

NETWORK SCIENCE

Assignment 2

STUDENT | Erfan Raoofian

ID | 47315221

INSTRUCTOR | Prof. Ifeoma Adaji



IRVING K. BARBER SCHOOL OF SCIENCE
DEPARTMENT OF COMPUTER SCIENCE
OKANAGAN CAMPUS

1 Abstract

In this assignment, a Graph is plotted in which the nodes are the students of a class, and the edges show which students like to sit next to each other. To be more specific, there would be a directed edge connecting student A to student B if student A likes to sit next to student B. The dataset used for this assignment is downloaded from the assignment page on canvas and I am using **grade 1** for this assignment.

2 Loading Data

As stated in the assignment guide file, these commands are used to load the data from the CSV files to the programming environment.

```
1 nodes <- read.csv("NodesGrade1.csv", header=TRUE)
2 edges <- read.csv("EdgesGrade1.csv", header=TRUE)
```

Then, a graph object is created from the loaded data using the command below:

```
1 grade1_graph <- graph_from_data_frame(edges, nodes, directed=TRUE)
```

3 Assignment Questions

In the following section of this report, I will answer some questions which were asked in the assignment.

3.1 Question 1: How many nodes and edges are in the network?

Number of nodes: Each node represents one and only one student. There are **35 students** in the class as there are 35 nodes in the graph. The number of nodes is checked using the command below:

```
1 vCount = vcount(grade1_graph) #vCount = 35
```

Number of edges: there would be a directed edge connecting student A to student B if student A likes to sit next to student B. It is possible to figure out the **number of edges** in the graph to be **67** using the command below:

```
1 eCount = ecount(grade1_graph) #eCount = 67
```

3.2 Question 2: Plot the network graph with some specific graph options stated in assignment file.

The graph is plotted using the code snippet below. First one is plotted with vertex_size = 8 in Figure 1 with original labels on the nodes, and the second version is plotted with vertex_size = 10 with gender specific labels on the nodes in Figure 2.

```
1 customizing <- layout_(grade1_graph, with_gem())
2 plot(grade1_graph, vertex.label=V(grade1_graph)$ gender.code,
3       vertex.size=10, edge.arrow.size=0.5,
4       vertex.color = V(grade1_graph)$ gender.code, layout=customizing)
```

This Question also asks to fill a table to indicate the number of edges within and between the two gender groups from the network plot. Instead of counting the numbers manually from the network

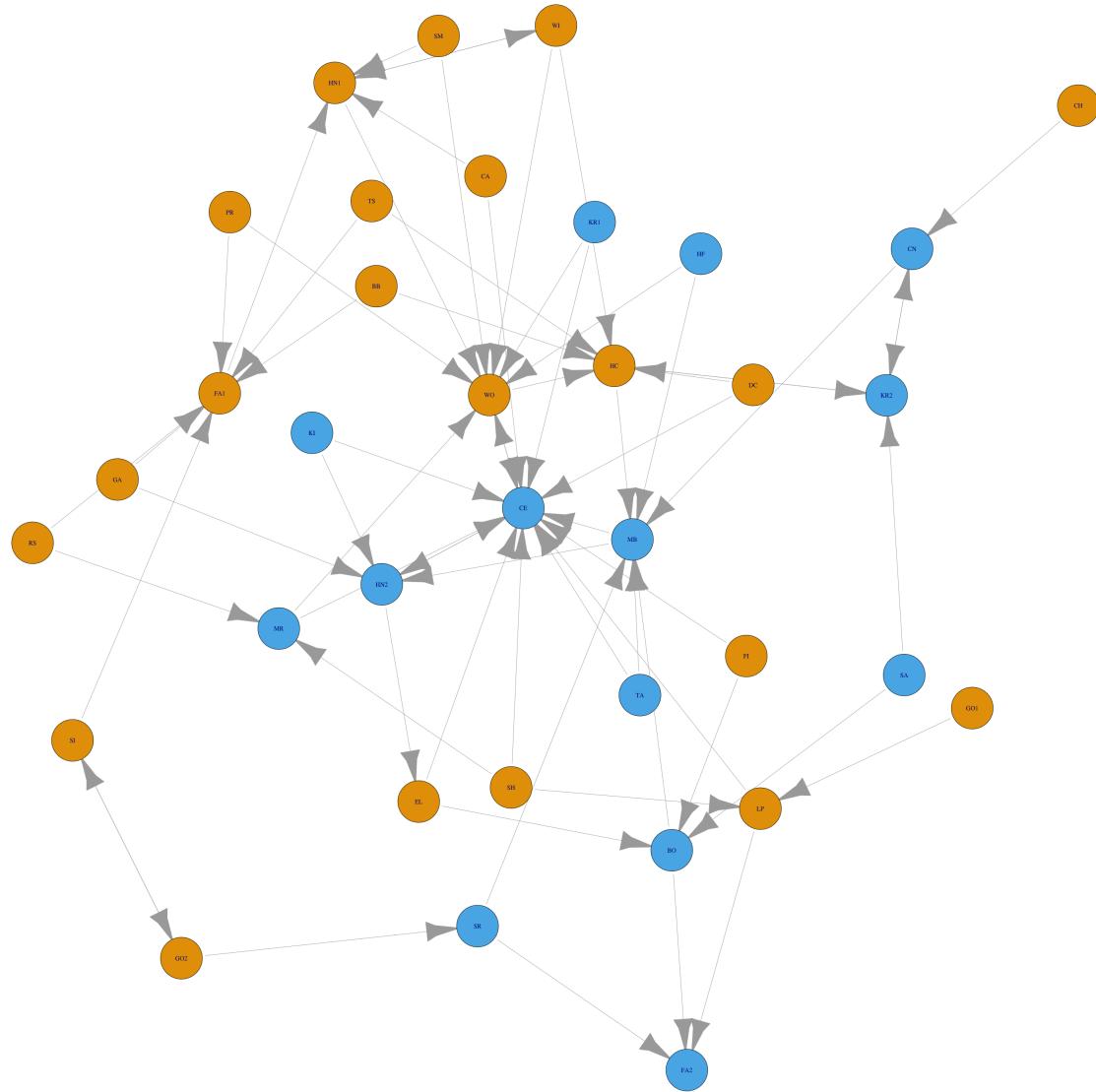


Figure 1: Network plot with vertex_size = 8

plot, I used some code to figure them out. the output of the code snippet below would be all the edges that their source is a node with gender code X.

```
1 from_X <- E(grade1_graph)[.from(V(grade1_graph)[gender.code==X])]
```

and the output of the code snippet below would be all the edges that their destination is a node with gender code Y.

```
1 to_Y <- E(grade1_graph)[.to(V(grade1_graph)[gender.code==Y])]
```

and the intersection of these two sets would be all the edges from nodes with gender code X to nodes with gender code Y which can be done using this command

```
1 X_to_Y = intersection(from_X, to_Y)
```

Having these tools, the numbers requested in the table can be calculated using the code snippet below.

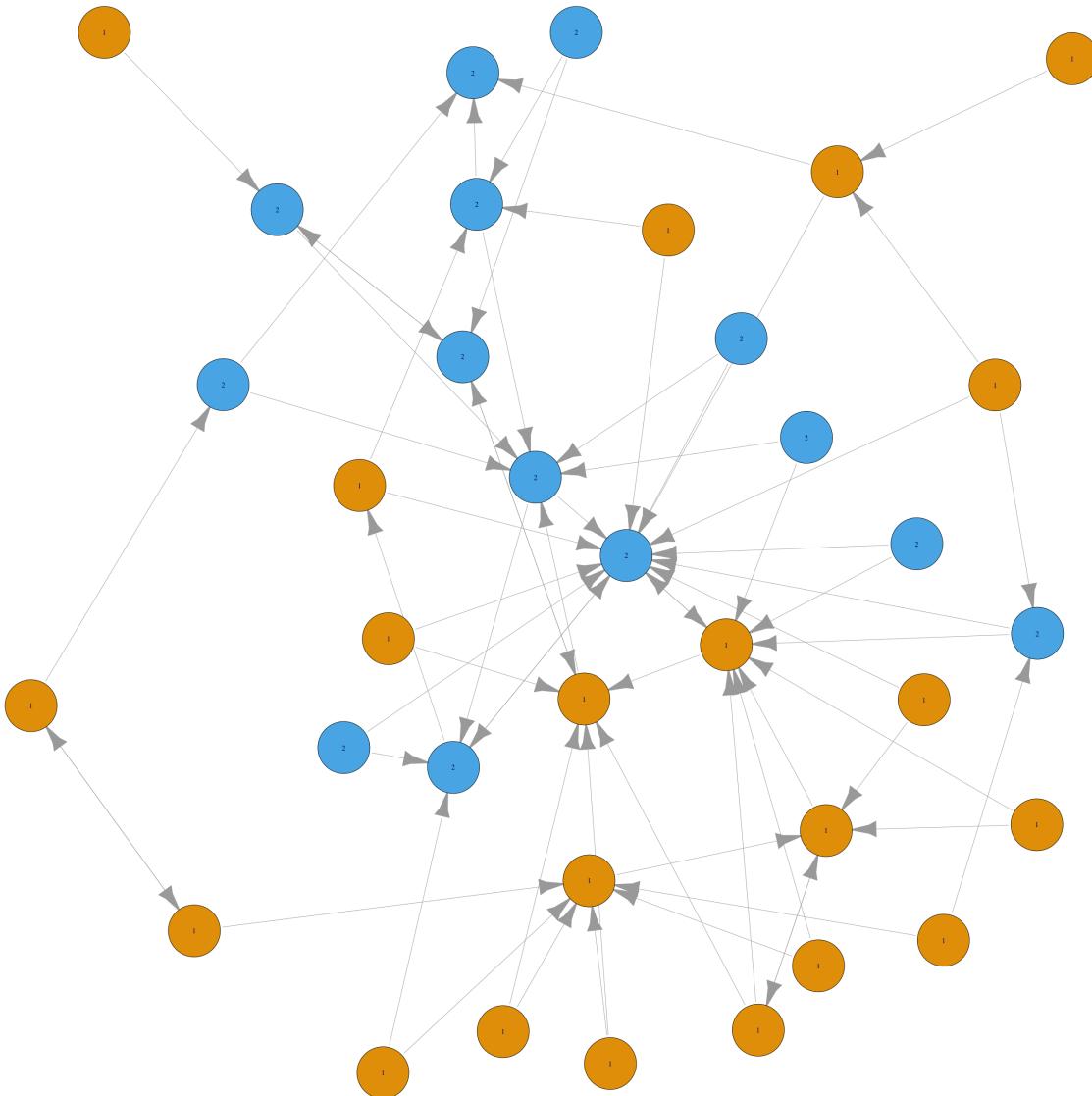


Figure 2: Network plot with vertex_size = 10

```
1 n_X_to_Y = length(X_to_Y)
```

With all these commands in hand, all the numbers requested in the table could be calculated by this snippet of code:

```
1 #relations from different genders
2 from_male <- E(grade1_graph)[.from(V(grade1_graph)[gender.code==1])]
3 from_female <- E(grade1_graph)[.from(V(grade1_graph)[gender.code==2])]
4 to_male <- E(grade1_graph)[.to(V(grade1_graph)[gender.code==1])]
5 to_female <- E(grade1_graph)[.to(V(grade1_graph)[gender.code==2])]

6
7
8 #relations from male to male
9 male_male = intersection(from_male, to_male)
10 n_mm = length(male_male) #24
```

	To Male	To Female	Total
From Male	24	17	41
From Female	6	20	26
Total	30	37	67

Table 1: Relations count between genders

```
11
12 #relations from male to female
13 male_female = intersection(from_male, to_female)
14 n_mf = length(male_female) #17
15
16 #relations from female to male
17 female_male = intersection(from_female, to_male)
18 n_fm = length(female_male) #6
19
20 #relations from female to female
21 female_female = intersection(from_female, to_female)
22 n_ff = length(female_female) #20
```

The completed values are shown in Table 1. There are 41 outgoing edges from nodes corresponding men and 26 outgoing edges from nodes corresponding women in the graph. Of the 41 edges originating from men, 24 want to sit next to the same gender (about %59), and of 26 edges originating women in the class, 20 want to sit next to the same gender(about %77). So if we select a random female from the class, The probability of her wanting to sit next to the same gender is about %18 more than if we select a random man from the class. Likewise, if we select a random female from the class, The probability of her wanting to sit next to the opposite gender is about %18 less than if we select a random man from the class.

3.3 Question 3: Compute and display using histograms the in- and out-degrees of students in your network.

This would be an easy task with this code.

```
1 deg.in <- degree(grade1_graph, mode = "in")
2 deg.out <- degree(grade1_graph, mode = "out")
3
4 in_degree_hist = hist(deg.in, col = "orange", border = "blue", labels =
5 TRUE)
6 out_degree_hist = hist(deg.out, col = "orange", border = "blue", labels
= TRUE)
```

The in-degree value shows the popularity of students. If student A wants to sit next to student B, here we can say that student B gets a popularity point. If a lot of students want to sit next to B, then student B is very popular among others. the degree that shows this popularity would be in-degree. The range of histogram for in-degree is from 0 to 14 which means that there exists at least one student that 12 or 13 students want to sit next to them. Unlike the out-degree histogram that ranges from 0 to 3. This, on the other hand, means that there is no student that has more than 3 preferences to sit next to. Both in and out degree histograms are depicted in Figure 3.



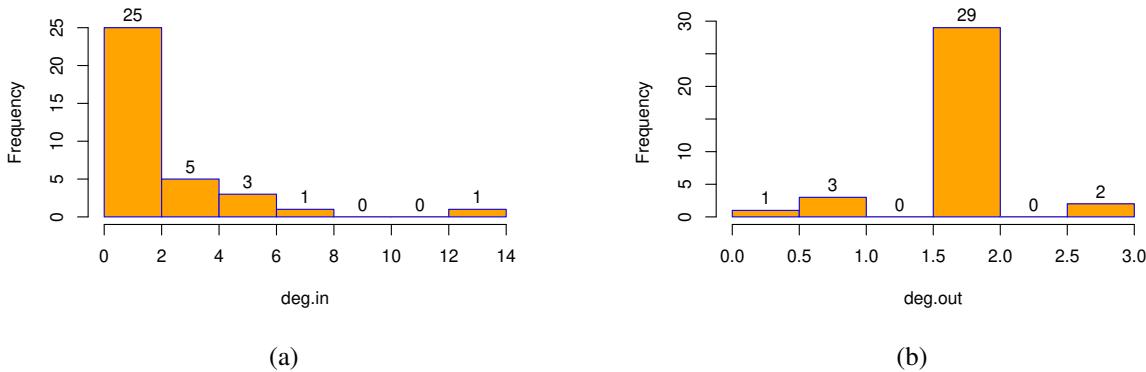


Figure 3: (a) In-degree histogram (b) Out-degree histogram

3.4 Question 4: Compare the mean popularity of males and females in the network. On average, who are more popular?

Here, I see the popularity measure of a gender group as the mean in-degree of that gender group. So, to calculate the popularity of male and female gender groups, this code snippet is used.

```

1 nodes[,4] <- deg.in
2 colnames(nodes)[4] <- "Popularity"
3
4 group_popularity = aggregate(Popularity ~ gender.code, data = nodes,
mean)

```

The popularity of men is 1.43 and the popularity of women is 2.64. This shows that popularity of women is much higher than men in the class.

3.5 Question 5: Compute the standard deviations of the “popularity degree” of males and females. Which gender group has more variability in popularity?

The standard deviation of the popularity of men is 2.42 and it is 3.5 for women. This shows that men have more similar popularity to each other compared to women in the class. In other words, there exist some women in the class who are much more popular than other women.

```

1 male_sd = sd(nodes[,4][which(nodes$gender.code==1)]) #2.42
2 Female_sd = sd(nodes[,4][which(nodes$gender.code==2)]) #3.5

```