

Exercise 1

2024-03-19

```
library(igraph)
```

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

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

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

```
library(dplyr)
```

```
##  
## Attaching package: 'dplyr'
```

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

```
## The following objects are masked from 'package:stats':  
##  
## filter, lag
```

```
## The following objects are masked from 'package:base':  
##  
## intersect, setdiff, setequal, union
```

```
library(tidyr)
```

```
##  
## Attaching package: 'tidyr'
```

```
## The following object is masked from 'package:igraph':  
##  
## crossing
```

```
library(tidyverse)
```

```
## — Attaching core tidyverse packages — tidyverse 2.0.0 —
## ✓ forcats    1.0.0    ✓ readr      2.1.4
## ✓ ggplot2    3.5.0    ✓ stringr    1.5.0
## ✓ lubridate  1.9.3    ✓ tibble     3.2.1
## ✓ purrr      1.0.2
```

```
## — Conflicts — tidyverse_conflicts() —
## ✖ lubridate::%--%()      masks igraph::%--%()
## ✖ tibble::as_data_frame() masks dplyr::as_data_frame(), igraph::as_data_frame()
## ✖ purrr::compose()      masks igraph::compose()
## ✖ tidyr::crossing()     masks igraph::crossing()
## ✖ dplyr::filter()       masks stats::filter()
## ✖ dplyr::lag()          masks stats::lag()
## ✖ purrr::simplify()     masks igraph::simplify()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```
library(tidygraph)
```

```
##
## Attaching package: 'tidygraph'
##
## The following object is masked from 'package:igraph':
##
##     groups
##
## The following object is masked from 'package:stats':
##
##     filter
```

```
library(RColorBrewer)
library(purrr)
library(ggplot2)
library(ggraph)
library(oaqc)
```

```
Connections <- read.csv("~/Desktop/McGill/Org Network Analysis/Connections.csv")
```

```
Connections <- Connections %>%
  mutate(label = str_c(
    `First.Name`,
    str_sub(`Last.Name`, 1, 1),
    row_number()
  )) %>%
  select(label, Company) %>%
  drop_na()
```

```
Connections %>%
  count()
```

```
##      n
## 1 1475
```

```
contact_count_by_company <- Connections %>%
  group_by(Company) %>%
  count(sort = TRUE) %>%
  filter(n > 10)

contact_count_by_company <- contact_count_by_company %>%
  filter(Company != "" & !is.na(Company))

contact_count_by_company
```

```
## # A tibble: 10 × 2
## # Groups:   Company [10]
##   Company                                n
##   <chr>                                <int>
## 1 Lahore University of Management Sciences    29
## 2 McGill University – Desautels Faculty of Management 18
## 3 Midas Safety                               17
## 4 McGill University                          13
## 5 EY                                           12
## 6 BAT                                           11
## 7 Freelance                                   11
## 8 HBL – Habib Bank Limited                   11
## 9 Self-employed                             11
## 10 Unilever                                  11
```

```
con_df <- Connections %>%
  filter(Company %in% contact_count_by_company$Company) %>%
  group_by(Company) %>%
  summarise(label_combinations = list(combn(label, 2, simplify = FALSE)), .groups =
'drop') %>%
  unnest(label_combinations) %>%
  transmute(
    from = map_chr(label_combinations, 1),
    to = map_chr(label_combinations, 2),
    Company = Company
  )

con_df
```

```
## # A tibble: 1,114 × 3
##   from      to      Company
##   <chr>    <chr>    <chr>
## 1 AdrianaI174 Fasih UrR184 BAT
## 2 AdrianaI174 VirginiaL254 BAT
## 3 AdrianaI174 RameenR281 BAT
## 4 AdrianaI174 MateoM293 BAT
## 5 AdrianaI174 AliyanR324 BAT
## 6 AdrianaI174 MuhammadA854 BAT
## 7 AdrianaI174 RoshnikN1074 BAT
## 8 AdrianaI174 RijaaF1121 BAT
## 9 AdrianaI174 ZohaF1257 BAT
## 10 AdrianaI174 Muhammad AmerF1336 BAT
## # i 1,104 more rows
```

```
set.seed(591)
```

```
graph_data <- con_df %>%
  slice_sample(prop = 0.4) %>%
  as_tbl_graph(directed = FALSE)
```

```
graph_data
```

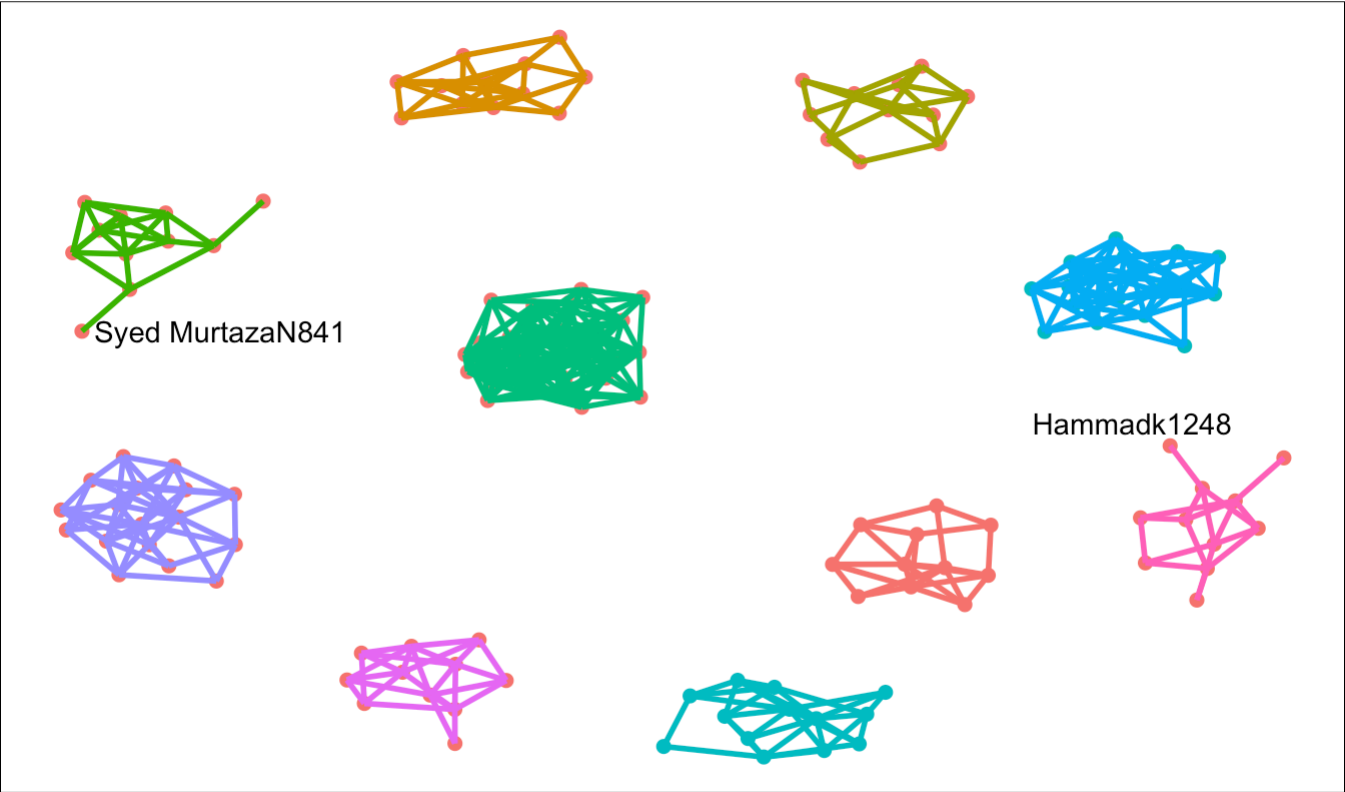
```
## # A tbl_graph: 144 nodes and 445 edges
## #
## # An undirected simple graph with 10 components
## #
## # Node Data: 144 × 1 (active)
##   name
##   <chr>
## 1 AlexanderD31
## 2 Sohaib BinM1117
## 3 MohsinB344
## 4 MohammadD740
## 5 HamzaB233
## 6 JavadN294
## 7 SonaliR166
## 8 HassanT502
## 9 ElizabethH296
## 10 YiyiY186
## # i 134 more rows
## #
## # Edge Data: 445 × 3
##   from    to Company
##   <int> <int> <chr>
## 1     1     60 McGill University
## 2     2     77 Lahore University of Management Sciences
## 3     3     11 Lahore University of Management Sciences
## # i 442 more rows
```

```
mcgill_names <- (con_df %>%  
  filter(Company %in% c(  
    "McGill University",  
    "McGill University - Desautels Faculty of Management"  
  )) %>%  
  select(from, to) %>%  
  pivot_longer(cols = c(from, to)) %>%  
  distinct() %>%  
  pull(value))
```

```
# Generate the layout and plot  
graph_layout <- ggraph(graph_data, layout = 'fr') + # replace 'fr' with your chosen  
layout  
  geom_node_point(size = 2, aes(color = name %in% mcgill_names)) +  
  geom_node_text(aes(label = name), repel = TRUE, max.overlaps = 2, check_overlap = T  
RUE) +  
  geom_edge_link(aes(color = Company), show.legend = TRUE, width = 1) +  
  theme_void() +  
  theme(legend.position = "bottom", panel.border = element_rect(fill = NA)) +  
  labs(  
    title = "LinkedIn Connection Network"  
  )  
  
# Explicitly print the plot  
print(graph_layout)
```

```
## Warning: ggrepel: 142 unlabeled data points (too many overlaps). Consider  
## increasing max.overlaps
```

LinkedIn Connection Network



- Company
- | | | | |
|-----|--------------------------|--|---------------|
| BAT | Freelance | Lahore University of Management Sciences | McGill Univer |
| EY | HBL - Habib Bank Limited | McGill University | Midas Safety |