Exercise_1

2023-03-14

STEP 2. Open the file in RStudio as a text file to clean up for import + import with `read_csv()`

Import the file using read_csv() function:

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.2.2
```

```
my_connections <- read_csv("C:/Users/ulyan/OneDrive - McGill University/Documents/MMA/Winter II 2023/Org
```

```
## Rows: 318 Columns: 6
## -- Column specification --------------------------------------------------------
## Delimiter: ","
## chr (6): First Name, Last Name, Email Address, Company, Position, Connected On
##
## i Use `spec()` to retrieve the full column specification for this data.
## i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

STEP 3. Get the count of your contacts by their current employer + total count

Import library:

```
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.2.2
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.2 --
## v ggplot2 3.3.6      v dplyr   1.0.10
## v tibble  3.1.8      v stringr 1.4.1
## v tidyr   1.2.1      v forcats 0.5.2
## v purrr   0.3.4
```

```
## Warning: package 'dplyr' was built under R version 4.2.2
```

```
## Warning: package 'forcats' was built under R version 4.2.2
```

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

```
library(dplyr)
```

Count of contacts by current employer (excluding NA rows):

```
contacts_by_employer <- drop_na(my_connections, "Company")
contacts_by_employer_count <- nrow(contacts_by_employer)

# Print the contacts by employer
print(contacts_by_employer_count)
```

```
## [1] 298
```

Total count of contacts:

```
total_count <- nrow(my_connections)

# Print the total count
print(total_count)
```

```
## [1] 318
```

STEP 4. Create nodes and edges dataframes to use with igraph

Install packages:

```
library(tidygraph)
```

```
## Warning: package 'tidygraph' was built under R version 4.2.2
```

```
##
## Attaching package: 'tidygraph'
```

```
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.2.2
```

```
##
## Attaching package: 'igraph'
```

```
## The following object is masked from 'package:tidygraph':
##
##     groups
```

```
## The following objects are masked from 'package:dplyr':
##
##     as_data_frame, groups, union

## The following objects are masked from 'package:purrr':
##
##     compose, simplify

## The following object is masked from 'package:tidyr':
##
##     crossing

## The following object is masked from 'package:tibble':
##
##     as_data_frame

## The following objects are masked from 'package:stats':
##
##     decompose, spectrum

## The following object is masked from 'package:base':
##
##     union
```

Pre-process data

Drop the "Email Address" column:

```
my_connections <- my_connections %>% select(-"Email Address")
```

Drop any rows with missing values in the remaining columns:

```
my_connections <- my_connections %>% drop_na()
```

Create a tidygraph object from the connections data

```
my_connections_graph <- my_connections %>%
  as_tbl_graph(nodes = c("First Name", "Last Name", "Company", "Position"),
               edges = c("First Name", "Last Name"),
               node_key = "name")
```

Extract the nodes and edges dataframes from the tidygraph object

Note: 'nodes_df' first lists all given names and then all last names. This also causes issues with 'edges_df'. I couldn't figure out how to fix this.

```
nodes_df <- my_connections_graph %>%
  activate(nodes) %>%
  as_tibble()

edges_df <- my_connections_graph %>%
  activate(edges) %>%
  as_tibble()
```

Create a new edges dataframe with name values in place of indexes:

```
edges_df2 <- edges_df %>%
  mutate(from = nodes_df$name[from],
         to = nodes_df$name[to])
```

STEP 5. Plot the resulting network

Create an igraph object from the nodes and edges dataframes:

```
my_igraph <- graph_from_data_frame(d = edges_df2, vertices = nodes_df)
```

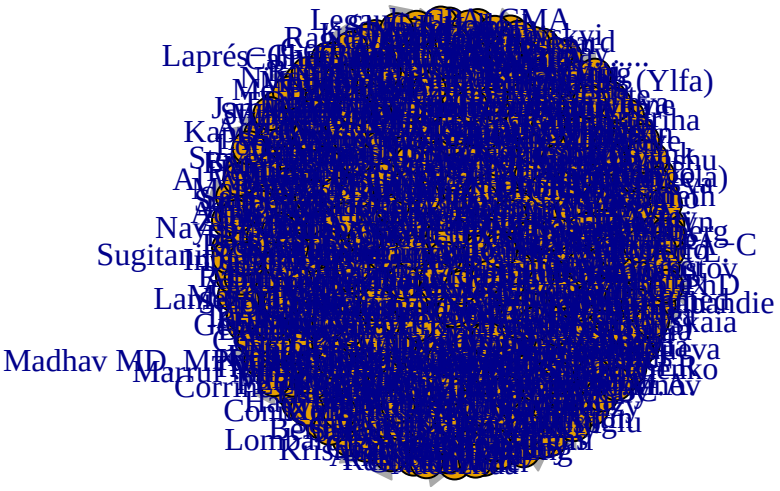
Plot the resulting network:

```
plot(my_igraph)
```

```
## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <f0>

## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <9f>

## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <87>

## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <ae>

## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <f0>

## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <9f>

## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <87>
```

```
## Warning in text.default(x, y, labels = labels, col = label.color, family
## = label.family, : conversion failure on 'Chouhan  ' in 'mbcsToSbcs': dot
## substituted for <b3>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : font metrics unknown for Unicode character U+1f1ee

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : font metrics unknown for Unicode character U+1f1f3

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <f0>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <9f>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <94>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <b8>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <e2>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <96>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <ab>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <ef>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <b8>

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : conversion failure on 'R  ' in 'mbcsToSbcs': dot substituted
## for <8f>
```

```
## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : font metrics unknown for Unicode character U+1f538

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : font metrics unknown for Unicode character U+25ab

## Warning in text.default(x, y, labels = labels, col = label.color, family =
## label.family, : font metrics unknown for Unicode character U+fe0f
```



Note: Because of the issues with 'nodes_df' and 'edges_df', plot is displayed incorrectly and looks very busy.