Section 001
CDS 101 *Introduction to Computational and Data Sciences*

Class 19: Introduction to Web Scraping II / Principles of Data Collection

April 3, 2018

General

Annoucements

- Reading schedule for next class (and beyond) to be posted soon.
- Don't forget to participate in the Question/Answer discussion for each Reading!
  - Answer post count reset for second half of course
  - Review the course syllabus for credit requirements
- Homework 3 on Web Scraping will be posted soon.

Review of Web Scraping Activity

Web scraping activity

On Thursday, March 29th, we worked on the following web-scraping activity:

Web scraping activity

On Thursday, March 29th, we worked on the following web-scraping activity:

Navigate to http://www.imdb.com/chart/tvmeter and scrape the list of the most popular TV shows. The result should be a tibble with 100 rows and 4 columns: rank, tv show name, year, and rating. The variables should be in this order.

Web scraping activity

On Thursday, March 29th, we worked on the following web-scraping activity:

> Navigate to http://www.imdb.com/chart/tvmeter and scrape the list of the most popular TV shows. The result should be a tibble with 100 rows and 4 columns: rank, tv show name, year, and rating. The variables should be in this order.

- The code blocks from the Top 250 Movies example worked for some, but not all of this exercise.

Web scraping activity

On Thursday, March 29th, we worked on the following web-scraping activity:

> Navigate to http://www.imdb.com/chart/tvmeter and scrape the list of the most popular TV shows. The result should be a tibble with 100 rows and 4 columns: rank, tv show name, year, and rating. The variables should be in this order.

- The code blocks from the Top 250 Movies example worked for some, but not all of this exercise.
- Primary objective was to use the SelectorGadget tool to modify the HTML nodes you needed to grab

Web scraping activity

On Thursday, March 29th, we worked on the following web-scraping activity:

> Navigate to http://www.imdb.com/chart/tvmeter and scrape the list of the most popular TV shows. The result should be a tibble with 100 rows and 4 columns: rank, tv show name, year, and rating. The variables should be in this order.

- The code blocks from the Top 250 Movies example worked for some, but not all of this exercise.
- Primary objective was to use the SelectorGadget tool to modify the HTML nodes you needed to grab
- How do you take the example code and modify it to work for this activity?

Scraping code: IMDB Top 250 Movies

```
page <- read_html("http://www.imdb.com/chart/top")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

Scraping code: IMDB Top 250 Movies

```
page <- read_html("http://www.imdb.com/chart/top")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

Scraping code: IMDB Top 250 Movies

```
page <- read_html("http://www.imdb.com/chart/tvmeter")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

Scraping code: IMDB Top 250 Movies

```
page <- read_html("http://www.imdb.com/chart/tvmeter")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

TV show titles

Let's check to see if it's actually necessary to change the `titles` code:

```
titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()
```

TV show titles

Let's check to see if it's actually necessary to change the `titles` code:

```
titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()
```

The length of the `titles` vector is:

```
length(titles)
```

```
## [1] 100
```

TV show titles

Let's check to see if it's actually necessary to change the `titles` code:

```
titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()
```

The length of the `titles` vector is:

```
length(titles)
```

```
## [1] 100
```

And the first 10 elements in `titles` are:

```
##  [1] "The Walking Dead"   "Roseanne"           "Grey's Anatomy"
##  [4] "Santa Clarita Diet" "Game of Thrones"    "The Terror"
##  [7] "Roseanne"           "Krypton"            "Homeland"
## [10] "Westworld"
```

TV show titles

Let's check to see if it's actually necessary to change the `titles` code:

```
titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()
```

The length of the `titles` vector is:

```
length(titles)
```

```
## [1] 100
```

And the first 10 elements in `titles` are:

```
##  [1] "The Walking Dead"   "Roseanne"           "Grey's Anatomy"
##  [4] "Santa Clarita Diet" "Game of Thrones"    "The Terror"
##  [7] "Roseanne"           "Krypton"            "Homeland"
## [10] "Westworld"
```

So far, so good!

TV show years

```
page <- read_html("http://www.imdb.com/chart/tvmeter")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

TV show years

Next, let's check if the `years` code works for us:

```
years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

TV show years

Next, let's check if the `years` code works for us:

```
years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

And the first few elements in `years` are:

```
## [1] "2010"   "2018"   "\n\n69" "2005"   "\n\n1"  "2017"   "\n\n20" "20
```

TV show years

Next, let's check if the `years` code works for us:

```
years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

And the first few elements in `years` are:

```
## [1] "2010"   "2018"   "\n\n69" "2005"   "\n\n1"  "2017"   "\n\n20" "20
```

Not so lucky this time.

TV show years

Next, let's check if the `years` code works for us:

```
years <- page %>%
  html_nodes(".secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

And the first few elements in `years` are:

```
## [1] "2010"   "2018"   "\n\n69" "2005"   "\n\n1"  "2017"   "\n\n20" "20
```

Not so lucky this time. Let's see how we can fix this.

SelectorGadget `years` demo

Follow along in Google Chrome

TV show years (revised)

Here's our revised `years` code based on our SelectorGadget work:

```
years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

TV show years (revised)

Here's our revised `years` code based on our SelectorGadget work:

```
years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

The first 10 elements in our revised `years` are:

```
##  [1] "2010" "2018" "2005" "2017" "2011" "2018" "1988" "2018" "2011" "2
```

TV show years (revised)

Here's our revised `years` code based on our SelectorGadget work:

```
years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

The first 10 elements in our revised `years` are:

```
##  [1] "2010" "2018" "2005" "2017" "2011" "2018" "1988" "2018" "2011" "2
```

Much better!

TV show years (revised)

Here's our revised `years` code based on our SelectorGadget work:

```
years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "")     # remove )
```

The first 10 elements in our revised `years` are:

```
##  [1] "2010" "2018" "2005" "2017" "2011" "2018" "1988" "2018" "2011" "2
```

Much better!

Note: We should append `%>% as.numeric()` to our `years` definition so that the years are interpreted by R as integers, not text.

TV show user scores

```
page <- read_html("http://www.imdb.com/chart/tvmeter")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

This seems promising...

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

This seems promising... however,

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

This seems promising... however, if we check the number of elements in `scores`:

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

This seems promising... however, if we check the number of elements in `scores`:

```
length(scores)
```

```
## [1] 99
```

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

This seems promising... however, if we check the number of elements in `scores`:

```
length(scores)
```

```
## [1] 99
```

This should be `100`, not `99`.

TV show user scores

Will the `scores` code work?

```
scores <- page %>%
  html_nodes(".article strong") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

This seems promising... however, if we check the number of elements in `scores`:

```
length(scores)
```

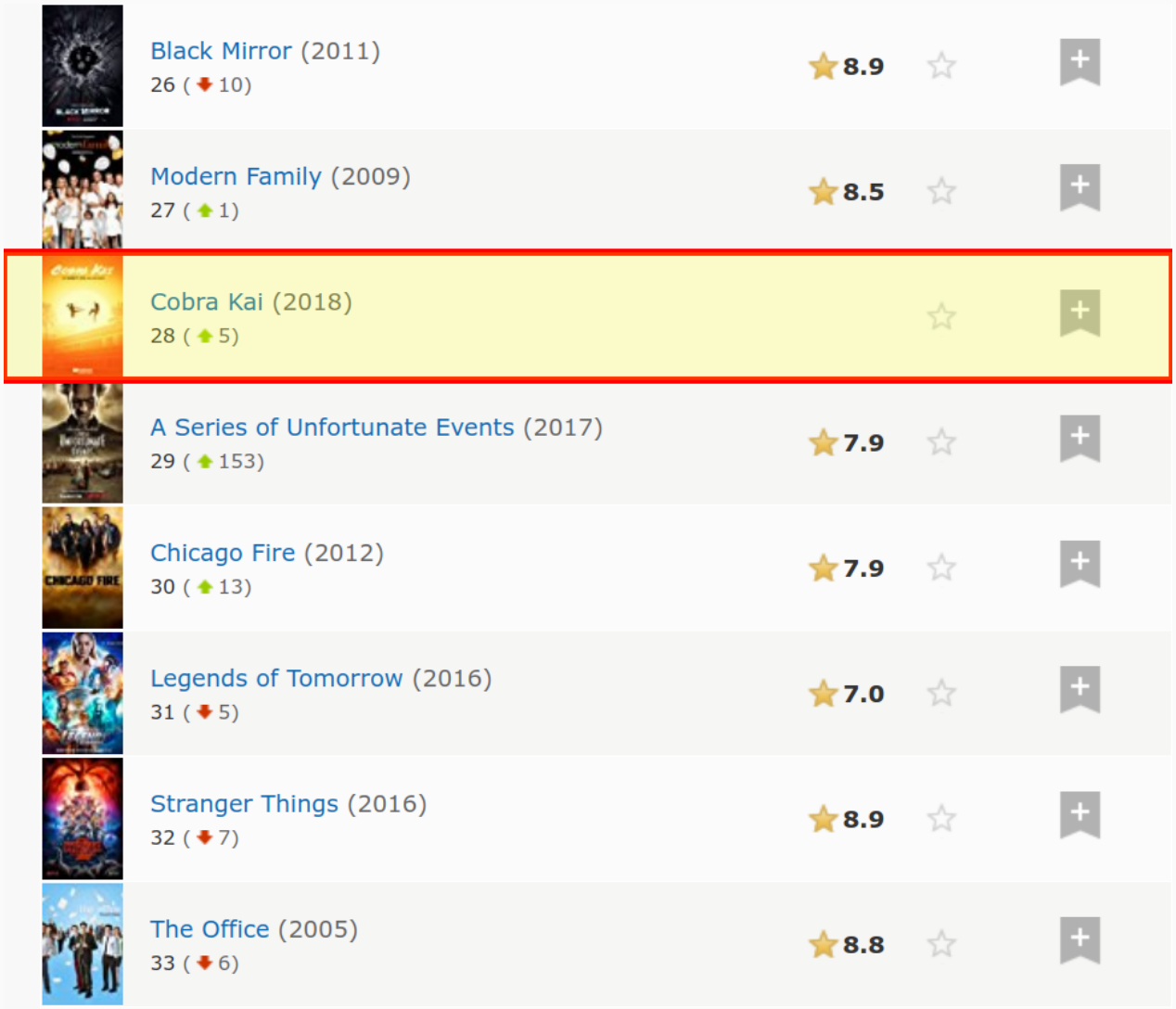
```
## [1] 99
```

This should be `100`, not `99`. What's going on?

SelectorGadget `scores` demo

Follow along in Google Chrome

Blank TV show scores



TV show user scores (revised)

Here's our revised `scores` code based on our SelectorGadget work that takes into account shows with a missing score:

```
scores <- page %>%
  html_nodes(".imdbRating") %>%
  html_text() %>%
  as.numeric()
```

TV show user scores (revised)

Here's our revised `scores` code based on our SelectorGadget work that takes into account shows with a missing score:

```
scores <- page %>%
  html_nodes(".imdbRating") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in our revised `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

TV show user scores (revised)

Here's our revised `scores` code based on our SelectorGadget work that takes into account shows with a missing score:

```
scores <- page %>%
  html_nodes(".imdbRating") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in our revised `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

That hasn't changed, and the number of elements in `scores` is:

```
length(scores)
```

```
## [1] 100
```

TV show user scores (revised)

Here's our revised `scores` code based on our SelectorGadget work that takes into account shows with a missing score:

```
scores <- page %>%
  html_nodes(".imdbRating") %>%
  html_text() %>%
  as.numeric()
```

The first 10 elements in our revised `scores` are:

```
##  [1] 8.4 7.3 7.6 7.7 9.5 8.8 7.0 7.3 8.4 8.9
```

That hasn't changed, and the number of elements in `scores` is:

```
length(scores)
```

```
## [1] 100
```

Much better!

Creating the data tibble

```
page <- read_html("http://www.imdb.com/chart/tvmeter")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_nodes(".imdbRating") %>%
  html_text() %>%
  as.numeric()

imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

TV show rank

The shows on the page are already sorted by rank.

TV show rank

The shows on the page are already sorted by rank.

It's easier to count from 1 to 100 and manually generate the ranking column rather than try and extract it from the page:

TV show rank

The shows on the page are already sorted by rank.

It's easier to count from 1 to 100 and manually generate the ranking column rather than try and extract it from the page:

```
ranks <- seq(from = 1, to = 100, by = 1)
```

TV show rank

The shows on the page are already sorted by rank.

It's easier to count from 1 to 100 and manually generate the ranking column rather than try and extract it from the page:

```
ranks <- seq(from = 1, to = 100, by = 1)
```

Let's do our sanity check:

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

TV show rank

The shows on the page are already sorted by rank.

It's easier to count from 1 to 100 and manually generate the ranking column rather than try and extract it from the page:

```
ranks <- seq(from = 1, to = 100, by = 1)
```

Let's do our sanity check:

```
##  [1]  1  2  3  4  5  6  7  8  9 10
```

No problem with how `seq()` works!

TV show tibble

We have everything we need, so let's take the original code for making the tibble:

TV show tibble

We have everything we need, so let's take the original code for making the tibble:

```
imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

TV show tibble

We have everything we need, so let's take the original code for making the tibble:

```
imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

and change the variable name to `imdb_tv_top_100`, put the columns in the correct order, and add in the ranks column:

TV show tibble

We have everything we need, so let's take the original code for making the tibble:

```
imdb_top_250 <- data_frame(
  title = titles, year = years, score = scores)
```

and change the variable name to `imdb_tv_top_100`, put the columns in the correct order, and add in the ranks column:

```
imdb_top_tv <- data_frame(
  rank = ranks, title = titles, year = years, score = scores)
```

Create a CSV file

Finally, let's save our work so that we don't need to always reconnect to the website:

Create a CSV file

Finally, let's save our work so that we don't need to always reconnect to the website:

```
imdb_top_tv %>%
  write_csv("2018-04-03T1238EST_imdb_tv.csv")
```

Create a CSV file

Finally, let's save our work so that we don't need to always reconnect to the website:

```
imdb_top_tv %>%
  write_csv("2018-04-03T1238EST_imdb_tv.csv")
```

Notice that the date and time that you scraped the data is part of the filename.

Create a CSV file

Finally, let's save our work so that we don't need to always reconnect to the website:

```
imdb_top_tv %>%
  write_csv("2018-04-03T1238EST_imdb_tv.csv")
```

Notice that the date and time that you scraped the data is part of the filename.

The list on this webpage changes frequently, so you want to document when you scraped!

Complete scraping code

```
page <- read_html("http://www.imdb.com/chart/tvmeter")

titles <- page %>%
  html_nodes(".titleColumn a") %>%
  html_text()

years <- page %>%
  html_nodes("a + .secondaryInfo") %>%
  html_text() %>%
  str_replace("\\(", "") %>% # remove (
  str_replace("\\)", "") %>% # remove )
  as.numeric()

scores <- page %>%
  html_nodes(".article strong") %>%
  html_nodes(".imdbRating") %>%
  html_text() %>%
  as.numeric()

ranks <- seq(from = 1, to = 100, by = 1)

imdb_top_tv <- data_frame(
  rank = ranks, title = titles, year = years, score = scores)
```

IMDB TV Table

rank	title	year	score
1	The Walking Dead	2010	8.4
2	Roseanne	2018	7.2
3	Grey's Anatomy	2005	7.6
4	Santa Clarita Diet	2017	7.7
5	Game of Thrones	2011	9.5
6	The Terror	2018	8.8
7	Roseanne	1988	7
8	Krypton	2018	7.3
9	Homeland	2011	8.4
10	Westworld	2016	8.9
...	...	...	...

Overview of data collection principles

Populations and samples

Research question: Can people become better, more efficient runners on their own, merely by running?


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Source: http://well.blogs.nytimes.com/2012/08/29/finding-your-ideal-running-form

Populations and samples


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Research question: Can people become better, more efficient runners on their own, merely by running?

Populations and samples


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Research question: Can people become better, more efficient runners on their own, merely by running?

Question: What is the population of interest?

Populations and samples


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Research question: Can people become better, more efficient runners on their own, merely by running?

Question: What is the population of interest?

Answer: All people

Populations and samples


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Research question: Can people become better, more efficient runners on their own, merely by running?

Question: What is the population of interest?

Answer: All people

Study Sample: Group of adult women who recently joined a running group

Populations and samples


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Research question: Can people become better, more efficient runners on their own, merely by running?

Question: What is the population of interest?

Answer: All people

Study Sample: Group of adult women who recently joined a running group

Question: Population to which results can be generalized?

Populations and samples


PHYS ED | AUGUST 29, 2012, 12:01 AM | 21 Comments

Finding Your Ideal Running Form

By GRETCHEN REYNOLDS

David De Lossy/Getty Images

Research question: Can people become better, more efficient runners on their own, merely by running?

Question: What is the population of interest?

Answer: All people

Study Sample: Group of adult women who recently joined a running group

Question: Population to which results can be generalized?

Answer: Adult women, if the data are randomly sampled

Anecdotal evidence and early smoking

- Anti-smoking research started in the 1930s and 1940s when cigarette smoking became increasingly popular. While some smokers seemed to be sensitive to cigarette smoke, others were completely unaffected.

Source: Brandt, *The Cigarette Century* (2009), Basic Books.

Anecdotal evidence and early smoking

- Anti-smoking research started in the 1930s and 1940s when cigarette smoking became increasingly popular. While some smokers seemed to be sensitive to cigarette smoke, others were completely unaffected.
- Anti-smoking research was faced with resistance based on **anecdotal evidence** such as "My uncle smokes three packs a day and he's in perfectly good health", evidence based on a limited sample size that might not be representative of the population.

Source: Brandt, *The Cigarette Century* (2009), Basic Books.

Anecdotal evidence and early smoking

- Anti-smoking research started in the 1930s and 1940s when cigarette smoking became increasingly popular. While some smokers seemed to be sensitive to cigarette smoke, others were completely unaffected.
- Anti-smoking research was faced with resistance based on **anecdotal evidence** such as "My uncle smokes three packs a day and he's in perfectly good health", evidence based on a limited sample size that might not be representative of the population.
- It was concluded that "smoking is a complex human behavior, by its nature difficult to study, confounded by human variability."

Source: Brandt, *The Cigarette Century* (2009), Basic Books.

Anecdotal evidence and early smoking

- Anti-smoking research started in the 1930s and 1940s when cigarette smoking became increasingly popular. While some smokers seemed to be sensitive to cigarette smoke, others were completely unaffected.
- Anti-smoking research was faced with resistance based on **anecdotal evidence** such as "My uncle smokes three packs a day and he's in perfectly good health", evidence based on a limited sample size that might not be representative of the population.
- It was concluded that "smoking is a complex human behavior, by its nature difficult to study, confounded by human variability."
- In time researchers were able to examine larger samples of cases (smokers), and trends showing that smoking has negative health impacts became much clearer.

Source: Brandt, *The Cigarette Century* (2009), Basic Books.

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?
- This is called a **census**.

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?
- This is called a **census**.
- There are problems with taking a census:

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?
- This is called a **census**.
- There are problems with taking a census:
- *It can be difficult to complete a census:* there always seem to be some individuals who are hard to locate or hard to measure. **And these difficult-to-find people may have certain characteristics that distinguish them from the rest of the population.**

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?
- This is called a **census**.
- There are problems with taking a census:
- *It can be difficult to complete a census:* there always seem to be some individuals who are hard to locate or hard to measure. **And these difficult-to-find people may have certain characteristics that distinguish them from the rest of the population.**
- Populations rarely stand still. Even if you could take a census, the population changes constantly, so it's never possible to get a perfect measure.

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?
- This is called a **census**.
- There are problems with taking a census:
- *It can be difficult to complete a census:* there always seem to be some individuals who are hard to locate or hard to measure. **And these difficult-to-find people may have certain characteristics that distinguish them from the rest of the population.**
- Populations rarely stand still. Even if you could take a census, the population changes constantly, so it's never possible to get a perfect measure.
- Taking a census may be more complex than sampling.

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Sampling from a population: Census

- Wouldn't it be better to just include everyone and "sample" the entire population?
- This is called a **census**.
- There are problems with taking a census:
- *It can be difficult to complete a census:* there always seem to be some individuals who are hard to locate or hard to measure. **And these difficult-to-find people may have certain characteristics that distinguish them from the rest of the population.**
- Populations rarely stand still. Even if you could take a census, the population changes constantly, so it's never possible to get a perfect measure.
- Taking a census may be more complex than sampling.

Illegal Immigrants Reluctant To Fill Out Census Form

by PETER O'DOWD

March 31, 2010 4:00 AM

from KJZZ

Listen to the Story

Morning Edition

3 min 48 sec

Playlist

Download

There is an effort underway to make sure Hispanics are accurately counted in the 2010 Census. Phoenix has some of the country's "hardest-to-count" districts. Some Latinos, especially illegal residents, fear that participating in the count will expose them to immigration raids or government harassment.

Source: http://www.npr.org/templates/story/story.php?storyId=125380052

Exploratory analysis to inference

- Sampling is natural

Exploratory analysis to inference

- Sampling is natural
- Think about sampling something you are cooking - you taste (examine) a small part of what you're cooking to get an idea about the dish as a whole.

Exploratory analysis to inference

- Sampling is natural
- Think about sampling something you are cooking - you taste (examine) a small part of what you're cooking to get an idea about the dish as a whole.
- When you taste a spoonful of soup and decide the spoonful you tasted isn't salty enough, that's **exploratory analysis**.

Exploratory analysis to inference

- Sampling is natural
- Think about sampling something you are cooking - you taste (examine) a small part of what you're cooking to get an idea about the dish as a whole.
- When you taste a spoonful of soup and decide the spoonful you tasted isn't salty enough, that's **exploratory analysis**.
- If you generalize and conclude that your entire soup needs salt, that's an **inference**.

Exploratory analysis to inference

- Sampling is natural
- Think about sampling something you are cooking - you taste (examine) a small part of what you're cooking to get an idea about the dish as a whole.
- When you taste a spoonful of soup and decide the spoonful you tasted isn't salty enough, that's **exploratory analysis**.
- If you generalize and conclude that your entire soup needs salt, that's an **inference**.
- For your inference to be valid, the spoonful you tasted (the sample) needs to be **representative** of the entire pot (the population).

Exploratory analysis to inference

- Sampling is natural
- Think about sampling something you are cooking - you taste (examine) a small part of what you're cooking to get an idea about the dish as a whole.
- When you taste a spoonful of soup and decide the spoonful you tasted isn't salty enough, that's **exploratory analysis**.
- If you generalize and conclude that your entire soup needs salt, that's an **inference**.
- For your inference to be valid, the spoonful you tasted (the sample) needs to be **representative** of the entire pot (the population).
- If your spoonful comes only from the surface and the salt is collected at the bottom of the pot, what you tasted is probably not representative of the whole pot.

Exploratory analysis to inference

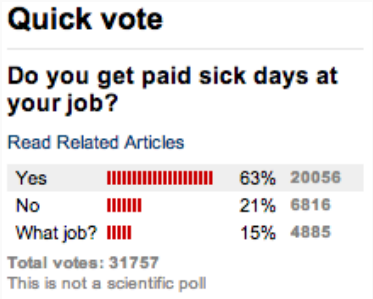
- Sampling is natural
- Think about sampling something you are cooking - you taste (examine) a small part of what you're cooking to get an idea about the dish as a whole.
- When you taste a spoonful of soup and decide the spoonful you tasted isn't salty enough, that's **exploratory analysis**.
- If you generalize and conclude that your entire soup needs salt, that's an **inference**.
- For your inference to be valid, the spoonful you tasted (the sample) needs to be **representative** of the entire pot (the population).
- If your spoonful comes only from the surface and the salt is collected at the bottom of the pot, what you tasted is probably not representative of the whole pot.
- If you first stir the soup thoroughly before you taste, your spoonful will more likely be representative of the whole pot.

Sampling bias

- **Non-response**: If only a small fraction of the randomly sampled people choose to respond to a survey, the sample may no longer be representative of the population.
- **Voluntary response**: Occurs when the sample consists of people who volunteer to respond because they have strong opinions on the issue. Such a sample will also not be representative of the population.

Sampling bias

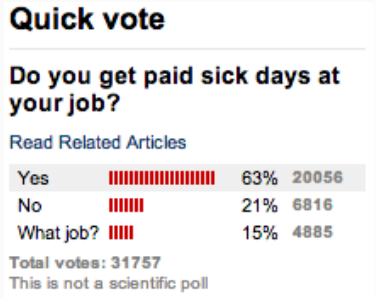
- **Non-response**: If only a small fraction of the randomly sampled people choose to respond to a survey, the sample may no longer be representative of the population.
- **Voluntary response**: Occurs when the sample consists of people who volunteer to respond because they have strong opinions on the issue. Such a sample will also not be representative of the population.



Source: cnn.com, Jan 14, 2012

Sampling bias

- **Non-response**: If only a small fraction of the randomly sampled people choose to respond to a survey, the sample may no longer be representative of the population.
- **Voluntary response**: Occurs when the sample consists of people who volunteer to respond because they have strong opinions on the issue. Such a sample will also not be representative of the population.



- **Convenience sample**: Individuals who are easily accessible are more likely to be included in the sample.

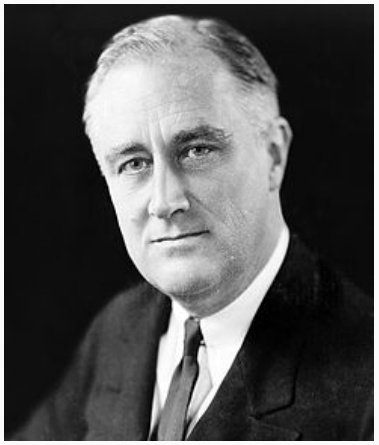
Source: cnn.com, Jan 14, 2012

Sampling bias example: Landon vs. FDR

A historical example of a biased sample yielding misleading results:

Sampling bias example: Landon vs. FDR

A historical example of a biased sample yielding misleading results:



In 1936, Landon sought the Republican presidential nomination opposing the re-election of FDR.

The Literary Digest Poll

- The Literary Digest polled about 10 million Americans, and got responses from about 2.4 million.

The Literary Digest Poll

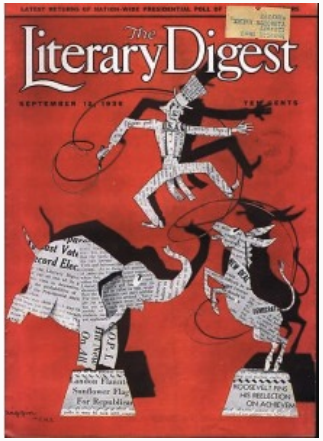
- The Literary Digest polled about 10 million Americans, and got responses from about 2.4 million.
- The poll showed that Landon would likely be the overwhelming winner and FDR would get only 43% of the votes.

The Literary Digest Poll

- The Literary Digest polled about 10 million Americans, and got responses from about 2.4 million.
- The poll showed that Landon would likely be the overwhelming winner and FDR would get only 43% of the votes.
- Election result: FDR won, with 62% of the votes.

The Literary Digest Poll

- The Literary Digest polled about 10 million Americans, and got responses from about 2.4 million.
- The poll showed that Landon would likely be the overwhelming winner and FDR would get only 43% of the votes.
- Election result: FDR won, with 62% of the votes.



- The magazine was completely discredited because of the poll, and was soon discontinued.

The Literary Digest Poll - what went wrong?

The magazine had surveyed:

The Literary Digest Poll - what went wrong?

The magazine had surveyed:

- Its own readers

The Literary Digest Poll - what went wrong?

The magazine had surveyed:

- Its own readers
- Registered automobile owners

The Literary Digest Poll - what went wrong?

The magazine had surveyed:

- Its own readers
- Registered automobile owners
- Registered telephone users

The Literary Digest Poll - what went wrong?

The magazine had surveyed:

- Its own readers
- Registered automobile owners
- Registered telephone users

These groups had incomes well above the national average of the day (remember, this is Great Depression era) which resulted in lists of voters far more likely to support Republicans than a truly **typical** voter of the time, i.e. the sample was not representative of the American population at the time.

Large samples are preferable, but...

- The Literary Digest election poll was based on a sample size of 2.4 million, which is huge, but since the sample was **biased**, the sample did not yield an accurate prediction.

Large samples are preferable, but...

- The Literary Digest election poll was based on a sample size of 2.4 million, which is huge, but since the sample was **biased**, the sample did not yield an accurate prediction.
- Back to the soup analogy: If the soup is not well stirred, it doesn't matter how large a spoon you have, it will still not taste right. If the soup is well stirred, a small spoon will suffice to test the soup.

"Correlation does not imply causation"

Explanatory and response variables

- To identify the explanatory variable in a pair of variables, identify which of the two is suspected of affecting the other:

Explanatory and response variables

- To identify the explanatory variable in a pair of variables, identify which of the two is suspected of affecting the other:
- explanatory variable $\xrightarrow{might\ affect}$ response variable

Explanatory and response variables

- To identify the explanatory variable in a pair of variables, identify which of the two is suspected of affecting the other:
- explanatory variable $\xrightarrow{might\ affect}$ response variable
- Labeling variables as explanatory and response does not guarantee the relationship between the two is actually causal, even if there is an association identified between the two variables. We use these labels only to keep track of which variable we suspect affects the other.

Observational studies and experiments

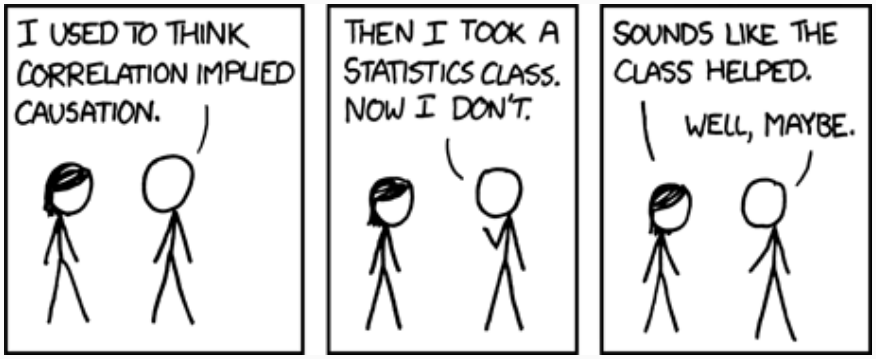
- **Observational study**: Researchers collect data in a way that does not directly interfere with how the data arise, i.e. they merely "observe", and can only establish an association between the explanatory and response variables.

Observational studies and experiments

- **Observational study**: Researchers collect data in a way that does not directly interfere with how the data arise, i.e. they merely "observe", and can only establish an association between the explanatory and response variables.
- **Experiment**: Researchers randomly assign subjects to various treatments in order to establish causal connections between the explanatory and response variables.

Observational studies and experiments

- **Observational study**: Researchers collect data in a way that does not directly interfere with how the data arise, i.e. they merely "observe", and can only establish an association between the explanatory and response variables.
- **Experiment**: Researchers randomly assign subjects to various treatments in order to establish causal connections between the explanatory and response variables.
- If you're going to walk away with one thing from the last few weeks of this class, let it be "correlation does not imply causation".



Source: http://xkcd.com/552/

Credits

These slides were adapted from the following sources:

- The Web Scraping slides and Mini HW 12 - Web Scraping assignment developed by Mine Çetinkaya-Rundel and made available under the CC BY 4.0 license.
- The Chapter 1 OpenIntro Statistics slides developed by Mine Çetinkaya-Rundel and made available under the CC BY-SA 3.0 license.