean up / enhance

May or may not be a lot of work depending on how messy the data are

- See if you like what you got:

```
glimpse(imdb_top_250)
```

```
## Observations: 250
## Variables: 3
## $ title <chr> "The Shawshank Redemption", "The Godfather", "The Godfathe…
## $ year  <dbl> 1994, 1972, 1974, 2008, 1957, 1993, 2003, 1994, 1966, 1999…
## $ score <dbl> 9.2, 9.1, 9.0, 9.0, 8.9, 8.9, 8.9, 8.9, 8.8, 8.8, 8.8, 8.8…
```

- Add a variable for rank

```
imdb_top_250 <- imdb_top_250 %>%
  mutate(
    rank = 1:nrow(imdb_top_250)
  )
```

title	year	score	rank
The Shawshank Redemption	1994	9.2	1
The Godfather	1972	9.1	2
The Godfather: Part II	1974	9	3
The Dark Knight	2008	9	4
12 Angry Men	1957	8.9	5
Schindler's List	1993	8.9	6
The Lord of the Rings: The Return of the King	2003	8.9	7
Pulp Fiction	1994	8.9	8
The Good, the Bad and the Ugly	1966	8.8	9
Fight Club	1999	8.8	10
...	...	...	...

Analyze

How would you go about answering this question: Which 1995 movies made the list?

```
imdb_top_250 %>%
  filter(year == 1995)
```

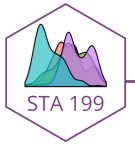
```
## # A tibble: 8 x 4
##   title                year score  rank
##   <chr>               <dbl> <dbl> <int>
## 1 Se7en                1995   8.6    20
## 2 The Usual Suspects   1995   8.5    32
## 3 Braveheart           1995   8.3    76
## 4 Toy Story            1995   8.3    81
## 5 Heat                 1995   8.2   121
## 6 Casino               1995   8.2   139
## 7 Before Sunrise       1995   8.1   194
## 8 La Haine             1995   8     228
```

Analyze

How would you go about answering this question: Which years have the most movies on the list?

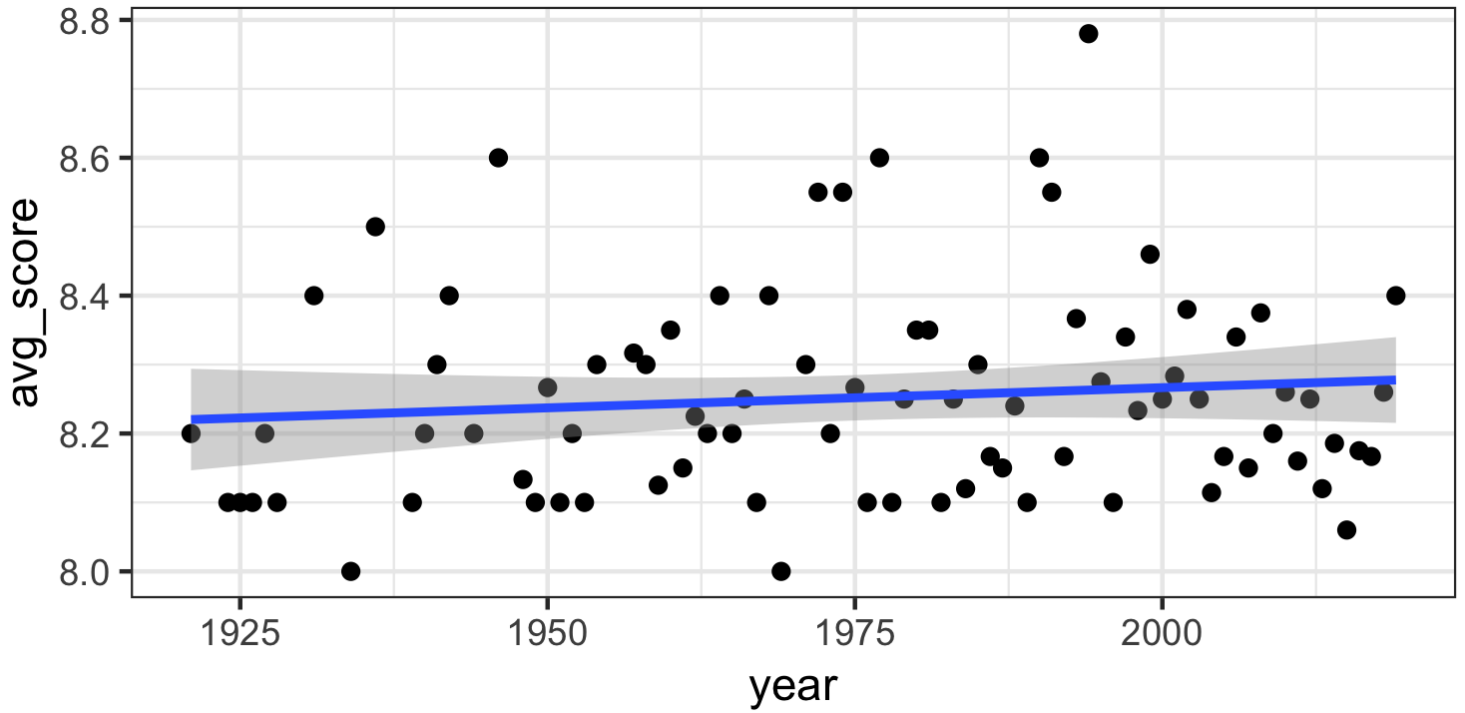
```
imdb_top_250 %>%
  group_by(year) %>%
  summarise(total = n()) %>%
  arrange(desc(total)) %>%
  head(5)
```

```
## # A tibble: 5 x 2
##    year total
##   <dbl> <int>
## 1  1995     8
## 2  2004     7
## 3  2014     7
## 4  1957     6
## 5  1998     6
```



Visualize

How would you go about creating this visualization: Visualize the average yearly score for movies that made it on the top 250 list over time.



Top Rated

- Which year has the highest average score for movies that made the Top 250?

- What is one reason we should write code to answer this question rather than look through the data?
- What is one reason we only want to print the year with the highest average rather than entire table?

Potential challenges

- Unreliable formatting at the source
- Data broken into many pages
- ...

Compare the display of information at raleigh.craigslist.org/search/apa to the list on the IMDB top 250 list.

What challenges can you foresee in scraping a list of the available apartments?

Application Exercise

Popular TV Shows

RStudio Cloud → Web scraping

1. Scrape the list of most popular TV shows on IMDB: http://www.imdb.com/chart/tvmeter

2. Examine each of the first three (or however many you can get through) tv show subpage to also obtain genre and runtime.

3. Time permitting, also try to get the following:

    - How many episodes so far
    - Certificate
    - First five plot keywords
    - Country
    - Language

Add this information to the data frame you created in step 1.