

# Regular Expressions

Instructor - Soumya Mukherjee

Content Credit- Dr. Matthew Beckman and Olivia Beck

July 27, 2023

## Chapter 17: Key Ideas

- Regular expressions allow us to match meaningful **patterns** in character strings
- Some popular uses:
  - detect whether a **pattern** is contained in a string (use `filter()` & `grep1()` )
  - substitute the elements of that **pattern** with something else (use `mutate()` & `gsub()` )
  - extract a component that matches the **pattern** (use `tidyr::extract()` )

## Some Exploits in the Land of RegEx

- Medtronic, Inc - quality monitoring for medical technology
  - Match key word or phrase in offline complaint data (uncommon)
  - Subset of complaint data and evaluate rate of some outcome over time
- PSU Men's Volleyball
  - Teams now have access to complete data for play in every match
  - Using RegEx to help parse the data to gain competitive advantage for PSU
  - (Sort of like Moneyball for Volleyball...)
- Scraping HTML data
  - We scraped the Men's Pole Vault World Records from Wikipedia ([https://en.wikipedia.org/wiki/Men%27s\\_pole\\_vault\\_world\\_record\\_progression](https://en.wikipedia.org/wiki/Men%27s_pole_vault_world_record_progression))
  - The footnotes in the `Date` column do not allow working with the dates directly, so we need to clean them up.
  - We can use RegEx to clean up

# How to Survive in the Land of Regex

- Step 1: Memorize the following special characters and their use: \d, \w, \S, [0-9], [^0-9], [[:lower:]], [[:alnum:]], \W, , ?, ., \$, %, |, \<, ^, \, {3}, \*, +, \s, \B, \>, \x

# How to Survive in the Land of Regex

- **NO!!!** absolutely no need to memorize all of it
- Use the RStudio Cheat Sheet: <https://www.rstudio.com/resources/cheatsheets/> (<https://www.rstudio.com/resources/cheatsheets/>)
- Use Google
- Just like everything else in (R) Programming:
  - Don't start from scratch
  - Find working code that does something similar
  - Make many iterations of small changes checking at each change that it didn't break
  - Keep going until the original code evolves into the thing you want!

## Pole Vault Demo

- live demo to clean up Pole Vault Records progression seen in web scraping



## Pole Vault Records Clean Up

Hide

```
webpage <- "https://en.wikipedia.org/wiki/Men%27s_pole_vault_world_record_progression"
```

```
table_list <-  
  webpage %>%  
  read_html(header=TRUE) %>%  
  html_nodes(css = "table") %>%  
  html_table(fill = TRUE)
```

```
PVRecords <- table_list[[2]] # convert list to data frame  
head(PVRecords, 3) # inspect the data
```

### Mark

<chr>

4.02 m (.mw-parser-output .frac{white-space:nowrap}.mw-parser-output .frac .num,.mw-parser-output .frac .den{font-size:80%;line-height:0;vertical-align:super}.mw-parser-output .frac .den{vertical-align:sub}.mw-parser-output .sr-only{border:0;clip:rect(0,0,0,0);clip-path:polygon(0px 0px,0px 0px,0px 0px);height:1px;margin:-1px;overflow:hidden;padding:0;position:absolute;width:1px}13 ft 2+¼ in)

4.09 m (13 ft 5 in)

4.12 m (13 ft 6 in)

3 rows | 1-1 of 6 columns

Hide

NA

### Tasks to clean up:

1. we should fix the variable name representing the number of world records achieved by each athlete
2. locate and replace all footnotes in the `Date` variable using `gsub()`
3. convert `Date` to a date class variable in R using a `lubridate` function
4. use `tidyr::extract()` to store the metric heights from the `Record` variable (make sure there are no spaces )

## Solutions

Hide

```
# locate and replace all footnotes in the `Date` column
PVMen <-
  PVRecords %>%
    rename(recordsBroken = `#[4]`) %>%
    mutate(Date = gsub(pattern = "\\[\\.\\]", replacement = "", x = Date)) %>%
    mutate(Date = lubridate::mdy(Date)) %>% #convert to date
    tidyr::extract(col = Mark, into = "Meters", regex = "(^\\d\\.\\.\\d)") %>%
    mutate(Meters = parse_number(Meters)) #convert to numeric(drops non-numeric characters)

PVMen %>%
  head()
```

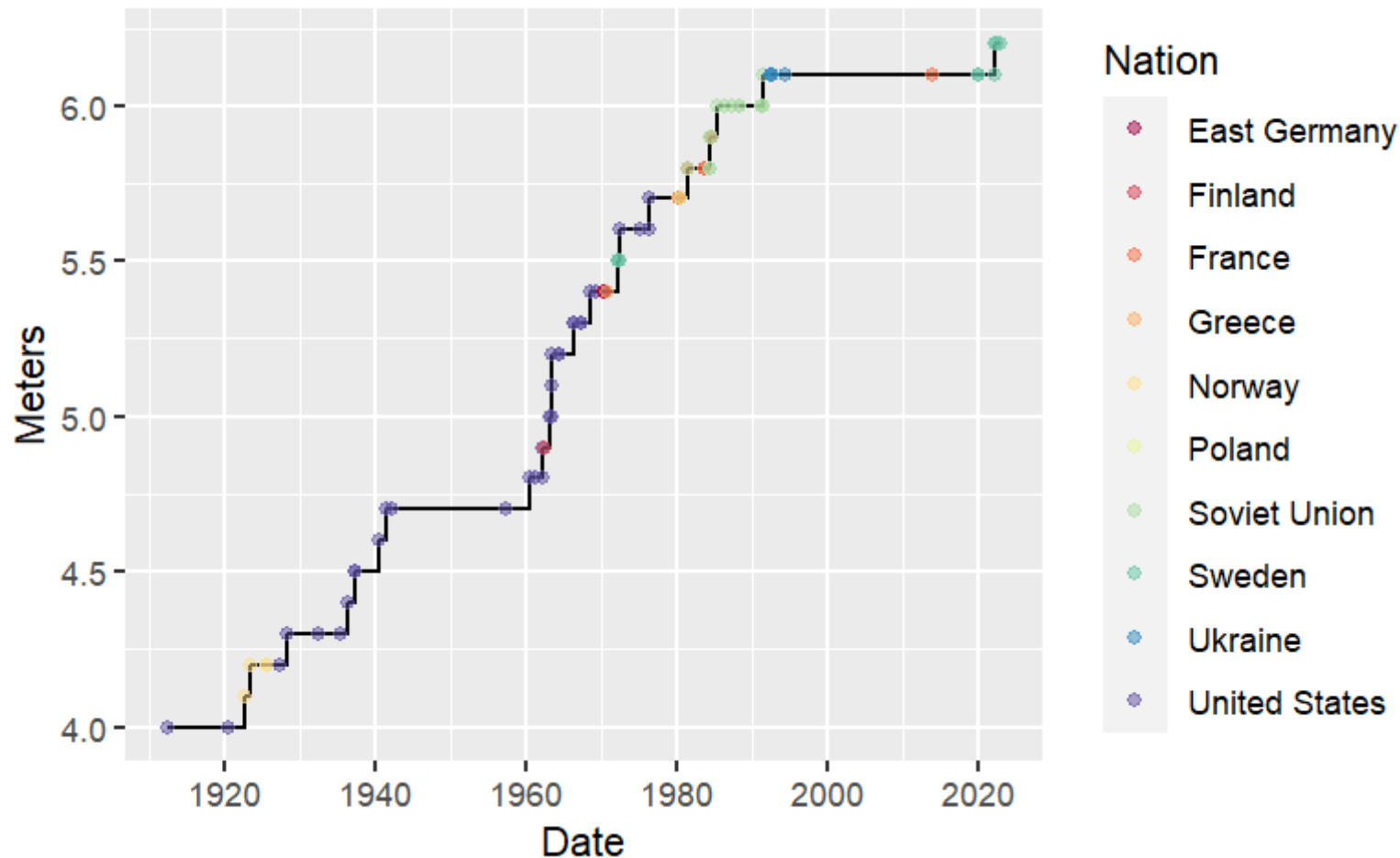
<b>Meters</b> <dbl>	<b>Athlete</b> <chr>	<b>Nation</b> <chr>	<b>Venue</b> <chr>	<b>Date</b> <date>	<b>recordsBroken</b> <int>
4.0	Marc Wright	United States	Cambridge, U.S.	1912-06-08	1
4.0	Frank Foss	United States	Antwerp, Belgium	1920-08-20	1
4.1	Charles Hoff	Norway	Copenhagen, Denmark	1922-09-22	1
4.2	Charles Hoff	Norway	Copenhagen, Denmark	1923-07-22	2
4.2	Charles Hoff	Norway	Oslo, Norway	1925-08-13	3
4.2	Charles Hoff	Norway	Turku, Finland	1925-09-27	4

6 rows

## Cool Graphs

Hide

```
PVMen %>%
  ggplot(aes(x = Date, y = Meters)) +
  geom_step() +
  geom_point(alpha = 0.5, aes(color = Nation))+
  scale_color_brewer(palette = "Spectral")
```



## Additional Resources

- <https://www.datacamp.com/tutorial/regex-r-regular-expressions-guide> (<https://www.datacamp.com/tutorial/regex-r-regular-expressions-guide>)
- <https://github.com/rstudio/cheatsheets/blob/main/strings.pdf> (<https://github.com/rstudio/cheatsheets/blob/main/strings.pdf>)

# Assignments

- Reading Quiz Chapter 17 (due 11:59pm Sunday , July 30)
- Activity: Statistics of Gene Expression (due 9:59am Sunday , July 30)