Data 100, Spring 2022 **February 11, 2022**

Discussion #4

Regular Expressions

Here's a complete list of metacharacters:

`. ^ $ * + ? { } [ ] \ | ( )`

Some reminders on what each can do (this is not exhaustive):

- **"^"** matches the position at the beginning of string (unless used for negation "[^]")
- **"$"** matches the position at the end of string character.
- **"?"** match preceding literal or sub-expression 0 or 1 times.
- **"+"** match preceding literal or sub-expression *one* or more times.
- **"*"** match preceding literal or sub-expression *zero* or more times
- **"."** match any character except new line.
- **"[ ]"** match any one of the characters inside, accepts a range, e.g., "[a-c]".
- **"( )"** used to create a sub-expression
- **"\d"** match any *digit* character. "\D" is the complement.
- **"\w"** match any *word* character (letters, digits, underscore). "\W" is the complement.
- **"\s"** match any *whitespace* character including tabs and newlines. \S is the complement.
- **"*?"** Non-greedy version of *. Not fully discussed in class.
- **"\b"** match boundary between words. Not discussed in class.
- **"+?"** Non-greedy version of +. Not discussed in class.
- **"{m,n}"** The preceding element or subexpression must occur between m and n times, inclusive.

Some useful `re` package functions:

- **re.split(pattern, string)** split the `string` at substrings that match the `pattern`. Returns a list.
- **re.sub(pattern, replace, string)** apply the `pattern` to `string` replacing matching substrings with `replace`. Returns a string.
- **re.findall(pattern, string)** Returns a list of all matches for the given `pattern` in the `string`.

Regular Expressions

1. Which strings contain a match for the following regular expression, `"1+1$"`? The character `"␣"` represents a single space.

   ○ A. `What␣is␣1+1` ○ B. `Make␣a␣wish␣at␣11:11` ○ C. `111␣Ways␣to␣Succeed`

2. Write a regular expression that matches strings (including the empty string) that only contain lowercase letters and numbers.

3. Given `sometext = "I’ve␣got␣10␣eggs,␣20␣gooses,␣and␣30␣giants."`, use `re.findall` to extract all the items and quantities from the string. The result should look like **`[’10 eggs’, ’20 gooses’, ’30 giants’]`**. You may assume that a space separates quantity and type, and that each item ends in s.

4. For each pattern specify the starting and ending position of the first match in the string. The index starts at zero and we are using closed intervals (both endpoints are included).

	`abcdefg`	`abcs!`	`ab␣abc`	`abc,␣123`
`abc*`	$[0, 2]$			
`[^\s]+`				
`ab.*c`				
`[a-z1,9]+`				

5. *(Bonus)* Given the following text in a variable `log`:

```
169.237.46.168 - - [26/Jan/2014:10:47:58 -0800]
"GET␣/stat141/Winter04/␣HTTP/1.1" 200 2585
"http://anson.ucdavis.edu/courses/"
```

   Fill in the regular expression in the variable `pattern` below so that after it executes, day is 26, month is Jan, and year is 2014.

```
pattern = ...
matches = re.findall(pattern, log)
day, month, year = matches[0]
```

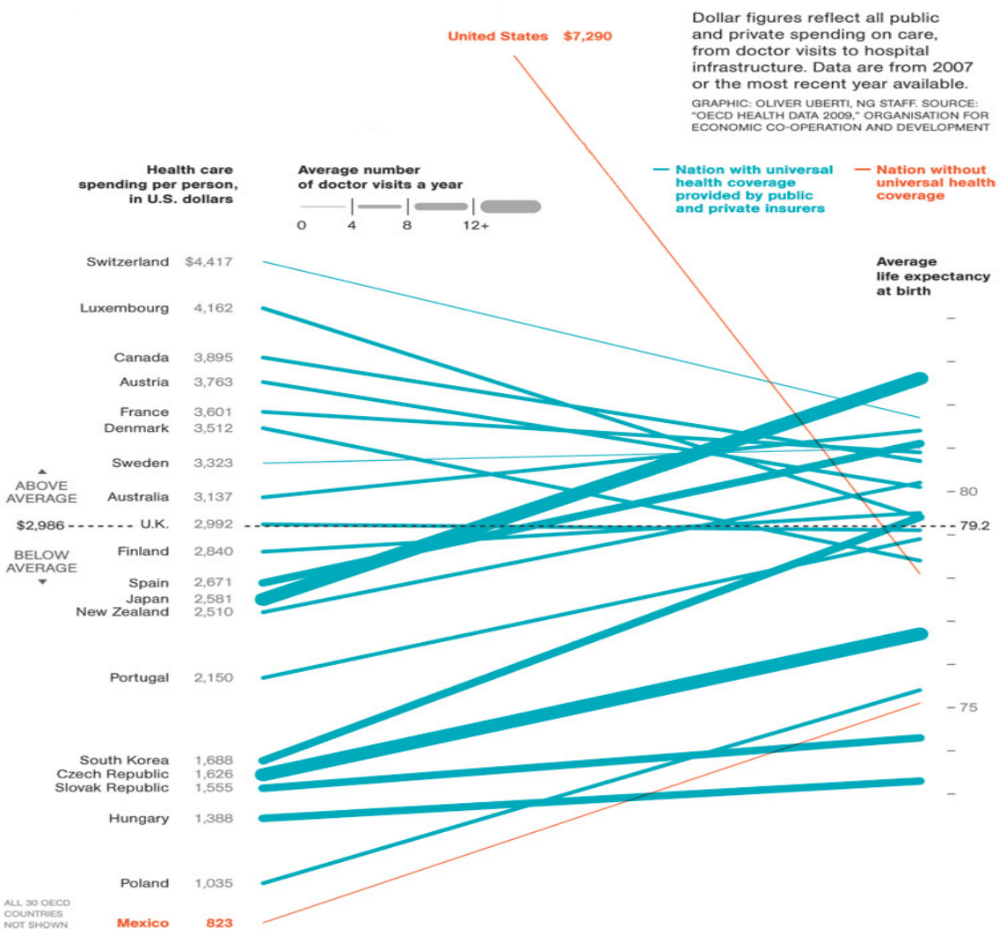
6. *(Bonus)* Given that `sometext` is a string, use `re.sub` to replace all clusters of non-vowel characters with a single period. For example `"a␣big␣moon,␣between␣us..."` would be changed to `"a.i.oo.e.ee.u."`.

7. *(Bonus)* Given the text:

   ```
   "<record>␣Josh␣Hug␣<hug@cs.berkeley.edu>␣Faculty␣</record>"
   "<record>␣Lisa␣Yan␣<lisa.yan@berkeley.edu>␣Instructor␣</record>"
   ```

   Which of the following matches exactly to the email addresses (including angle brackets)?
   ○ A. `<.*@.*>` ○ B. `<[^>]*@[^>]*>` ○ C. `<.*@\w+\..*>`

Data Visualization



Dollar figures reflect all public and private spending on care, from doctor visits to hospital infrastructure. Data are from 2007 or the most recent year available.

GRAPHIC: OLIVER UBERTI, NG STAFF. SOURCE: "OECD HEALTH DATA 2009," ORGANISATION FOR ECONOMIC CO-OPERATION AND DEVELOPMENT

8. The first part of the discussion will be centered on the above visualization.

   (a) Five variables are being represented visually in this graphic. What are they and what are their feature types (ie qualitative, quantitative, nominal, ordinal)?

   (b) How are the variables represented in the graphic, e.g., the variable `XXX` is mapped to the x-axis, the variable `WWW` is mapped to the y-axis, the variable `ZZZ` is conveyed through color, etc.?

(c) How can we figure out how to interpret the visual qualities of the plot, e.g., how do we know what a color represents?

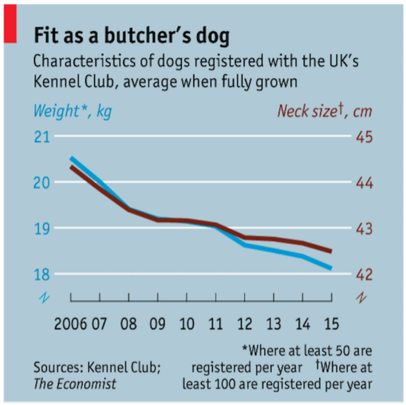
(d) What purpose does the comment at the top right of the plot serve?

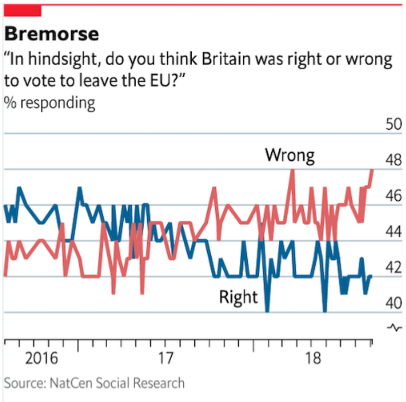
(e) Make 3 observations about the figure. Describe the feature that you are basing your observation on.

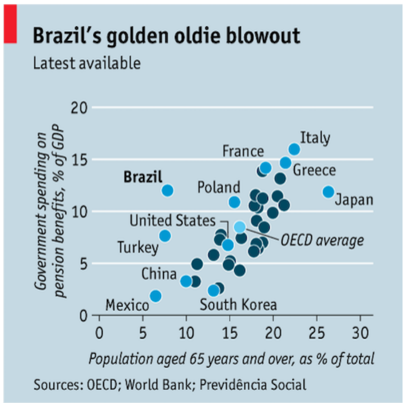
For example, South Korea's expenditure on health care is comparable to Eastern European countries (and among the lowest of all countries plotted), but the life expectancy is much higher than the Eastern European countries. In the plot we see that the left endpoint of South Korea's line segment is near the Eastern European countries, but the slope of the line segment is much steeper.

(f) Consider the steep negative slope and narrowness of the line segment that represents the data for the United States. What systemic, social, or societal issues might explain this?

9. Creating visualizations that represent data accurately and that support the narrative we wish to create is no easy task. Even the journalists and editors at *The Economist*, a newspaper known for it's compelling, data-driven articles, have been known to make blunders. Three of their ill-thought-out plots are presented below. Consider what aspects of the visualizations are misleading, and think of ways in which you can remedy them.







Hint: The datapoints in the rightmost plot are shaded based on whether or not they are labeled.