Homework_3

Kenan Sooklall

2/10/2021

#1. Using the 173 majors listed in fivethirtyeight.com's College Majors dataset [https://fivethirtyeight.com/features/the-economic-guide-to-picking-a-college-major/], provide code that identifies the majors that contain either "DATA" or "STATISTICS"

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.0 --

## v ggplot2 3.3.3     v purrr   0.3.4
## v tibble  3.0.6     v dplyr   1.0.4
## v tidyr   1.1.2     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1

## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
df <- read.csv('https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/majors-lis
df %>% filter(across(Major, ~ grepl('DATA|STATISTICS', .)))
```

```
##   FOD1P                                          Major          Major_Category
## 1  6212 MANAGEMENT INFORMATION SYSTEMS AND STATISTICS               Business
## 2  2101      COMPUTER PROGRAMMING AND DATA PROCESSING Computers & Mathematics
## 3  3702               STATISTICS AND DECISION SCIENCE Computers & Mathematics
```

#2 Write code that transforms the data below:

```
df <- data.frame(x1=c("bell pepper", "bilberry", "blackberry", "blood orange"), x2=c("blueberry", "canta
                 x3=c("elderberry", "lime", "lychee", "mulberry"))
```

[1] "bell pepper" "bilberry" "blackberry" "blood orange"

[5] "blueberry" "cantaloupe" "chili pepper" "cloudberry"

[9] "elderberry" "lime" "lychee" "mulberry"

[13] "olive" "salal berry"

Into a format like this:

```
vect <- as.vector(as.matrix(df[,c("x1", "x2", "x3")]))
vect
```

```
##  [1] "bell pepper"  "bilberry"     "blackberry"   "blood orange" "blueberry"
##  [6] "cantaloupe"   "chili pepper" "cloudberry"   "elderberry"   "lime"
## [11] "lychee"       "mulberry"
```

c("bell pepper", "bilberry", "blackberry", "blood orange", "blueberry", "cantaloupe", "chili pepper", "cloudberry", "elderberry", "lime", "lychee", "mulberry", "olive", "salal berry")

The two exercises below are taken from R for Data Science, 14.3.5.1 in the on-line version:

#3 Describe, in words, what these expressions will match:

Regex checked on https://regexr.com/ and https://regex101.com/

(.)\1\1

Matches a text that is repeated 3 times like Will match: (111)1(222)2)(333)(aaa)z(bbb)d(ccc)

(.)(.)\2\1

Matches any character for the first two steps then the number 2 and 1 separated by a backslash Will match: ab\2\1 AND bb\2\1 Will not match: xy\3\1

(..)\1

Two capture groups, matches the first pair then the (\1) will match those pairs. A total of 4 characters will match Will match: (2222)(3333) Will not match: aaac

(.).\1.\1

Matches any first two character then (\1) exactly then any other character then a (\1) Will match: qb\1f\1 ab\1c\1 Will not match: qb\1f\2 ab\4d\2

(.)(.)(.).*\3\2\1

Matches any first 3 character then an infinate amount of characters followed by (\3)(\2)(\1) exactly Will match: abcwgasdhrhrana\3\2\1 Will not match: dgasdhgaheghjlhlasg\4\2\1

#4 Construct regular expressions to match words that:

Start and end with the same character.

"(^.).*\1$"

Contain a repeated pair of letters (e.g. "church" contains "ch" repeated twice.)

"(..).*\1"

Contain one letter repeated in at least three places (e.g. "eleven" contains three "e"s.)

([a-z]).\|*1*.\1