

# Assignment 2 - Social Network Analysis

## Part I

Start by installing the “igraph” package. Once you have installed igraph, load the package.

Now upload the data file “discipline-data.csv” as a data frame called “D1”. Each row is a disciplinary action from a teacher to a student so the first line shows that teacher “E” sent student “21” to the principal. It also shows the gender of both the teacher and student and the student’s main elective field of study (“major”) and the field that the teacher instructs in (“t.expertise”).

Before you proceed, you will need to change the data type of the student id variable. Since it is a number R will automatically think it is an integer and code it as such (look at the list of variables by clicking on the data frame arrow in the Data pane. Here you will see the letters “int” next to the stid variable, that stands for integer). However, in this case we are treating the variable as a category, there is no numeric meaning in the variable. So we need to change the format to be a category, what R calls a “factor”. We can do this with the following code:

```
library(igraph)

##
## Attaching package: 'igraph'

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

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

D1 <- read.csv("discipline-data.csv")
D1$stid <- as.factor(D1$stid)
```

igraph requires data to be in a particular structure. There are several structures that it can use but we will be using a combination of an “edge list” and a “vertex list”. As you might imagine the edge list contains a list of all the relationships between students and teachers and any characteristics of those edges that we might be interested in. There are two essential variables in the edge list a “from” variable and a “to” variable that describe the relationships between vertices (a disciplinary action is given “from” and teacher “to” a student). While the vertex list contains all the characteristics of those vertices, in our case gender and major.

So let’s convert our data into an edge list!

First we will isolate the variables that are of interest: tid and stid

```
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
```

```
D2 <- select(D1, tid, stid)
```

Since our data represents every time a teacher sends a student to the principal there are multiple rows when the same teacher sends the same student. We want to collapse these into a single row, with a variable that shows how many times a teacher-student pair appears.

```
EDGE <- count(D2, tid, stid)

names(EDGE) <- c("from", "to", "count")
```

EDGE is your edge list. Now we need to make the vertex list, a list of all the teachers and students and their characteristics in our network.

```
#First we will separate the teachers from our original data frame
V.TCH <- select(D1, tid, t.gender, t.expertise)
#Remove all the repeats so that we just have a list of each teacher and their characteristics
V.TCH <- unique(V.TCH)
#Add a variable that describes that they are teachers
V.TCH$group <- "teacher"

#Now repeat this process for the students
V.STD <- select(D1, stid, s.gender, s.major)
V.STD <- unique(V.STD)
V.STD$group <- "student"

#Make sure that the student and teacher data frames have the same variables names
names(V.TCH) <- c("id", "gender", "topic", "group")
names(V.STD) <- c("id", "gender", "topic", "group")

#Bind the two data frames together (you will get a warning because the teacher data frame has 5 types of
VERTEX <- bind_rows(V.TCH, V.STD)
```

```
## Warning in bind_rows_(x, .id): Unequal factor levels: coercing to character
```

```
## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector
```

```
## Warning in bind_rows_(x, .id): binding character and factor vector,
## coercing into character vector
```

```
## Warning in bind_rows_(x, .id): Unequal factor levels: coercing to character
```

```
## Warning in bind_rows_(x, .id): binding character and factor vector,
```

```
## coercing into character vector
```

```
## Warning in bind_rows(x, .id): binding character and factor vector,  
## coercing into character vector
```

Now we have both a Vertex and Edge list it is time to plot our graph!

```
#Load the igraph package
```

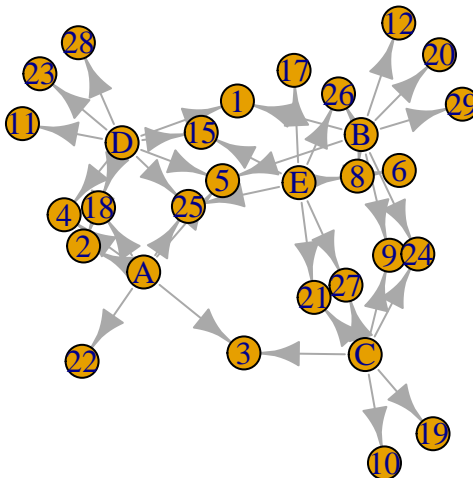
```
library(igraph)
```

```
#First we will make an object that contains the graph information using our two dataframes EDGE and VER
```

```
g <- graph.data.frame(EDGE, directed=TRUE, vertices=VERTEX)
```

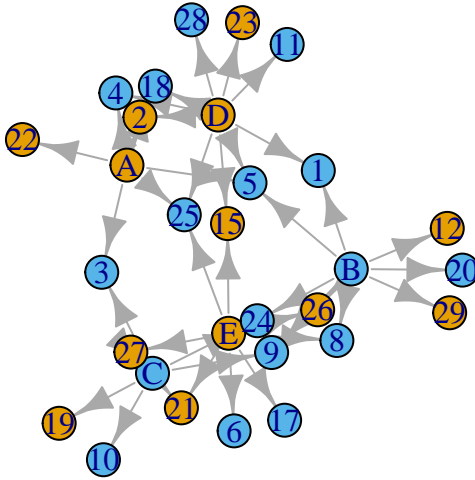
```
#Now we can plot our graph using the force directed graphing technique - our old friend Fruchterman-Rei
```

```
plot(g,layout=layout.fruchterman.reingold)
```



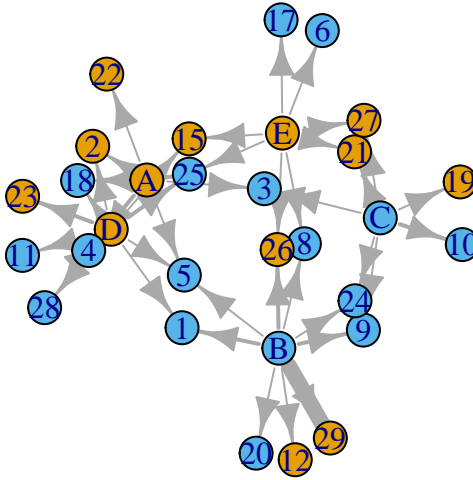
```
#There are many ways to change the attributes of the graph to represent different characteristics of th
```

```
plot(g,layout=layout.fruchterman.reingold, vertex.color=VERTEX$gender)
```



*#We can change the thickness of the edge according to the number of times a particular teacher has sent*

```
plot(g,layout=layout.fruchterman.reingold, vertex.color=VERTEX$gender, edge.width=EDGE$count)
```



## Part II

In Part II your task is to look up in the igraph documentation and create a graph that sizes the student vertices in terms of the number of disciplinary actions they have recieved, and the teachers in terms of the number of disciplinary actions they have given out.

```
V.TCH.SIZE <- D2 %>%
  group_by(tid) %>%
  summarize(size = n())
names(V.TCH.SIZE) <- c("id", "size")

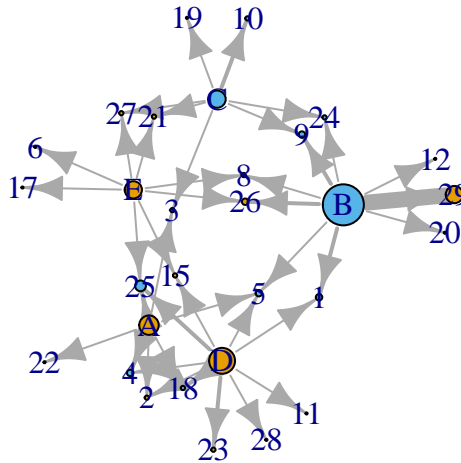
V.STD.SIZE <- D2 %>%
  group_by(stdid) %>%
  summarize(size = n())
names(V.STD.SIZE) <- c("id", "size")

V.SIZE <- rbind(V.TCH.SIZE, V.STD.SIZE)

VERTEX <- merge(VERTEX, V.SIZE, x.by = "id", y.by = "size")

g <- graph.data.frame(EDGE, directed=TRUE, vertices=VERTEX)

plot(g, layout=layout.fruchterman.reingold, vertex.label = VERTEX$id, vertex.color=VERTEX$gender, vertex
```



### Part III

Now practice with data from our class. Please create a **person-network** with the data set `hudk4050-classes.csv`. To create this network you will need to create a person-class matrix using the `tidyr` functions and then create a person-person matrix using `t()`. You will then need to plot a matrix rather than a data frame using `igraph`.

Once you have done this, also look up how to generate the following network metrics: betweenness centrality and dregree. **Who is the most central person in the network?**

```
library(tidyr)
```

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

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

```
library(dplyr)
library(tibble)
```

```
##
## Attaching package: 'tibble'
```

```

## The following object is masked from 'package:igraph':
##
##      as_data_frame

library(stringr)
library(igraph)
library(reshape2)

##
## Attaching package: 'reshape2'

## The following object is masked from 'package:tidyr':
##
##      smiths

# Import data
table <- read.table("hudk4050-classes.txt", header = TRUE, sep = ",")

# Crop data
classdata <- table[-c(1:2),c(20:26)]
person_class <- gather(classdata, "QID", "CID", 2:7)

## Warning: attributes are not identical across measure variables;
## they will be dropped

person_class <- person_class[,-2]
person_class <- na_if(person_class, "")
person_class <- na.omit(person_class)

# Drop incomplete data
person_class <- subset(person_class, str_detect(person_class[,2], "[A-Z]+\\s*[0-9]+"))

# Format data
person_class[,2] <- gsub("([A-Z]+)\\s*([0-9]+)", "\\1 \\2", person_class[,2])

# Modify ambiguous data
person_class[person_class == 'QMSS-G 5072' | person_class == 'G 5072'] <- 'QMSS 5072'
person_class[person_class == 'QMSS G 5015'] <- 'QMSS 5015'
person_class[person_class == 'QMSS GR 5067' | person_class == 'G 5067'] <- 'QMSS 5067'

person_person <- merge(person_class, person_class, by = 'CID')
names(person_person) <- c('CID', 'FROM', 'TO')
person_person <- subset(person_person, !FROM == TO)

person_person.edge <- count(person_person, FROM, TO)
names(person_person.edge) <- c('FROM', 'TO', 'count')

person_person.matrix <- pivot_wider(person_person.edge, names_from = FROM, values_from = count)
person_person.matrix %>% remove_rownames %>% column_to_rownames(var='TO')

##
ad3671 adw2184 ajz2123 art2172 bc2824 bkd2115

```

## adw2184	2	NA	1	1	1	2
## ajz2123	1	1	NA	1	1	1
## art2172	1	1	1	NA	1	1
## bc2824	1	1	1	1	NA	1
## bkd2115	2	2	1	1	1	NA
## cc97760n@pace.edu	2	2	1	1	1	2
## ch3460	1	2	1	1	1	2
## CL02112	1	1	1	1	1	1
## cy2535	2	2	3	1	1	2
## dm3487	1	2	1	1	1	2
## HG2527	2	2	1	1	1	2
## hw2663	1	1	1	1	1	1
## IB2445	1	1	1	1	1	1
## jc5230	2	2	1	1	1	2
## jg4191	1	2	1	1	1	2
## jh4175	1	1	1	1	1	1
## js5498	2	2	1	1	1	2
## jz3101	1	1	1	1	1	2
## kdp2124	1	2	1	1	1	1
## la2738	1	1	1	1	1	1
## ld2882	1	2	1	1	1	2
## ll3358	2	3	1	1	1	2
## lm3477	1	2	1	1	1	2
## lr2956	3	3	1	1	1	2
## mh3054	1	NA	NA	NA	NA	NA
## mw3399	1	1	2	1	1	1
## nx2150	1	2	1	1	1	2
## ql2360	1	1	2	1	1	1
## sj2562	2	2	1	1	1	2
## ss5851	1	2	1	1	1	2
## wz2508	2	2	1	1	1	2
## xc2496	2	2	1	1	1	2
## xn2135	1	1	1	1	1	1
## xql2001	2	2	1	1	1	3
## XY2418	1	2	1	1	1	2
## xz2840	1	2	1	1	1	2
## XZ2910	2	2	2	1	1	2
## yl3984	1	1	1	1	1	1
## yl4232	1	1	1	1	1	1
## yl4284	1	1	1	1	1	1
## ym2775	1	2	1	1	1	2
## yq2257	1	1	1	1	1	1
## yx2531	2	2	2	1	1	2
## yz3413	2	2	1	1	1	2
## yz3679	3	3	1	1	1	2
## yz3730	2	2	1	1	1	2
## ZC2323	1	1	1	1	1	1
## zg2338	1	2	1	1	1	2
## zz2641	3	3	1	1	1	2
## zz2726	2	2	2	1	1	2
## ad3671	NA	2	1	1	1	2
## RW2796	NA	1	NA	NA	NA	1
##	cc97760n@pace.edu ch3460 CL02112 cy2535 dm3487 HG2527					
## adw2184		2	2	1	2	2



## ajz2123	1	1	1	3	1	1		
## art2172	1	1	1	1	1	1		
## bc2824	1	1	1	1	1	1		
## bkd2115	2	2	1	2	2	2		
## cc97760n@pace.edu	NA	1	1	2	1	3		
## ch3460	1	NA	1	1	4	1		
## CL02112	1	1	NA	1	1	1		
## cy2535	2	1	1	NA	1	2		
## dm3487	1	4	1	1	NA	1		
## HG2527	3	1	1	2	1	NA		
## hw2663	1	1	1	1	1	1		
## IB2445	2	1	1	1	1	2		
## jc5230	3	1	1	2	1	3		
## jg4191	1	4	1	1	4	1		
## jh4175	1	1	1	1	2	1		
## js5498	3	1	1	2	1	3		
## jz3101	1	3	1	1	3	1		
## kdp2124	2	2	1	1	2	2		
## la2738	1	1	1	1	1	1		
## ld2882	1	4	1	1	4	1		
## ll3358	3	2	1	2	2	3		
## lm3477	1	4	1	1	4	1		
## lr2956	2	2	1	2	2	2		
## mh3054	NA	NA	NA	NA	NA	NA		
## mw3399	1	1	1	2	1	1		
## nx2150	1	4	1	1	4	1		
## ql2360	2	1	1	2	1	2		
## sj2562	3	1	1	2	1	3		
## ss5851	1	4	1	1	4	1		
## wz2508	3	1	1	2	1	3		
## xc2496	3	1	1	2	1	4		
## xn2135	1	1	1	1	1	1		
## xql2001	2	1	1	2	1	2		
## XY2418	1	4	1	1	4	1		
## xz2840	1	4	1	1	5	1		
## XZ2910	2	1	1	3	1	2		
## yl3984	1	1	1	1	1	1		
## yl4232	1	1	1	1	1	1		
## yl4284	1	1	1	1	1	1		
## ym2775	1	4	1	1	4	1		
## yq2257	1	1	1	2	1	1		
## yx2531	2	1	1	3	1	2		
## yz3413	2	1	1	2	1	2		
## yz3679	2	2	1	2	2	2		
## yz3730	3	1	1	2	1	3		
## ZC2323	1	1	1	1	1	1		
## zg2338	1	4	1	1	4	1		
## zz2641	2	2	1	2	2	2		
## zz2726	2	1	1	3	1	2		
## ad3671	2	1	1	2	1	2		
## RW2796	NA	3	NA	NA	3	NA		
##	hw2663	IB2445	jc5230	jg4191	jh4175	js5498	jz3101	kdp2124
## adw2184	1	1	2	2	1	2	1	2
## ajz2123	1	1	1	1	1	1	1	1

## art2172	1	1	1	1	1	1	1	1
## bc2824	1	1	1	1	1	1	1	1
## bkd2115	1	1	2	2	1	2	2	1
## cc97760n@pace.edu	1	2	3	1	1	3	1	2
## ch3460	1	1	1	4	1	1	3	2
## CL02112	1	1	1	1	1	1	1	1
## cy2535	1	1	2	1	1	2	1	1
## dm3487	1	1	1	4	2	1	3	2
## HG2527	1	2	3	1	1	3	1	2
## hw2663	NA	1	1	1	1	1	1	1
## IB2445	1	NA	2	1	1	2	1	2
## jc5230	1	2	NA	1	1	3	1	2
## jg4191	1	1	1	NA	1	1	3	2
## jh4175	1	1	1	1	NA	1	1	1
## js5498	1	2	3	1	1	NA	1	2
## jz3101	1	1	1	3	1	1	NA	1
## kdp2124	1	2	2	2	1	2	1	NA
## la2738	1	1	1	1	1	1	1	1
## ld2882	1	1	1	4	1	1	3	2
## ll3358	1	2	3	2	1	3	1	3
## lm3477	1	1	1	4	1	1	3	2
## lr2956	1	1	2	2	1	2	1	2
## mh3054	NA	NA	NA	NA	NA	NA	NA	NA
## mw3399	1	1	1	1	2	1	1	1
## nx2150	1	1	1	4	1	1	3	2
## ql2360	1	3	2	1	1	2	1	2
## sj2562	1	2	3	1	1	3	1	2
## ss5851	1	1	1	4	1	1	3	2
## wz2508	1	2	3	1	1	3	1	2
## xc2496	1	2	3	1	1	3	1	2
## xn2135	1	1	1	1	1	1	1	1
## xql2001	1	1	2	1	2	2	1	1
## XY2418	1	1	1	4	1	1	3	2
## xz2840	1	1	1	4	2	1	3	2
## XZ2910	1	1	2	1	1	2	1	1
## yl3984	1	1	1	1	1	1	1	1
## yl4232	1	1	1	1	1	1	1	1
## yl4284	1	1	1	1	1	1	1	1
## ym2775	1	1	1	4	1	1	3	2
## yq2257	1	1	1	1	1	1	1	1
## yx2531	1	1	2	1	1	2	1	1
## yz3413	1	1	2	1	1	2	1	1
## yz3679	1	2	2	2	1	2	1	2
## yz3730	1	2	3	1	1	3	1	2
## ZC2323	1	1	1	1	1	1	1	1
## zg2338	1	1	1	4	1	1	3	2
## zz2641	1	1	2	2	1	2	1	2
## zz2726	1	2	2	1	1	2	1	1
## ad3671	1	1	2	1	1	2	1	1
## RW2796	NA	NA	NA	3	NA	NA	2	1
##	la2738	ld2882	ll3358	lm3477	lr2956	mh3054	mw3399	nx2150
## adw2184	1	2	3	2	3	NA	1	2
## ajz2123	1	1	1	1	1	NA	2	1
## art2172	1	1	1	1	1	NA	1	1

## bc2824	1	1	1	1	1	NA	1	1
## bkd2115	1	2	2	2	2	NA	1	2
## cc97760n@pace.edu	1	1	3	1	2	NA	1	1
## ch3460	1	4	2	4	2	NA	1	4
## CL02112	1	1	1	1	1	NA	1	1
## cy2535	1	1	2	1	2	NA	2	1
## dm3487	1	4	2	4	2	NA	1	4
## HG2527	1	1	3	1	2	NA	1	1
## hw2663	1	1	1	1	1	NA	1	1
## IB2445	1	1	2	1	1	NA	1	1
## jc5230	1	1	3	1	2	NA	1	1
## jg4191	1	4	2	4	2	NA	1	4
## jh4175	1	1	1	1	1	NA	2	1
## js5498	1	1	3	1	2	NA	1	1
## jz3101	1	3	1	3	1	NA	1	3
## kdp2124	1	2	3	2	2	NA	1	2
## la2738	NA	1	1	1	1	NA	1	1
## ld2882	1	NA	2	4	2	NA	1	4
## ll3358	1	2	NA	2	3	NA	1	2
## lm3477	1	4	2	NA	2	NA	1	4
## lr2956	1	2	3	2	NA	NA	1	2
## mh3054	NA	NA	NA	NA	NA	NA	NA	NA
## mw3399	1	1	1	1	1	NA	NA	1
## nx2150	1	4	2	4	2	NA	1	NA
## ql2360	1	1	2	1	1	NA	2	1
## sj2562	1	1	3	1	2	NA	1	1
## ss5851	1	4	2	4	2	NA	1	4
## wz2508	1	1	3	1	2	NA	1	1
## xc2496	1	1	3	1	2	NA	1	1
## xn2135	1	1	1	1	1	NA	1	1
## xql2001	1	1	2	1	2	NA	2	1
## XY2418	1	4	2	4	2	NA	1	4
## xz2840	1	4	2	4	2	NA	1	4
## XZ2910	1	1	2	1	2	NA	2	1
## yl3984	1	1	1	1	1	NA	1	1
## yl4232	1	1	1	1	1	NA	1	1
## yl4284	1	1	1	1	1	NA	1	1
## ym2775	1	4	2	4	2	NA	1	4
## yq2257	1	1	1	1	1	NA	1	1
## yx2531	1	1	2	1	2	NA	2	1
## yz3413	2	1	2	1	2	NA	1	1
## yz3679	1	2	3	2	3	1	1	2
## yz3730	1	1	3	1	2	NA	1	1
## ZC2323	1	1	1	1	1	NA	1	1
## zg2338	1	4	2	4	2	NA	1	4
## zz2641	1	2	3	2	4	NA	1	2
## zz2726	1	1	2	1	2	NA	2	1
## ad3671	1	1	2	1	3	1	1	1
## RW2796	NA	3	1	3	1	NA	NA	3
##	ql2360	RW2796	sj2562	ss5851	wz2508	xc2496	xn2135	xql2001
## adw2184	1	1	2	2	2	2	1	2
## ajz2123	2	NA	1	1	1	1	1	1
## art2172	1	NA	1	1	1	1	1	1
## bc2824	1	NA	1	1	1	1	1	1

## bkd2115	1	1	2	2	2	2	1	3
## cc97760n@pace.edu	2	NA	3	1	3	3	1	2
## ch3460	1	3	1	4	1	1	1	1
## CL02112	1	NA	1	1	1	1	1	1
## cy2535	2	NA	2	1	2	2	1	2
## dm3487	1	3	1	4	1	1	1	1
## HG2527	2	NA	3	1	3	4	1	2
## hw2663	1	NA	1	1	1	1	1	1
## IB2445	3	NA	2	1	2	2	1	1
## jc5230	2	NA	3	1	3	3	1	2
## jg4191	1	3	1	4	1	1	1	1
## jh4175	1	NA	1	1	1	1	1	2
## js5498	2	NA	3	1	3	3	1	2
## jz3101	1	2	1	3	1	1	1	1
## kdp2124	2	1	2	2	2	2	1	1
## la2738	1	NA	1	1	1	1	1	1
## ld2882	1	3	1	4	1	1	1	1
## ll3358	2	1	3	2	3	3	1	2
## lm3477	1	3	1	4	1	1	1	1
## lr2956	1	1	2	2	2	2	1	2
## mh3054	NA	NA	NA	NA	NA	NA	NA	NA
## mw3399	2	NA	1	1	1	1	1	2
## nx2150	1	3	1	4	1	1	1	1
## ql2360	NA	NA	2	1	2	2	1	1
## sj2562	2	NA	NA	1	3	3	1	2
## ss5851	1	3	1	NA	1	1	1	1
## wz2508	2	1	3	1	NA	3	1	2
## xc2496	2	NA	3	1	3	NA	1	2
## xn2135	1	NA	1	1	1	1	NA	1
## xql2001	1	NA	2	1	2	2	1	NA
## XY2418	1	3	1	4	1	1	1	1
## xz2840	1	3	1	4	1	1	1	1
## XZ2910	2	NA	2	1	2	2	1	2
## y13984	1	NA	1	1	1	1	1	1
## y14232	1	NA	1	1	1	1	3	1
## y14284	1	NA	1	1	1	1	1	1
## ym2775	1	3	1	4	1	1	1	1
## yq2257	1	NA	1	1	1	1	1	1
## yx2531	2	NA	2	1	2	2	1	2
## yz3413	1	NA	2	1	2	2	1	2
## yz3679	2	1	2	2	2	2	1	2
## yz3730	2	NA	3	1	3	3	1	2
## ZC2323	1	NA	1	1	1	1	1	1
## zg2338	1	3	1	4	1	1	1	1
## zz2641	1	1	2	2	2	2	1	2
## zz2726	2	NA	2	1	2	2	1	2
## ad3671	1	NA	2	1	2	2	1	2
## RW2796	NA	NA	NA	3	1	NA	NA	NA
##	XY2418	xz2840	XZ2910	y13984	y14232	y14284	ym2775	yq2257
## adw2184	2	2	2	1	1	1	2	1
## ajz2123	1	1	2	1	1	1	1	1
## art2172	1	1	1	1	1	1	1	1
## bc2824	1	1	1	1	1	1	1	1
## bkd2115	2	2	2	1	1	1	2	1

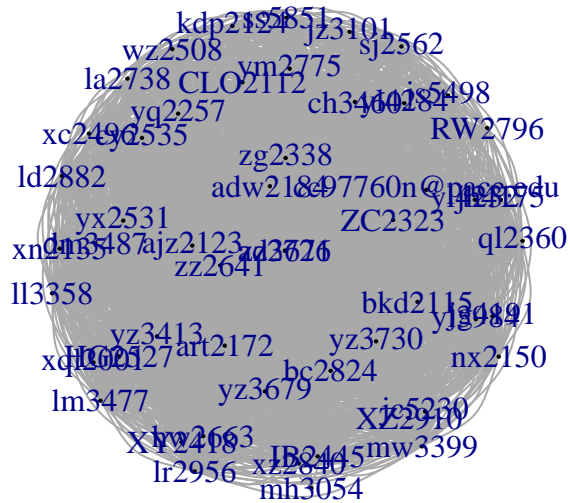
## cc97760n@pace.edu	1	1	2	1	1	1	1	1
## ch3460	4	4	1	1	1	1	4	1
## CL02112	1	1	1	1	1	1	1	1
## cy2535	1	1	3	1	1	1	1	2
## dm3487	4	5	1	1	1	1	4	1
## HG2527	1	1	2	1	1	1	1	1
## hw2663	1	1	1	1	1	1	1	1
## IB2445	1	1	1	1	1	1	1	1
## jc5230	1	1	2	1	1	1	1	1
## jg4191	4	4	1	1	1	1	4	1
## jh4175	1	2	1	1	1	1	1	1
## js5498	1	1	2	1	1	1	1	1
## jz3101	3	3	1	1	1	1	3	1
## kdp2124	2	2	1	1	1	1	2	1
## la2738	1	1	1	1	1	1	1	1
## ld2882	4	4	1	1	1	1	4	1
## ll3358	2	2	2	1	1	1	2	1
## lm3477	4	4	1	1	1	1	4	1
## lr2956	2	2	2	1	1	1	2	1
## mh3054	NA	NA	NA	NA	NA	NA	NA	NA
## mw3399	1	1	2	1	1	1	1	1
## nx2150	4	4	1	1	1	1	4	1
## ql2360	1	1	2	1	1	1	1	1
## sj2562	1	1	2	1	1	1	1	1
## ss5851	4	4	1	1	1	1	4	1
## wz2508	1	1	2	1	1	1	1	1
## xc2496	1	1	2	1	1	1	1	1
## xn2135	1	1	1	1	3	1	1	1
## xql2001	1	1	2	1	1	1	1	1
## XY2418	NA	4	1	1	1	1	4	1
## xz2840	4	NA	1	1	1	1	4	1
## XZ2910	1	1	NA	1	1	1	1	1
## yl3984	1	1	1	NA	1	1	1	1
## yl4232	1	1	1	1	NA	1	1	1
## yl4284	1	1	1	1	1	NA	1	1
## ym2775	4	4	1	1	1	1	NA	1
## yq2257	1	1	1	1	1	1	1	NA
## yx2531	1	1	3	1	1	1	1	1
## yz3413	1	1	2	1	1	1	1	1
## yz3679	2	2	2	1	1	1	2	1
## yz3730	1	1	3	1	1	1	1	1
## ZC2323	1	1	1	1	1	1	1	1
## zg2338	4	4	1	1	1	1	4	1
## zz2641	2	2	2	1	1	1	2	1
## zz2726	1	1	3	1	1	1	1	1
## ad3671	1	1	2	1	1	1	1	1
## RW2796	3	3	NA	NA	NA	NA	3	NA
##	yx2531	yz3413	yz3679	yz3730	ZC2323	zg2338	zz2641	zz2726
## adw2184	2	2	3	2	1	2	3	2
## ajz2123	2	1	1	1	1	1	1	2
## art2172	1	1	1	1	1	1	1	1
## bc2824	1	1	1	1	1	1	1	1
## bkd2115	2	2	2	2	1	2	2	2
## cc97760n@pace.edu	2	2	2	3	1	1	2	2

## ch3460	1	1	2	1	1	4	2	1
## CL02112	1	1	1	1	1	1	1	1
## cy2535	3	2	2	2	1	1	2	3
## dm3487	1	1	2	1	1	4	2	1
## HG2527	2	2	2	3	1	1	2	2
## hw2663	1	1	1	1	1	1	1	1
## IB2445	1	1	2	2	1	1	1	2
## jc5230	2	2	2	3	1	1	2	2
## jg4191	1	1	2	1	1	4	2	1
## jh4175	1	1	1	1	1	1	1	1
## js5498	2	2	2	3	1	1	2	2
## jz3101	1	1	1	1	1	3	1	1
## kdp2124	1	1	2	2	1	2	2	1
## la2738	1	2	1	1	1	1	1	1
## ld2882	1	1	2	1	1	4	2	1
## ll3358	2	2	3	3	1	2	3	2
## lm3477	1	1	2	1	1	4	2	1
## lr2956	2	2	3	2	1	2	4	2
## mh3054	NA	NA	1	NA	NA	NA	NA	NA
## mw3399	2	1	1	1	1	1	1	2
## nx2150	1	1	2	1	1	4	2	1
## ql2360	2	1	2	2	1	1	1	2
## sj2562	2	2	2	3	1	1	2	2
## ss5851	1	1	2	1	1	4	2	1
## wz2508	2	2	2	3	1	1	2	2
## xc2496	2	2	2	3	1	1	2	2
## xn2135	1	1	1	1	1	1	1	1
## xql2001	2	2	2	2	1	1	2	2
## XY2418	1	1	2	1	1	4	2	1
## xz2840	1	1	2	1	1	4	2	1
## XZ2910	3	2	2	3	1	1	2	3
## yl3984	1	1	1	1	1	1	1	1
## yl4232	1	1	1	1	1	1	1	1
## yl4284	1	1	1	1	1	1	1	1
## ym2775	1	1	2	1	1	4	2	1
## yq2257	1	1	1	1	1	1	1	1
## yx2531	NA	2	2	2	1	1	2	3
## yz3413	2	NA	2	2	1	1	2	2
## yz3679	2	2	NA	2	1	2	3	2
## yz3730	2	2	2	NA	1	1	2	2
## ZC2323	1	1	1	1	NA	1	1	1
## zg2338	1	1	2	1	1	NA	2	1
## zz2641	2	2	3	2	1	2	NA	2
## zz2726	3	2	2	2	1	1	2	NA
## ad3671	2	2	3	2	1	1	3	2
## RW2796	NA	NA	1	NA	NA	3	1	NA

```

person_person.graph <- graph_from_data_frame(person_person.edge, directed=FALSE)
plot(person_person.graph, layout=layout.sphere, vertex.label = V(person_person.graph)$FROM, vertex.size

```



### To Submit Your Assignment

Please submit your assignment by first “knitting” your RMarkdown document into an html file and then comit, push and pull request both the RMarkdown file and the html file.