

Week 3 Assignment

Warner Alexis

Please deliver links to an R Markdown file (in GitHub and rpubs.com) with solutions to the problems below. You may work in a small group, but please submit separately with names of all group participants in your submission.

#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"

```
## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr      1.1.3      v readr      2.1.4
## v forcats    1.0.0      v stringr    1.5.0
## v ggplot2    3.4.3      v tibble     3.2.1
## v lubridate  1.9.2      v tidyr      1.3.0
## v purrr      1.0.2
## -- Conflicts ----- tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()     masks stats::lag()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

I am going to provide code that identifies the majors that contain either "DATA" or "STATISTICS"

```
# read the data in GitHub
major <- read.csv("https://raw.githubusercontent.com/fivethirtyeight/data/master/college-majors/majors.csv",
                  stringsAsFactors = F, header = T)
data_m <- major %>% filter(str_detect(Major, "DATA|STATISTICS"))

gt_data <- gt(data_m)
# Create two additional footnotes, using the
# `columns` and `where` arguments of `data_cells()`
gt_data |>
  tab_header(
    title = "The Data Science and Technology Majors",
    subtitle = "The Only DATA and STATISTICS Majors"
  )
```

The Data Science and Technology Majors
The Only DATA and STATISTICS Majors

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

```
# Show the gt Table
gt_data
```

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

```
d <- "(.)(.)\\2\\1"
str_detect(abc,d)
```

```
## [1] FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE FALSE
```

```
str_match(abc,d)
```

```
##      [,1]    [,2] [,3]
## [1,] NA     NA   NA
## [2,] NA     NA   NA
## [3,] NA     NA   NA
## [4,] NA     NA   NA
## [5,] NA     NA   NA
## [6,] NA     NA   NA
## [7,] "aaaa" "a"   "a"
## [8,] "bbbb" "b"   "b"
## [9,] NA     NA   NA
```

“(.)\\2\\1” would match any match contain in the second parenthesis which mean it will match any four of the same letters. Examples: “aaaa”, “aabbbbcccc”— It will only match a and b

```
j <- c("(.)\\1")
str_detect(abc,j)
```

```
## [1] TRUE FALSE FALSE TRUE TRUE TRUE FALSE FALSE FALSE
```

```
str_match(abc,j)
```

```
##      [,1]      [,2]
## [1,] "bc\\001" "bc"
## [2,] NA       NA
## [3,] NA       NA
## [4,] "z\\001\\001" "z\\001"
## [5,] "Z\\001\\001" "Z\\001"
## [6,] "b\\001\\001" "b\\001"
## [7,] NA       NA
## [8,] NA       NA
## [9,] NA       NA
```

“(.)\\1” will match any of the last two characters in a group string folowed by “\\1”. For Example: “abc\\1” – bc will be selected “acgdefstwrhyg9.\\1” — g9 will be selected.

```
c_1 <- c("cdcacdabbb11","dgdgdfg","abacgwabda","trtrtrtrtr")
p <- c("(.)\\.\\1\\.\\1")
str_detect(c_1,p)
```

```
## [1] TRUE TRUE FALSE TRUE
```

```
str_match(c_1,p)
```

```
##      [,1]    [,2]
## [1,] "cdcac" "c"
## [2,] "dgdgd" "d"
## [3,] NA     NA
## [4,] "trtrt" "t"
```

“(.)\\.\\1\\.\\1” will match string characters only where their first letter is identical after every other string characters. For example: “cdcacdabbb11” will match “c” “trtrtrtr” will match “t”

```
ch <- c("bcbdbcbdbcbd", "cdcacdbbb11", "dfdfhjdfh", "spsdlkjpsps", "000550050005")
l <- c("(.) (.) (.) .*\\3\\2\\1")
str_detect(ch, l)
```

```
## [1] TRUE FALSE FALSE TRUE TRUE
```

```
str_match(ch, l)
```

```
##      [,1]      [,2] [,3] [,4]
## [1,] "bcbdbcbdbcbd" "d"  "b"  "c"
## [2,] NA           NA   NA   NA
## [3,] NA           NA   NA   NA
## [4,] "spsdlkjpsps" "s"  "p"  "s"
## [5,] "00055005000" "0"  "0"  "0"
```

“(.) (.) (.) .*\\3\\2\\1” will only match characters that repeat three times in a string group. For Examples : “bcbdbcbdbcbd” will match dbc “spsdlkjpsps” will match “s” “p” “s”

4 Construct regular expressions to match words that: Start and end with the same character. Contain a repeated pair of letters (e.g. “church” contains “ch” repeated twice.) Contain one letter repeated in at least three places (e.g. “eleven” contains three “e”s.)

```
words <- c("alababa", "cardiac", "chaotic", "clementine", "blueberry", "guava", "jujube")
str_view(words, "^ (.) .*\\1$", match = T)
```

```
## [1] | <alababa>
## [2] | <cardiac>
## [3] | <chaotic>
```

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

```
v <- c("(.) .*\\1 .*\\1")
str_detect(words, v)
```

```
## [1] TRUE FALSE FALSE TRUE FALSE FALSE FALSE
```

```
str_view(words, v, match = T)
```

```
## [1] | <alababa>
## [4] | cl<ementine>
```

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

```
k <- c("(.) .*\\1 .*\\1")
str_match(fruit, k)
```

```
##      [,1]      [,2]
## [1,] NA           NA
## [2,] NA           NA
## [3,] NA           NA
## [4,] "anana"      "a"
## [5,] "ell peppe"  "e"
## [6,] NA           NA
## [7,] NA           NA
## [8,] NA           NA
## [9,] "ood o"      "o"
## [10,] NA          NA
## [11,] NA          NA
## [12,] NA          NA
## [13,] NA          NA
```

##	[14,]	NA	NA
##	[15,]	NA	NA
##	[16,]	NA	NA
##	[17,]	"pepp"	"p"
##	[18,]	"ementine"	"e"
##	[19,]	NA	NA
##	[20,]	NA	NA
##	[21,]	"ranberr"	"r"
##	[22,]	NA	NA
##	[23,]	NA	NA
##	[24,]	NA	NA
##	[25,]	NA	NA
##	[26,]	NA	NA
##	[27,]	NA	NA
##	[28,]	NA	NA
##	[29,]	"elderbe"	"e"
##	[30,]	NA	NA
##	[31,]	NA	NA
##	[32,]	NA	NA
##	[33,]	NA	NA
##	[34,]	NA	NA
##	[35,]	NA	NA
##	[36,]	NA	NA
##	[37,]	NA	NA
##	[38,]	NA	NA
##	[39,]	NA	NA
##	[40,]	NA	NA
##	[41,]	NA	NA
##	[42,]	"iwi frui"	"i"
##	[43,]	NA	NA
##	[44,]	NA	NA
##	[45,]	NA	NA
##	[46,]	NA	NA
##	[47,]	NA	NA
##	[48,]	NA	NA
##	[49,]	NA	NA
##	[50,]	NA	NA
##	[51,]	NA	NA
##	[52,]	NA	NA
##	[53,]	NA	NA
##	[54,]	NA	NA
##	[55,]	NA	NA
##	[56,]	"apaya"	"a"
##	[57,]	NA	NA
##	[58,]	NA	NA
##	[59,]	NA	NA
##	[60,]	NA	NA
##	[61,]	NA	NA
##	[62,]	"pineapp"	"p"
##	[63,]	NA	NA
##	[64,]	NA	NA
##	[65,]	NA	NA
##	[66,]	"e mangostee"	"e"
##	[67,]	NA	NA

##	[68,]	NA	NA
##	[69,]	NA	NA
##	[70,]	"raspberr"	"r"
##	[71,]	"redcurr"	"r"
##	[72,]	NA	NA
##	[73,]	NA	NA
##	[74,]	NA	NA
##	[75,]	NA	NA
##	[76,]	"rawberr"	"r"
##	[77,]	NA	NA
##	[78,]	NA	NA
##	[79,]	NA	NA
##	[80,]	NA	NA