M2 Assignment 1

```
library(data.table)
library(tidyverse)
library(magrittr)
library(tidygraph)
library(ggraph)
library(igraph)
```

First just loading the data.

```
advice <- read.table("https://raw.githubusercontent.com/SDS-AAU/SDS-master/master/00_data/network_krackhard/Krack-High-Tec-edgelist-Advice.txt", header=FALSE)
friendship <-  read.table("https://raw.githubusercontent.com/SDS-AAU/SDS-master/master/00_data/network_krackhard/Krack-High-Tec-edgelist-Friendship.txt", header= FALSE)
reportsto <- read.table("https://raw.githubusercontent.com/SDS-AAU/SDS-master/master/00_data/network_krackhard/Krack-High-Tec-edgelist-ReportsTo.txt", header= FALSE)
attributes <- read_csv("https://raw.githubusercontent.com/SDS-AAU/SDS-master/master/00_data/network_krackhard/Krack-High-Tec-Attributes.csv")
```

Before starting answering the assignment, the columns are renamed in the following way - explained in an example with the "advice" dataset. The first column in the "advice" dataset contains the ID of 21 persons and each person ID occours 21 times in a raw. The second column contains the same ID numbers but this time from 1:21 repeatedly - the persons in this column is the persons of which the persons in column one can seek advice from. In this way every person "meets" all other persons in a row at some point in the dataset. Therefore, the columns are renamed as: column one =" from" and column two= "to". The last column is an indicator variable that takes the value 1 if the person in column 1 has seeked advice from the person in column 2, which is why it is renamed as "presence". The idea for the rest of the dataset is the same and therefore no further elaboration is made.

At last the "attributes" dataset contain information about the managers, where:
ID is the numeric ID of the manager
AGE is the age of the managers (in years)
TENURE is the length of service or tenure (in years)
LEVEL is the level in the corporate hierarchy (coded 1,2 and 3; 1 = CEO, 2 = Vice President, 3 = manager)
DEPT is the department (coded 1,2,3,4 with the CEO in department 0, i.e. not in a department)

```
advice <- advice %>% rename("from" = "V1", "to" = "V2", "presence"="V3")
friendship <- friendship %>% rename("from" = "V1", "to" = "V2", "presence"="V3")
reportsto <- reportsto %>% rename("from" = "V1", "to" = "V2", "presence"="V3")
```

Throughout the assignment the CEO (department 0) is included in the departments since I find it interesting to explore his/hers role in the network.

Question 1: Creating a network

Constructing a tibble and graph object, where the edge-list only consist of the connections (i.e. where presence=1).

```
g_advice <- advice %>%
  filter(presence==1) %>% #filtering for the observations where no edge occours
  select(-presence) %>%#After the filtering above is made, this can be removed sin
ce all values in this variable =1 now.
  as_tbl_graph(directed = TRUE)

g_friendship <- friendship %>%
  filter(presence==1) %>%
  select(-presence) %>%
  as_tbl_graph(directed = TRUE)

g_reportsto <- reportsto %>%
  filter(presence==1) %>%
  select(-presence) %>%
  as_tbl_graph(directed = TRUE)
```

For convenience the person ID in the node characteristics in the dataset "attributes" is renamed from "ID" to "name" in order to merge dataset together nicely later in the assignment.

```
attributes <- attributes %>% rename("name"="ID")
```

Now the node characteristics are attached to the node-list with a mutuate function for all 3 node-lists:

```
set.seed(1337)
g_advice <- g_advice %N>%
  mutate(name=name %>% as.numeric()) %>%
  left_join(attributes, by="name")

g_friendship <- g_friendship %N>%
  mutate(name=name %>% as.numeric()) %>%
  left_join(attributes, by="name")

g_reportsto <- g_reportsto %N>%
  mutate(name=name %>% as.numeric()) %>%
  left_join(attributes, by="name")
```

Taking a look of the edge- and nodes-list:

```
g_advice
```

```
## # A tbl_graph: 21 nodes and 190 edges
## #
## # A directed simple graph with 1 component
## #
## # Node Data: 21 x 5 (active)
##    name   AGE TENURE LEVEL  DEPT
##   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     1    33   9.33     3     4
## 2     2    42  19.6      2     4
## 3     3    40  12.8      3     2
## 4     4    33   7.5      3     4
## 5     5    32   3.33     3     2
## 6     6    59  28        3     1
## # … with 15 more rows
## #
## # Edge Data: 190 x 2
##    from    to
##   <int> <int>
## 1     1     2
## 2     1     4
## 3     1     8
## # … with 187 more rows
```

```
g_friendship
```

```
## # A tbl_graph: 21 nodes and 102 edges
## #
## # A directed simple graph with 1 component
## #
## # Node Data: 21 x 5 (active)
##    name   AGE TENURE LEVEL  DEPT
##   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     1    33   9.33     3     4
## 2     2    42  19.6      2     4
## 3     3    40  12.8      3     2
## 4     4    33   7.5      3     4
## 5     5    32   3.33     3     2
## 6     6    59  28        3     1
## # … with 15 more rows
## #
## # Edge Data: 102 x 2
##    from    to
##   <int> <int>
## 1     1     2
## 2     1     4
```

```
## 3     1     7
## # … with 99 more rows

g_reportsto

## # A tbl_graph: 21 nodes and 20 edges
## #
## # A rooted tree
## #
## # Node Data: 21 x 5 (active)
##    name   AGE TENURE LEVEL  DEPT
##   <dbl> <dbl>  <dbl> <dbl> <dbl>
## 1     1    33   9.33     3     4
## 2     2    42  19.6      2     4
## 3     3    40  12.8      3     2
## 4     4    33   7.5      3     4
## 5     5    32   3.33     3     2
## 6     6    59  28        3     1
## # … with 15 more rows
## #
## # Edge Data: 20 x 2
##    from    to
##   <int> <int>
## 1     1     2
## 2     2    21
## 3     3    13
## # … with 17 more rows
```

It is seen that the edge-list only consists of the observation where an edge exists. At the same time the node characteristics have been attached to the node-lists. Note to reader: All interesting visualizations is provided in question 4.

Question 2: Analysis

A: Network level characteristics

Here different measures of the network will be investigated such as density, transistivity and so on.

Density:

The density in a network measures all the actual connections (the ones that actullay exsists in the network) of all the possible connections (the ones that could have been).

```
edge_density(g_advice)

## [1] 0.452381
```

```
edge_density(g_friendship)
```

```
## [1] 0.2428571
```

```
edge_density(g_reportsto)
```

```
## [1] 0.04761905
```

From this it can be concluded that the managers have a rather large amount of people from whom they seek advice but have a smaller amout of people how they labelled as friends. At the same time the amout of people they reports back to is very small and in generel this structure also seems to be reasonable just intuitively.

Transistivity:

The transistivity measures the amout of closed triangles, in this way it is possible to make an indication of how clustered the network seems to be which is way it is called the clustering coefficient.

```
transitivity(g_advice, type ="global")
```

```
## [1] 0.7345088
```

```
transitivity(g_friendship, type ="global")
```

```
## [1] 0.4714946
```

```
transitivity(g_reportsto, type ="global")
```

```
## [1] 0
```

First looking at the advice network, there is a large tendency to clusters (a measure of 0.73), which means that the same people seek advice with each orther in clusters. The same is true in a smaller extent for the friendship network while the report network has no closed triangles.

Reciprocity:

The reciprocity measures how likely it is that an edge in one direction is being reciplicatied in the other direction.

```
reciprocity(g_advice)
```

```
## [1] 0.4736842
```

```
reciprocity(g_friendship)
```

```
## [1] 0.4509804
```

```
reciprocity(g_reportsto)
```

```
## [1] 0
```

These values show that in all most half of the cases a person which seeks advice with another person, will also be sought out to give advice the orther way around. The same is true for friends i.e. if a person calls another person a friend, it is in half of the cases likely that the first person sees the other person as a friend too. At last there is no chance that the person one person reports to, also reports back.

In generel it seems as if the advice and friendship network tend to be more dynamic in the way that it involves more people which has connected crisscrossing. Where the reporting network on the opposite tends to go one way.

The remaining questions in 2:

The reciprocity of advice and friendship: The reciprocity of advice would intuitively make sense in the way, that if you seek advice with a person, it most be assumed that this person knowns more about the subject that you or is maybe higher placed than you. Therefore, the person you sought adviced with is just as likely to seek advice from you (probably on another subject) than from another mananger in the firm. Looking at the reciprocity of the friendship network a greater skepticism/wonder arises. The value of 0.45 do not seems reasonably, but at the same time the word "friends" are highly subjective, and the definition can therefore vary from person to person, which is probably why the reciprocity is so low.

Friends of the friends: In order to answer the question of rather the friends of your friends also tends to be your friend the transistivity measure is used because it provide the amout of closed triangles in a network. The measure for friendship is calculated earlier as 0.4714946, which means that in almost half of the cases the friend of your friend is also your friend.

The likelihood of being in a friendship or advice- seeking relationship? Here the measure of density is used because this is a measure of actual connections over possible connections. The higher the value the more connections and thereby a bigger tendency to use the network. Earlier the density of the advice and friendship network was calculated as 0.45238 and 0.2428571 respectively which is why it can be concluded that people are more likely to use the advice seeking relationship than friendship.

B: Node level characteristics

The most popular in the network: In order to find the most popular in the two requested networks the centrality degree is calculated since it is an unweighted network. The centrality degree is given the argument "in" since the interst is being a friend and being the person, which gives the advice.

```
g_advice <- g_advice %N>%
  mutate(centrality_dgr = centrality_degree(weights = NULL, mode="in"))
g_friendship <- g_friendship %N>%
  mutate(centrality_dgr = centrality_degree(weights = NULL, mode="in"))
```

Just arranging the table descendig for advice.

```
bind_cols(g_advice %N>%
            select(name, centrality_dgr, DEPT, LEVEL) %>%
            arrange(desc(centrality_dgr)) %>%
            as_tibble()) %>%
head()
```

```
## # A tibble: 6 x 4
##    name centrality_dgr  DEPT LEVEL
##   <dbl>          <dbl> <dbl> <dbl>
## 1     2             18     4     2
## 2    18             15     3     2
## 3    21             15     1     2
## 4     1             13     4     3
## 5     7             13     0     1
## 6    11             11     3     3
```

In the table it can be seen that it is the person which the ID=2, working in department 4 and being a VP, that is the most popular person in the advice given network i.e. the person most pepole seek advice from, here 18 people.

The same is done for the friendship network.

```
bind_cols(g_friendship %N>%
            select(name, centrality_dgr, DEPT, LEVEL) %>%
            arrange(desc(centrality_dgr)) %>%
            as_tibble()) %>%
head()
```

```
## # A tibble: 6 x 4
##    name centrality_dgr  DEPT LEVEL
##   <dbl>          <dbl> <dbl> <dbl>
## 1     2             10     4     2
## 2     1              8     4     3
## 3    12              8     1     3
## 4     5              6     2     3
## 5    11              6     3     3
## 6    17              6     1     3
```

This is the same person as above, ID=2, which 10 people labelling him/her to be a friend. But the second to the fifth most popular persons are managers.

Are managers in higher hirarchy more popular as friend, and advice giver? This is answered by the use of centrality degree calculated above. By grouping by level and taking the mean of the centrality degree over the corporate levels, the following two tables appear.

```
g_advice %N>%
select(LEVEL, centrality_dgr) %>%
```

```
as_tibble() %>%
group_by(LEVEL) %>%
summarise(mean_centralitydgr = mean(centrality_dgr)) %>%
arrange(desc(mean_centralitydgr)) %>%
head()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 3 x 2
##   LEVEL mean_centralitydgr
##   <dbl>              <dbl>
## 1     2              14.5
## 2     1              13
## 3     3               7.44
```

When it comes to advice seeking relationship it seems as if the hierarchy matters. The level of 1 (CEO) and 2 (VP's) are just about twice as high as level 3 (the managers). But at the same time the VP's are more popular advice giver than the CEO.

```
g_friendship %N>%
select(LEVEL, centrality_dgr) %>%
as_tibble() %>%
group_by(LEVEL) %>%
summarise(mean_centralitydgr = mean(centrality_dgr)) %>%
arrange(desc(mean_centralitydgr)) %>%
head()
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
## # A tibble: 3 x 2
##   LEVEL mean_centralitydgr
##   <dbl>              <dbl>
## 1     2               6
## 2     3               4.69
## 3     1               3
```

The popularity in the friendship network differs a bit. Here the CEO has the smallest amout of people labelling him/her as a friend while the managers (level 3) and VP's (level 2) are second and first. This seems reasonable since the VP's probably have a larger contact surface than the managers and at the same the CEO of a company will often not include themselves in the social relationship in the same way as the employees.

C: Relational Characteristics

The assortativity coefficient measures the in what degree connected nodes have the same label, if it is high it means that they have the same label and vice versa. Therefore it can be used to answer if managers from the same department, hierarchy, age and tenure more likely to become friends or give advice? Starting with giving advice:

```r
assortativity(g_advice, V(g_advice)$DEPT, directed = TRUE)
```

```
## [1] 0.1075871
```

```r
assortativity(g_advice, V(g_advice)$LEVEL, directed = TRUE)
```

```
## [1] 0.05539745
```

```r
assortativity(g_advice, V(g_advice)$AGE, directed = TRUE)
```

```
## [1] 0.0387598
```

```r
assortativity(g_advice, V(g_advice)$TENURE, directed = TRUE)
```

```
## [1] 0.1552188
```

There is no extreme connection between any of the measures, but there for sure no connection between age and advice or level and advice at all. The connection for department and tenure is larger, but not doesn't look significant i.e. managers from same department, hierarchy, age and tenure don't seem to seek advice with each in a larger expected than with the rest of the employees.

Moving on to friendship and the connection here:

```r
assortativity(g_friendship, V(g_friendship)$DEPT, directed = TRUE)
```

```
## [1] 0.1511577
```

```r
assortativity(g_friendship, V(g_friendship)$LEVEL, directed = TRUE)
```

```
## [1] 0.2592447
```

```r
assortativity(g_friendship, V(g_friendship)$AGE, directed = TRUE)
```

```
## [1] 0.1002871
```

```r
assortativity(g_friendship, V(g_friendship)$TENURE, directed = TRUE)
```

```
## [1] -0.09456003
```

Here there seems to be a connection between friendship and working at the same level i.e. people tend to be friends with people in the same level as themselves. As above the rest of the relationsship do not seem significant and there is a negativ relationship connected to the tenure, but again it is so small that it is probably insignificant.

Are friends more likely to give each other advice? Again, the assortativity measure is used:

```
assortativity(g_friendship, V(g_advice), directed = TRUE)

## [1] -0.08119351
```

It doesn't seems to be the case.

In generel it can be difficult to make an assessment of how large or small the measures used above are. Therefore it is often compared to the same measures for a random network at the same size. But I find it unnecessary to create here in order to answer the questions in the measure.

Question 3: Aggregated Networks

For convivence the dataset with attributes is made smaller.

```
attributesdept <- attributes %>%
  select(name, DEPT)
```

The advice and smaller attributes dataset is joint in order to merge the department of the people in "from" AND "to" into the datset. At the same time at new variable containing the type=advice is added in order to color the graphs later in an awesome way.

```
advice1<- advice %>%
  left_join(attributesdept, by=c("from"="name")) %>%
  rename("dept_from"="DEPT") %>%
  left_join(attributesdept, by=c("to"="name")) %>%
  rename("dept_to"="DEPT") %>%
  filter(presence==1)  %>%
  select(-from, -to, -presence) %>%
  add_column(type="advice")
```

The same as above is done for the friendship network.

```
friend1<- friendship %>%
  left_join(attributesdept, by=c("from"="name")) %>%
  rename("dept_from"="DEPT") %>%
  left_join(attributesdept, by=c("to"="name")) %>%
  rename("dept_to"="DEPT") %>%
  filter(presence==1) %>%
  select(-from, -to, -presence) %>%
  add_column(type="friends")
```

The two new dataset containing the departments is binded together.

```
department <- rbind(advice1, friend1)
```

And a graph obejct is made.

```
g_dept <- department %>%
  as_tbl_graph(directed = TRUE)
```

Just inspecting the obejct.

```
g_dept
```

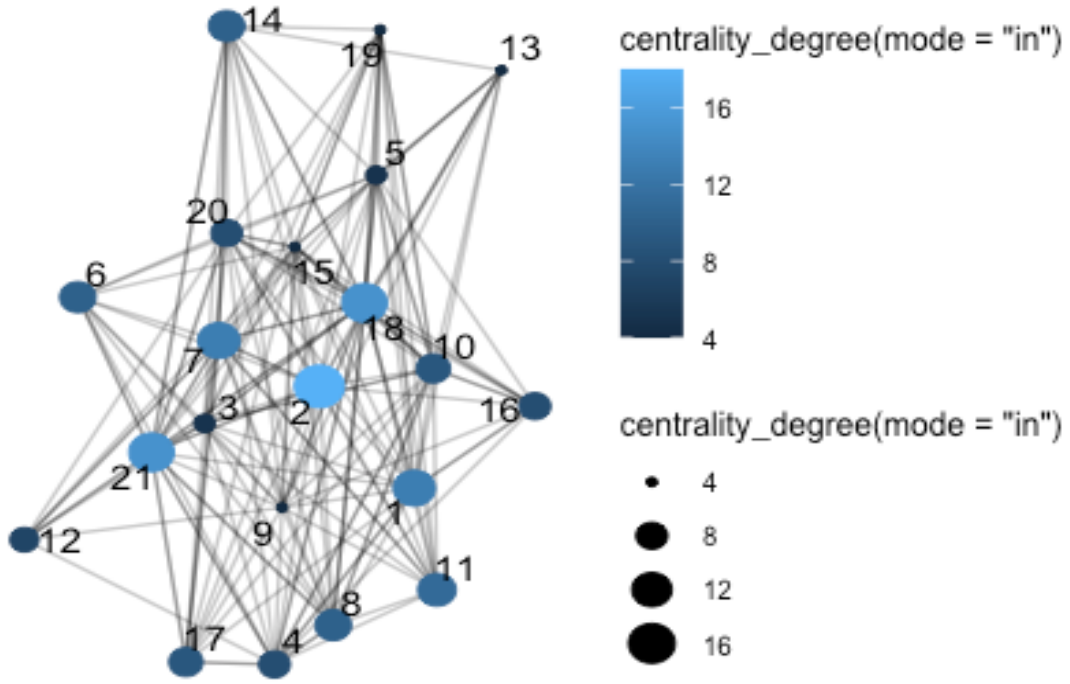
```
## # A tbl_graph: 5 nodes and 292 edges
## #
## # A directed multigraph with 1 component
## #
## # Node Data: 5 x 1 (active)
##   name
##   <chr>
## 1 4
## 2 2
## 3 1
## 4 0
## 5 3
## #
## # Edge Data: 292 x 3
##    from    to type
##   <int> <int> <chr>
## 1     1     1 advice
## 2     1     1 advice
## 3     1     3 advice
## # … with 289 more rows
```

And seeing that the nodes is the departments and the edges are the number of cross departmental friendships/advice relationships as asked in the assignment.

Question 4: Visualization

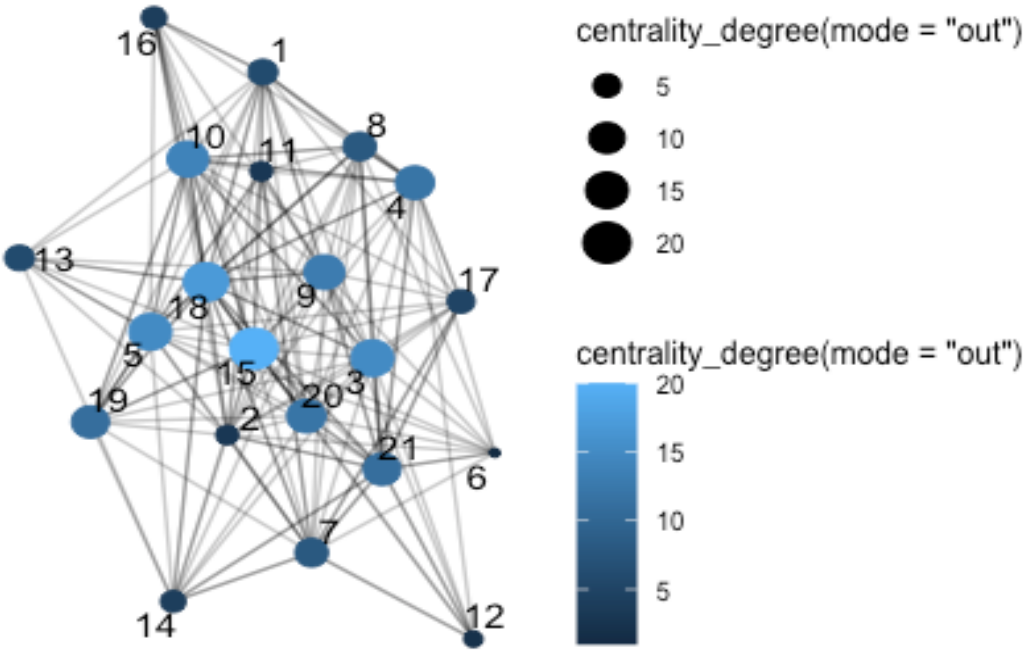
Now for some awesome insights using visualizations. First a visualization of the advice network where the nodes are colored and sized by the centrality degree (using the “in” as mode)

```
g_advice %>% ggraph(layout = "nicely") +
  geom_edge_link(alpha = 0.25) +
  geom_node_point(aes(size= centrality_degree(mode="in"), color= centrality_degree
(mode="in"))) +
  geom_node_text(aes(label = name),
                 repel = TRUE) +
  theme_graph()
```



The graph supports the earlier findings, that the persons with ID 2, 18 and 21 is the most important/popular in the network while 13, 15 and 19 are the less popular. It is also possible to plot the same network but where the nodes are colored and sized by the centrality degree with the mode “out”.

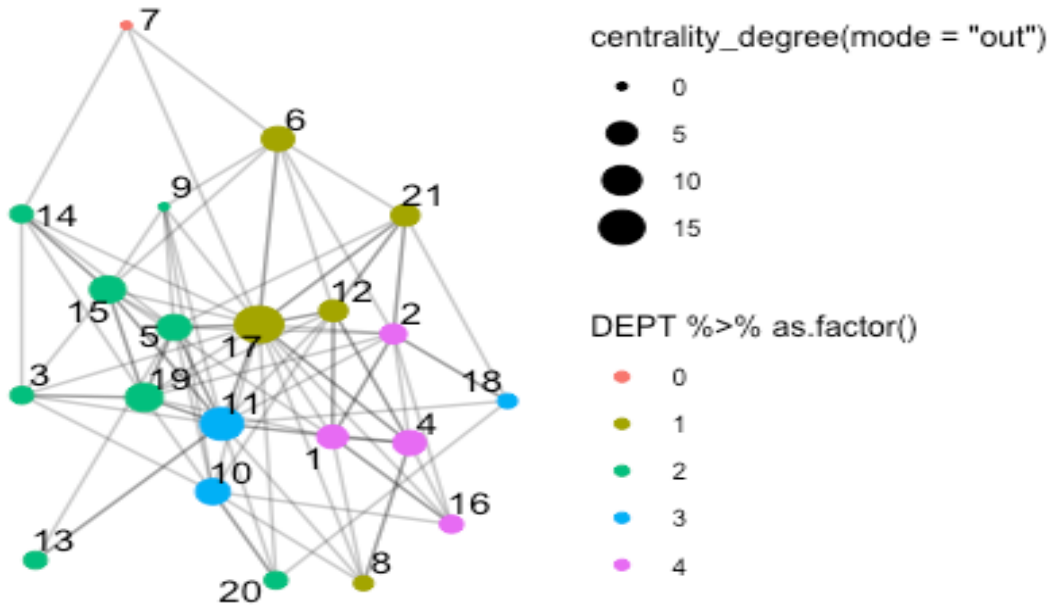
```
g_advice %>% ggraph(layout = "nicely") +
  geom_edge_link(alpha = 0.25) +
  geom_node_point(aes(size= centrality_degree(mode="out"), color= centrality_degre
e(mode="out"))) +
  geom_node_text(aes(label = name),
                 repel = TRUE) +
  theme_graph()
```



From this it is seen that just because a person given alot of advices, it doen't mean that they don't also seek advice. Looking at person 18, this person are also seeking alot of advice from other. But at the same time person 15 seek a lot of adivce but is rarely asked, this could mean that it is a new employee.

The same plot but for the friendship network and colored by department in stead.

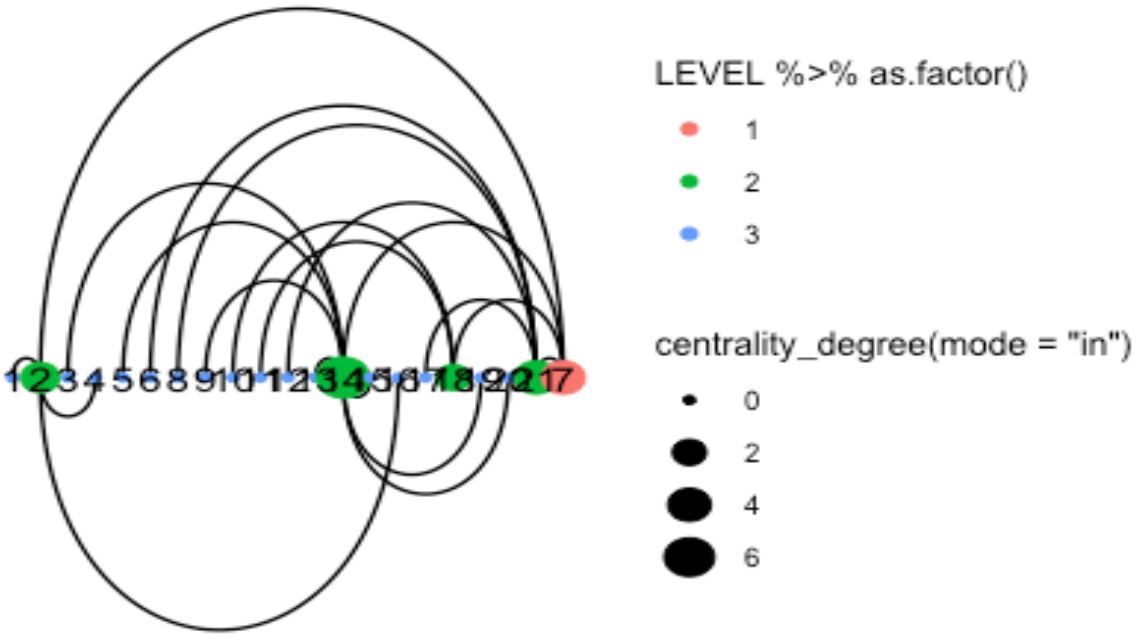
```
g_friendship %>% ggraph(layout = "nicely") +
  geom_edge_link(alpha = 0.25) +
  geom_node_point(aes(size= centrality_degree(mode="out"), color= DEPT %>% as.fact
or())) +
  geom_node_text(aes(label = name),
                 repel = TRUE) +
  theme_graph()
```



There doesn't seems to be a connection between which department a person is placed in and the likelihood that they call someone a friend. The only thing that catches the eye is that the CEO who don't name any in the network as a friend.

Now looking at the report network where the nodes are sized by centrality degree and colored by level.

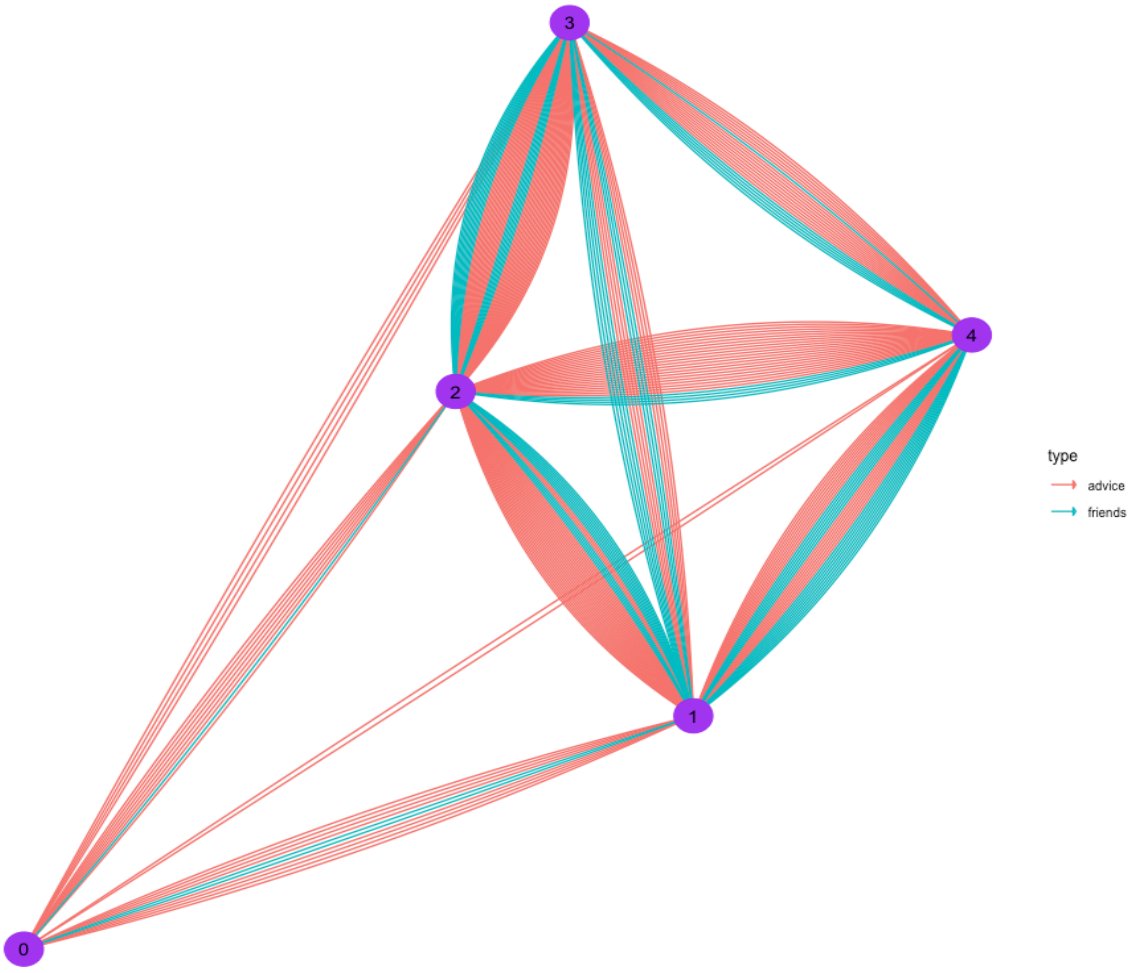
```
g_reportsto %>% ggraph(layout = 'linear') +
  geom_edge_arc() +
  geom_node_point(aes(size = centrality_degree(mode="in"),
                      color = LEVEL %>% as.factor()),
                  show.legend = TRUE) +
   geom_node_text(aes(label = name)) +
    theme_graph()
```



As expected, the most important people in the reporting network is VP's (green) or the CEO (red). It is seen that the people which are VP's are the only ones whom reports directly to the CEO. Number 14 has the most important role in this network measured on centrality.

For a insight in the aggregated dataset a "fan" graph is used, coloring by type.

```
set.seed(1337)
g_dept %>%
  ggraph(layout = 'fr') +
  geom_edge_fan(aes(color = type),
                arrow = arrow(angle = 60, length = unit(0.1, 'cm'),type = "closed"
),
                alpha = 1) +
  geom_node_point(col = 'purple', size=10) +
  geom_node_text(aes(label = name)) +
  theme_graph()
```

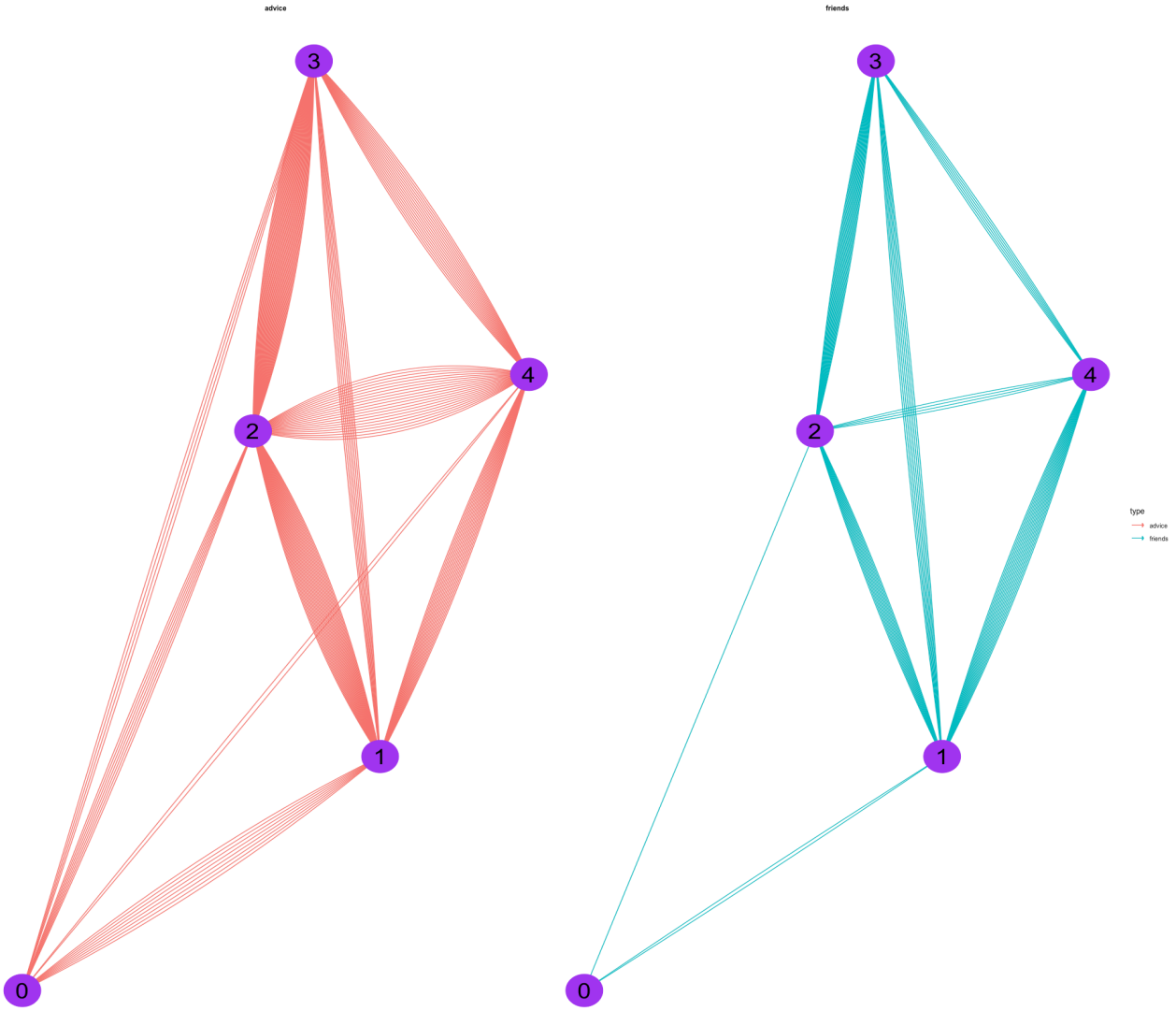


The red is representing the advice network and the blue is the friendship network. As seen in the plot that the CEO i node 0 is primarily a part of the advice seeking network and not the friendship (unfortunately since the network contains alot of edge it is not possible to seen that direction of the network). Some departments have are larger friendship relationship than others - as 1 and 4, where 2 and 4 have more of an advice based relationship.

The two networks plottet separately.

```
set.seed(1337)
g_dept %>%
  ggraph(layout = 'fr') +
  geom_edge_fan(aes(color = type),
                arrow = arrow(angle = 60, length = unit(0.1, 'cm'),type = "closed"
),
```

```
                alpha = 1) +
geom_node_point(col = 'purple', size=20) +
geom_node_text(aes(label = name), size=10) +
theme_graph() +
facet_edges(~type)
```



With a spilt up of the networks it is clear to see that the advice network is larger than the friendship network which would make sence in a workplace - every department are connected in the advice network wihich is not the case in the friendship network. In both network department 3 and 5 have little contact with each other.