Untitled

Read the data

```
data = read.csv('Connections.csv', encoding = 'UTF-8')
```

```
library(dplyr)
```

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

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

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

```
data %>% group_by(Company) %>% count() %>% arrange(desc(n))
```

```
## # A tibble: 792 x 2
## # Groups:   Company [792]
##    Company                                                  n
##    <chr>                                                <int>
##  1 "Belcorp"                                               64
##  2 "McGill University - Desautels Faculty of Management"   23
##  3 "Banco de Crédito BCP"                                  19
##  4 "Interbank"                                             19
##  5 ""                                                      18
##  6 "Scotiabank"                                            16
##  7 "Entel Perú"                                            15
##  8 "Rappi"                                                 13
##  9 "Alicorp"                                               12
## 10 "EY"                                                    12
## # ... with 782 more rows
```

```
#library(tidygraph)
#library(ggraph)

# not being used for now
```

```
names(data)
```

```
data$Last_name_initial = substr(data$Last.Name, 1,1)


data = data %>% mutate(
  name = paste(First.Name, Last_name_initial, sep = " ")
)
```

```r
data$index <- 1:nrow(data)

unique(data$Company)

df1 <- data.frame()

# iterate through each company and generate the edges
for (company in unique(data$Company) ){

company_edges = data %>% filter(Company==company) %>% pull(index)

edges = expand.grid(company_edges, company_edges)

# remove the connections with themselves
edges = edges %>% filter(Var1 != Var2)

# append to main df1
df1 <- rbind(df1, edges)

#print(dim(df1))

}

# nodes = data$index
# edges = df1

df1$temp <- apply(df1, 1, function(x) paste(sort(x), collapse=""))

df1 = df1[!duplicated(df1$temp), 1:2] # remove duplicate (each combination appears once)
```

```r
library(tidygraph)
```

```
## Warning: package 'tidygraph' was built under R version 4.1.3
```

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

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

```r
#install.packages('network')
library(network)
```

```
## Warning: package 'network' was built under R version 4.1.3
```

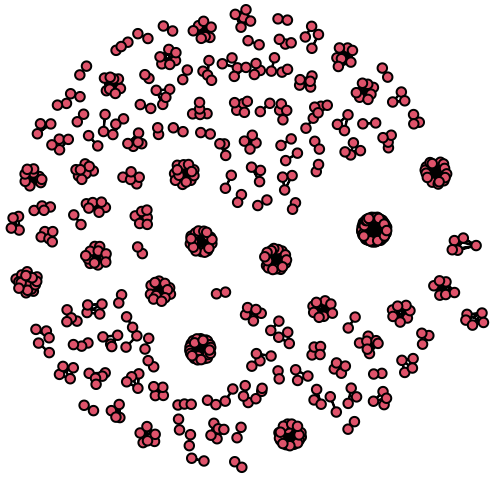
```
##
## 'network' 1.17.1 (2021-06-12), part of the Statnet Project
## * 'news(package="network")' for changes since last version
## * 'citation("network")' for citation information
## * 'https://statnet.org' for help, support, and other information
```

```r
linkedin_network <- network(df1, attr = data,  matrix.type = "edgelist",directed=FALSE)

linkedin_network
```

```
##  Network attributes:
##   vertices = 635
##   directed = FALSE
##   hyper = FALSE
##   loops = FALSE
##   multiple = FALSE
##   bipartite = FALSE
##   total edges= 3860
##     missing edges= 0
##     non-missing edges= 3860
##
##  Vertex attribute names:
##     vertex.names
##
##  Edge attribute names not shown
```

```
plot(linkedin_network )
```



```
#to do: print names of nodes
```