orgb e1

2024-03-19

Data

```
library(readr)
connections <- read.csv("~/Desktop/orgb/Connections.csv")
#drop email column because of missing value
connections <- connections %>% select(-`Email.Address`)
```

Contacts count by current employer/ contact of each company count

```
contacts_count <- connections %>%
  group_by(Company) %>%
  summarise(Count = n())

#print number
print(contacts_count)
```

```
## # A tibble: 520 x 2
##    Company                           Count
##    <chr>                             <int>
##  1 ""                                   41
##  2 " (R)"                                1
##  3 "5Y Capital"                          1
##  4 "ADY SERVICES LTD"                    1
##  5 "AIESEC"                              1
##  6 "AIESEC in Ottawa"                    1
##  7 "AIG"                                 2
##  8 "ALDO Group"                          2
##  9 "AMERICAN EAGLE OUTFITTERS INC."      1
## 10 "AN Design Communications"            1
## # i 510 more rows
```

#Total Contacts Count

```
total_count <- nrow(connections)
print(paste("Total contacts:", total_count))
```

```
## [1] "Total contacts: 850"
```

#Create Nodes and Edges

```r
# node
nodes <- connections %>%
  transmute(id = row_number(),
            label = paste(`First.Name`, substr(iconv(`Last.Name`, "UTF-8", "ASCII//TRANSLIT//IGNORE"),
```

```r
# edges
edges <- connections %>%
  mutate(id = row_number()) %>%  # Generate a id for each row
  select(id, Company) %>%
  group_by(Company) %>%
  filter(n() > 1) %>%
  summarise(pairs = list(combn(id, 2, simplify = FALSE))) %>%
  unnest(pairs) %>%
  transmute(from = map_dbl(pairs, 1), to = map_dbl(pairs, 2))
```

```r
library(ggraph)
```

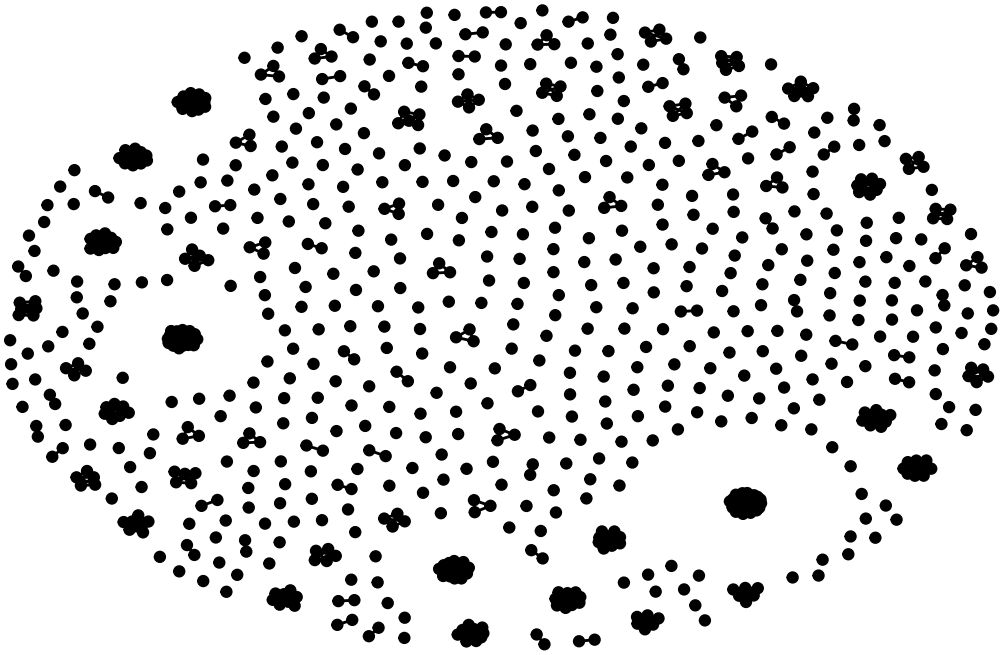
```
## Loading required package: ggplot2
```

```r
# tbl_graph object from nodes and edges
network <- tbl_graph(nodes = nodes, edges = edges)

# graph using tidygraph and visualize with ggraph
ggraph(network, layout = 'fr') +
  geom_edge_link() +
  geom_node_point() +
  geom_node_text(aes(label = label), repel = TRUE) +
  theme_graph()
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on ’<e6><b3> _’ in ’mbcsToSbcs’: dot substituted for <b3>

## Warning: ggrepel: 850 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```

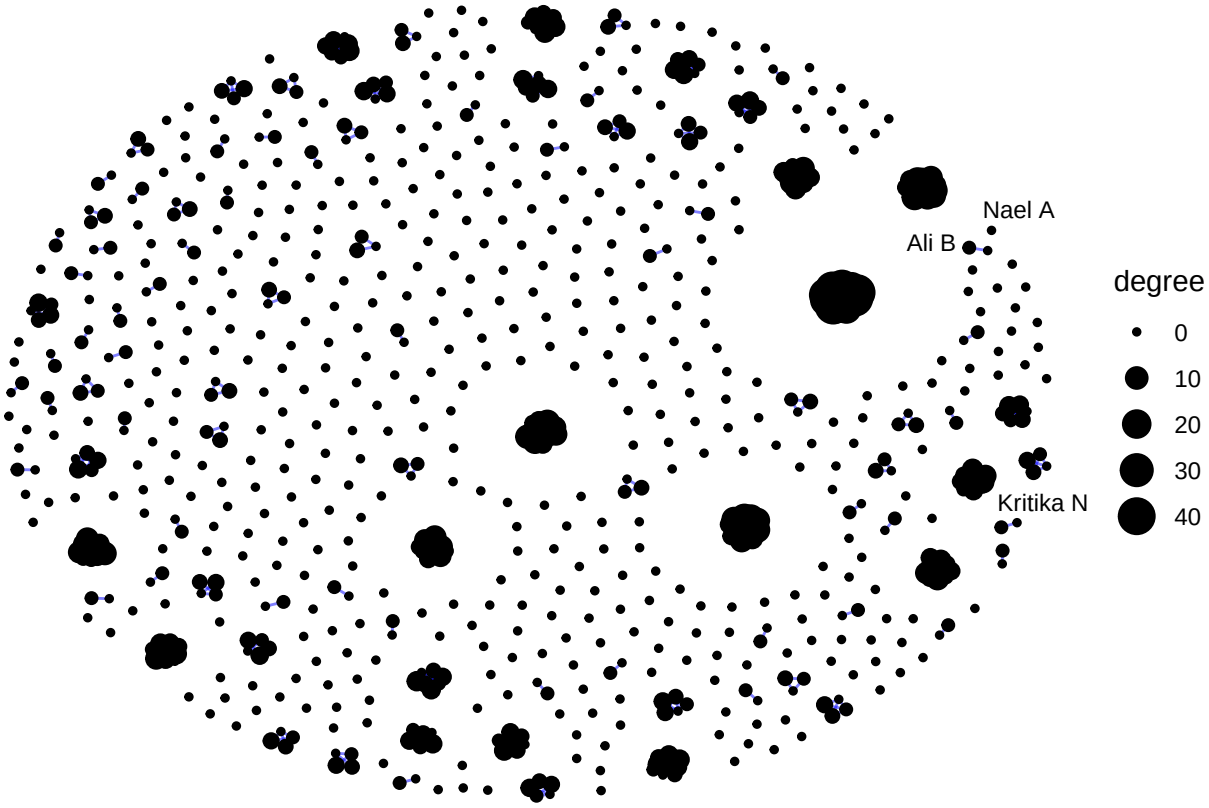


```r
# degree column that counts the number of connections each node has
network <- network %>%
  mutate(degree = centrality_degree())

# Visualize the network with node size representing the degree
ggraph(network, layout = 'fr') +
  geom_edge_link(color = 'blue', alpha = 0.5) +
  geom_node_point(aes(size = degree)) +
  geom_node_text(aes(label = label), repel = TRUE, size = 3) +
  theme_void()
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <ef>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on 'Jingwen (Hennie<ef><bc> H' in 'mbcsToSbcs': dot
## substituted for <bc>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>
```

```
## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <e6>

## Warning in grid.Call(C_textBounds, as.graphicsAnnot(x$label), x$x, x$y, :
## conversion failure on '<e6><b3> _' in 'mbcsToSbcs': dot substituted for <b3>

## Warning: ggrepel: 847 unlabeled data points (too many overlaps). Consider
## increasing max.overlaps
```



#Mcgill Network

```
# Filter Mcgill or Desautels networks
filtered_connections <- connections %>%
  filter(grepl("McGill|Desautels", Company))
```

```
## Warning: There were 6 warnings in ‘filter()‘.
## The first warning was:
## i In argument: ‘grepl("McGill|Desautels", Company)‘.
## Caused by warning in ‘grepl()‘:
## ! unable to translate ’___<e7><b4>Ipsos China’ to a wide string
## i Run ‘dplyr::last_dplyr_warnings()‘ to see the 5 remaining warnings.
```

```
# nodes for Mcgill nwtwork
filtered_nodes <- filtered_connections %>%
  transmute(id = row_number(),
            label = paste(`First.Name`, substr(iconv(`Last.Name`, "UTF-8", "ASCII//TRANSLIT//IGNORE"),


# edges for Mcgill contacts
filtered_edges <- filtered_connections %>%
  mutate(id = row_number()) %>%
  select(id, Company) %>%
  group_by(Company) %>%
  filter(n() > 1) %>%
  summarise(pairs = list(combn(id, 2, simplify = FALSE))) %>%
```

```
  unnest(pairs) %>%
  transmute(from = map_dbl(pairs, 1), to = map_dbl(pairs, 2))

#tbl_graph object from filtered nodes and edges
filtered_network <- tbl_graph(nodes = filtered_nodes, edges = filtered_edges)

# degree column that counts the number of connections each node has
filtered_network <- filtered_network %>%
  mutate(degree = centrality_degree())

# Visualize the Mcgill network
ggraph(filtered_network, layout = 'fr') +
  geom_edge_link(color = 'blue', alpha = 0.5) +
  geom_node_point(aes(size = degree)) +
  geom_node_text(aes(label = label), repel = TRUE, size = 3) +
  theme_void()
```