Network Science
Final Assignment

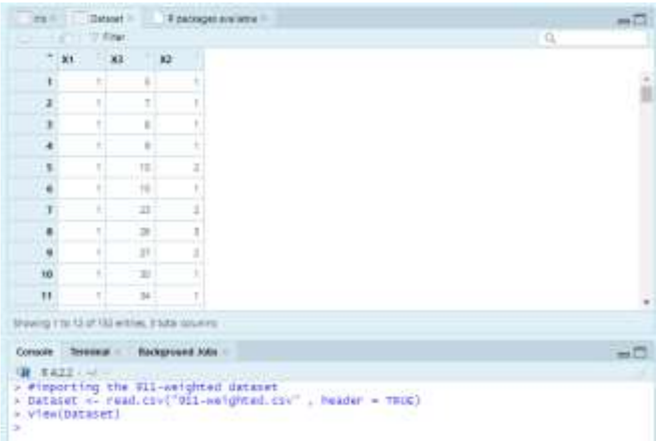
Name: Anas Ali Wassan

Roll No: BSIT/2k19/18

Teacher Name: Sir Abdul Waheed Mahesar

- **Dataset:**

In this Assignment, we will be using the **911-weighted** dataset. So we need to import the dataset first in R.



- **Packages:**

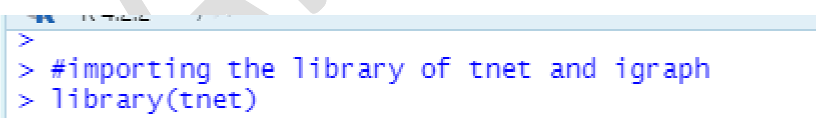
We will find the structural properties of the dataset in R. To obtain this we will be using the **igraph** and **tnet** and **bipartite** packages in R.

- **Structural Properties of Dataset:**

This package will provide the function for calculating various structural properties of the network such as degree, density, local and global clustering, betweenness, and closeness

- **Degree:**

In the SECTION of the network, the degree is a measure of how many connections an individual node or vertex has with other nodes in the network.



```
Console   Terminal   Background Jobs
R 4.2.2 · ~/
Type ?tnet for help.
> library(igraph)
> Degree <- degree_w(Dataset)
> Degree
      node degree output
 [1,]    1     16     23
 [2,]    2      3      6
 [3,]    3      0      0
 [4,]    4      6     10
 [5,]    5      1      1
 [6,]    6      0      0
 [7,]    7      0      0
 [8,]    8      6      6
 [9,]    9      3      3
[10,]   10      0      0
```

- **<u>Density:</u>**

In network science, density refers to the number of connections or edges in a network compared to the total possible number of connections

```
Console   Terminal   Background Jobs
R 4.2.2 · ~/
[55,]   55   1   1
[56,]   56   0   0
[57,]   57   0   0
[58,]   58   3   3
[59,]   59   0   0
[60,]   60   0   0
[61,]   61   5   5
[62,]   62   3   3
> #creating an igraph object
> g <- graph.data.frame(Dataset , directed = FALSE)
> networkDensity <-graph.density(g)
> networkDensity
[1] 0.08038075
>
```

- **<u>Global Clustering.</u>**

In network science, global clustering refers to the overall level of clustering in a network, which is a measure of how interconnected the nodes in a network are.

```
[1] 0.08038075
> globalClustering <- transitivity(g)
> globalClustering
[1] 0.3595506
>
```

- **<u>Local Clustering.</u>**

In network science, local clustering refers to the clustering of a particular node or group of nodes in a network. It is a measure of how closely connected a node or group of nodes is to other Similar nodes in the network.

```
> localClustering <- cluster_optimal(g)
> localClustering
IGRAPH clustering optimal, groups: 6, mod: 0.54
+ groups:
  $`1`
   [1] "1"  "17" "25" "26" "27" "31" "33" "55" "62" "10" "23" "34" "38" "47" "56"

  $`2`
   [1] "2"  "16" "18" "20" "21" "29" "40" "58" "3"  "6"  "19"

  $`3`
  [1] "4"  "5"  "9"  "14" "30" "7"
```

- **Betweenness and closeness.:**

In network science, betweenness and closeness are two measures of the centrality of a node in a network. Centrality refers to the importance or influence of a node in the network. Betweenness is a measure of the number of times a node lies on the shortest path between two other nodes in the network.

```
  + ... omitted several groups/vertices
> #finding the betweenes of and closeness of the network
> networkCloseness <- closeness(g)
> networkBetweenness <- betweenness(g)
>
> |
```

2. Projecting the dataset from one mode to two mode.

We have to check rather the dataset network is one mode or two, we will do this by using the **bipartite** package in R.

Here is an example to check whether a dataset represents a two-mode network in R.

```
> net = network(Datset , matrixFormat="adjacency" , directed = FALSE)
Error in as.network(x, directed = directed, hyper = hyper, loops = loops,
  object 'Datset' not found
> net = network(Dataset , matrixFormat="adjacency" , directed = FALSE)
> is_bipartite = is.bipartite(net)
> is_bipartite
[1] FALSE
>
```

If the **is_bipartite** return **True** it means the network dataset is two mode but **is_bipartite** returns **false** it means the network dataset is not a two-mode network.

- **<u>Network Analysis with respect to all matrices.</u>**

We are using dataset of the 911 Terrorist attack that happened in the US.
There are total **62** nodes in the dataset and these nodes show the number of people who include in the attack

	X1	X3	X2
143	58	20	1
144	58	21	1
145	61	35	1
146	61	44	1
147	61	47	1
148	61	54	1
149	61	60	1
150	62	3	1
151	62	29	1
152	62	38	1

Node 31 has the highest degree in the network Which represents the person Marwan Al-shehhi it means Marwan Al-shehhi has the highest number of connections in his network.

[25,]	25	9	17
[26,]	26	3	5
[27,]	27	2	4
[28,]	28	0	0
[29,]	29	4	5
[30,]	30	1	1
[31,]	31	18	30
[32,]	32	2	5
[33,]	33	3	9
[34,]	34	0	0
[35,]	35	3	3
[36,]	36	2	2
[37,]	37	0	0
[38,]	38	0	0
[39,]	39	0	0

Muhammad Atta who is node 1 which has the highest number of betweenness in the network

```
> networkBetweenness
          1           2           4           5
1041.946923    9.119048   91.830506    4.474950
         12          13          14          15
  12.997222    1.000000    1.392857   27.946825
         20          21          22          24
 322.344048   10.266667   41.036183    0.000000
         29          30          31          32
  25.446176    4.474950  194.120344   21.600000
         40          41          42          43
   0.000000    0.000000  192.433333  460.730586
         54          55          58          61
   6.500000    0.000000    0.000000   61.500000
         23          34          38          47
  60.000000    0.000000   20.633333   53.686081
         39          49           6          19
  13.083333    2.900000    0.000000    0.000000
         44          45          51          52
   0.000000    0.000000    0.000000    0.000000
>
```

According to Eigen centrality values three nodes (1,31, 62) are the most active members in their network

```
> evcent(net)
 [1] 0.399472868 0.049098990 0.198846763 0.190514248 0.085611788 0.239707255
 [7] 0.044290668 0.089723080 0.036471287 0.163076976 0.095713296 0.029623719
[13] 0.014234187 0.069751899 0.139421303 0.075797470 0.100915769 0.034513568
[19] 0.231469476 0.205271248 0.155332723 0.174848946 0.138901378 0.408713345
[25] 0.052924361 0.177108469 0.025122005 0.006836054 0.029623719 0.011882472
[31] 0.018141795 0.059011786 0.009370280 0.014822084 0.009429062 0.011736577
[37] 0.045877316 0.028707359 0.022958443 0.259961882 0.208980380 0.047013458
[43] 0.120947735 0.047013458 0.158349598 0.079913337 0.146513882 0.202613444
[49] 0.034843139 0.011178320 0.015353454 0.028707359 0.016408317 0.002135085
[55] 0.002135085 0.002135085 0.009646973 0.006945023 0.008326286 0.006945023
[61] 0.006945023 0.009646973
```

- **<u>Network analysis metrics namely *Assortativity* and *Eigen vector centrality*.</u>**

Assortativity and eigenvector centrality are both metrics used in network analysis, which is a field of study that involves the use of mathematical and statistical techniques to understand the structure and behavior of complex networks.

<u>Assortativity</u>

Assortativity, also known as assortative mixing, is a measure of how nodes in a network are connected to each other based on their attributes or characteristics. In other words, it is a measure of how nodes with similar attributes tend to be connected to each other. For example, in a social network, nodes with similar ages or interests may be more likely to be connected to each other.

We will find the assortativity degree using the **igraph** function ‘assortativity_degree’.

```
Console   Terminal ×   Background Jobs ×
R 4.2.2 · ~/
[1] FALSE
>
> #finding the assortavity degree
Error in file(out, "wt") : cannot open the connection
Error in file(out, "wt") : cannot open the connection
Error in file(out, "wt") : cannot open the connection
Error in file(out, "wt") : cannot open the connection
Error in file(out, "wt") : cannot open the connection
Error in file(out, "wt") : cannot open the connection
> tempdir()
[1] "C:\\Users\\Sa\\AppData\\Local\\Temp\\RtmpQzHqOq"
> dir.create(tempdir())
> assortativity.degree(g)
[1] -0.09869433
```

<u>Eigenvector centrality</u>

Eigenvector centrality, on the other hand, is a measure of the importance or influence of a node in a network. This measure is based on the idea that a node is important if it is connected to other important nodes. In other words, the centrality of a node is determined by the centrality of its neighbors. For example, in a social network, a node with many connections to highly influential individuals may be considered to be highly central

We will find the assortativity degree using the **igraph** function 'evcent()' which takes the graph input.

```
Console   Terminal ×   Background Jobs ×
R 4.2.2 · ~/
> #finding the Eigenvector degree
> evcent(net)
 [1] 0.399472868 0.049098990 0.198846763 0.190514248 0.085611788 0.239707255
 [7] 0.044290668 0.089723080 0.036471287 0.163076976 0.095713296 0.029623719
[13] 0.014234187 0.069751899 0.139421303 0.075797470 0.100915769 0.034513568
[19] 0.231469476 0.205271248 0.155332723 0.174848946 0.138901378 0.408713345
[25] 0.052924361 0.177108469 0.025122005 0.006836054 0.029623719 0.011882472
[31] 0.018141795 0.059011786 0.009370280 0.014822084 0.009429062 0.011736577
[37] 0.045877316 0.028707359 0.022958443 0.259961882 0.208980380 0.047013458
[43] 0.120947735 0.047013458 0.158349598 0.079913337 0.146513882 0.202613444
[49] 0.034843139 0.011178320 0.015353454 0.028707359 0.016408317 0.002135085
[55] 0.002135085 0.002135085 0.009646973 0.006945023 0.008326286 0.006945023
[61] 0.006945023 0.009646973
> |
```