Homework #10

Sang Doan

11/8/2020

Problem 1

An `Email` object contains the sender's and the recipient's names and the full file path.

```
'../raw_data/enron_dataset/arnold-j/sent/35.' %>%
  Email
```

```
## [1] "John Arnold"
## [2] "Margaret Allen"
## [3] "../raw_data/enron_dataset/arnold-j/sent/35."
```

Problem 2

```
Cuilla <- Employee('cuilla-m')
get_name.Employee(Cuilla)
```

```
## [1] "Martin Cuilla"
```

```
get_number_of_emails.Employee(Cuilla)
```

```
## [1] 16
```

```
get_email_filename.Employee(Cuilla, 8)
```

```
## [1] "../raw_data/enron_dataset/cuilla-m/sent/8."
```

Problem 3

Functions starting with `map` are from package `purrr`. They are generally faster than the `base::apply` family.

```
Emails <- c('jones-t', 'shackleton-s', 'sager-e', 'taylor-m', 'stclair-c') %>%
  map(function(.thisE) Employee(.thisE))

names <- map_chr(1:5, function(x) Emails[[x]]$name) %>%
  print
```

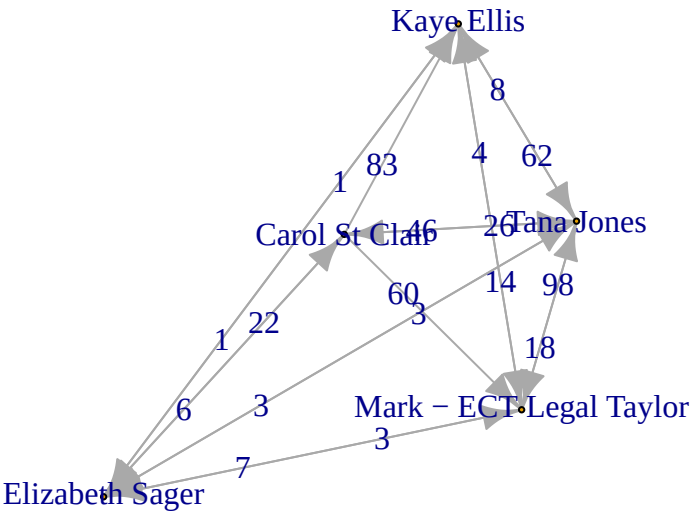
```
## [1] "Tana Jones"              "Kaye Ellis"
## [3] "Elizabeth Sager"         "Mark - ECT Legal Taylor"
## [5] "Carol St Clair"
```

```
mat <- sapply(1:5, function(x) {

  dat <- setDT(Emails[[x]]$emails) %>%
    transpose
```

```
  count <- map_int(1:5, function(x) dat[V2 == names[x], length(V2)])

  return(count)
}) %>%
  print
```

```
##      [,1] [,2] [,3] [,4] [,5]
## [1,]    1   62    3   98   26
## [2,]    8  181    1    4    0
## [3,]    3    1    1    7    6
## [4,]   18   14    3    1    0
## [5,]   46   83   22   60    0
```

```
gra <- graph_from_adjacency_matrix(mat, mode = 'directed', weighted = TRUE, diag = FALSE)
plot(
  gra,
  vertex.size = 0,
  vertex.label = names,
  edge.label = edge_attr(gra)$weight
)
```



Code

config.R

```
source('analysis.R')

library(tidyverse)
library(magrittr)
library(data.table)
library(igraph)
```

analysis.R

```
# Problem 1
Email <- function(filename) {
```

```
  email <- readLines(filename)
  out <- email[which(str_detect(email, 'X-From: |X-To: '))] %>% str_remove('X-From: |X-To: ')
  out[3] <- filename

  return(out) #Vector out: [1] sender, [2] recipient, [3] full path
}

from.Email <- function(em) em[1]
to.Email <- function(em) em[2]

# Problem 2
Employee <- function(directory_name) {
  allEmails <- list.files(
    paste('../raw_data/enron_dataset/', directory_name, sep = ''),
    full.names = T, recursive = T
  ) %>%
  str_sort(numeric = TRUE) %>%
    # Sort file names in correct order (1 then 2, not 1 then 10)
  purrr::map(function(.filename) Email(.filename))
    # Map a list of emails

  thisEmployee <- list(
    name = from.Email(allEmails[[1]]),
    emails = allEmails
  )

  return(thisEmployee)
}

get_number_of_emails.Employee <- function(e) length(e$emails)
get_email_filename.Employee <- function(e, i) e$emails[[i]][3]
get_name.Employee <- function(e) e$name
```