class_4_lab

Thomas Zwiller

2025-03-26

1. Loading the igraph package

```
#Loading the libraries I need
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.4.1
```

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

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

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

```
library(ggraph)
```

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

2. Loading in the ego-centric data

```
nodes <- read.csv('/Users/TomTheIntern/Desktop/Mendoza/Mod 4/Networks/Lab 2/nodelist.csv')
summary(nodes)
```

```
##        ID            Name                Age           Gender
##  Min.   : 1.00   Length:12          Min.   :21.00   Length:12
##  1st Qu.: 3.75   Class :character   1st Qu.:23.00   Class :character
##  Median : 6.50   Mode  :character   Median :36.50   Mode  :character
##  Mean   : 6.50                      Mean   :38.00
##  3rd Qu.: 9.25                      3rd Qu.:45.75
##  Max.   :12.00                      Max.   :65.00
```

```
edges <- read.csv('/Users/TomTheIntern/Desktop/Mendoza/Mod 4/Networks/Lab 2/edgelist.csv')
summary(edges)
```

```
##     ego_num         alter_num          ego                alter
##  Min.   : 1.000   Min.   : 1.000   Length:40          Length:40
##  1st Qu.: 2.750   1st Qu.: 2.750   Class :character   Class :character
##  Median : 5.000   Median : 5.000   Mode  :character   Mode  :character
##  Mean   : 5.575   Mean   : 5.575
##  3rd Qu.: 9.000   3rd Qu.: 9.000
##  Max.   :12.000   Max.   :12.000
##     type              strength
##  Length:40          Min.   :1.00
##  Class :character   1st Qu.:2.00
##  Mode  :character   Median :4.00
##                     Mean   :3.45
##                     3rd Qu.:4.25
##                     Max.   :5.00
```

3 Creating the igraph object

```
#making the directed igraph
net <- graph_from_data_frame(edges, directed = T, vertices = nodes)
net
```

```
## IGRAPH 9d9e9ba DN-- 12 40 --
## + attr: name (v/c), Name (v/c), Age (v/n), Gender (v/c), ego (e/c),
## | alter (e/c), type (e/c), strength (e/n)
## + edges from 9d9e9ba (vertex names):
##  [1] 1 ->2  2 ->1  1 ->5  5 ->1  1 ->3  3 ->1  1 ->4  4 ->1  1 ->6  6 ->1
## [11] 1 ->7  7 ->1  1 ->10 10->1  2 ->4  4 ->2  2 ->3  3 ->2  4 ->3  3 ->4
## [21] 4 ->6  6 ->4  4 ->5  5 ->4  3 ->5  5 ->3  10->9  9 ->10 10->11 11->10
## [31] 10->12 12->10 9 ->12 12->9  11->12 12->11 6 ->7  7 ->6  7 ->8  8 ->7
```

```
#making the un-directed igraph
un_net <- graph_from_data_frame(edges, directed = F, vertices = nodes)
un_net
```

```
## IGRAPH 577a655 UN-- 12 40 --
## + attr: name (v/c), Name (v/c), Age (v/n), Gender (v/c), ego (e/c),
## | alter (e/c), type (e/c), strength (e/n)
## + edges from 577a655 (vertex names):
##  [1] 1 --2  1 --2  1 --5  1 --5  1 --3  1 --3  1 --4  1 --4  1 --6  1 --6
## [11] 1 --7  1 --7  1 --10 1 --10 2 --4  2 --4  2 --3  2 --3  3 --4  3 --4
## [21] 4 --6  4 --6  4 --5  4 --5  3 --5  3 --5  9 --10 9 --10 10--11 10--11
## [31] 10--12 10--12 9 --12 9 --12 11--12 11--12 6 --7  6 --7  7 --8  7 --8
```

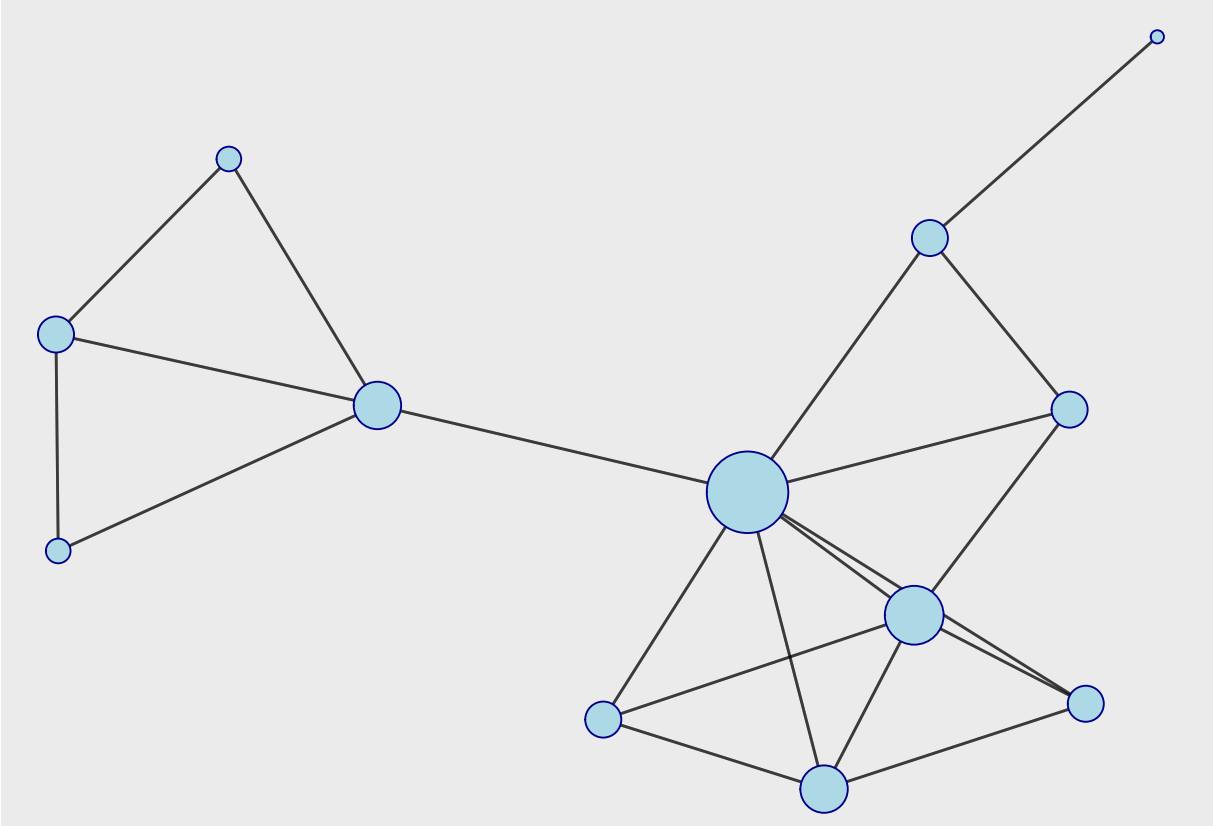
4. Choose two centrality measures and apply them to your network I am using Degree and Eigenvector as my two centrality measures

5. Visualize your network (one graph for each measure) with node size representing your chosen measures

Degree

```
ggraph(net) +
  geom_edge_link0(color = "black", alpha = .5) +
  geom_node_point(fill = "lightblue", color = "darkblue", shape = 21,
                  size = igraph::degree(net, mode = "all"))
```

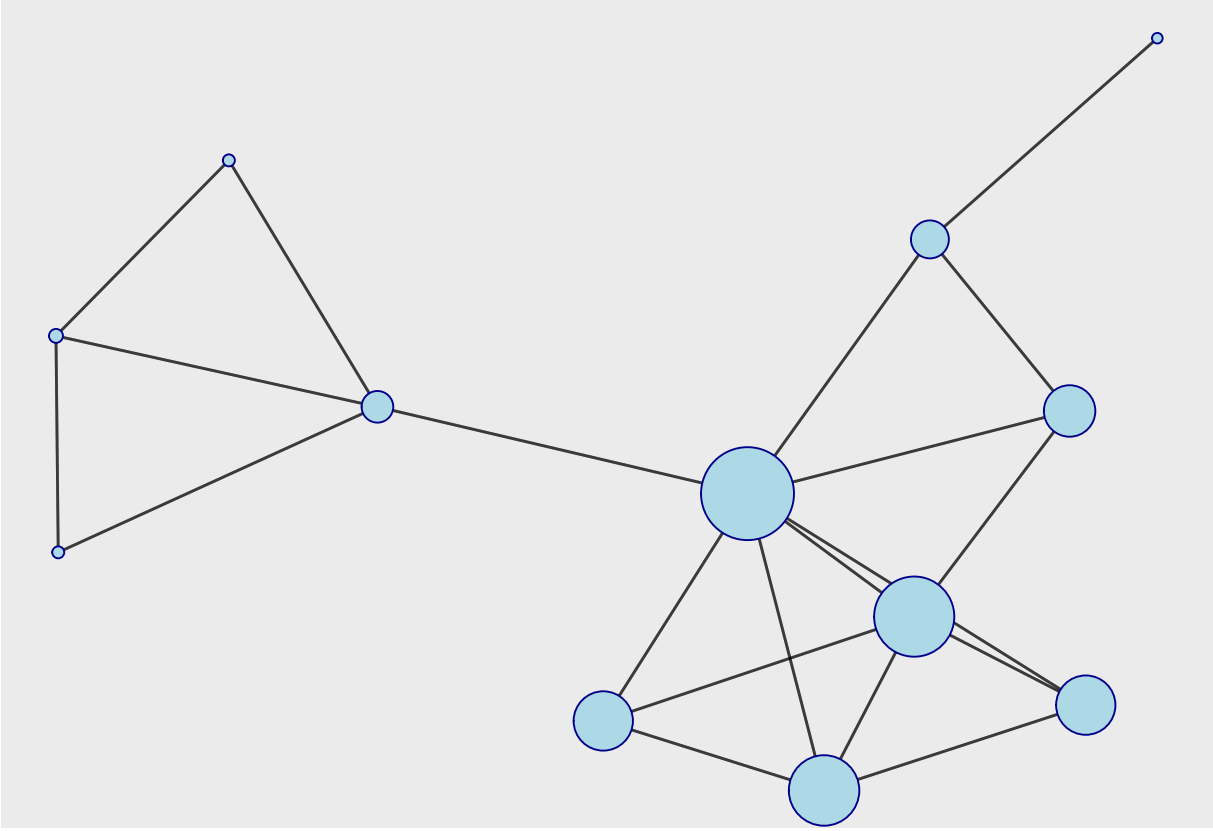
```
## Using "stress" as default layout
```



Eigenvector

```
ggraph(un_net) +
  geom_edge_link(color = "black", alpha = .5) +
  geom_node_point(fill = "lightblue", color = "darkblue", shape = 21,
                  size = igraph::eigen_centrality(un_net)$vector*16)
```

```
## Using "stress" as default layout
```



6. Describe what the measure tells you about the people in your network

What Degree tells me about the people in my network Unsurprisingly, I had the most degrees, which made my node the biggest. Generally, the different communities were relatively the same size:

the football network (my co-workers) were all consistent because they were connected. my girlfriend's family generally tended to be smaller, as they only had 2-3 connections compared to football having 4-5

my res-life coworkers were the smallest, with my co-workers girlfriend having just one connection, making it the smallest, and about as close to an isolet as possible.

What Eigenvector tells me about the people in my network

The Eigenvector measure looks relatively similar to the degree measure, though there are a few key differences. My girlfriend has a much bigger circle in the nodes, but because most of her connections were to other relatively un-connected nodes, she had a smaller representation.

Comparatively, the football network was actually bigger because they all knew me, and each other, which helped improve their score.

My res-life co-workers were relatively smaller than their node score, however my co-worker who works with the ND bad (and would theoretically know Marcus Freeman) had a bit of a higher score.