Prep3S24

BilalTariq

2024-02-16

Reminder: Prep assignments are to be completed individually. Upload a final copy of the .Rmd and renamed .pdf to your private repo, and submit the renamed pdf to Gradescope before the deadline (Sunday night, 2/18/24, by midnight).

Reading

The associated reading for the week is Sections 6.4, 8.5-8.7, and 8.9-8.10. This reading explores data intake (getting data into R from a variety of data formats), and ethics issues.

Remember, I recommend you code along with the book examples. You can try out the code yourself - just be sure to load the mdsr package and any other packages referenced. You can get the code in R script files (basically, files of just R code, not like a .Rmd) from the book website.

1 - Data Intake Basics

part a - Many of our data sets have been provided as .csv files. What does .csv stand for?

Solution: It is a non-proprietary (comma separated text format) that allows for data exchange between multiple software.

part b - Name one web-related data-table friendly format.

Solution: HTML Format

part c - The *haven* package is designed to help import files from certain other apps into R. From its help file, list three apps it can import data files from.

Hint: You may need to install *haven* if you want to look this up within R. Searching it on the web is also fine for this problem.

Solution: SAS, SPSS and STATA

part d - In Chapter 19, in our unit on text analysis, we will use the *aRxiv* package. From its help file, this package is an "Interface to the arXiv API". What does this mean?

Hint: arXiv is a website that hosts scholarly articles, often pre-prints, that are not peer-reviewed. (So, a later version of the paper might be peer-reviewed and published elsewhere.) The key to answering the question posed is the API part. Put another way, what does this package help you to do in relation to accessing the data stored by arXiv?

Solution: This package allows us to access data from the arXiv website.

2 - Web scraping

In Section 6.4.1.2, the *rvest* package is used to scrape a Wikipedia page. BUT WAIT! While we may have the technical ability to scrape a webpage, that doesn't necessarily mean we are *allowed* to scrape it.

Before scraping a web page, you should always check whether doing so is allowed.

Sometimes this information is listed on the page or in an EULA.

If you're unsure of the permissions for a particular domain, you can use the handy `paths_allowed()` function within the *robotstxt* package.

> part a - Check the permissions for the Wikipedia page using the code below. If the code returns "TRUE", then that indicates a bot has permission to access the page. Do you (via R) have permission to access the page?

Solution:

```
# Define url to use again
url <- "https://en.wikipedia.org/wiki/Mile_run_world_record_progression"

# Check bot permissions
paths_allowed(url)
```

```
en.wikipedia.org
```

```
[1] TRUE
```

Yes, we do have permission to access the page as it returns the value TRUE.

> part b - Now, use the code chunk below to follow along with the code in Section 6.4.1.2 to scrape the tables from the Wikipedia page on *Mile run world record progression*. Use `length(tables)` to identify how many tables are in the object you created called `tables`. How many tables are there?

Solution:

```
url <- "https://en.wikipedia.org/wiki/Mile_run_world_record_progression"
tables <- url %>%
  read_html() %>%
  html_elements("table")
```

```
length(tables)
```

```
[1] 13
```

Next, look at the Wikipedia page. We want to work with the table toward the bottom titled "Women Indoor IAAF era" that shows four records: one for Mary Decker, two for Doina Melinte, and one for Genzebe Dibaba.

> part c - From your `tables` object created in part b, create a dataframe called `women_indoor` that includes this "Women Indoor IAAF era" table data.

Hint: You can use the same code as used in the textbook to create the `amateur` and `records` tables, except you'll need to update the table number that's `plucked`.

Solution:

```
women_indoor <- tables %>%
  purrr::pluck(11) %>%
  html_table() %>%
  mutate(
    Time = ms(Time),
    Type = "Indoor"
  )
```

> part d - Use `kable()` to display the table from part c. Who holds the indoor one-mile world record for IAAF women, and what was her time?

Solution: The indoor one-mile world record for IAAF women is held by Genzebe Dibaba and she has a time of 4:13.31.

```
kable(women_indoor, booktabs = TRUE)
```

Time	Athlete	Nationality	Date	Venue	Type
4M 20.5S	Mary Decker	United States	February 19, 1982	San Diego United States	Indoor
4M 18.86S	Doina Melinte	Romania	February 13, 1988	East Rutherford United States	Indoor
4M 17.14S	Doina Melinte	Romania	February 9, 1990	East Rutherford United States	Indoor
4M 13.31S	Genzebe Dibaba	Ethiopia	February 17, 2016	Stockholm Sweden	Indoor

```
glimpse(women_indoor)
```

```
Rows: 4
Columns: 6
$ Time        <Period> 4M 20.5S, 4M 18.86S, 4M 17.14S, 4M 13.31S
$ Athlete     <chr> "Mary Decker", "Doina Melinte", "Doina Melinte", "Genze~
$ Nationality <chr> "United States", "Romania", "Romania", "Ethiopia"
$ Date        <chr> "February 19, 1982", "February 13, 1988", "February 9, 199~
$ Venue       <chr> "San Diego  United States", "East Rutherford  United State~
$ Type        <chr> "Indoor", "Indoor", "Indoor", "Indoor"
```

part e - Perform the necessary wrangling as directed to answer the question below.

Read through all the desired items before beginning!

- Create a dataframe called `women_outdoor` that contains the table for "Women's IAAF era" (starting with Anne Smith's record and ending with Faith Kipyegon's record).
- Combine `women_indoor` and `women_outdoor` into one dataframe called `women_records` using the `bind_rows()` function.
- Include a variable called `Type` in this new dataframe to indicate whether a particular observation corresponds to an indoor record or an outdoor record (Hint: create `Type` separately in each dataframe before combining).
- Finally, arrange `women_records` by ascending time, drop the `Venue` variable, and display the table using `kable()`.
- Use your wrangled data set to answer this question: Who holds the fastest record, and was it from an indoor or outdoor event?

Solution: The fastest record was from an Outdoor event by Faith Kipyegon

```
women_outdoor <- tables %>%
  purrr::pluck(9) %>%
  html_table() %>%
  mutate(
    Time = ms(Time, quiet = FALSE, roll = TRUE),
    Type = "Outdoor"
  ) %>%
  arrange(Time)


women_outdoorIndoor <- bind_rows(women_outdoor, women_indoor) %>%
  select(Time, Athlete, Nationality, Date, Type)
```

```
kable(women_outdoorIndoor)
```

Time	Athlete	Nationality	Date	Type
4M 7.64S	Faith Kipyegon	Kenya	21 July 2023[11]	Outdoor
4M 12.33S	Sifan Hassan	Netherlands	12 July 2019	Outdoor
4M 12.56S	Svetlana Masterkova	Russia	14 August 1996[10]	Outdoor
4M 15.61S	Paula Ivan	Romania	10 July 1989[10]	Outdoor
4M 16.71S	Mary Decker-Slaney	United States	21 August 1985[10]	Outdoor
4M 17.44S	Maricica Puică	Romania	9 September 1982[10]	Outdoor
4M 18.08S	Mary Decker-Tabb	United States	9 July 1982[10]	Outdoor
4M 20.89S	Lyudmila Veselkova	Soviet Union	12 September 1981[10]	Outdoor
4M 21.7S	Mary Decker	United States	26 January 1980[10]	Outdoor
4M 22.1S	Natalia Mărășescu	Romania	27 January 1979[10]	Outdoor
4M 23.8S	Natalia Mărășescu	Romania	21 May 1977[10]	Outdoor
4M 29.5S	Paola Pigni	Italy	8 August 1973[10]	Outdoor
4M 35.3S	Ellen Tittel	West Germany	20 August 1971[10]	Outdoor
4M 36.8S	Maria Gommers	Netherlands	14 June 1969[10]	Outdoor
4M 37S	Anne Smith	United Kingdom	3 June 1967[10]	Outdoor
4M 20.5S	Mary Decker	United States	February 19, 1982	Indoor
4M 18.86S	Doina Melinte	Romania	February 13, 1988	Indoor
4M 17.14S	Doina Melinte	Romania	February 9, 1990	Indoor
4M 13.31S	Genzebe Dibaba	Ethiopia	February 17, 2016	Indoor

3 - Ethics

As we wrap up the chapter on ethics, what are three major takeaways from Chapter 8 that had an impact on how you think about approaching your work as a budding data scientist?

Solution: