

STA 445

Constant Yaokumah

2023-10-27

```
library(stringr)
library(refinr)
library(dplyr)
library(lubridate)
library(mosaicData)
```

Q1

```
strings <- c("seed", "banana", "pens", "pineapple")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, 'a') )
```

```
##      string result
## 1      seed  FALSE
## 2    banana   TRUE
## 3      pens  FALSE
## 4 pineapple   TRUE
```

Return TRUE if “strings” contain ‘a’ in any word, otherwise FALSE

Q1b

```
strings <- c("able", "banana", "I am able", "pineapple")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, 'ab') )
```

```
##      string result
## 1      able   TRUE
## 2    banana  FALSE
## 3 I am able   TRUE
## 4 pineapple  FALSE
```

Return TRUE if ” strings” contain ‘ab’ in that order in any word, otherwise FALSE

Q1 c

```
strings <- c("able", "banana", "pepper", "pineapple")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '[ab]') )
```

```
##      string result
## 1     able    TRUE
## 2    banana    TRUE
## 3    pepper   FALSE
## 4 pineapple    TRUE
```

Return TRUE if " strings" contain 'ab' , 'b' or 'a', order doesn't matter in any word, otherwise FALSE

Q d

```
strings <- c("able", "banana", "peppera", "pineapple")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '^[ab]') )
```

```
##      string result
## 1     able    TRUE
## 2    banana    TRUE
## 3    peppera  FALSE
## 4 pineapple  FALSE
```

Return TRUE if " strings" contain 'ab' , 'b' or 'a', must begin the word, order doesn't matter in any word, otherwise FALSE

Q e

```
strings <- c(" 457 Aaron ", "banana", "pepper", " 2345 Amen ")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '\\d+\\s[aA]') )
```

```
##      string result
## 1 457 Aaron    TRUE
## 2   banana  FALSE
## 3   pepper  FALSE
## 4 2345 Amen    TRUE
```

Return TRUE if "strings" matches one or more digits, any white space and either lowercase letter 'a' or uppercase letter 'A', otherwise FALSE ## Q f

```
strings <- c(" 457 Aaron ", "banana", "pepper", " 2345 Amen ", "5678 a")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '\\d+\\s*[aA]') )
```

```
##      string result
## 1    457 Aaron    TRUE
## 2      banana FALSE
## 3      pepper FALSE
## 4 2345   Amen    TRUE
## 5      5678 a    TRUE
```

Return TRUE if “strings” matches one or more digits, zero or more white space and either lowercase letter ‘a’ or uppercase letter ‘A’, otherwise FALSE

Q g

```
strings <- c("Aaron457 ", "banana", "pepper", " 2345   Amen ")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '.*') )
```

```
##      string result
## 1   Aaron457    TRUE
## 2     banana    TRUE
## 3     pepper    TRUE
## 4 2345   Amen    TRUE
```

Return TRUE if “strings” contain any character with zero or more repetitions of previous, otherwise FALSE

Q h

```
strings <- c("aybar", "cdbar", "5abbar", "Kabbar")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '^\\w{2}bar') )
```

```
##   string result
## 1  aybar    TRUE
## 2  cdbar    TRUE
## 3 5abbar FALSE
## 4 Kabbar FALSE
```

Return TRUE if “strings” matches the start of a string, any alphanumeric character and exactly two word character expected with literal string ‘bar’, otherwise FALSE

Q i

```
strings <- c("foo.bar", "xybar", "Kabbar")
data.frame( string = strings ) %>%
  mutate( result = str_detect(string, '(foo\\.bar)|(\\w{2}bar)') )
```

```
##      string result
## 1 foo.bar    TRUE
## 2  xybar     TRUE
## 3 Kabbar    FALSE
```

Return TRUE if “strings” matches ‘foo.bar’ or matches the start of a string, any alphanumeric character and exactly two word character expected with literal string ‘bar’, otherwise FALSE

Q2

```
file.names <- c('S123.P2.C10_20120621_213422.jpg',
               'S10.P1.C1_20120622_050148.jpg',
               'S187.P2.C2_20120702_023501.jpg')
file.name <- str_replace_all(file.names, pattern = '_', replacement = '.')
file.name.split <- data.frame(str_split_fixed(file.name, pattern = '\\.', n = 6))
year_mon_day <- (file.name.split[,4])
hour_min_sec <- (file.name.split[,5])
Year <- str_sub(year_mon_day, start = 1, end = 4)
Month <- str_sub(year_mon_day, start = 5, end = 6)
Day <- str_sub(year_mon_day, start = 7, end = 8)
Hour <- str_sub(hour_min_sec, start = 1, end = 2)
Minute <- str_sub(hour_min_sec, start = 3, end = 4)
Second <- str_sub(hour_min_sec, start = 5, end = 6)
Site <- file.name.split[,1]
Plot <- file.name.split[,2]
Camera <- file.name.split[,3]
file.name.final <- data.frame(Site,Plot,Camera,Year,Month,Day,Hour,Minute,Second)
file.name.final
```

```
##      Site Plot Camera Year Month Day Hour Minute Second
## 1 S123    P2     C10 2012    06  21   21     34     22
## 2  S10    P1      C1 2012    06  22    5     01     48
## 3 S187    P2      C2 2012    07  02    2     35     01
```

Q3

```
Gettysburg <- ('Four score and seven years ago our fathers brought forth on this
continent, a new nation, conceived in Liberty, and dedicated to the proposition
that all men are created equal.
```

```
Now we are engaged in a great civil war, testing whether that nation, or any
nation so conceived and so dedicated, can long endure. We are met on a great
battle-field of that war. We have come to dedicate a portion of that field, as
a final resting place for those who here gave their lives that that nation might
live. It is altogether fitting and proper that we should do this.
```

```
But, in a larger sense, we can not dedicate -- we can not consecrate -- we can
not hallow -- this ground. The brave men, living and dead, who struggled here,
have consecrated it, far above our poor power to add or detract. The world will
```

little note, nor long remember what we say here, but it can never forget what they did here. It is for us the living, rather, to be dedicated here to the unfinished work which they who fought here have thus far so nobly advanced. It is rather for us to be here dedicated to the great task remaining before us -- that from these honored dead we take increased devotion to that cause for which they gave the last full measure of devotion -- that we here highly resolve that these dead shall not have died in vain -- that this nation, under God, shall have a new birth of freedom -- and that government of the people, by the people, for the people, shall not perish from the earth.')

```
words <- str_replace_all(Gettysburg, pattern = '\\-\\-|\\-|\\.|\\.', replacement = '')
words <- str_split(words, "\\s+")[1]
length_word <- nchar(words)
mean_word_length <- mean(length_word)
paste("Mean_word_length:", mean_word_length)
```

```
## [1] "Mean_word_length: 4.23985239852399"
```

12 Q1

```
library(lubridate)
mdy("September 13, 2010.")
```

```
## [1] "2010-09-13"
```

```
mdy("Sept 13, 2010") #b) Sept 13, 2010.
```

```
## [1] NA
```

Return NA because the Sept is not the default short of September

```
mdy("Sep 13, 2010.") #c) Sep 13, 2010.
```

```
## [1] "2010-09-13"
```

```
mdy("S 13, 2010.") #d) S 13, 2010. Comment on the month abbreviation needs.
```

```
## [1] NA
```

Return NA because the S is not the default short of September")

```
dmy("07-Dec-1941.") #e) 07-Dec-1941.
```

```
## [1] "1941-12-07"
```

```
mdy("1-5-1998")#f) 1-5-1998. Comment on why you might be wrong.
```

```
## [1] "1998-01-05"
```

This might be wrong because the month could be the day and vice versa

```
dmy("21-5-1998")#g) 21-5-1998. Comment on why you know you are correct.
```

```
## [1] "1998-05-21"
```

```
ymd_hm("2020-May-5 10:30 Am")
```

```
## [1] "2020-05-05 10:30:00 UTC"
```

```
ymd_hm("2020-May-5 10:30 Am", tz = "America/Los_Angeles")
```

```
## [1] "2020-05-05 10:30:00 PDT"
```

```
ymd_hm("2020-May-5 10:30 Am", tz = "America/Puerto_Rico")
```

```
## [1] "2020-05-05 10:30:00 AST"
```

Q2

```
birth_date <- ymd("1998-09-07")
current_date <- Sys.Date()
birthday_64th <- birth_date + years(64)
age_years <- ceiling(as.numeric(difftime(current_date, birth_date, units = "days") / 365))-1
next_birthday <- birth_date + years(age_years + 1)
days_until_next_birthday <- as.numeric(difftime(next_birthday, current_date, units = "days"))
months_until_next_birthday <- as.integer(days_until_next_birthday / 30)
days_remaining_until_next_birthday <- days_until_next_birthday %% 30

paste("birth_date", birth_date)
```

```
## [1] "birth_date    1998-09-07"
```

```
paste("current_date      ",current_date)
```

```
## [1] "current_date      2023-10-27"
```

```
paste("birthday_64th     ",birthday_64th)
```

```
## [1] "birthday_64th     2062-09-07"
```

```
paste("age_years         ",age_years)
```

```
## [1] "age_years         25"
```

```
paste("next_birthday     ",next_birthday)
```

```
## [1] "next_birthday     2024-09-07"
```

```
paste("days_until_next_birthday ",days_until_next_birthday)
```

```
## [1] "days_until_next_birthday    316"
```

```
paste("months_until_next_birthday ",months_until_next_birthday)
```

```
## [1] "months_until_next_birthday    10"
```

```
paste("days_remaining_until_next_birthday ",days_remaining_until_next_birthday)
```

```
## [1] "days_remaining_until_next_birthday    16"
```

Q 3

```
Arizona_time <- with_tz(ymd_hm("2015-05-08 15:00 ", tz = "America/Phoenix"), "America/Phoenix")
```

```
Auckland_time <- with_tz(Arizona_time, "Pacific/Auckland")
```

```
Arizona_time
```

```
## [1] "2015-05-08 15:00:00 MST"
```

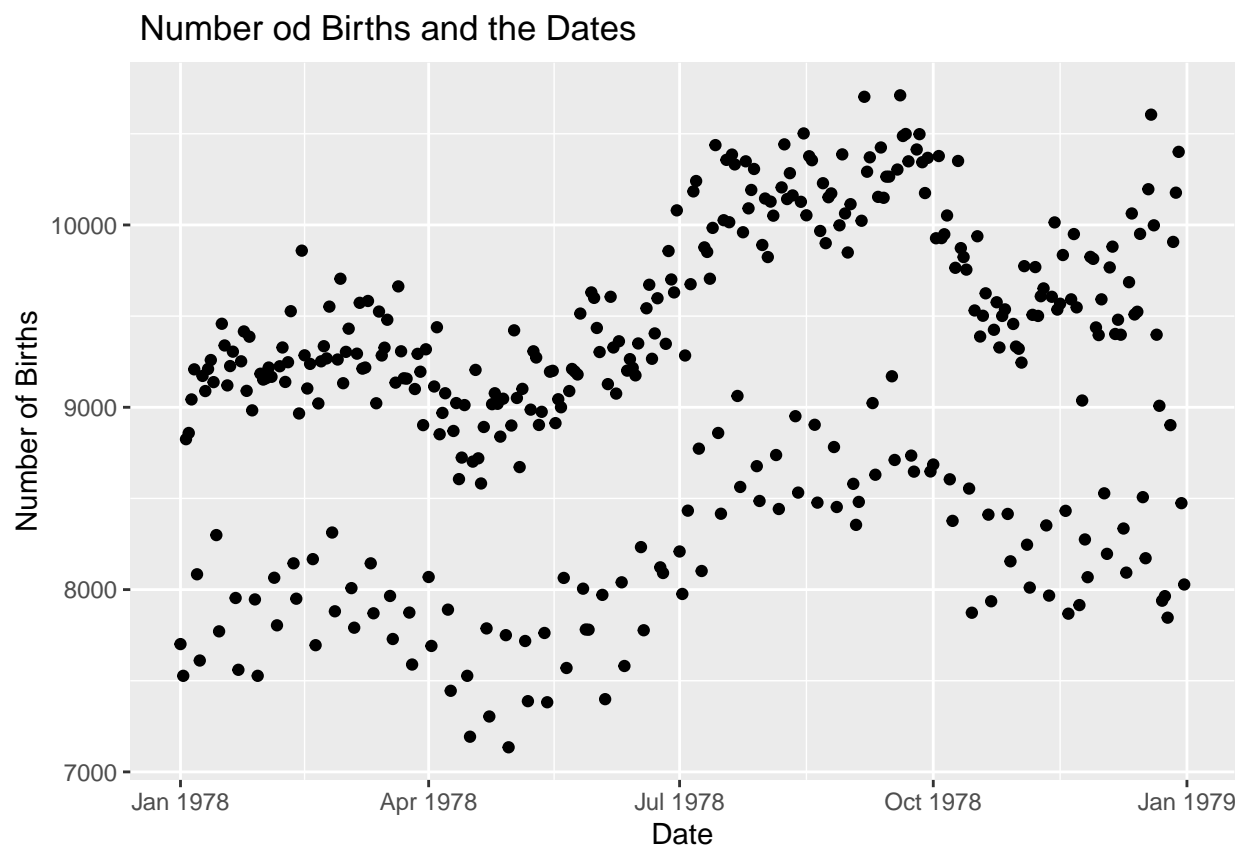
```
Auckland_time
```

```
## [1] "2015-05-09 10:00:00 NZST"
```

Q5

```
library(mosaicData)
library(ggplot2)
library(dplyr)

data(Births78)
Birth_78 <- subset(Births78, select = c(date, births))
ggplot(Birth_78, aes(x = date, y = births)) + geom_point() + labs(x = "Date", y = "Number of Births",
```



```
Birth_78 <- Birth_78 %>% mutate(dow = wday(date, label = TRUE))
ggplot(Birth_78, aes(x = date, y = births, color = dow)) + geom_point() + labs(x = "Date", y = "Number of Births",
```