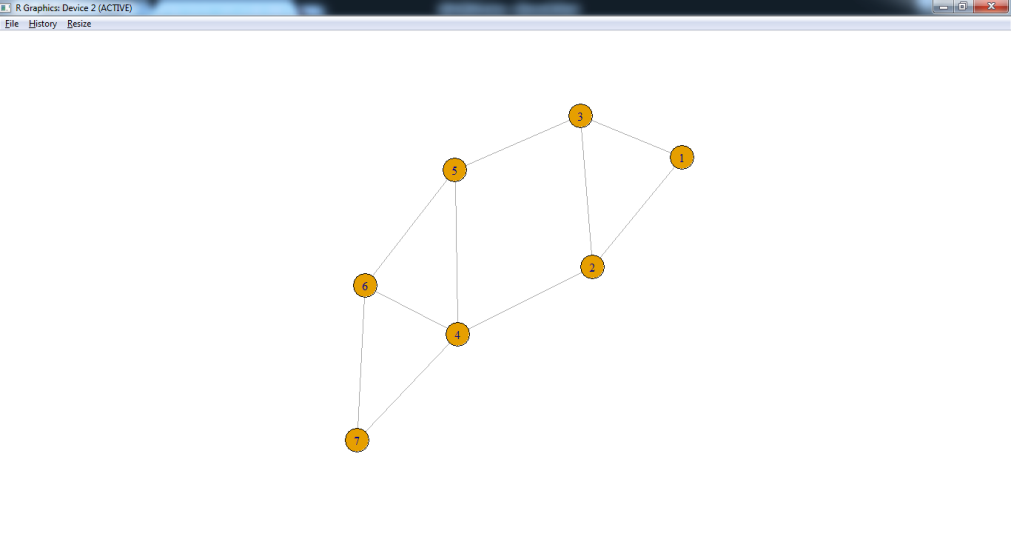
**SNA PRACTICAL**

**Practical No.1:Write a program to compute the following for a given network:**

1. **Number of Edges**
2. **Number of Nodes**
3. **Degree Of Nodes**
4. **Node with lowest degree**
5. **Node with highest degree**
6. **The agjacency list**
7. **Matrix of the Graph**

**CODE:**

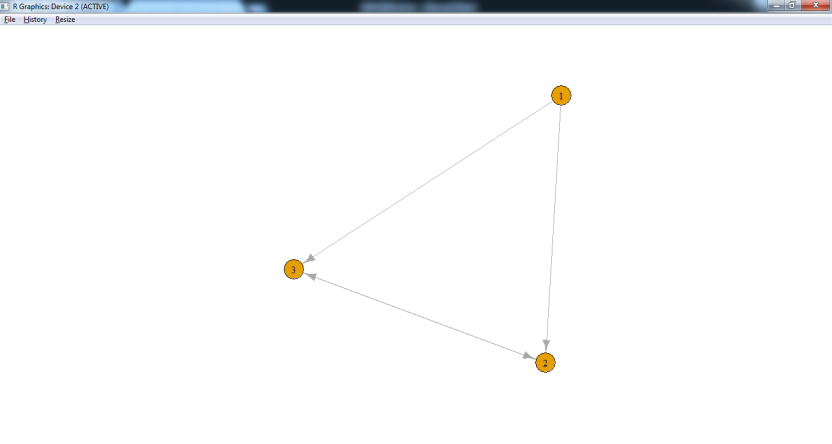
|  |
| --- |
| R version 3.4.1 (2017-06-30) -- "Single Candle"  Copyright (C) 2017 The R Foundation for Statistical Computing  Platform: x86\_64-w64-mingw32/x64 (64-bit)  R is free software and comes with ABSOLUTELY NO WARRANTY.  You are welcome to redistribute it under certain conditions.  Type 'license()' or 'licence()' for distribution details.  R is a collaborative project with many contributors.  Type 'contributors()' for more information and  'citation()' on how to cite R or R packages in publications.  Type 'demo()' for some demos, 'help()' for on-line help, or  'help.start()' for an HTML browser interface to help.  Type 'q()' to quit R.  [Workspace loaded from ~/8/.RData]  Loading required package: rCUR  Loading required package: MASS  Loading required package: Matrix  Loading required package: lattice  > install.packages("igraph");  Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’  (as ‘lib’ is unspecified)  also installing the dependencies ‘magrittr’, ‘pkgconfig’  Warning in install.packages :  cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.4/PACKAGES.rds': HTTP status was '404 Not Found'  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/magrittr\_1.5.zip'  Content type 'application/zip' length 155834 bytes (152 KB)  downloaded 152 KB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/pkgconfig\_2.0.1.zip'  Content type 'application/zip' length 20036 bytes (19 KB)  downloaded 19 KB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/igraph\_1.2.1.zip'  Content type 'application/zip' length 8315334 bytes (7.9 MB)  downloaded 7.9 MB  package ‘magrittr’ successfully unpacked and MD5 sums checked  package ‘pkgconfig’ successfully unpacked and MD5 sums checked  package ‘igraph’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\student\AppData\Local\Temp\RtmpUJlvNX\downloaded\_packages  Warning messages:  1: R graphics engine version 12 is not supported by this version of RStudio. The Plots tab will be disabled until a newer version of RStudio is installed.  2: package ‘rCUR’ was built under R version 3.4.4  3: package ‘lattice’ was built under R version 3.4.4  >  > library(igraph)  Attaching package: ‘igraph’  The following objects are masked from ‘package:stats’:  decompose, spectrum  The following object is masked from ‘package:base’:  union  Warning message:  package ‘igraph’ was built under R version 3.4.4  > g<-graph.formula(1-2,1-3,2-3,2-4,3-5,4-5,4-6,4-7,5-6,6-7)  **Names of Edges & Nodes**  > V(g)  + 7/7 vertices, named, from 4e27f4d:  [1] 1 2 3 4 5 6 7  > E(g)  + 10/10 edges from 4e27f4d (vertex names):  [1] 1--2 1--3 2--3 2--4 3--5 4--5 4--6 4--7 5--6 6--7  **Plotting of Graph**  > plot(g) |
|  |
| |  | | --- | | > | |



**Directed Graph**

dg <-graph.formula(1-+2,1-+3,2++3)

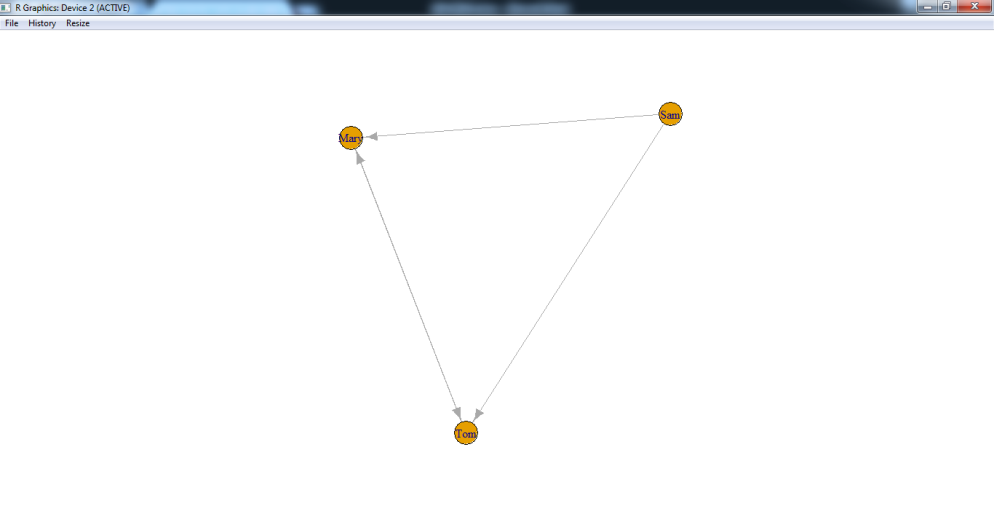
plot(dg)



Graph With Names

> dg1 <-graph.formula(Sam-+Mary,Sam-+Tom,Mary++Tom)

> plot(dg1)



**Number of vertices/nodes**

> vcount(g)

[1] 7

**Number Of Edges**

|  |
| --- |
| > ecount(g)  [1] 10 |
|  |
| |  | | --- | |  | | > degree(g)  1 2 3 4 5 6 7  2 3 3 4 3 3 2  > degree(dg,mode="in")  1 2 3  0 2 2  > degree(dg,mode="out")  1 2 3  2 1 1 | |  | | |  | | --- | |  | | |

**Node with lowest**

> V(dg)$name[degree(dg)==min(degree(dg))]

[1] "1"

**Node with highest**

> V(dg)$name[degree(dg)==max(degree(dg))]

[1] "2" "3"

**To find neighbors/adjacency list**

|  |
| --- |
| > neighbors(g,5)  + 3/7 vertices, named, from 346eba0:  [1] 3 4 6  > neighbors(g,2)  + 3/7 vertices, named, from 346eba0:  [1] 1 3 4  > get.adjlist(dg)  $`1`  + 2/3 vertices, named, from 49edc37:  [1] 2 3  $`2`  + 3/3 vertices, named, from 49edc37:  [1] 1 3 3  $`3`  + 3/3 vertices, named, from 49edc37:  [1] 1 2 2 |
|  |
| |  | | --- | | > | |

**Matrix of the Graph**

|  |
| --- |
| > get.adjacency(g)  7 x 7 sparse Matrix of class "dgCMatrix"  1 2 3 4 5 6 7  1 . 1 1 . . . .  2 1 . 1 1 . . .  3 1 1 . . 1 . .  4 . 1 . . 1 1 1  5 . . 1 1 . 1 .  6 . . . 1 1 . 1  7 . . . 1 . 1 . |
|  |
| |  | | --- | | > | |

**Practical No. 2:Perform following tasks: (i) View data collection forms and/or import onemode/two-mode datasets; (ii) Basic Networks matrices transformations**

**CODE:**

**File name:** “Media-Example-NODES.csv” and “Media-Example-EDGES.csv”

>nodes <- read.csv("D:/MSC 17/SNA//Dataset1-Media-Example-NODES.csv", header=T, as.is=T)

> links <- read.csv("D:/MSC 17/SNA//Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)

> head(nodes)

id media media.type type.label audience.size

1 s01 NY Times 1 Newspaper 20

2 s02 Washington Post 1 Newspaper 25

3 s03 Wall Street Journal 1 Newspaper 30

4 s04 USA Today 1 Newspaper 32

5 s05 LA Times 1 Newspaper 20

6 s06 New York Post 1 Newspaper 50

> head(links)

from to weight type

1 s01 s02 10 hyperlink

2 s01 s02 12 hyperlink

3 s01 s03 22 hyperlink

4 s01 s04 21 hyperlink

5 s04 s11 22 mention

6 s05 s15 21 mention

> nodes2 <- read.csv("D:/MSC 17/SNA//Dataset2-Media-User-Example-NODES.csv", header=T, as.is=T)

> links2 <- read.csv("D:/MSC 17/SNA//Dataset2-Media-User-Example-EDGES.csv", header=T, row.names=1)

> head(nodes2)

id media media.type media.name audience.size

1 s01 NYT 1 Newspaper 20

2 s02 WaPo 1 Newspaper 25

3 s03 WSJ 1 Newspaper 30

4 s04 USAT 1 Newspaper 32

5 s05 LATimes 1 Newspaper 20

6 s06 CNN 2 TV 56

> head(links2)

U01 U02 U03 U04 U05 U06 U07 U08 U09 U10 U11 U12 U13 U14 U15 U16 U17 U18 U19 U20

s01 1 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0

s02 0 0 0 1 1 0 0 0 0 0 0 0 0 0 0 0 0 0 0 1

s03 0 0 0 0 0 1 1 1 1 0 0 0 0 0 0 0 0 0 0 0

s04 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0 0 0

s05 0 0 0 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0 0 0

s06 0 0 0 0 0 0 0 0 0 0 0 0 1 1 0 0 1 0 0 0

**Reading data from a text file**

> read.table("D:/MSC 17/SNA//mydata.txt")

V1 V2 V3

1 100 a1 b1

2 200 a2 b2

3 300 a3 b3

4 400 a4 b4

Basic Networks matrices transformations

> net <- graph.data.frame(d=links, vertices=nodes, directed=T)

> m=as.matrix(net)

> get.adjacency(m)

17 x 17 sparse Matrix of class "dgCMatrix"

[[ suppressing 17 column names ‘s01’, ‘s02’, ‘s03’ ... ]]

s01 . 2 1 1 . . . . . . . . . . 1 . .

s02 1 . 1 . . . . . 1 1 . . . . . . .

s03 1 . . 1 1 . . 1 . 1 1 1 . . . . .

s04 . . 1 . . 1 . . . . 1 1 . . . . 1

s05 1 1 . . . . . . 1 . . . . . 1 . .

s06 . . . . . 1 . . . . . . . . . 1 1

s07 . . 1 . . . . 1 . 1 . . . 1 . . .

s08 . . 1 . . . 1 . 2 . . . . . . . .

s09 . . . . . . . . . 1 . . . . . . .

s10 . . 1 . . . . . . . . . . . . . .

s11 . . . . . . . . . . . . . . . . .

s12 . . . . . 1 . . . . . . 1 1 . . .

s13 . . . . . . . . . . . 1 . . . . 1

s14 . . . . . . . . . . 1 . 1 . . . .

s15 2 . . 1 . 1 . . . . . . . . . . .

s16 . . . . . 1 . . . . . . . . . . 1

s17 . . . 1 . . . . . . . . . . . . .

**Practical No. 3:Compute the following node level measures:**

**(i) Density;**

**(ii) Degree;**

**(iii) Reciprocity;**

**(iv) Transitivity;**

**(v) Centralization;**

**(vi) Clustering.**

**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

Warning message:

package ‘igraph’ was built under R version 3.4.4

**Density-:**

> vcount(g)

[1] 7

> ecount(g)

[1] 10

> ecount(g)/(vcount(g)\*(vcount(g)-1))

[1] 0.2380952

> degree(net)

s01 s02 s03 s04 s05 s06 s07 s08 s09 s10 s11

10 7 13 9 5 8 5 6 5 5 3

s12 s13 s14 s15 s16 s17

6 4 4 6 3 5

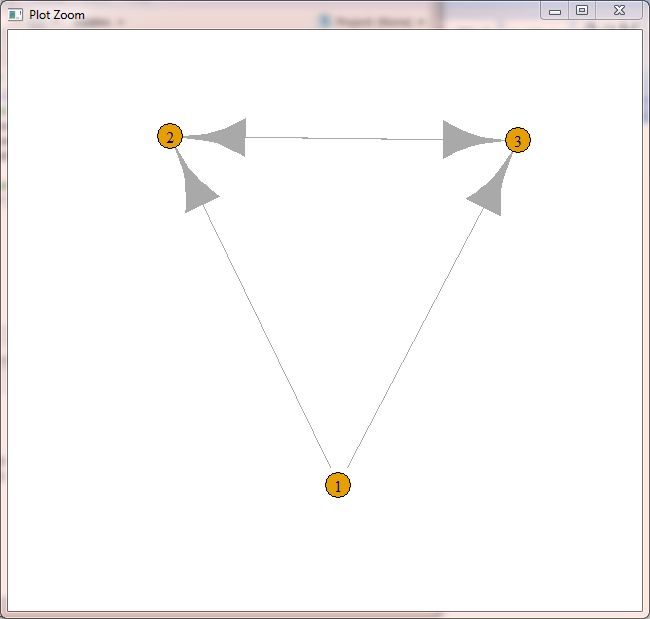
**Reciprocity-:**

> dg <-graph.formula(1-+2,1-+3,2++3)

> plot(dg)

> reciprocity(dg)

[1] 0.5

****

**Formula As per Textbook-:**

> dyad.census(dg)

$mut

[1] 1

$asym

[1] 2

$null

[1] 0

> 2\*dyad.census(dg)$mut/ecount(dg)

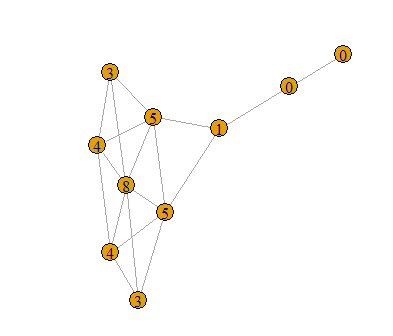
[1] 0.5

**Transitivity-:**

> kite <-graph.famous("Krackhardt Kite")

> atri <- adjacent.triangles(kite)

> plot(kite,vertex.label=atri)

****

> transitivity(kite,type="local")

[1] 0.6666667 0.6666667 1.0000000 0.5333333

[5] 1.0000000 0.5000000 0.5000000 0.3333333

[9] 0.0000000 NaN

**Formula As per Textbook-:**

> adjacent.triangles(kite)/(degree(kite)\*(degree(kite)-1)/2)

[1] 0.6666667 0.6666667 1.0000000 0.5333333

[5] 1.0000000 0.5000000 0.5000000 0.3333333

[9] 0.0000000 NaN

**Centralization-:**

**Degree of Centrality-:**

> centralization.degree(net,mode="in",normalized = T)

$res

[1] 5 3 6 4 1 5 1 2 4 4 3 3 2 2 2 1 4

$centralization

[1] 0.1838235

$theoretical\_max

[1] 272

**Closeness Centralization-:**

> closeness(net,mode="all",weights=NA)

s01 s02 s03 s04

0.03333333 0.03030303 0.04166667 0.03846154

s05 s06 s07 s08

0.03225806 0.03125000 0.03030303 0.02857143

s09 s10 s11 s12

0.02564103 0.02941176 0.03225806 0.03571429

s13 s14 s15 s16

0.02702703 0.02941176 0.03030303 0.02222222

s17

0.02857143

> centralization.closeness(net, mode="all", normalized=T)

$res

[1] 0.5333333 0.4848485 0.6666667 0.6153846

[5] 0.5161290 0.5000000 0.4848485 0.4571429

[9] 0.4102564 0.4705882 0.5161290 0.5714286

[13] 0.4324324 0.4705882 0.4848485 0.3555556

[17] 0.4571429

$centralization

[1] 0.3753596

$theoretical\_max

[1] 7.741935

**Betweeness Centrality-:**

> betweenness(net, directed=T, weights=NA)

s01 s02 s03 s04

26.857143 6.238095 126.511905 92.642857

s05 s06 s07 s08

13.000000 20.333333 1.750000 21.000000

s09 s10 s11 s12

1.000000 15.000000 0.000000 33.500000

s13 s14 s15 s16

20.000000 4.000000 5.666667 0.000000

s17

58.500000

> edge.betweenness(net, directed=T, weights=NA)

[1] 6.619048 6.619048 11.785714 8.333333

[5] 6.500000 11.166667 21.333333 4.250000

[9] 4.250000 16.000000 64.476190 9.500000

[13] 3.261905 3.261905 15.000000 1.000000

[17] 15.000000 17.000000 16.750000 2.000000

[21] 1.250000 8.000000 12.500000 4.000000

[25] 26.000000 18.000000 14.500000 17.000000

[29] 7.500000 4.500000 2.738095 23.000000

[33] 11.000000 31.000000 9.011905 18.000000

[37] 28.500000 0.000000 3.000000 6.500000

[41] 17.000000 8.666667 74.500000 11.750000

[45] 34.000000 4.500000 6.333333 8.809524

[49] 5.333333 3.000000 28.000000 10.000000

> centralization.betweenness(net, directed=T, normalized=T)

$res

[1] 26.857143 6.238095 126.511905

[4] 92.642857 13.000000 20.333333

[7] 1.750000 21.000000 1.000000

[10] 15.000000 0.000000 33.500000

[13] 20.000000 4.000000 5.666667

[16] 0.000000 58.500000

$centralization

[1] 0.4439329

$theoretical\_max

[1] 3840

**Eigenvector Centrality-:**

> centralization.evcent(net, directed=T, normalized=T)

$vector

[1] 0.7694528 0.5623895 1.0000000 0.8569443

[5] 0.3049992 0.9285033 0.1025656 0.3362816

[9] 0.4696841 0.6510633 0.6361813 0.6479337

[13] 0.2674341 0.2289017 0.3277070 0.2831928

[17] 0.7125008

$value

[1] 3.278697

$options

$options$bmat

[1] "I"

$options$n

[1] 17

$options$which

[1] "LR"

$options$nev

[1] 1

$options$tol

[1] 0

$options$ncv

[1] 0

$options$ldv

[1] 0

$options$ishift

[1] 1

$options$maxiter

[1] 1000

$options$nb

[1] 1

$options$mode

[1] 1

$options$start

[1] 1

$options$sigma

[1] 0

$options$sigmai

[1] 0

$options$info

[1] 0

$options$iter

[1] 7

$options$nconv

[1] 1

$options$numop

[1] 31

$options$numopb

[1] 0

$options$numreo

[1] 18

$centralization

[1] 0.4946416

$theoretical\_max

[1] 16

**Clustering-:**

install(igraph)

# let's generate two networks and merge them into one graph.

> g2 <- barabasi.game(50, p=2, directed=F)

> g1 <- watts.strogatz.game(1, size=100, nei=5, p=0.05)

> g <- graph.union(g1,g2)

#Let's remove multi-edges and loops

> g <- simplify(g)

# 1st we calculate the edge betweenness,

> ebc <- edge.betweenness.community(g, directed=F)

> mods <- sapply(0:ecount(g), function(i)

+ {

+ g2 <- delete.edges(g, ebc$removed.edges[seq(length=i)])

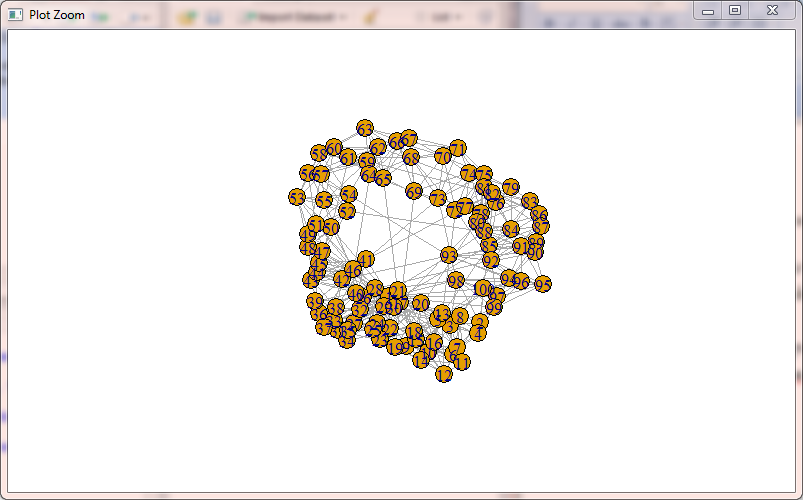
+ cl <- clusters(g2)$membership

+ modularity(g,cl)

+ })

>

> plot(g)

****

# Now, let's color the nodes according to their membership

> g2<-delete.edges(g, ebc$removed.edges[seq(length=which.max(mods)-1)])

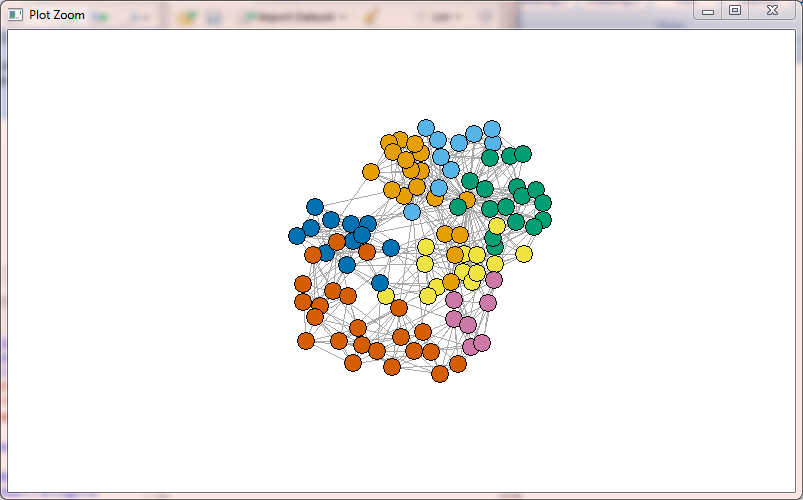
> V(g)$color=clusters(g2)$membership

# Let's choose a layout for the graph

> g$layout <- layout.fruchterman.reingold

# plot it

> plot(g, vertex.label=NA)

****

# fastgreedy.community agorithm

> fc <- fastgreedy.community(g)

> com<-community.to.membership(g, fc$merges, steps= which.max(fc$modularity)-1)

Error in community.to.membership(g, fc$merges, steps = which.max(fc$modularity) - :

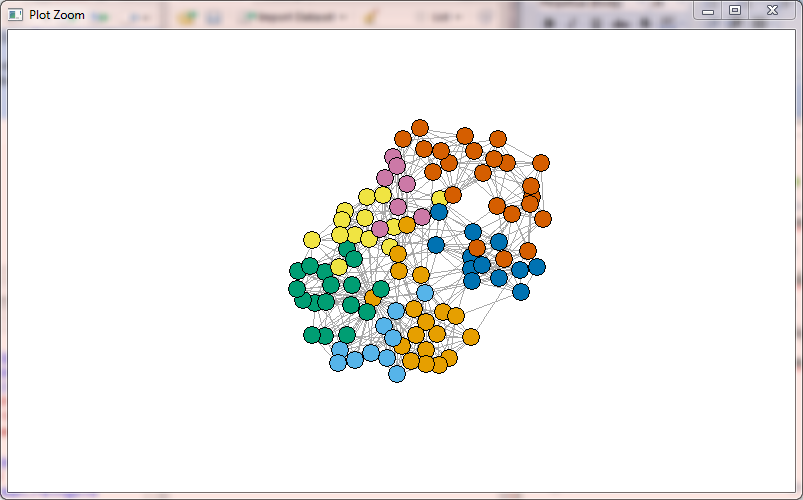
could not find function "community.to.membership"

> V(g)$color <- com$membership+1

Error: object 'com' not found

> g$layout <- layout.fruchterman.reingold

> plot(g, vertex.label=NA)

****

**Practical No.4:For a given network find the following:**

**(i) Length of the shortest path from a given node to another node;**

**(ii) the density of the graph;**

**(iii) Draw egocentric network of node G with chosen configuration parameters.**

**CODE:**

> install.packages("igraph")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/igraph\_1.2.1.zip'

Content type 'application/zip' length 8313645 bytes (7.9 MB)

downloaded 7.9 MB

package ‘igraph’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpQ5A4tE\downloaded\_packages

> install.packages("sna")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/sna\_2.4.zip'

Content type 'application/zip' length 884429 bytes (863 KB)

downloaded 863 KB

package ‘sna’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpQ5A4tE\downloaded\_packages

> install.packages("Dominance")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/Dominance\_1.1.0.zip'

Content type 'application/zip' length 62926 bytes (61 KB)

downloaded 61 KB

package ‘Dominance’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpQ5A4tE\downloaded\_packages

> install.packages("igraph")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/igraph\_1.2.1.zip'

Content type 'application/zip' length 8313645 bytes (7.9 MB)

downloaded 7.9 MB

package ‘igraph’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpQ5A4tE\downloaded\_packages

> install.packages("sna")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/sna\_2.4.zip'

Content type 'application/zip' length 884429 bytes (863 KB)

downloaded 863 KB

package ‘sna’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpQ5A4tE\downloaded\_packages

> install.packages("Dominance")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/Dominance\_1.1.0.zip'

Content type 'application/zip' length 62926 bytes (61 KB)

downloaded 61 KB

package ‘Dominance’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpQ5A4tE\downloaded\_packages

> library(igraph)

Attaching package: ‘igraph’

The following objects are masked from ‘package:stats’:

decompose, spectrum

The following object is masked from ‘package:base’:

union

Warning message:

package ‘igraph’ was built under R version 3.4.4

(i) LENGTH OF THE SHORTEST PATH FROM A GIVEN NODE TO ANOTHER NODE

> matt <- as.matrix(read.table(text=

+ "node R S T U

+ R 7 5 0 0

+ S 7 0 0 2

+ T 0 6 0 0

+ U 4 0 1 0", header=T))

> nms <- matt[,1 ]

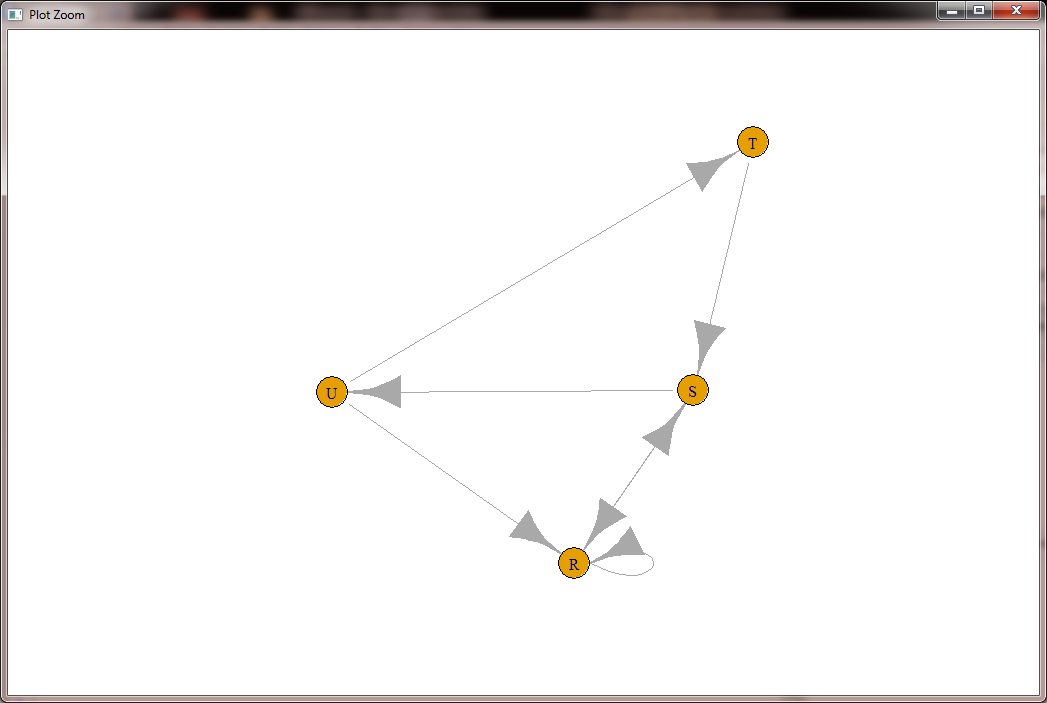
> matt <- matt[, -1]

> colnames(matt) <- rownames(matt) <- nms

> matt[is.na(matt)] <- 0

> g <- graph.adjacency(matt, weighted=TRUE)

> plot(g)



> s.paths <- shortest.paths(g, algorithm = "dijkstra")

> print(s.paths)

R S T U

R 0 5 5 4

S 5 0 3 2

T 5 3 0 1

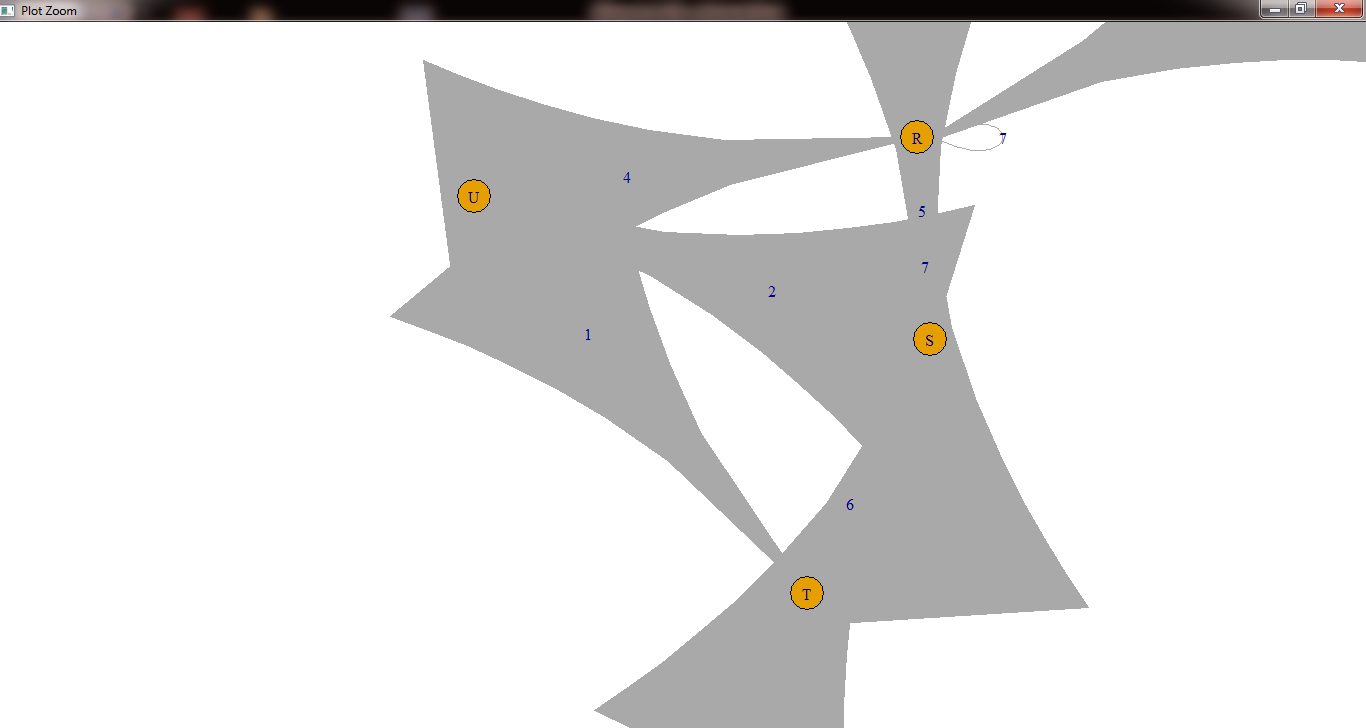
U 4 2 1 0

> shortest.paths(g,v="R",to="S")

S

R 5

> plot(g,edge.label=E(g)$weight)

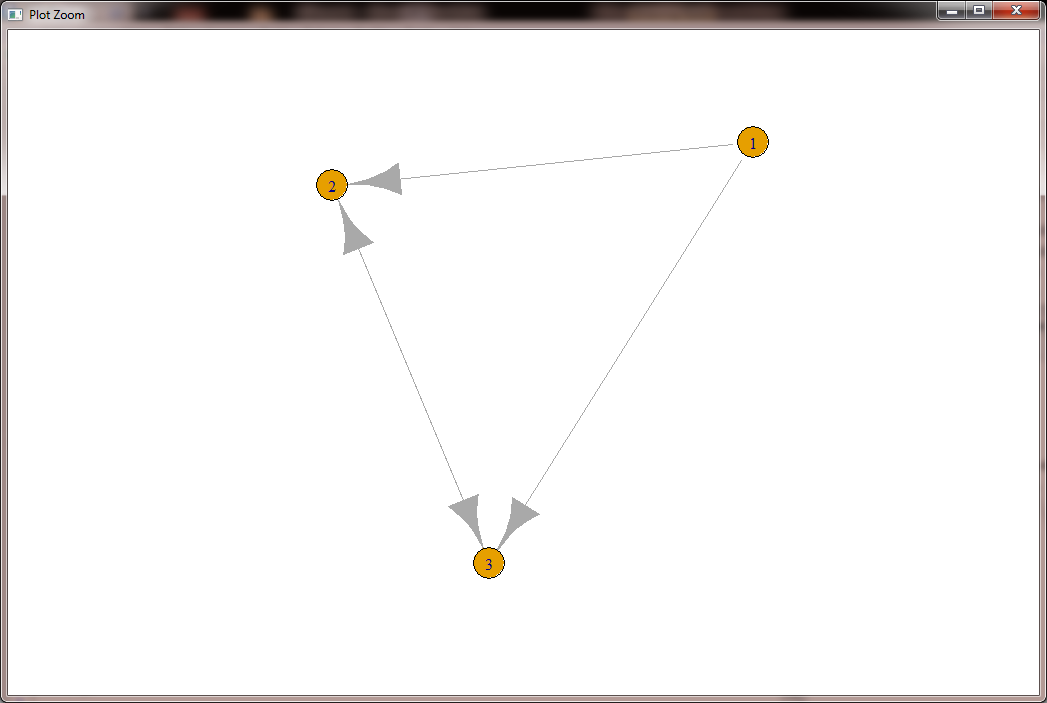


**THE DENSITY OF THE GRAPH**

The density of a graph is the ratio of the number of edges and the number of possible edges.

> dg <- graph.formula(1-+2, 1-+3, 2++3)

> plot(dg)



> graph.density(dg, loops=TRUE)

[1] 0.4444444

i.e(4/9=0.444)

WITHOUT CONSIDERING LOOPS

> graph.density(simplify(dg), loops=FALSE)

[1] 0.6666667

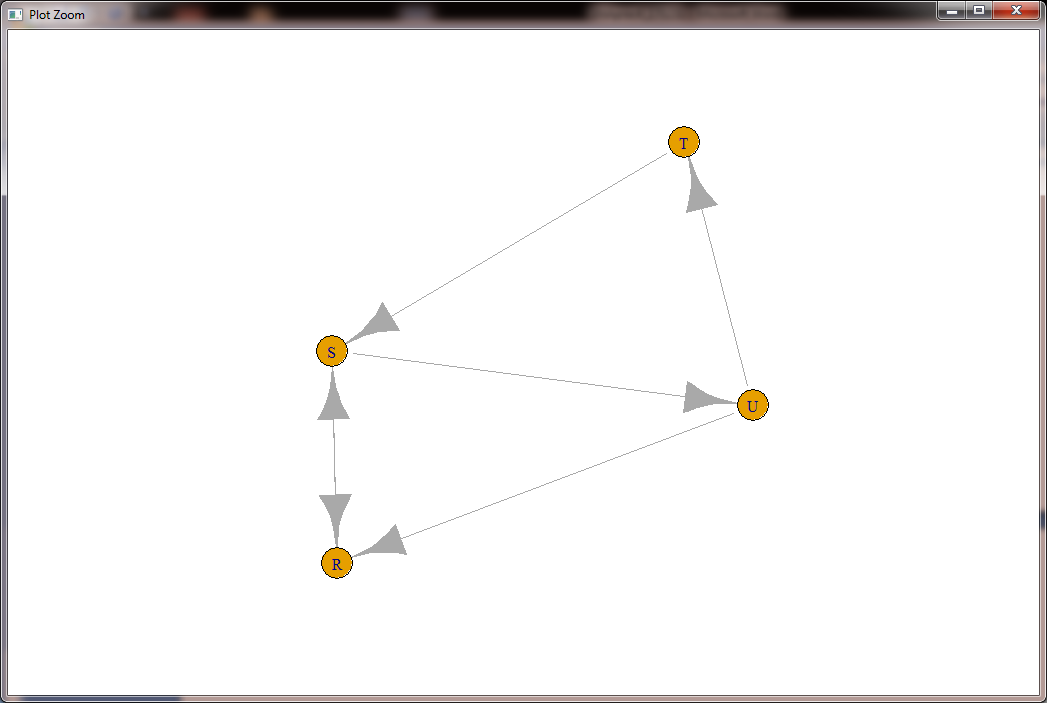
i.e. 4/6 =0.666

**OR**

**WITHOUT LOOP**

> g <- graph.formula(R+-U,R++S,S-+U,U-+T,T-+S)

> plot(g)



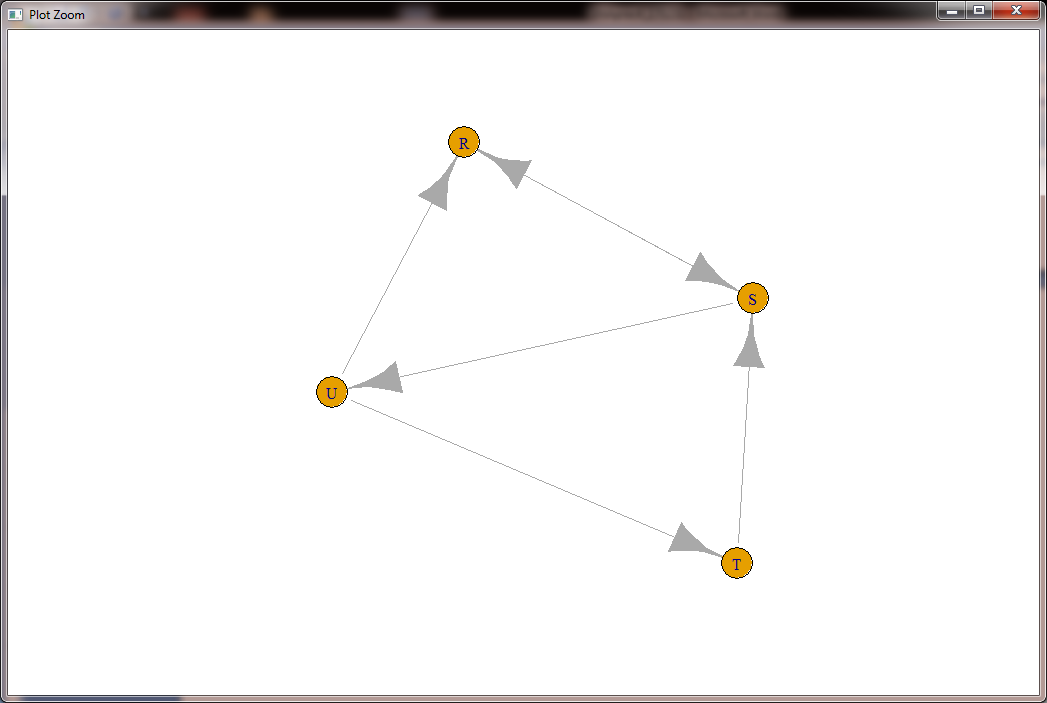
> graph.density(g,loops=FALSE)

[1] 0.5

**WITH LOOP**

> g <- graph.formula(R+-U,R++S,S-+U,U-+T,T-+S,R++R)

> plot(g)

****

> graph.density(g,loops=TRUE)

[1] 0.375

**OTHER EXAMPLE**

> matt <- as.matrix(read.table(text="node V1 V2 V3 v4

+ V1 100 200 400 300

+ V2 200 300 300 100

+ V3 300 100 200 400

+ V4 400 400 100 300",header=T))

> nms <- matt[,1 ]

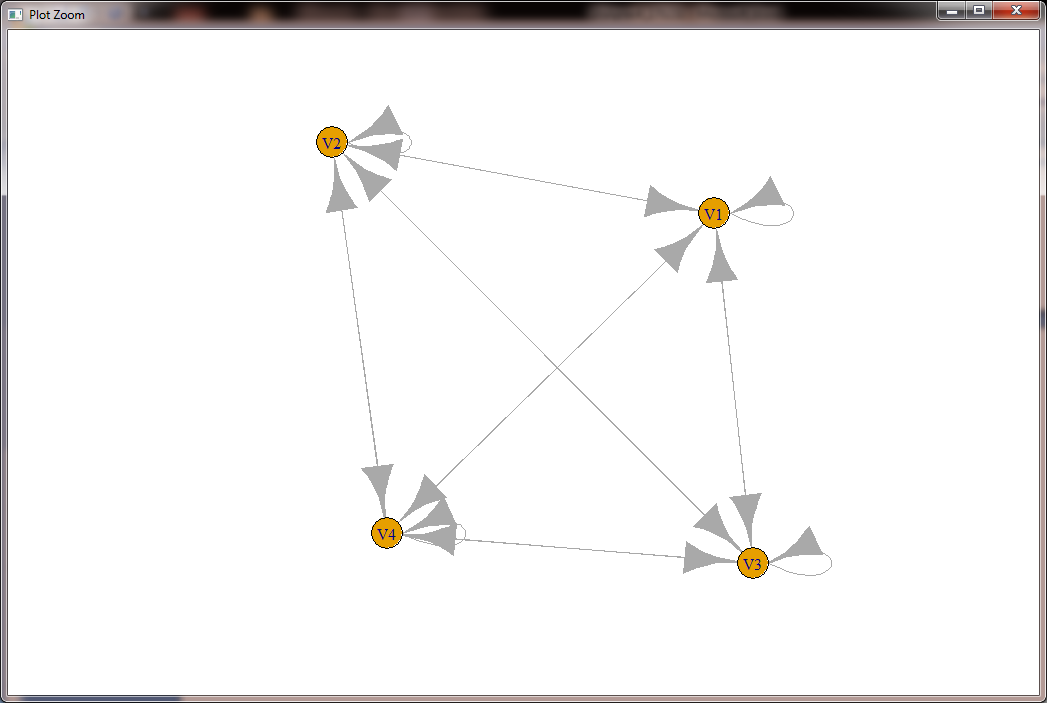
> matt <- matt[, -1]

> colnames(matt) <- rownames(matt) <- nms

> matt[is.na(matt)] <- 0

> g <- graph.adjacency(matt, weighted=TRUE)

> plot(g)



> s.paths <- shortest.paths(g, algorithm = "dijkstra")

> print(s.paths)

V1 V2 V3 V4

V1 0 200 300 300

V2 200 0 100 100

V3 300 100 0 100

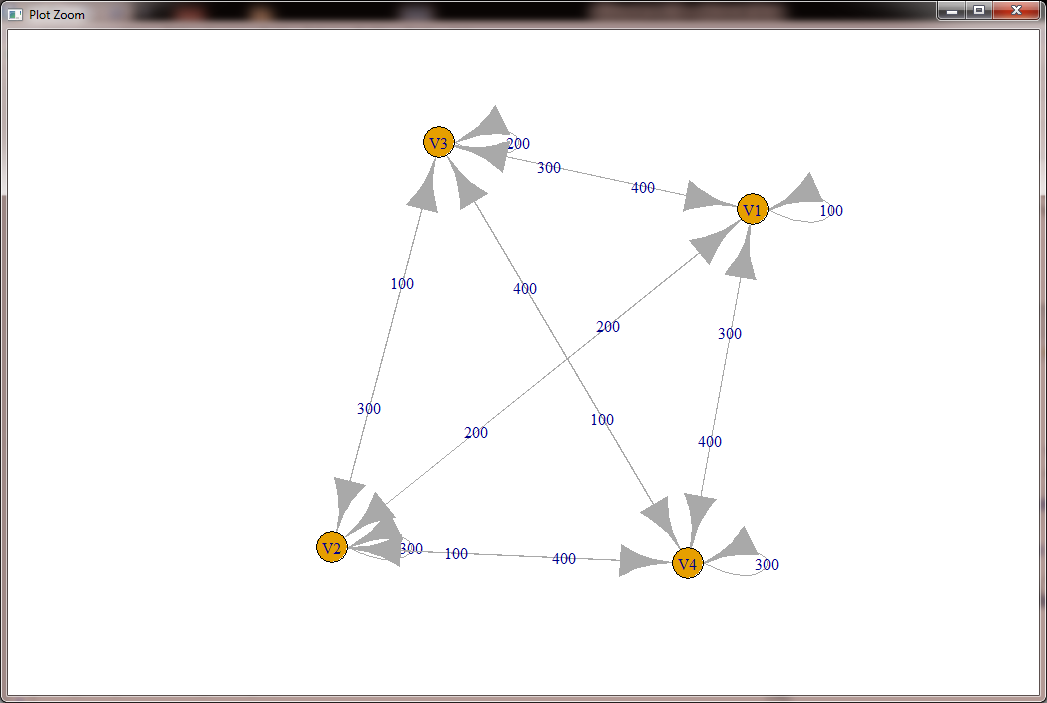
V4 300 100 100 0

> shortest.paths(g, v="V1", to="V2")

V2

V1 200

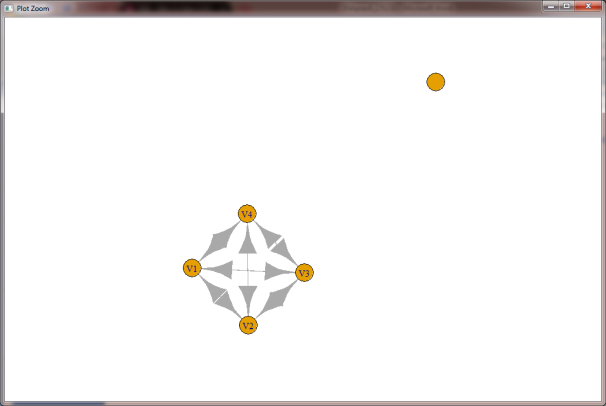
> plot(g, edge.label=E(g)$weight)



**The density of the graph**

> dg <- graph.formula(V1++V1,V1++V2,V1++V3,V1++V4,V2++V1,V2++V2,V2++V3,V2++V4,V3++V1,V3++V2,V3++V3,V3++V4,)

> plot(dg)



Without considering loops

> graph.density(dg, loops=TRUE)

[1] 0.48

> graph.density(simplify(dg), loops=FALSE)

[1] 0.6

**Practical No.5:Write a program to distinguish between:**

**i) a network as a sociogram (or “network graph”)**

**ii) a network as a matrix,&**

**iii) a network as an edge list.**

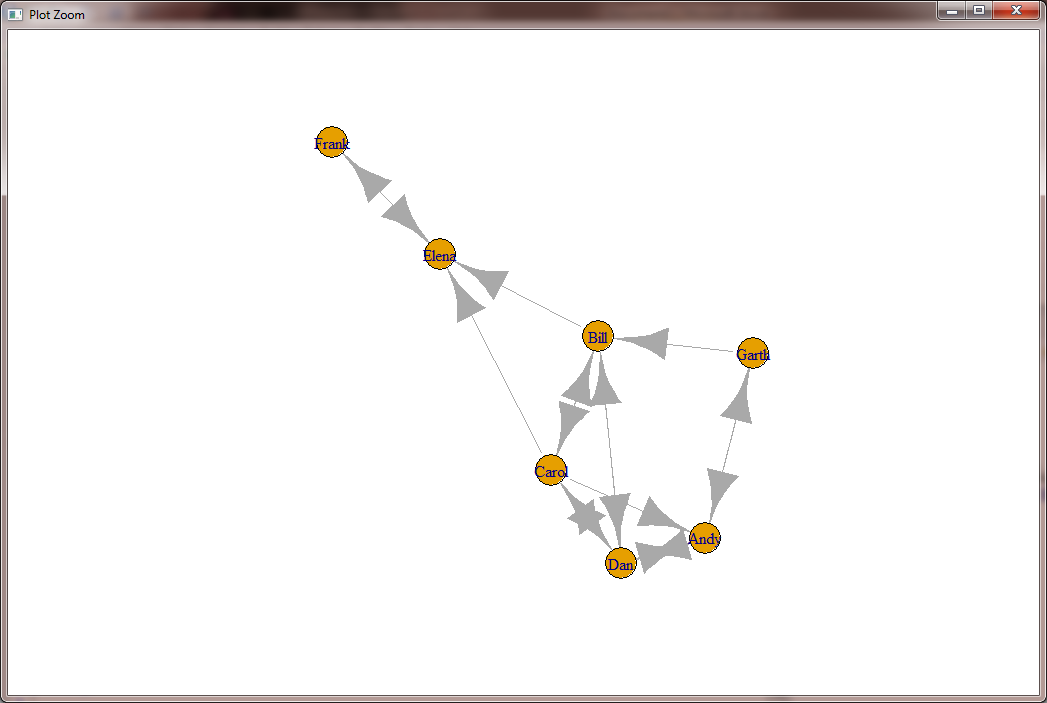
**using 3 distinct networks representatives of each.**

**CODE:**

**A network as a graph**

> ng<-graph.formula(Andy++Garth,Garth-+Bill,Bill-+Elena,Elena++Frank,Carol-+Andy,Carol-+Elena,Carol++Dan,Carol++Bill,Dan++Andy,Dan++Bill)

> plot(ng)



**A network as a matrix**

> get.adjacency(ng)

7 x 7 sparse Matrix of class "dgCMatrix"

Andy Garth Bill Elena Frank Carol Dan

Andy . 1 . . . . 1

Garth 1 . 1 . . . .

Bill . . . 1 . 1 1

Elena . . . . 1 . .

Frank . . . 1 . . .

Carol 1 . 1 1 . . 1

Dan 1 . 1 . . 1 .

**A network as an edge list**

> E(ng)

+ 16/16 edges from 02488bd (vertex names):

[1] Andy ->Garth Andy ->Dan Garth->Andy Garth->Bill Bill ->Elena Bill ->Carol

[7] Bill ->Dan Elena->Frank Frank->Elena Carol->Andy Carol->Bill Carol->Elena

[13] Carol->Dan Dan ->Andy Dan ->Bill Dan ->Carol

> get.adjedgelist(ng,mode="in")

$Andy

+ 3/16 edges from 02488bd (vertex names):

[1] Garth->Andy Carol->Andy Dan ->Andy

$Garth

+ 1/16 edge from 02488bd (vertex names):

[1] Andy->Garth

$Bill

+ 3/16 edges from 02488bd (vertex names):

[1] Garth->Bill Carol->Bill Dan ->Bill

$Elena

+ 3/16 edges from 02488bd (vertex names):

[1] Bill ->Elena Frank->Elena Carol->Elena

$Frank

+ 1/16 edge from 02488bd (vertex names):

[1] Elena->Frank

$Carol

+ 2/16 edges from 02488bd (vertex names):

[1] Bill->Carol Dan ->Carol

$Dan

+ 3/16 edges from 02488bd (vertex names):

[1] Andy ->Dan Bill ->Dan Carol->Dan

**Practical No.6:Write a program to exhibit**

**i) structural equivalence,**

**ii) automatic equivalence,**

**iii) regular equivalence from a network.**

**CODE:**

> install.packages("igraph")

Warning in install.packages :

cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/src/contrib/PACKAGES.rds': HTTP status was '404 Not Found'

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

Warning in install.packages :

cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.4/PACKAGES.rds': HTTP status was '404 Not Found'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/igraph\_1.2.1.zip'

Content type 'application/zip' length 8314961 bytes (7.9 MB)

downloaded 7.9 MB

package ‘igraph’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpyaPm8n\downloaded\_packages

Warning messages:

1: package ‘rCUR’ was built under R version 3.4.4

2: package ‘lattice’ was built under R version 3.4.4

> library(igraph)

Attaching package: ‘igraph’

The following objects are masked from ‘package:stats’:

decompose, spectrum

The following object is masked from ‘package:base’:

union

Warning message:

package ‘igraph’ was built under R version 3.4.4

> install.packages("sna")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/sna\_2.4.zip'

Content type 'application/zip' length 883637 bytes (862 KB)

downloaded 862 KB

package ‘sna’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\RtmpyaPm8n\downloaded\_packages

> library(sna)

Loading required package: statnet.common

Attaching package: ‘statnet.common’

The following object is masked from ‘package:base’:

order

Loading required package: network

network: Classes for Relational Data

Version 1.13.0.1 created on 2015-08-31.

copyright (c) 2005, Carter T. Butts, University of California-Irvine

Mark S. Handcock, University of California -- Los Angeles

David R. Hunter, Penn State University

Martina Morris, University of Washington

Skye Bender-deMoll, University of Washington

For citation information, type citation("network").

Type help("network-package") to get started.

Attaching package: ‘network’

The following objects are masked from ‘package:igraph’:

%c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices, get.edge.attribute, get.edges, get.vertex.attribute,

is.bipartite, is.directed, list.edge.attributes, list.vertex.attributes, set.edge.attribute, set.vertex.attribute

sna: Tools for Social Network Analysis

Version 2.4 created on 2016-07-23.

copyright (c) 2005, Carter T. Butts, University of California-Irvine

For citation information, type citation("sna").

Type help(package="sna") to get started.

Attaching package: ‘sna’

The following objects are masked from ‘package:igraph’:

betweenness, bonpow, closeness, components, degree, dyad.census, evcent, hierarchy, is.connected, neighborhood,

triad.census

Warning messages:

1: package ‘sna’ was built under R version 3.4.4

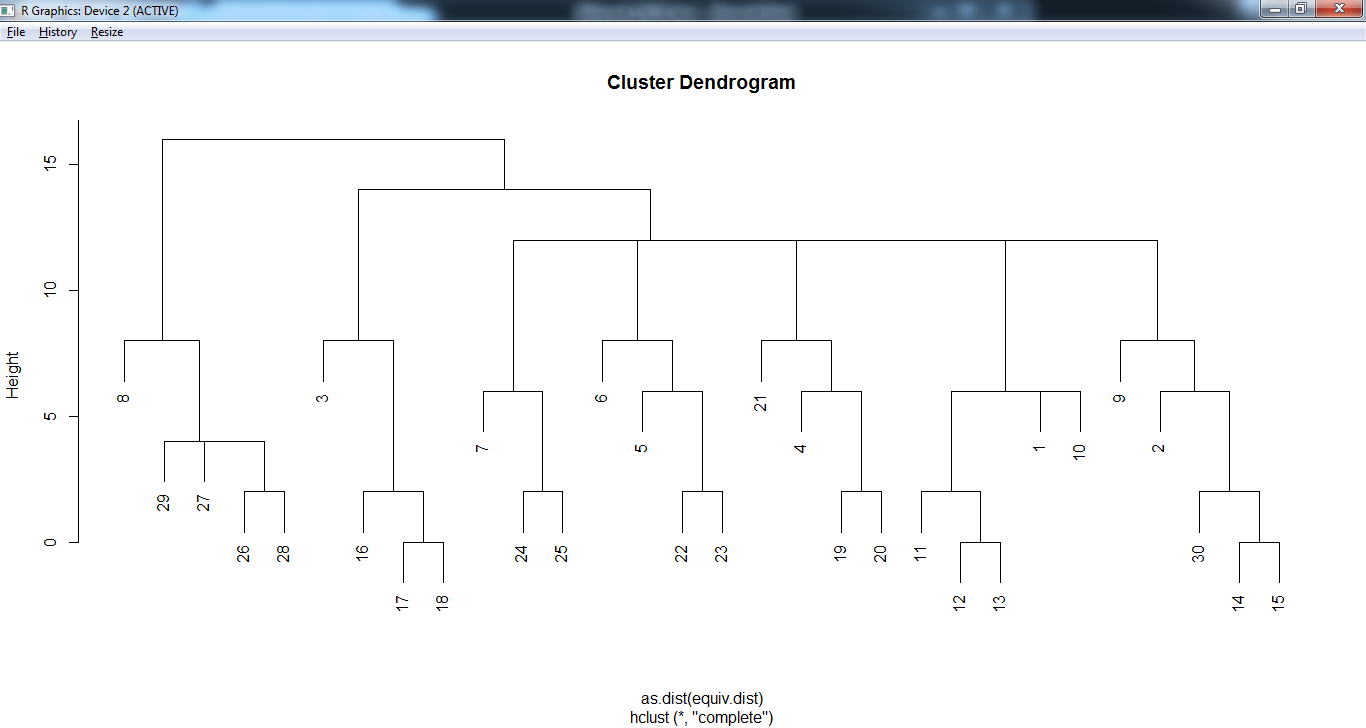
2: package ‘statnet.common’ was built under R version 3.4.4

3: package ‘network’ was built under R version 3.4.4

> links2 <- read.csv("D:/MSC 17/SNA//Dataset2-Media-User-Example-EDGES.csv", header=T, row.names=1)

> eq <-equiv.clust(links2)

> plot(eq)

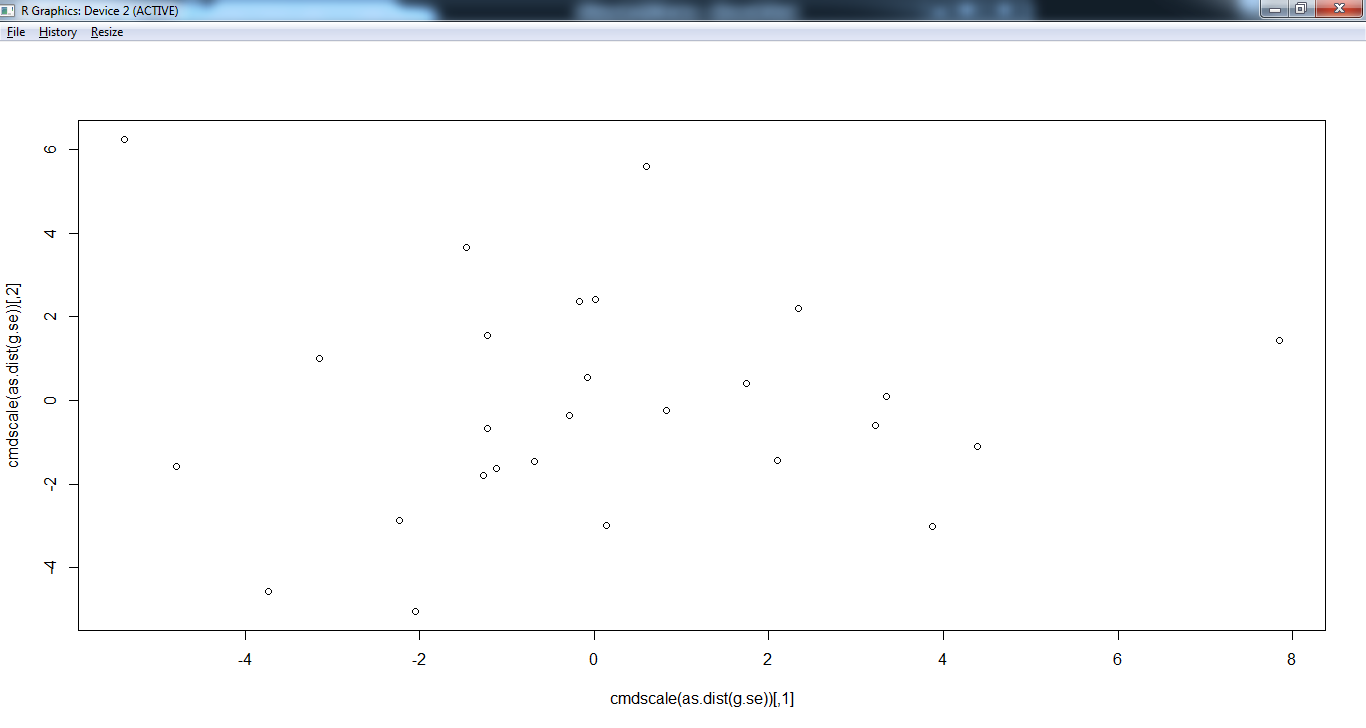


**Get Structural Equivalence distances**

> g.se<-sedist(links2)

**Plot a metric MDS of vertex positions in two dimensions**

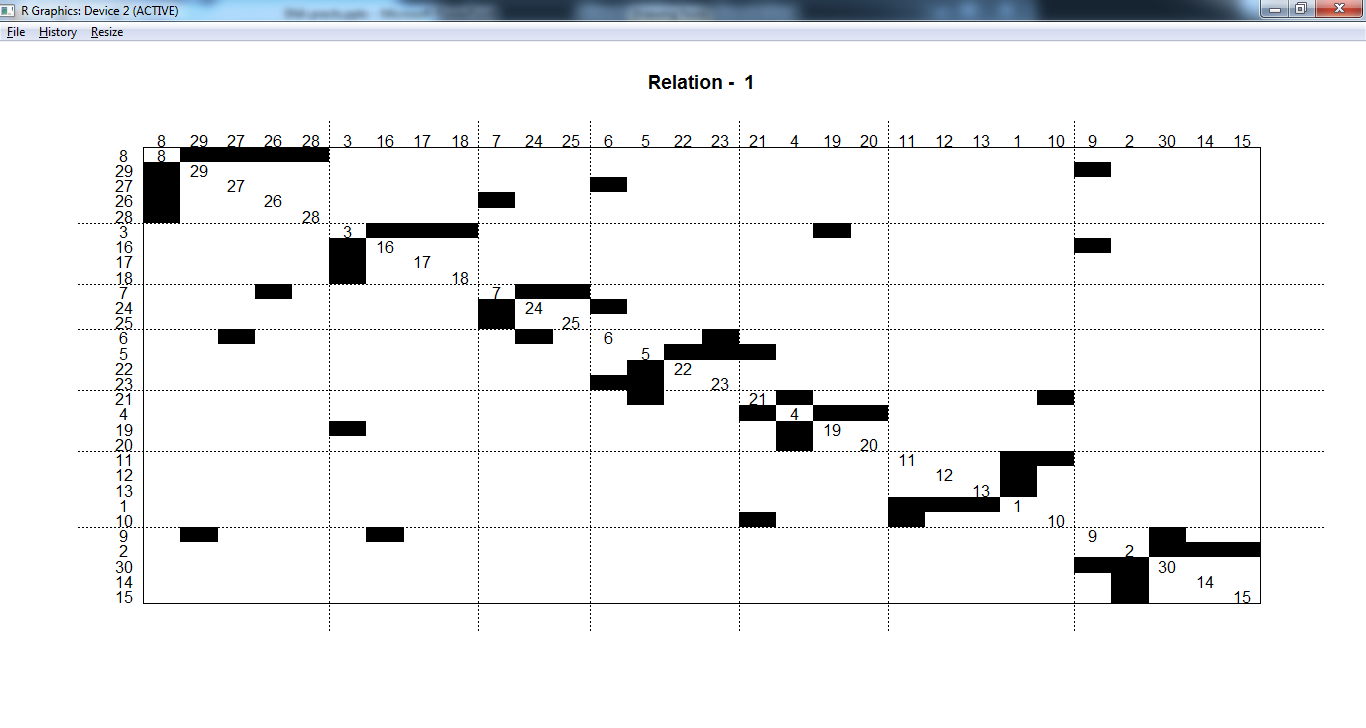
> plot(cmdscale(as.dist(g.se)))



**Blockmodeling**

> b<-blockmodel(links2,eq,h=10)

> plot(b)



**Practical No.7:Create sociograms for the persons-by-persons network and the committee-bycommittee network for a given relevant problem. Create one-mode network and two-node network for the same.**

**CODE:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| > install.packages("sna")  Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’  (as ‘lib’ is unspecified)  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/sna\_2.4.zip'  Content type 'application/zip' length 884077 bytes (863 KB)  downloaded 863 KB  package ‘sna’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\student\AppData\Local\Temp\RtmpI78qUQ\downloaded\_packages  > library(sna)  Loading required package: statnet.common  Attaching package: ‘statnet.common’  The following object is masked from ‘package:base’:  order  Loading required package: network  network: Classes for Relational Data  Version 1.13.0.1 created on 2015-08-31.  copyright (c) 2005, Carter T. Butts, University of California-Irvine  Mark S. Handcock, University of California -- Los Angeles  David R. Hunter, Penn State University  Martina Morris, University of Washington  Skye Bender-deMoll, University of Washington  For citation information, type citation("network").  Type help("network-package") to get started.  sna: Tools for Social Network Analysis  Version 2.4 created on 2016-07-23.  copyright (c) 2005, Carter T. Butts, University of California-Irvine  For citation information, type citation("sna").  Type help(package="sna") to get started.  Warning messages:  1: package ‘sna’ was built under R version 3.4.4  2: package ‘statnet.common’ was built under R version 3.4.4  3: package ‘network’ was built under R version 3.4.4  > install.packages("igraph")  Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’  (as ‘lib’ is unspecified)  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/igraph\_1.2.1.zip'  Content type 'application/zip' length 8315647 bytes (7.9 MB)  downloaded 7.9 MB  package ‘igraph’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\student\AppData\Local\Temp\RtmpI78qUQ\downloaded\_packages  > library(igraph)  Attaching package: ‘igraph’  The following objects are masked from ‘package:sna’:  betweenness, bonpow, closeness, components, degree, dyad.census, evcent, hierarchy, is.connected, neighborhood,  triad.census  The following objects are masked from ‘package:network’:  %c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices, get.edge.attribute, get.edges, get.vertex.attribute,  is.bipartite, is.directed, list.edge.attributes, list.vertex.attributes, set.edge.attribute, set.vertex.attribute  The following objects are masked from ‘package:stats’:  decompose, spectrum  The following object is masked from ‘package:base’:  union  Warning message:  package ‘igraph’ was built under R version 3.4.4  >  > install.packages("Dominance")  Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’  (as ‘lib’ is unspecified)  also installing the dependencies ‘gtools’, ‘XLConnectJars’, ‘rJava’, ‘chron’, ‘gdata’, ‘XLConnect’  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/gtools\_3.8.1.zip'  Content type 'application/zip' length 273740 bytes (267 KB)  downloaded 267 KB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/XLConnectJars\_0.2-15.zip'  Content type 'application/zip' length 14788644 bytes (14.1 MB)  downloaded 14.1 MB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/rJava\_0.9-10.zip'  Content type 'application/zip' length 726100 bytes (709 KB)  downloaded 709 KB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/chron\_2.3-52.zip'  Content type 'application/zip' length 113264 bytes (110 KB)  downloaded 110 KB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/gdata\_2.18.0.zip'  Content type 'application/zip' length 1187178 bytes (1.1 MB)  downloaded 1.1 MB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/XLConnect\_0.2-15.zip'  Content type 'application/zip' length 9750110 bytes (9.3 MB)  downloaded 9.3 MB  trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/Dominance\_1.1.0.zip'  Content type 'application/zip' length 63013 bytes (61 KB)  downloaded 61 KB  package ‘gtools’ successfully unpacked and MD5 sums checked  package ‘XLConnectJars’ successfully unpacked and MD5 sums checked  package ‘rJava’ successfully unpacked and MD5 sums checked  package ‘chron’ successfully unpacked and MD5 sums checked  package ‘gdata’ successfully unpacked and MD5 sums checked  package ‘XLConnect’ successfully unpacked and MD5 sums checked  package ‘Dominance’ successfully unpacked and MD5 sums checked  The downloaded binary packages are in  C:\Users\student\AppData\Local\Temp\RtmpI78qUQ\downloaded\_packages  **Console:**  > data(data\_Network\_1)  > bytes= "00111111111000000000"  > Sociogram(data\_Network\_1,bytes)  $sociogram  IGRAPH bf9b4ed D--- 12 463 --  + attr: color (v/c), size (v/n), label (v/n), color (e/c), width (e/n), curved (e/n)  + edges from bf9b4ed:  [1] 6-> 8 8-> 2 5-> 7 6->10 6->11 8->10 8-> 5 10-> 3 2-> 6 6-> 2 4-> 9 3-> 9 6-> 2 6->11 7->11 8-> 5 2-> 7 6-> 4 6-> 5  [20] 10-> 1 9-> 4 9-> 8 2-> 6 3-> 8 3->11 6->11 10-> 5 8->11 6-> 2 10-> 8 2-> 3 9->11 8-> 1 3-> 9 3->11 3-> 9 2-> 6 6-> 2  [39] 7-> 8 6-> 2 6-> 2 2-> 6 2-> 6 3-> 5 6-> 2 9-> 3 6->10 6-> 9 6->11 6-> 5 10-> 7 9-> 8 6-> 7 5-> 3 5->11 6-> 2 2-> 6  [58] 3-> 9 9-> 3 3-> 9 9-> 3 3-> 9 9-> 3 6-> 2 7-> 9 4-> 9 9-> 5 6-> 2 6-> 7 7-> 8 1-> 3 1-> 9 6-> 2 2-> 6 6-> 2 2-> 6  [77] 3-> 9 9-> 3 6-> 2 2-> 6 10-> 3 10-> 9 3-> 9 9-> 3 1->11 11-> 1 6-> 2 10-> 1 6-> 2 2-> 6 10-> 5 10->11 6-> 1 2-> 1 10-> 9  [96] 10-> 3 11-> 2 4->10 9-> 8 3-> 8 10-> 4 6-> 2 2-> 6 3-> 5 2-> 7 9->11 2->11 2-> 6 6-> 2 6-> 2 2-> 6 11-> 2 2-> 6 6-> 2  [115] 6->11 10-> 8 10-> 1 10-> 5 4-> 3 4-> 8 6-> 2 2-> 6 6-> 3 10-> 5 2-> 9 4-> 5 5-> 4 8-> 1 11-> 2 1->10 6-> 7 1-> 6 10-> 1  [134] 1->10 3-> 9 9-> 3 6->10 2-> 6 1-> 8 1-> 4 7-> 4 10-> 1 5->10 6-> 2 2-> 6 11-> 2 11-> 6 11-> 1 5-> 4 4-> 5 9-> 5 9-> 3  + ... omitted several edges  $counts\_circles  [1] 58 97 0 86 75 49 0 0 135 56 101 104 89 74 2 0  $count\_interactions  [1] 4 3 1 8 9 5 6 6 24 28 6 14 28 9 2 6 4 6 2 8 7 10 24 6 2 9 5 4 28 4 2 5 6 14 2 14 24 28 6 28 28 24 24 5 28  [46] 15 8 2 9 2 3 10 5 3 2 28 24 14 15 14 15 14 15 28 2 6 4 28 5 6 1 1 28 24 28 24 14 15 28 24 6 5 14 15 2 3 28 8 28 24  [91] 5 2 4 4 5 6 11 5 10 6 7 28 24 5 4 5 5 24 28 28 24 11 24 28 9 4 8 5 4 6 28 24 2 5 4 3 3 6 11 5 5 3 8 5 14  [136] 15 8 24 5 3 3 8 3 28 24 11 10 3 3 3 4 15 14 24 28 6 5 5 3 2 2 5 3 3 6 4 6 4 24 6 6 9 2 8 5 4 15 5 8 7  [181] 4 5 5 6 6 28 5 4 6 1 6 2 2 4 2 8 3 6 6 4 2 6 15 4 6 5 1 15 6 3 6 4 10 5 3 3 6 2 6 5 6 4 5 1 3  [226] 6 5 5 3 9 2 2 5 4 2 6 5 5 5 4 5 28 9 5 2 4 5 4 4 6 5 8 10 5 6 6 1 6 10 4 8 9 3 2 1 5 5 6 6 4  [271] 3 8 10 5 6 6 3 3 6 4 7 4 6 11 4 10 2 28 4 4 10 3 10 8 6 2 8 2 7 6 3 8 3 6 8 1 28 1 2 4 10 14 4 3 4  [316] 4 8 6 3 5 5 4 7 6 2 4 6 24 6 15 4 8 6 24 10 5 3 4 6 4 1 6 8 2 24 28 2 4 14 4 8 8 3 3 3 3 5 2 10 2  [361] 3 8 10 4 5 2 6 4 7 5 6 7 24 2 7 2 2 3 2 3 4 6 1 1 6 5 6 3 8 6 24 2 4 4 5 6 7 8 4 7 7 2 2 6 2  [406] 4 6 4 3 7 7 8 11 11 11 4 10 6 4 1 4 4 6 6 8 4 2 8 4 4 3 6 4 4 8 28 6 6 3 3 4 15 10 10 11 2 2 14 15 6  [451] 6 4 3 15 14 10 8 10 8 11 5 5 11  $line\_size  [1] 0.3333333 9.3333333      **Rotate -90 degree**    **Rotate 90 degree**    **Rotate -45 degree**     |  | | --- | | > print(data\_Network\_1)  Name Beschreibung item.number dominance.order  1 1 Pferd1 1 1  2 2 Pferd2 2 2  3 3 Pferd3 3 NA  4 4 Pferd4 4 5  5 5 Pferd5 5 10  6 6 Pferd6 6 3  7 7 Pferd7 7 6  8 8 Pferd8 8 NA  9 9 Pferd9 9 4  10 10 Pferd10 10 9  11 11 Pferd11 11 8  12 12 Pferd12 12 <NA>  13 13 Pferd13 13 <NA>  14 14 Pferd14 14 <NA>  15 15 Pferd15 15 <NA>  16 16 Pferd16 16 NA  17 NA <NA> NA <NA>  18 NA <NA> NA <NA>  19 NA <NA> NA <NA>  20 NA <NA> NA <NA>  21 NA <NA> NA <NA>  22 NA <NA> NA <NA>  23 NA <NA> NA <NA>  24 NA <NA> NA <NA>  25 NA <NA> NA <NA>  26 NA <NA> NA <NA>  27 NA <NA> NA <NA>  28 NA <NA> NA <NA>  29 NA <NA> NA <NA>  30 NA <NA> NA <NA>  31 NA <NA> NA <NA>  32 NA <NA> NA <NA>  33 NA <NA> NA <NA>  34 NA <NA> NA <NA>  35 NA <NA> NA <NA>  36 NA <NA> NA <NA>  37 NA <NA> NA <NA>  38 NA <NA> NA <NA>  39 NA <NA> NA <NA>  40 NA <NA> NA <NA>  41 NA <NA> NA <NA>  42 NA <NA> NA <NA>  43 NA <NA> NA <NA>  44 NA <NA> NA <NA>  45 NA <NA> NA <NA>  46 NA <NA> NA <NA>  47 NA <NA> NA <NA>  48 NA <NA> NA <NA>  49 NA <NA> NA <NA>  50 NA <NA> NA <NA>  51 NA <NA> NA <NA>  52 NA <NA> NA <NA>  53 NA <NA> NA <NA>  54 NA <NA> NA <NA>  55 NA <NA> NA <NA>  56 NA <NA> NA <NA>  57 NA <NA> NA <NA>  58 NA <NA> NA <NA>  59 NA <NA> NA <NA>  60 NA <NA> NA <NA>  61 NA <NA> NA <NA>  62 NA <NA> NA <NA>  age sex action.from. action.to kind.of.action  1 NA 2 4 9 11  2 NA 1 9 4 11  3 NA 1 4 12 11  4 NA 1 12 4 11  5 NA 1 5 9 11  6 NA 1 9 5 11  7 NA 1 5 9 1  8 NA 1 9 5 2  9 NA 1 9 4 1  10 NA 1 4 9 2  11 NA 1 5 9 1  12 NA 1 9 5 2  13 NA 1 9 4 1  14 NA 1 4 9 2  15 NA 1 11 4 1  16 NA NA 4 11 2  17 <NA> <NA> 9 11 3  18 <NA> <NA> 11 9 9  19 <NA> <NA> 2 10 10  20 <NA> <NA> 2 10 11  21 <NA> <NA> 10 2 11  22 <NA> <NA> 11 2 3  23 <NA> <NA> 11 2 9  24 <NA> <NA> 6 10 3  25 <NA> <NA> 9 13 3  26 <NA> <NA> 13 9 9  27 <NA> <NA> 9 14 3  28 <NA> <NA> 14 9 9  29 <NA> <NA> 13 6 14  30 <NA> <NA> 6 13 18  31 <NA> <NA> 5 13 17  32 <NA> <NA> 13 5 18  33 <NA> <NA> 11 13 3  34 <NA> <NA> 11 13 9  35 <NA> <NA> 11 6 3  36 <NA> <NA> 11 6 4  37 <NA> <NA> 6 11 6  38 <NA> <NA> 11 6 9  39 <NA> <NA> 13 4 3  40 <NA> <NA> 4 13 9  41 <NA> <NA> 2 9 3  42 <NA> <NA> 10 4 5  43 <NA> <NA> 4 10 9  44 <NA> <NA> 10 4 5  45 <NA> <NA> 4 10 9  46 <NA> <NA> 13 4 5  47 <NA> <NA> 4 13 9  48 <NA> <NA> 5 4 5  49 <NA> <NA> 4 5 9  50 <NA> <NA> 9 2 3  51 <NA> <NA> 5 12 3  52 <NA> <NA> 12 5 9  53 <NA> <NA> 4 11 8  54 <NA> <NA> 11 4 9  55 <NA> <NA> 5 11 5  56 <NA> <NA> 11 5 9  57 <NA> <NA> 4 12 3  58 <NA> <NA> 9 2 3  59 <NA> <NA> 2 14 8  60 <NA> <NA> 14 2 9  61 <NA> <NA> 2 9 1  62 <NA> <NA> 9 2 2  time test.2.kind.of.action  1 <NA> 3  2 2009-06-07 03:30:00 3  3 <NA> 3  4 <NA> 3  5 <NA> 3  6 <NA> 3  7 <NA> 1  8 <NA> 0  9 <NA> 0  10 <NA> 2  11 <NA> 1  12 <NA> 0  13 <NA> 0  14 <NA> 2  15 <NA> 1  16 <NA> 2  17 <NA> 3  18 <NA> 9  19 <NA> 3  20 <NA> 3  21 <NA> 3  22 <NA> 3  23 <NA> 9  24 <NA> 3  25 <NA> 3  26 <NA> 9  27 <NA> 3  28 <NA> 9  29 <NA> 5  30 <NA> 9  31 <NA> 8  32 <NA> 9  33 <NA> 3  34 <NA> 9  35 <NA> 3  36 <NA> 4  37 <NA> 6  38 <NA> 9  39 <NA> 3  40 <NA> 9  41 <NA> 3  42 <NA> 5  43 <NA> 9  44 <NA> 5  45 <NA> 9  46 <NA> 5  47 <NA> 9  48 1899-12-31 19:53:20 5  49 <NA> 9  50 <NA> 3  51 <NA> 3  52 <NA> 9  53 <NA> 8  54 <NA> 9  55 <NA> 5  56 <NA> 9  57 <NA> 3  58 <NA> 3  59 <NA> 8  60 <NA> 9  61 <NA> 1  62 <NA> 0  test.3.kind.of.acttion name.of.action  1 3 leading  2 3 following  3 3 approach  4 3 bite  5 3 threat to bite  6 3 kick  7 1 threat to kick  8 2 chase  9 0 retreat  10 2 grooming approach  11 1 grooming  12 2 interv. approach  13 0 interv. bite  14 2 interv. threat to bite  15 1 interv. kick  16 2 interv. threat to kick  17 3 interv. chase  18 9 interv. retreat  19 3 interv. grooming approach  20 3 interv. grooming  21 3 <NA>  22 3 <NA>  23 9 <NA>  24 3 <NA>  25 3 <NA>  26 9 <NA>  27 3 <NA>  28 9 <NA>  29 5 <NA>  30 9 <NA>  31 8 <NA>  32 9 <NA>  33 3 <NA>  34 9 <NA>  35 3 <NA>  36 4 <NA>  37 6 <NA>  38 9 <NA>  39 3 <NA>  40 9 <NA>  41 3 <NA>  42 5 <NA>  43 9 <NA>  44 5 <NA>  45 9 <NA>  46 5 <NA>  47 9 <NA>  48 5 <NA>  49 9 <NA>  50 3 <NA>  51 3 <NA>  52 9 <NA>  53 8 <NA>  54 9 <NA>  55 5 <NA>  56 9 <NA>  57 3 <NA>  58 3 <NA>  59 8 <NA>  60 9 <NA>  61 1 <NA>  62 2 <NA>  action.number classification weighting  1 1 1 1  2 2 2 -1  3 3 1 1  4 4 1 1  5 5 1 1  6 6 1 1  7 7 1 1  8 8 1 1  9 9 2 -1  10 10 1 1  11 11 1 1  12 12 1 1  13 13 1 1  14 14 1 1  15 15 1 1  16 16 1 1  17 17 1 1  18 18 2 -1  19 19 1 1  20 20 1 1  21 NA NA NA  22 NA NA NA  23 NA NA NA  24 NA NA NA  25 NA NA NA  26 NA NA NA  27 NA NA NA  28 NA NA NA  29 NA NA NA  30 NA NA NA  31 NA NA NA  32 NA NA NA  33 NA NA NA  34 NA NA NA  35 NA NA NA  36 NA NA NA  37 NA NA NA  38 NA NA NA  39 NA NA NA  40 NA NA NA  41 NA NA NA  42 NA NA NA  43 NA NA NA  44 NA NA NA  45 NA NA NA  46 NA NA NA  47 NA NA NA  48 NA NA NA  49 NA NA NA  50 NA NA NA  51 NA NA NA  52 NA NA NA  53 NA NA NA  54 NA NA NA  55 NA NA NA  56 NA NA NA  57 NA NA NA  58 NA NA NA  59 NA NA NA  60 NA NA NA  61 NA NA NA  62 NA NA NA  [ reached getOption("max.print") -- omitted 738 rows ] | |  | | |  | | --- | |  | | |
|  |
| |  | | --- | |  | |

**Practical No.8:Displaying Bipartite network in the graph format**

**CODE:**

> install.packages("sna")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

Warning in install.packages :

cannot open URL 'http://www.stats.ox.ac.uk/pub/RWin/bin/windows/contrib/3.4/PACKAGES.rds': HTTP status was '404 Not Found'

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/sna\_2.4.zip'

Content type 'application/zip' length 885178 bytes (864 KB)

downloaded 864 KB

package ‘sna’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\Rtmpc1mZmq\downloaded\_packages

Warning messages:

1: R graphics engine version 12 is not supported by this version of RStudio. The Plots tab will be disabled until a newer version of RStudio is installed.

2: package ‘rCUR’ was built under R version 3.4.4

3: package ‘lattice’ was built under R version 3.4.4

> library(sna)

Loading required package: statnet.common

Attaching package: ‘statnet.common’

The following object is masked from ‘package:base’:

order

Loading required package: network

network: Classes for Relational Data

Version 1.13.0.1 created on 2015-08-31.

copyright (c) 2005, Carter T. Butts, University of California-Irvine

Mark S. Handcock, University of California -- Los Angeles

David R. Hunter, Penn State University

Martina Morris, University of Washington

Skye Bender-deMoll, University of Washington

For citation information, type citation("network").

Type help("network-package") to get started.

sna: Tools for Social Network Analysis

Version 2.4 created on 2016-07-23.

copyright (c) 2005, Carter T. Butts, University of California-Irvine

For citation information, type citation("sna").

Type help(package="sna") to get started.

Warning messages:

1: package ‘sna’ was built under R version 3.4.4

2: package ‘statnet.common’ was built under R version 3.4.4

3: package ‘network’ was built under R version 3.4.4

> install.packages("igraph")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/igraph\_1.2.1.zip'

Content type 'application/zip' length 8311197 bytes (7.9 MB)

downloaded 7.9 MB

package ‘igraph’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

C:\Users\student\AppData\Local\Temp\Rtmpc1mZmq\downloaded\_packages

> library(igraph)

Attaching package: ‘igraph’

The following objects are masked from ‘package:sna’:

betweenness, bonpow, closeness, components, degree, dyad.census, evcent, hierarchy, is.connected, neighborhood,

triad.census

The following objects are masked from ‘package:network’:

%c%, %s%, add.edges, add.vertices, delete.edges, delete.vertices, get.edge.attribute, get.edges, get.vertex.attribute,

is.bipartite, is.directed, list.edge.attributes, list.vertex.attributes, set.edge.attribute, set.vertex.attribute

The following objects are masked from ‘package:stats’:

decompose, spectrum

The following object is masked from ‘package:base’:

union

Warning message:

package ‘igraph’ was built under R version 3.4.4

> install.packages("Dominance")

Installing package into ‘C:/Users/student/Documents/R/win-library/3.4’

(as ‘lib’ is unspecified)

trying URL 'https://cran.rstudio.com/bin/windows/contrib/3.4/Dominance\_1.1.0.zip'

Content type 'application/zip' length 62975 bytes (61 KB)

downloaded 61 KB

package ‘Dominance’ successfully unpacked and MD5 sums checked

The downloaded binary packages are in

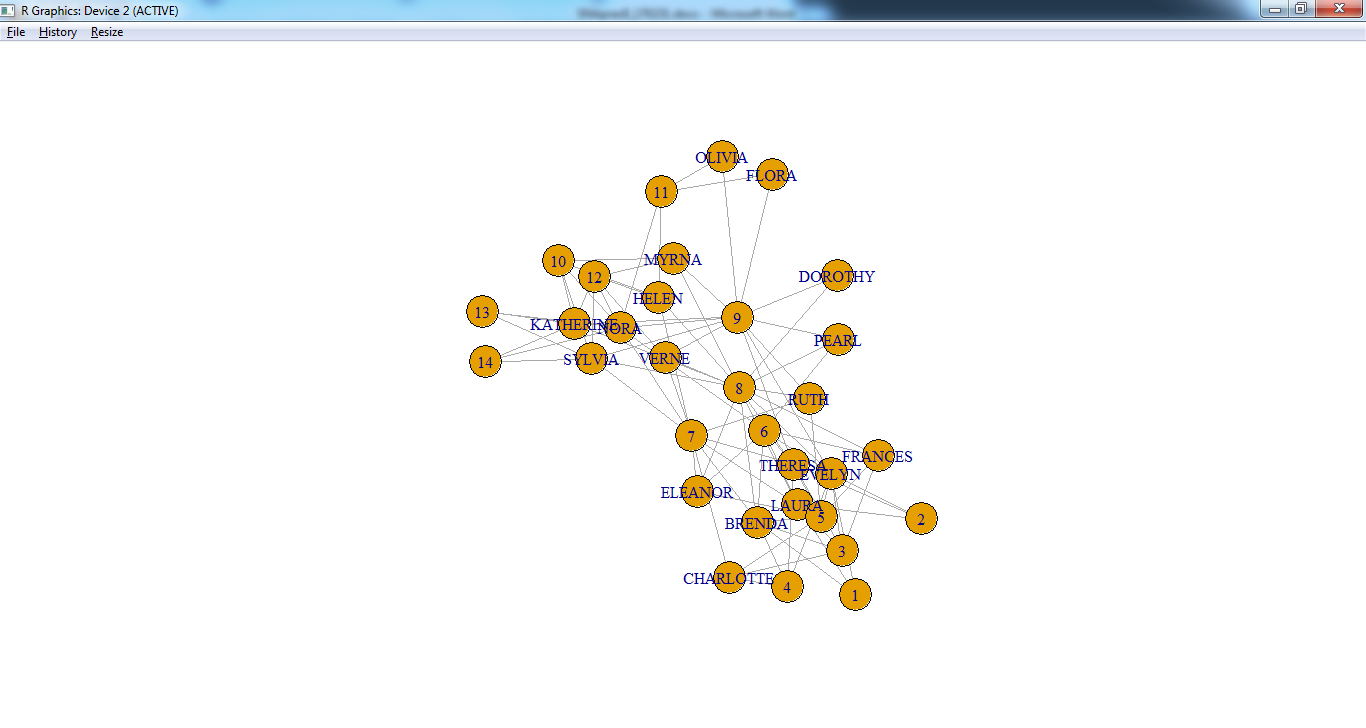
C:\Users\student\AppData\Local\Temp\Rtmpc1mZmq\downloaded\_packages

**Console:**

> davis <- read.csv(file.choose("D:/MSC 17/SNA//davis.csv"), header=FALSE)

> g <- graph.data.frame(davis, directed=FALSE)

> plot(g)



> bipartite.mapping(g)

$res

[1] TRUE

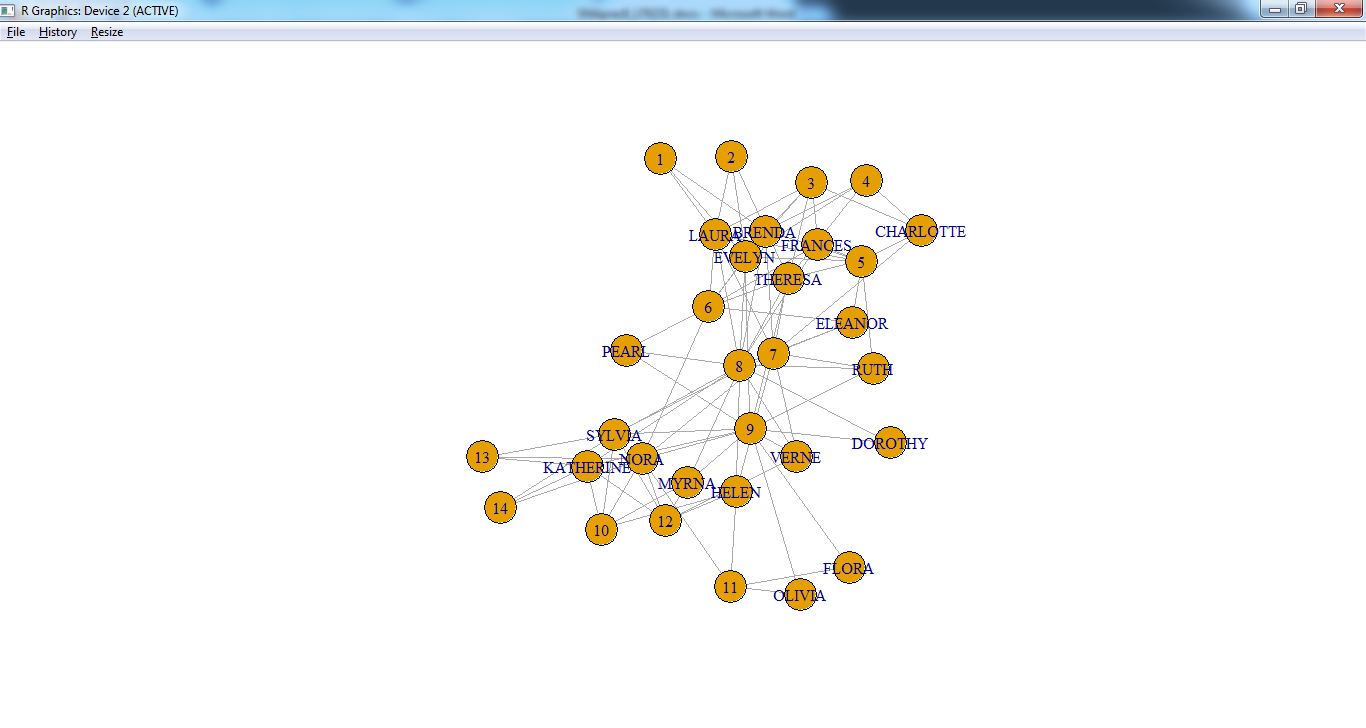
$type

[1] FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE FALSE TRUE TRUE TRUE TRUE TRUE

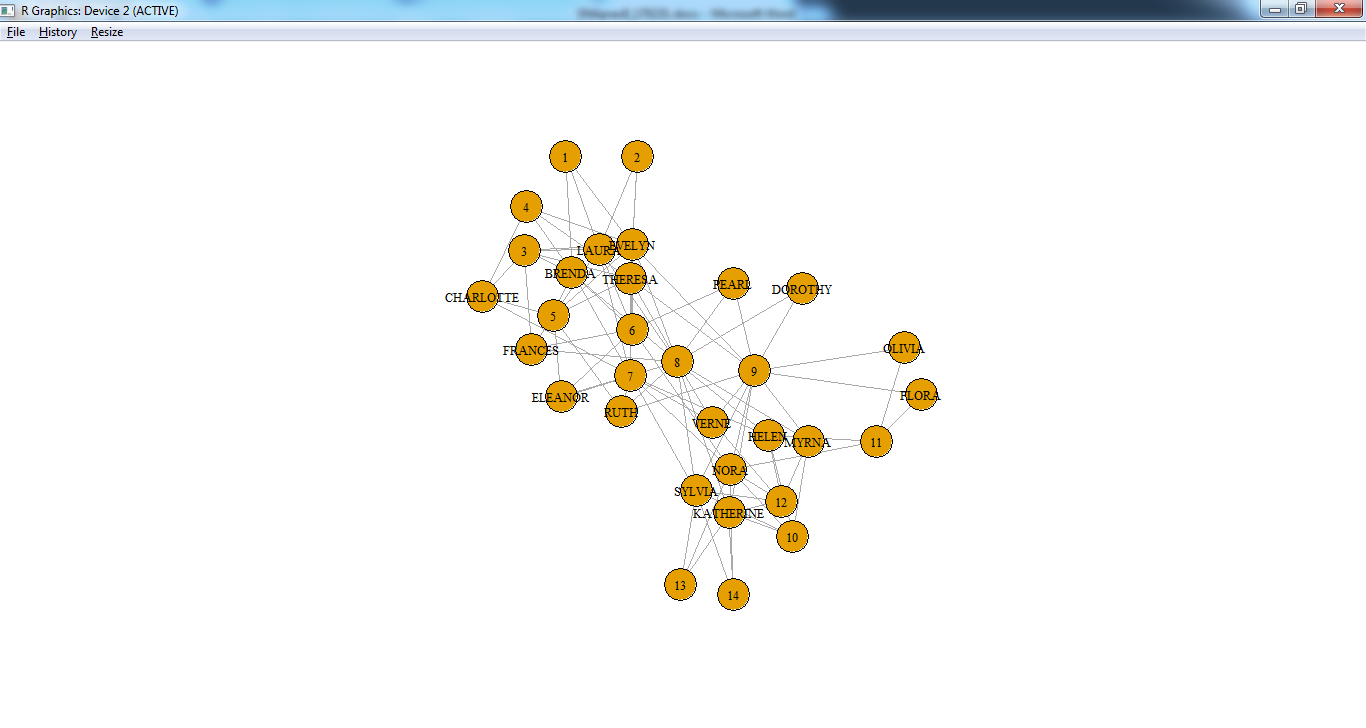
[24] TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE TRUE

> V(g)$type <-bipartite\_mapping(g)$type

> plot(g)



> plot(g,vertex.label.cex=0.8,vertex.label.color="black")

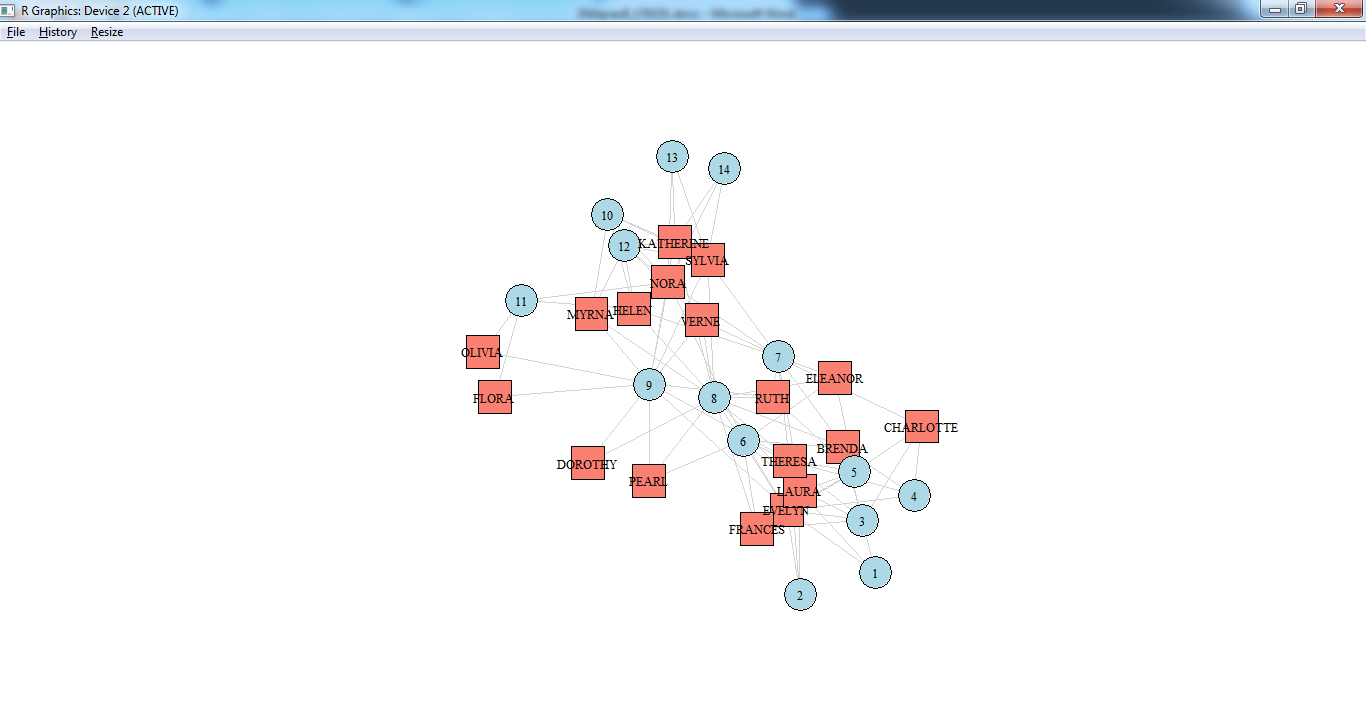


> V(g)$color <-ifelse(V(g)$type,"lightblue","salmon")

> V(g)$shape <-ifelse(V(g)$type,"circle","square")

> E(g)$color <-"lightgray"

> plot(g,vertex.label.cex=0.8,vertex.label.color="black")



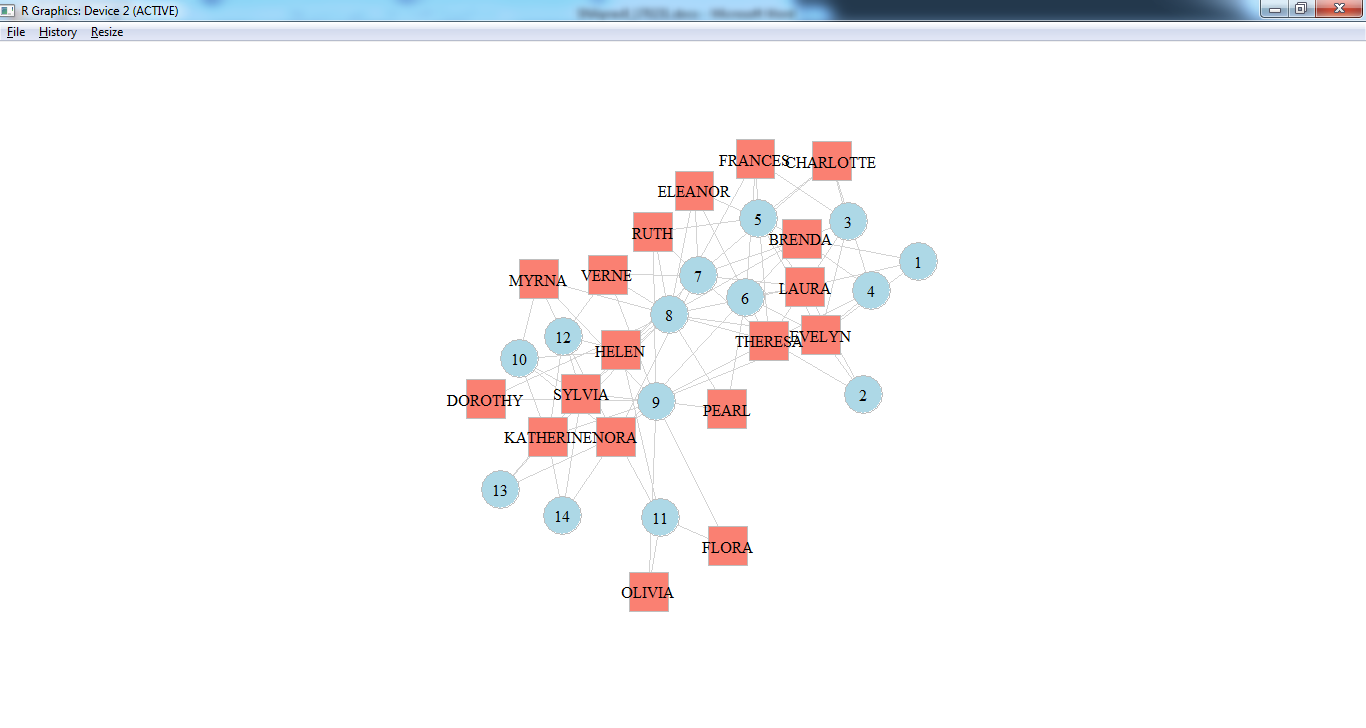
> V(g)$label.color <-"black"

> V(g)$label.cex <-1

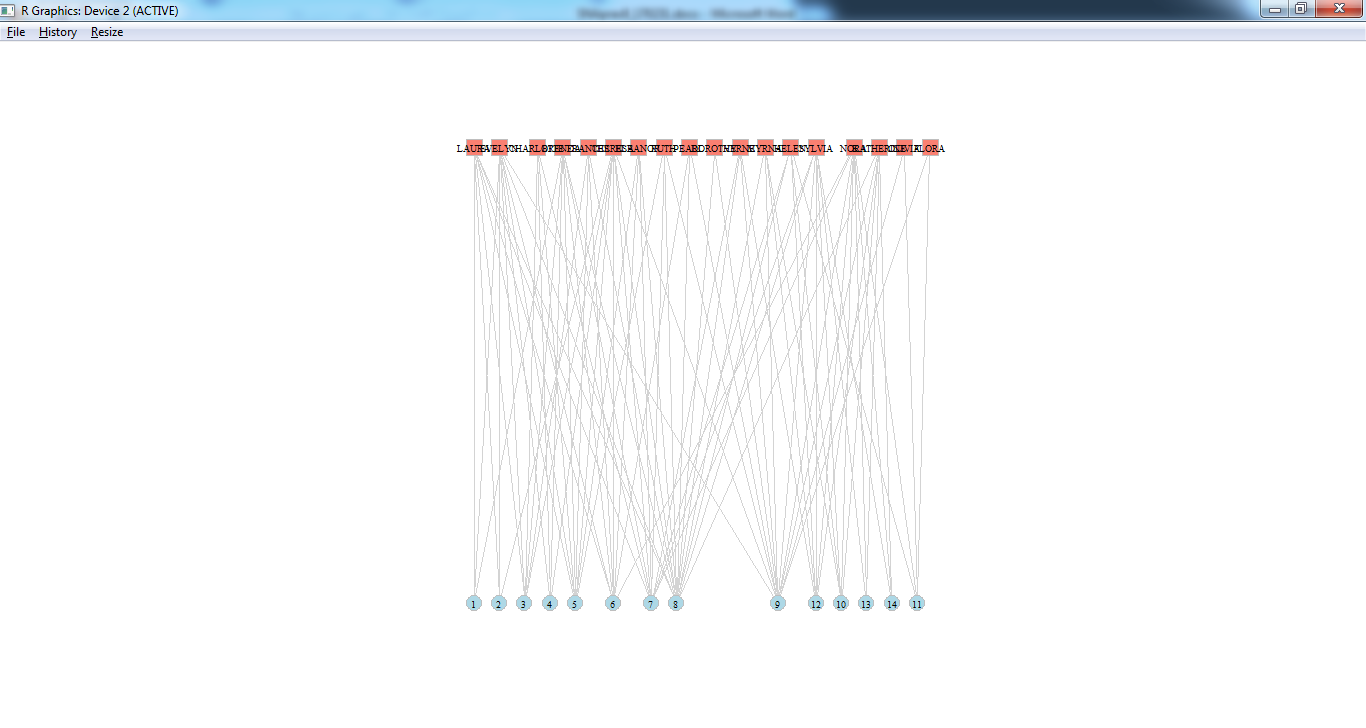
> V(g)$frame.color <-"gray"

> V(g)$size <- 18

> plot(g, layout =layout\_with\_graphopt)



> plot(g, layout=layout.bipartite, vertex.size=7, vertex.label.cex=0.6)



**Practical No.9:Perform SVD analysis of a network.**

**CODE:**

> A = as.matrix(data.frame(c(4,7,-1,8), c(-5,-2,4,2), c(-1,3,-3,6)))

> A

c.4..7...1..8. c..5...2..4..2. c..1..3...3..6.

[1,] 4 -5 -1

[2,] 7 -2 3

[3,] -1 4 -3

[4,] 8 2 6

# The singular value decomposition of the matrix is computed using the svd() function.

|  |
| --- |
| > A.svd<-svd(A)  > A.svd  $d  [1] 13.161210 6.999892 3.432793  $u  [,1] [,2] [,3]  [1,] -0.2816569 0.7303849 -0.42412326  [2,] -0.5912537 0.1463017 -0.18371213  [3,] 0.2247823 -0.4040717 -0.88586638  [4,] -0.7214994 -0.5309048 0.04012567  $v  [,1] [,2] [,3]  [1,] -0.8557101 0.01464091 -0.5172483  [2,] 0.1555269 -0.94610374 -0.2840759  [3,] -0.4935297 -0.32353262 0.8073135  Thus the above matrix AA can be factorized as the following:    #Singular Value Decomposition Step-by-Step  SVD can be performed step-by-step with R by calculating ATAATA and AATAAT then finding the eigenvalues and eigenvectors of the matrices.    > ATA <- t(A) %\*% A  > ATA  c.4..7...1..8. c..5...2..4..2. c..1..3...3..6.  c.4..7...1..8. 130 -22 68  c..5...2..4..2. -22 49 -1  c..1..3...3..6. 68 -1 55  # The V component of the singular value decomposition is then found by calculating the eigenvectors of the resultant ATAATA matrix.  > ATA.e <- eigen(ATA)  > v.mat <- ATA.e$vectors  > v.mat  [,1] [,2] [,3]  [1,] 0.8557101 -0.01464091 -0.5172483  [2,] -0.1555269 0.94610374 -0.2840759  [3,] 0.4935297 0.32353262 0.8073135  Here we see the VV matrix is the same as the output of the svd() but with some sign changes. These sign changes can happen, as mentioned earlier, as the eigenvector scaled by −1−1 is still the same eigenvector, just scaled. We will alter the signs of our calculated VV to match the output of the svd() function.  > v.mat[,1:2]<-v.mat[,1:2]\*-1  > v.mat  [,1] [,2] [,3]  [1,] -0.8557101 0.01464091 -0.5172483  [2,] 0.1555269 -0.94610374 -0.2840759  [3,] -0.4935297 -0.32353262 0.8073135    > AAT <- A%\*%t(A)  > AAT  [,1] [,2] [,3] [,4]  [1,] 42 35 -21 16  [2,] 35 62 -24 70  [3,] -21 -24 26 -18  [4,] 16 70 -18 104  #The eigenvectors are again found for the computed AATAAT matrix.  > AAT.e <- eigen(AAT)  > u.mat<- AAT.e$vectors  > u.mat  [,1] [,2] [,3] [,4]  [1,] -0.2816569 0.7303849 -0.42412326 -0.4553316  [2,] -0.5912537 0.1463017 -0.18371213 0.7715340  [3,] 0.2247823 -0.4040717 -0.88586638 0.0379443  [4,] -0.7214994 -0.5309048 0.04012567 -0.4426835  There are four eigenvectors in the resulting matrix; however, we are only interested in the non-zero eigenvalues and their respective eigenvectors. Therefore, we can remove the last eigenvector from the matrix which gives us the UU matrix. Note the eigenvalues of AATAAT and ATAATA are the same except the 00 eigenvalue in the AATAAT matrix.  > u.mat <- u.mat[,1:3]  As mentioned earlier, the singular values rr are the square roots of the non-zero eigenvalues of the AATAAT and ATAATA matrices.  > r<-sqrt(ATA.e$values)  > r<-r\*diag(length(r))[,1:3]  > r  [,1] [,2] [,3]  [1,] 13.16121 0.000000 0.000000  [2,] 0.00000 6.999892 0.000000  [3,] 0.00000 0.000000 3.432793  Our answers align with the output of the svd() function. We can also show that the matrix AA is indeed equal to the components resulting from singular value decomposition.  > svd.matrix<-u.mat%\*%r%\*%t(v.mat)  > svd.matrix  [,1] [,2] [,3]  [1,] 4 -5 -1  [2,] 7 -2 3  [3,] -1 4 -3  [4,] 8 2 6 |
|  |
| |  | | --- | |  | |