



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



INDEX

SR. NO.	Practicals	Date	Signature
1.	Write a program to compute the following for a given a network:(i) number of edges, (ii) number of nodes; (iii) degree of node; (iv)node with lowest degree; (v) the adjacency list; (vi) matrix of the graph.	03/07/24	
2.	Perform following tasks: (i) View data collection forms and/or import one-mode/two-mode datasets; (ii) Basic Networks matrices transformations	10/07/24	
3.	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.	28/07/24	
4.	Compute the following node level measures: (i) Density; (ii) Degree; (iii) Reciprocity; (iv) Transitivity; (v) Centralization; (vi) Clustering.	1/08/24	
5.	Write a program to distinguish between a network as a matrix, a network as an edge list, and a network as a sociogram (or “network graph”) using 3 distinct networks representatives of each.	12/08/24	
6.	Write a program to exhibit structural equivalence, automorphic equivalence, and regular equivalence from a network.	26/08/24	
7.	Perform SVD analysis of a network.	3/09/24	
8.	Displaying Bipartite network in the graph format.	17/09/24	
9.	Hamming distance.	05/10/24	



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 1

Aim: Write a program to compute the following for a given a network: (i) number of edges, (ii) number of nodes; (iii) degree of node; (iv) node with lowest degree; (v) the adjacency list; (vi) matrix of the graph.

Name: Vishal Waikar

Roll No.: KFPMSCCS021

Date: 03/07/24

Sign:

Code:

```
>library(igraph)
>g <- graph.formula(1-2, 1-3, 2-3, 2-4, 3-5, 4-5, 4-6, 4-7, 5-6, 6-7)
```

Name of Edges & Nodes

```
> V(g)
> E(g)
```

Plotting the graph

```
> plot(g)
```

```
~/tmp/RtmpHaCX3M/downloaded_packages'
Warning message:
In download.file(url, destfile = f, quiet = TRUE) :
  URL 'https://cran.r-project.org/CRAN_mirrors.csv': status was 'Could not connect to server'
> library(igraph)

Attaching package: 'igraph'

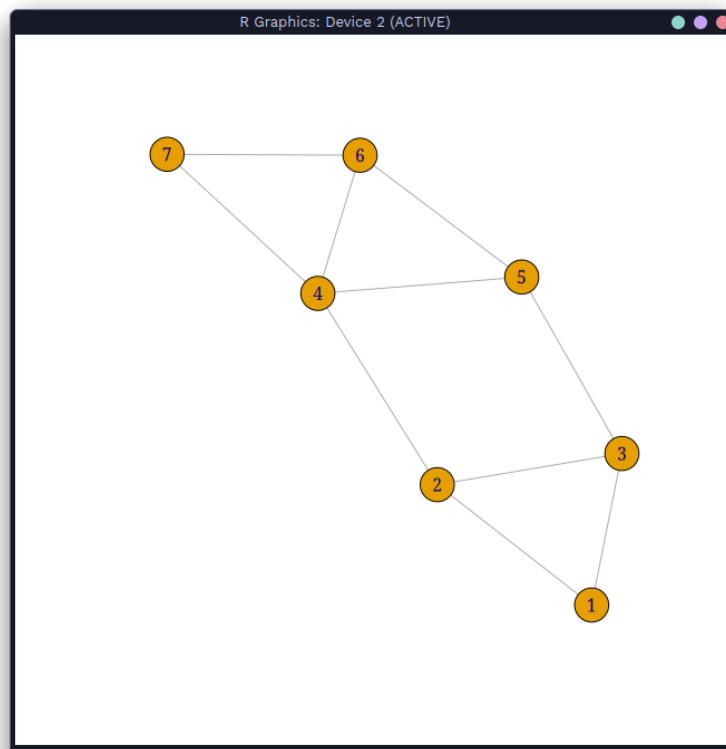
The following objects are masked from 'package:stats':

  decompose, spectrum

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

  union

> library(igraph)
> g <- graph.formula(1-2, 1-3, 2-3, 2-4, 3-5, 4-5, 4-6, 4-7, 5-6, 6-7)
> V(g)
+ 7/7 vertices, named, from 0d64e00:
[1] 1 2 3 4 5 6 7
> E(g)
+ 10/10 edges from 0d64e00 (vertex names):
[1] 1--2 1--3 2--3 2--4 3--5 4--5 4--6 4--7 5--6 6--7
> plot(g)
> █
```



Directed graph

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

```
~: R — Konsole

'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.

> dg <- graph.formula(1-+2, 1-+3, 2++3)
Error in graph.formula(1 - +2, 1 - +3, 2 + +3) :
  could not find function "graph.formula"
> library(igraph)

Attaching package: 'igraph'

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

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

> library(igraph)
>
> 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)
```

```
~ : R — Konsole
Natural language support but running in an English locale

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.

> library(igraph)

Attaching package: 'igraph'

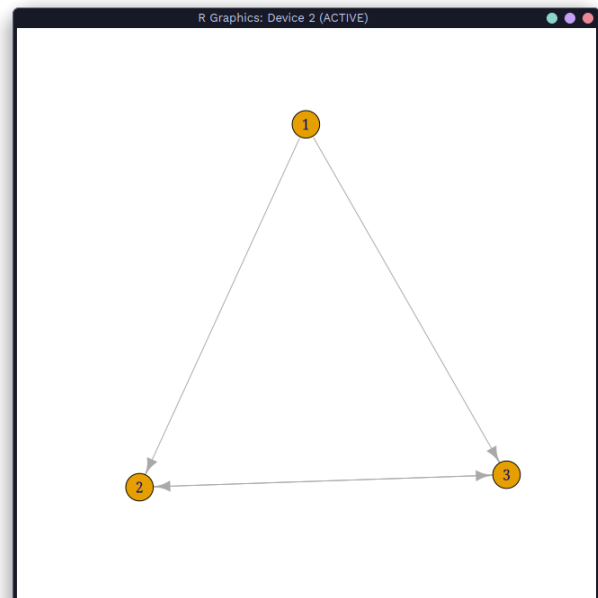
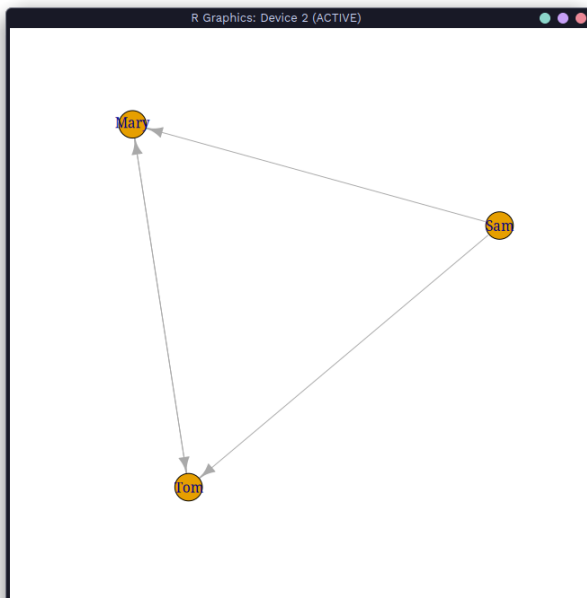
The following objects are masked from 'package:stats':

    decompose, spectrum

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

    union

> library(igraph)
> dg1 <- graph.formula(Sam->Mary, Sam->Tom, Mary->Tom)
> plot(dg1)
> 
```



Number of vertices/node:

> vcount(g)

Number of edges/dyad/ties:

> ecount(g)

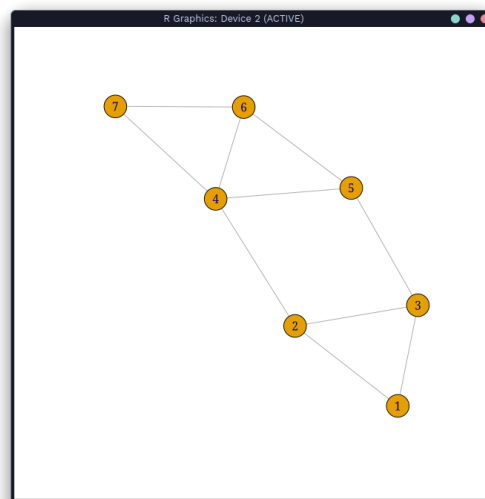
```
~ : R — Konsole

decompose, spectrum

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

  union

> library(igraph)
> dgl <- graph.formula(Sam--Mary, Sam--Tom, Mary++Tom)
> plot(dgl)
>
>
> g <- graph.formula(1-2, 1-3, 2-3, 2-4, 3-5, 4-5, 4-6,4-7, 5-6, 6-7)
> V(g)
+ 7/7 vertices, named, from 18df649:
[1] 1 2 3 4 5 6 7
> E(g)
+ 10/10 edges from 18df649 (vertex names):
[1] 1--2 1--3 2--3 2--4 3--5 4--5 4--6 4--7 5--6 6--7
> vcount(g)
[1] 7
> ecount(g)
[1] 10
>
>
>
```



Degree of a Node



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



> degree(g)

In-degree

> degree(dg, mode="in")

Out-degree

> degree(dg, mode="out")

Node with lowest degree

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

Node with highest degree

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

```
~ : R — Konsole
> ecount(g)
[1] 10
>
>
> degree(g)
1 2 3 4 5 6 7
2 3 3 4 3 3 2
>
> degree(dg, mode="in")
Error: object 'dg' not found
> dg <- graph.formula(1--2, 1--3, 2++3)
> degree(dg, mode="in")
1 2 3
0 2 2
>
> degree(dg, mode="out")
1 2 3
2 1 1
>
> V(dg)$name[degree(dg)==min(degree(dg))]
[1] "1"
>
> V(dg)$name[degree(dg)==max(degree(dg))]
[1] "2" "3"
>
> neighbors(g,5)
```

To find neighbours / adjacency list:

> neighbors(g,5)
> neighbors(g,2)
> get.adjlist(dg)
> get.adjacency(g)

```
~ : R — Konsole

union

> g <- graph.formula(1-2, 1-3, 2-3, 2-4, 3-5, 4-5, 4-6, 4-7, 5-6, 6-7)
>
> neighbors(g,5)
+ 3/7 vertices, named, from 2ae182b:
[1] 3 4 6
>
> neighbors(g,2)
+ 3/7 vertices, named, from 2ae182b:
[1] 1 3 4
> dg <- graph.formula(1-+2, 1-+3, 2-+3)
>
> get.adjlist(dg)
$`1`
+ 2/3 vertices, named, from 7607267:
[1] 2 3

$`2`
+ 3/3 vertices, named, from 7607267:
[1] 1 3 3

$`3`
+ 3/3 vertices, named, from 7607267:
[1] 1 2 2
```

Adjacency Matrix

> get.adjacency(g)

```
~ : R — Konsole

> get.adjacency(g)
7 x 7 sparse Matrix of class "dgMatrix"
 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 .
Warning message:
'get.adjacency()' was deprecated in igraph 2.0.0.
Please use 'as_adjacency_matrix()' instead.
This warning is displayed once every 8 hours.
Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
> get.adjacency(g)
7 x 7 sparse Matrix of class "dgMatrix"
 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 .
>
```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 2

Aim: Perform following tasks:

- (i) View data collection forms and/or import one-mode/two-mode datasets;**
- (ii) Basic Networks matrices transformations**

Name: Vishal Waikar

Roll No.: KFPMSCCS021

Date: 10/07/24

Sign:

Note:

Where your working directory is set at this moment

> getwd()

R now knows exactly in which folder you're working.

> setwd("<location of your dataset>")

Reading data from a csv file

> nodes <- read.csv("Dataset1-Media-Example-NODES.csv", header=T, , as.is=T)

> head(nodes)

> links <- read.csv("Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)

> head(links)

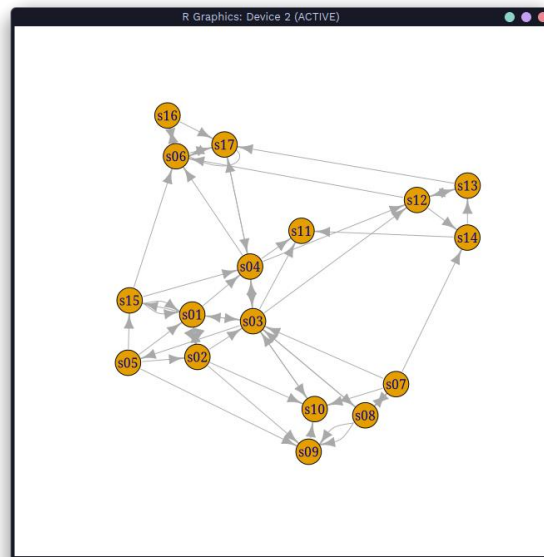
```
netsci2016 : R — Konsole
[1] "/home/stxari"
> setwd("/home/stxari/Downloads/netsci2016")
>
> >nodes <- read.csv("Dataset1-Media-Example-NODES.csv", header=T, , as.is=T)
Error: unexpected '>' in ">"
> nodes <- read.csv("Dataset1-Media-Example-NODES.csv", header=T, , as.is=T)
> head(nodes)
  id      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
> links <- read.csv("Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)
Error: unexpected invalid token in "links <- read.csv("
> links <- read.csv("Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)
> 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
>
```

Basic Networks matrices transformations


```
> net <- graph.data.frame(d=links, vertices=nodes, directed=T)
> m=as.matrix(net)
> get.adjacency(m)
> plot(net)
```

```
netscix2016 : R — Konsole

>
>
>
>
>
>
>
> net <- graph.data.frame(d=links, vertices=nodes, directed=T)
Warning message:
`graph.data.frame()` was deprecated in igraph 2.0.0.
i Please use `graph_from_data_frame()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> m=as.matrix(net)
> get.adjacency(m)
Error in `ensure_igraph()`:
! Must provide a graph object (provided wrong object type).
Run `rlang::last_trace()` to see where the error occurred.
Warning message:
`get.adjacency()` was deprecated in igraph 2.0.0.
i Please use `as_adjacency_matrix()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> plot(net)
>
```





HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 3

Aim: 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.

Name: Vishal Waikar

Roll No.: KFPMSCCS021

Date: 28/07/24

Sign:

Density

> vcount(g)

> ecount(g)

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

```
~ : R — Konsole
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.

> library(igraph)

Attaching package: 'igraph'

The following objects are masked from 'package:stats':

    decompose, spectrum

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

    union

> g <- graph.formula(1-2, 1-3, 2-3, 2-4, 3-5, 4-5, 4-6, 4-7, 5-6, 6-7)
> vcount(g)
Error: unexpected symbol in "vcount(g)"
> vcount(g)
[1] 7
> ecount(g)
[1] 10
> ecount(g)/(vcount(g)*(vcount(g)-1))
[1] 0.2380952
>
```

Degree

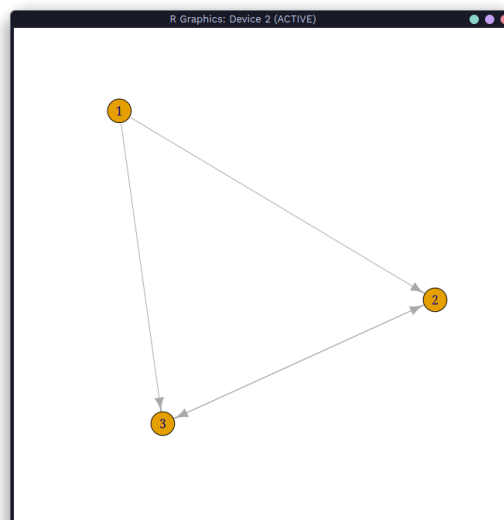
degree(net)

```
netscix2016 : R — Konsole

Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> nodes <- read.csv("Dataset1-Media-Example-NODES.csv", header=T, , as.is=T)
Error in file(file, "rt") : cannot open the connection
In addition: Warning message:
In file(file, "rt") :
  cannot open file 'Dataset1-Media-Example-NODES.csv': No such file or directory
> setwd("/home/stxari/Downloads/netscix2016")
> nodes <- read.csv("Dataset1-Media-Example-NODES.csv", header=T, , as.is=T)
> links <- read.csv("Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)
Error: unexpected invalid token in "links <- read.csv("
> links <- read.csv("Dataset1-Media-Example-EDGES.csv", header=T, as.is=T)
> net <- graph.data.frame(d=links, vertices=nodes, directed=T)
> m=as.matrix(net)
> get.adjacency(m)
Error in `ensure_igraph()` :
! Must provide a graph object (provided wrong object type).
Run `rlang::last_trace()` to see where the error occurred.
Warning message:
`get.adjacency()` was deprecated in igraph 2.0.0.
i Please use `as_adjacency_matrix()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> degree(net)
s01 s02 s03 s04 s05 s06 s07 s08 s09 s10 s11 s12 s13 s14 s15 s16 s17
10  7  13  9  5  8  5  6  5  5  3  6  4  4  6  3  5
> 
```

Reciprocity

```
> dg <- graph.formula(1->2, 1->3, 2->3)
> plot(dg)
> reciprocity(dg)
```





HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Formula as per text book

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

```
netscix2016 : R — Konsole

>
>
>
>
> dg <- graph.formula(1--2, 1-+3, 2++3)
> plot(dg)
> reciprocity(dg)
[1] 0.5
> dyad.census(dg)
$mut
[1] 1

$asym
[1] 2

$null
[1] 0

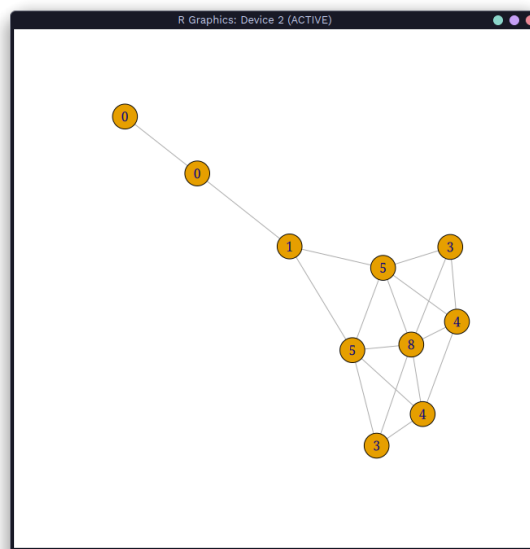
Warning message:
`dyad.census()` was deprecated in igraph 2.0.0.
i Please use `dyad_census()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> 2*dyad.census(dg)$mut/ecount(dg)
[1] 0.5
>
```

Transitivity

```
> kite <- graph.famous("Krackhardt_Kite")
> atri <- adjacent.triangles(kite)
> plot(kite, vertex.label=atri)
> transitivity(kite, type="local")
> adjacent.triangles(kite) / (degree(kite) * (degree(kite)-1)/2)
```

```
netscix2016 : R — Konsole

`dyad.census()` was deprecated in igraph 2.0.0.
i Please use `dyad_census()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> 2*dyad.census(dg)$mut/ecount(dg)
[1] 0.5
>
>
>
>
>
>
> kite <- graph.famous("Krackhardt_Kite")
> atri <- adjacent.triangles(kite)
Warning message:
`adjacent.triangles()` was deprecated in igraph 2.0.0.
i Please use `count_triangles()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> plot(kite, vertex.label=atri)
> transitivity(kite, type="local")
[1] 0.6666667 0.6666667 1.0000000 0.5333333 1.0000000 0.5000000 0.5000000
[8] 0.3333333 0.0000000      NaN
> adjacent.triangles(kite) / (degree(kite) * (degree(kite)-1)/2)
[1] 0.6666667 0.6666667 1.0000000 0.5333333 1.0000000 0.5000000 0.5000000
[8] 0.3333333 0.0000000      NaN
> █
```



Centralization



HSNC UNIVERSITY, MUMBAI

KISHINCHAND CHELLARAM COLLEGE

M. Sc Computer Science Semester 3 (2024 – 25)



Degree of centrality

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

Closeness Centralization

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

Betweenness Centrality

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

Eigenvector centrality

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

Clustering

```
> plot(kite)
> get.adjedgelist(kite, mode = c("all", "out", "in", "total"))
```

```
netscix2016 : R — Konsole
Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
> centralization.betweenness(net, directed=T, normalized=T)

$res
[1] 26.857143 6.238095 126.511905 92.642857 13.000000 20.333333
[7] 1.750000 21.000000 1.000000 15.000000 0.000000 33.500000
[13] 20.000000 4.000000 5.666667 0.000000 58.500000

$centralization
[1] 0.4439329

$theoretical_max
[1] 3840

Warning message:
'centralization.betweenness()' was deprecated in igraph 2.0.0.
i Please use 'centr_betw()' instead.
This warning is displayed once every 8 hours.
Call 'lifecycle::last_lifecycle_warnings()' to see where this warning was generated.
> centralization.evcent(net, directed=T, normalized=T)

$vector
[1] 0.7694528 0.5623895 1.0000000 0.8569443 0.3049992 0.9285033 0.1025656
[8] 0.3362816 0.4696841 0.6510633 0.6361813 0.6479337 0.2674341 0.2289017
[15] 0.3277070 0.2831928 0.7125008
```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



```
netscix2016 : R — Konsole

Warning message:
`centralization.evcent()` was deprecated in igraph 2.0.0.
i Please use `centr_eigen()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> plot(kite)

> get.adjedgelist(kite, mode = c("all", "out", "in", "total"))

[[1]]
+ 4/18 edges from dc8536a:
[1] 1--2 1--3 1--4 1--6

[[2]]
+ 4/18 edges from dc8536a:
[1] 1--2 2--4 2--5 2--7

[[3]]
+ 3/18 edges from dc8536a:
[1] 1--3 3--4 3--6

[[4]]
+ 6/18 edges from dc8536a:
[1] 1--4 2--4 3--4 4--5 4--6 4--7
```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 4

Aim: Compute the following node level measures:

(i) Density; (ii) Degree; (iii) Reciprocity; (iv) Transitivity; (v) Centralization; (vi) Clustering.

Name: Vishal Waikar

Roll No.: KFPMSCCS021

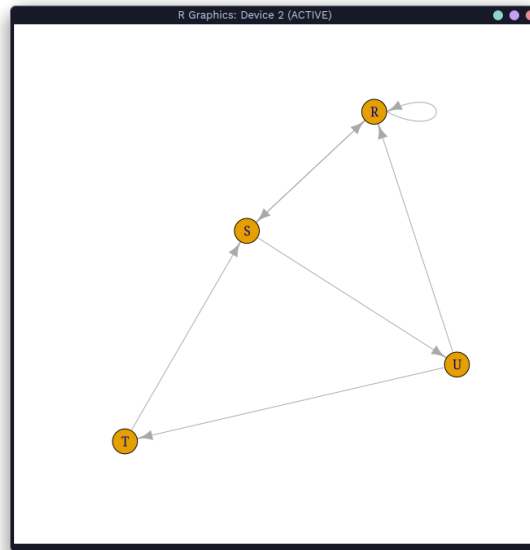
Date: 1/08/24

Sign:

Length of the shortest path from a given node to another node.

```
> library(igraph)
> 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)
```

```
~ : R — Konsole
>
>
> library(igraph)
> 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)
Warning message:
`graph.adjacency()` was deprecated in igraph 2.0.0.
Please use `graph_from_adjacency_matrix()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> plot(g)
>
```

```

> s.paths <- shortest.paths(g, algorithm = "dijkstra")
> print(s.paths)
> shortest.paths(g, v="R", to="S")
> plot(g, edge.label=E(g)$weight)

```

```

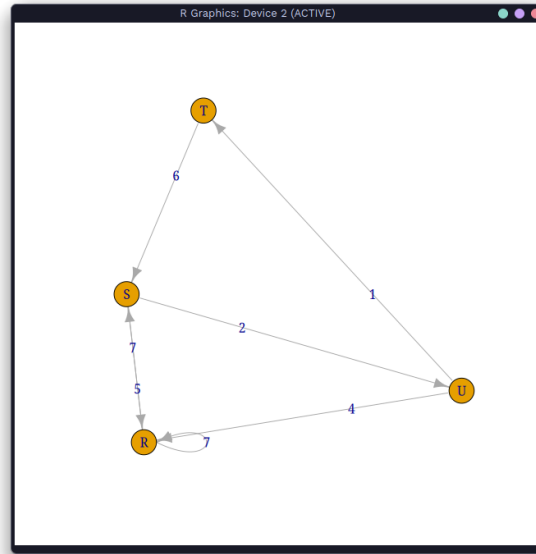
~ : R — Konsole

> g <- graph.adjacency(matt, weighted=TRUE)
Warning message:
`graph.adjacency()` was deprecated in igraph 2.0.0.
i Please use `graph_from_adjacency_matrix()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> plot(g)
> s.paths <- shortest.paths(g, algorithm = "dijkstra")
Warning message:
`shortest.paths()` was deprecated in igraph 2.0.0.
i Please use `distances()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> 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)$weight)
Error: unexpected '>' in ">"
> plot(g, edge.label=E(g)$weight)
>

```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



The density of the graph

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

```
> library(igraph)
> dg <- graph.formula(1-+2, 1-+3, 2++3)
> plot(dg)
> graph.density(dg, loops=TRUE)
```

- Without considering loops

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



```
~ : R — Konsole

Type 'q()' to quit R.

> library(igraph)
Attaching package: 'igraph'

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

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

> dg <- graph.formula(1-+2, 1-+3, 2++3)
> plot(dg)
> graph.density(dg, loops=TRUE)
[1] 0.4444444
Warning message:
`graph.density()` was deprecated in igraph 2.0.0.
i Please use `edge_density()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> graph.density(simplify(dg), loops=FALSE)
[1] 0.6666667
>
```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 5

Aim: Write a program to distinguish between a network as a matrix, a network as an edge list, and a network as a sociogram (or “network graph”) using 3 distinct networks representatives of each.

Name: Vishal Waikar

Roll No.: KFPMSCCS021

Date: 12/08/24

Sign:

A network as a graph

> library(igraph)

> 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)

```
Platform: x86_64-pc-linux-gnu
~ : R — Konsole

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.

Natural language support but running in an English locale

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.

> library(igraph)

Attaching package: 'igraph'

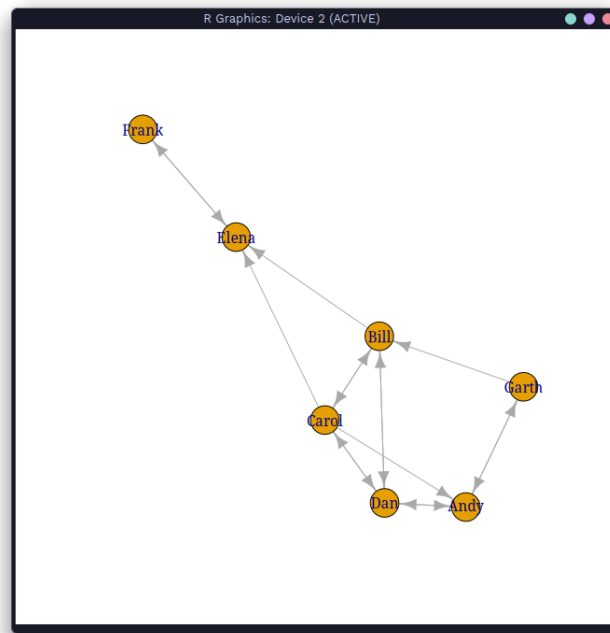
The following objects are masked from 'package:stats':

    decompose, spectrum

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

    union

> 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)

```
~ : R — Konsole

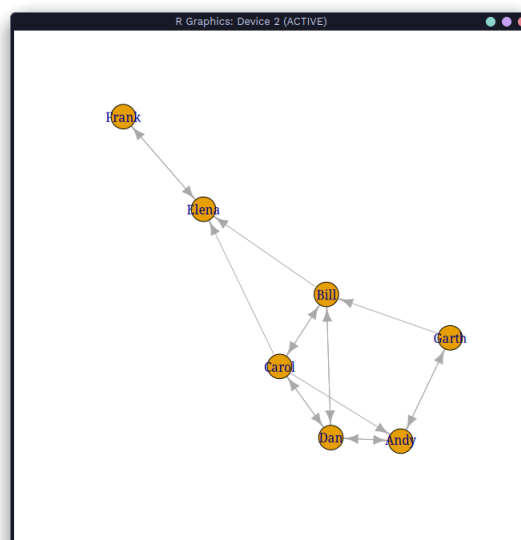
The following objects are masked from 'package:stats':

  decompose, spectrum

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

  union

> 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)
>
>
>
> get.adjacency(ng)7 x 7 sparse Matrix of class "dgCMatrix"
Error: unexpected numeric constant in "get.adjacency(ng)7"
> 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   .
Warning message:
`get.adjacency()` was deprecated in igraph 2.0.0.
Please use `as_adjacency_matrix()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> 
```



A network as an edge list



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



> E(ng)

```
~ : R — Konsole

union

> 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)
>
>
>
> get.adjacency(ng) 7 x 7 sparse Matrix of class "dgCMatrix"
Error: unexpected numeric constant in "get.adjacency(ng)7"
> 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    .
Warning message:
`get.adjacency()` was deprecated in igraph 2.0.0.
i Please use `as_adjacency_matrix()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> E(ng)
+ 16/16 edges from 20ff9cc (vertex names):
[1] Andy ->Garth Andy ->Dan  Garth->Andy  Garth->Bill  Bill ->Elena
[6] Bill ->Carol Bill ->Dan  Elena->Frank Frank->Elena Carol->Andy
[11] Carol->Bill  Carol->Elena Carol->Dan  Dan ->Andy  Dan ->Bill
[16] Dan ->Carol
> 
```



HSNC UNIVERSITY, MUMBAI

KISHINCHAND CHELLARAM COLLEGE

M. Sc Computer Science Semester 3 (2024 – 25)



```
~ : R — Konsole

[16] Dan ->Carol
> get.adjedgelist(ng,mode="in")
$Andy
+ 3/16 edges from 20ff9cc (vertex names):
[1] Garth->Andy Carol->Andy Dan ->Andy

$Garth
+ 1/16 edge from 20ff9cc (vertex names):
[1] Andy->Garth

$Bill
+ 3/16 edges from 20ff9cc (vertex names):
[1] Garth->Bill Carol->Bill Dan ->Bill

$Elena
+ 3/16 edges from 20ff9cc (vertex names):
[1] Bill ->Elena Frank->Elena Carol->Elena

$Frank
+ 1/16 edge from 20ff9cc (vertex names):
[1] Elena->Frank

$Carol
+ 2/16 edges from 20ff9cc (vertex names):
[1] Bill->Carol Dan ->Carol

$Dan
+ 3/16 edges from 20ff9cc (vertex names):
[1] Andy ->Dan Bill ->Dan Carol->Dan

Warning message:
`get.adjedgelist()` was deprecated in igraph 2.0.0.
Please use `as_adj_edge_list()` instead.
```




HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 6

Aim: Write a program to exhibit structural equivalence, automorphic equivalence, and regular equivalence from a network.

Name: Vishal Waikar

Roll No.: KFPMSCCS021

Date: 26/08/24

Sign:

Code:

```
> library(sna)
> library(igraph)
> links2 <- read.csv("/mnt/HDD/Collage/Msc/SEM3/SNA/netscix2016/Dataset2-Media-User-Example-EDGES.csv", header=T, row.names=1)
> eq<-equiv.clust(links2)
> plot(eq)
```

```
~ : R — Konsole

> 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

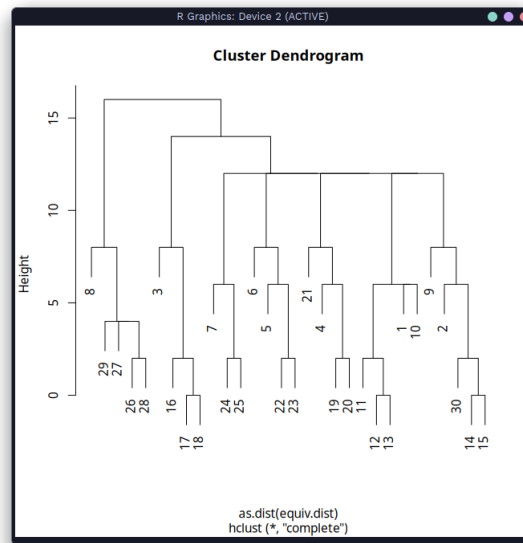
The following objects are masked from 'package:stats':

  decompose, spectrum

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

  union

> links2 <- read.csv("/mnt/HDD/Collage/Msc/SEM3/SNA/netscix2016/Dataset2-Media-User-Example-EDGES.csv", head
r=T, row.names=1)
Error: unexpected invalid token in "links2 <- read.csv("
> links2 <- read.csv("/mnt/HDD/Collage/Msc/SEM3/SNA/netscix2016/Dataset2-Media-User-Example-EDGES.csv", head
r=T, row.names=1)
> eq<-equiv.clust(links2)
> plot(eq)
>
```

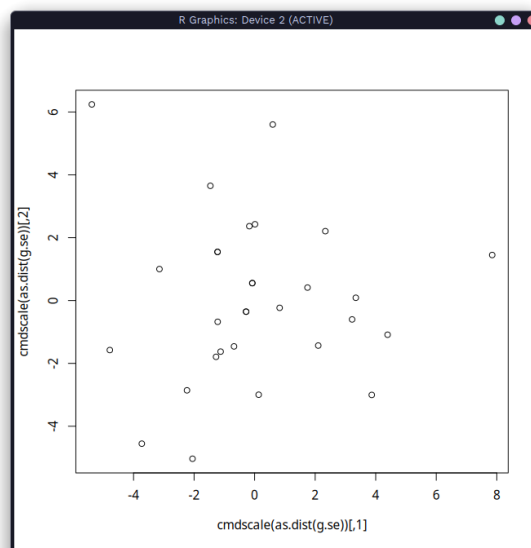


Get Structural Equivalence distances

```
>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)
```

```
~ : R — Konsole

The following objects are masked from 'package:sna':

  betweenness, bonpow, closeness, components, degree, dyad.census,
  event, 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

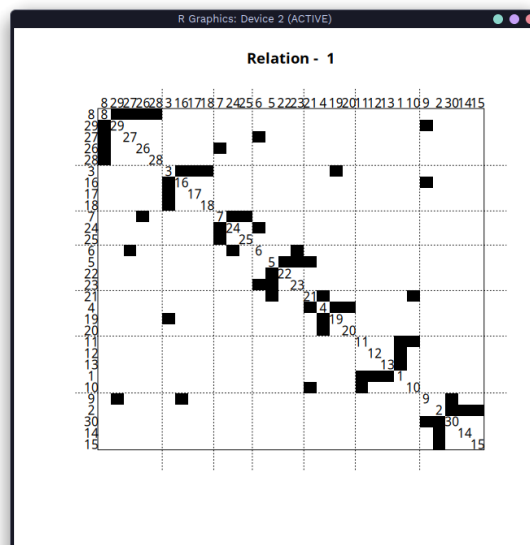
The following objects are masked from 'package:stats':

  decompose, spectrum

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

  union

> links2 <- read.csv("/mnt/HDD/Collage/Msc/SEM3/SNA/netscix2016/Dataset2-Media-User-Example-EDGES.csv", head=
r=T, row.names=1)
Error: unexpected invalid token in "links2 <- read.csv("
> links2 <- read.csv("/mnt/HDD/Collage/Msc/SEM3/SNA/netscix2016/Dataset2-Media-User-Example-EDGES.csv", head=
r=T, row.names=1)
> eq<-equiv.clust(links2)
> plot(eq)
> g.se<-sedist(links2)
> plot(cmdscale(as.dist(g.se)))
> b<-blockmodel(links2,eq,h=10)
> plot(b)
> |
```





HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 7	
Aim: Perform SVD analysis of a network.	
Name: Vishal Waikar	Roll No.: KFPMSCCS021
Date: 3/09/24	Sign:

Code:

```
> library(igraph)
> a <- matrix(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, , 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1,
1, 1), 9, 4)
> print(a)
```



HSNC UNIVERSITY, MUMBAI

KISHINCHAND CHELLARAM COLLEGE

M. Sc Computer Science Semester 3 (2024 – 25)



```
~ : R — Konsole

Type 'q()' to quit R.

> library(igraph)

Attaching package: 'igraph'

The following objects are masked from 'package:stats':

    decompose, spectrum

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

    union

> a <- matrix(c(1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0, 0, 0, 1, 1, 1, 0, 0, 0, 0, 0, 0, 0,
0, 0, 1, 1, 1), 9, 4)
> print(a)
      [,1] [,2] [,3] [,4]
[1,] 1    1    0    0
[2,] 1    1    0    0
[3,] 1    1    0    0
[4,] 1    0    1    0
[5,] 1    0    1    0
[6,] 1    0    1    0
[7,] 1    0    0    1
[8,] 1    0    0    1
[9,] 1    0    0    1
> svd(a)
$d
[1] 3.464102e+00 1.732051e+00 1.732051e+00 1.922963e-16
$u
      [,1]      [,2]      [,3]      [,4]
```

```
~ : R — Konsole

[1,] 1    1    0    0
[2,] 1    1    0    0
[3,] 1    1    0    0
[4,] 1    0    1    0
[5,] 1    0    1    0
[6,] 1    0    1    0
[7,] 1    0    0    1
[8,] 1    0    0    1
[9,] 1    0    0    1
> svd(a)
$d
[1] 3.464102e+00 1.732051e+00 1.732051e+00 1.922963e-16
$u
      [,1]      [,2]      [,3]      [,4]
[1,] -0.3333333  0.4714045 -1.741269e-16  7.760882e-01
[2,] -0.3333333  0.4714045 -3.692621e-16 -1.683504e-01
[3,] -0.3333333  0.4714045 -5.301858e-17 -6.077378e-01
[4,] -0.3333333 -0.2357023 -4.082483e-01  6.774193e-17
[5,] -0.3333333 -0.2357023 -4.082483e-01  6.774193e-17
[6,] -0.3333333 -0.2357023 -4.082483e-01  6.774193e-17
[7,] -0.3333333 -0.2357023  4.082483e-01  5.194768e-17
[8,] -0.3333333 -0.2357023  4.082483e-01  5.194768e-17
[9,] -0.3333333 -0.2357023  4.082483e-01  5.194768e-17
$V
      [,1]      [,2]      [,3] [,4]
[1,] -0.8660254  0.0000000 -4.378026e-17  0.5
[2,] -0.2886751  0.8164966 -2.509507e-16 -0.5
[3,] -0.2886751 -0.4082483 -7.071068e-01 -0.5
[4,] -0.2886751 -0.4082483  7.071068e-01 -0.5
> |
```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 8

Aim: Displaying Bipartite network in the graph format.

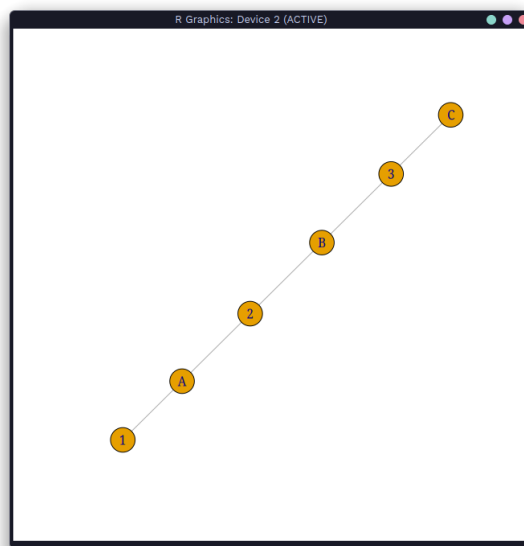
Name: Vishal Waikar

Roll No.: KFPMSCCS021

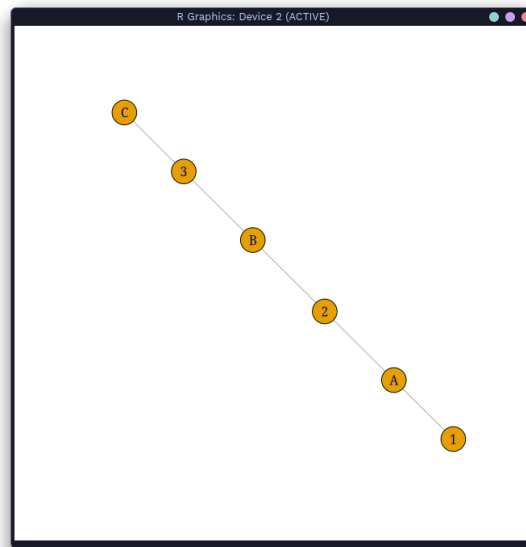
Date: 17/09/24

Sign:

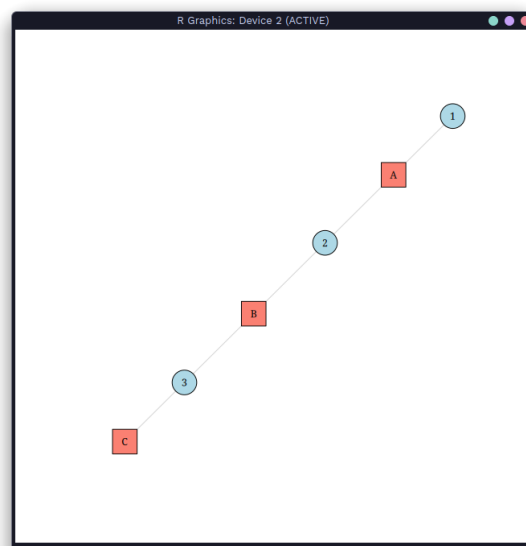
```
> library(igraph)
> davis <- read.csv("/mnt/HDD/Collage/Msc/SEM3/SNA/netscix2016/csv.csv")
> g <- graph.data.frame(davis, directed=FALSE)
> plot(g)
```



```
> bipartite.mapping(g)
> V(g)$type <- bipartite_mapping(g)$type
> plot(g)
```

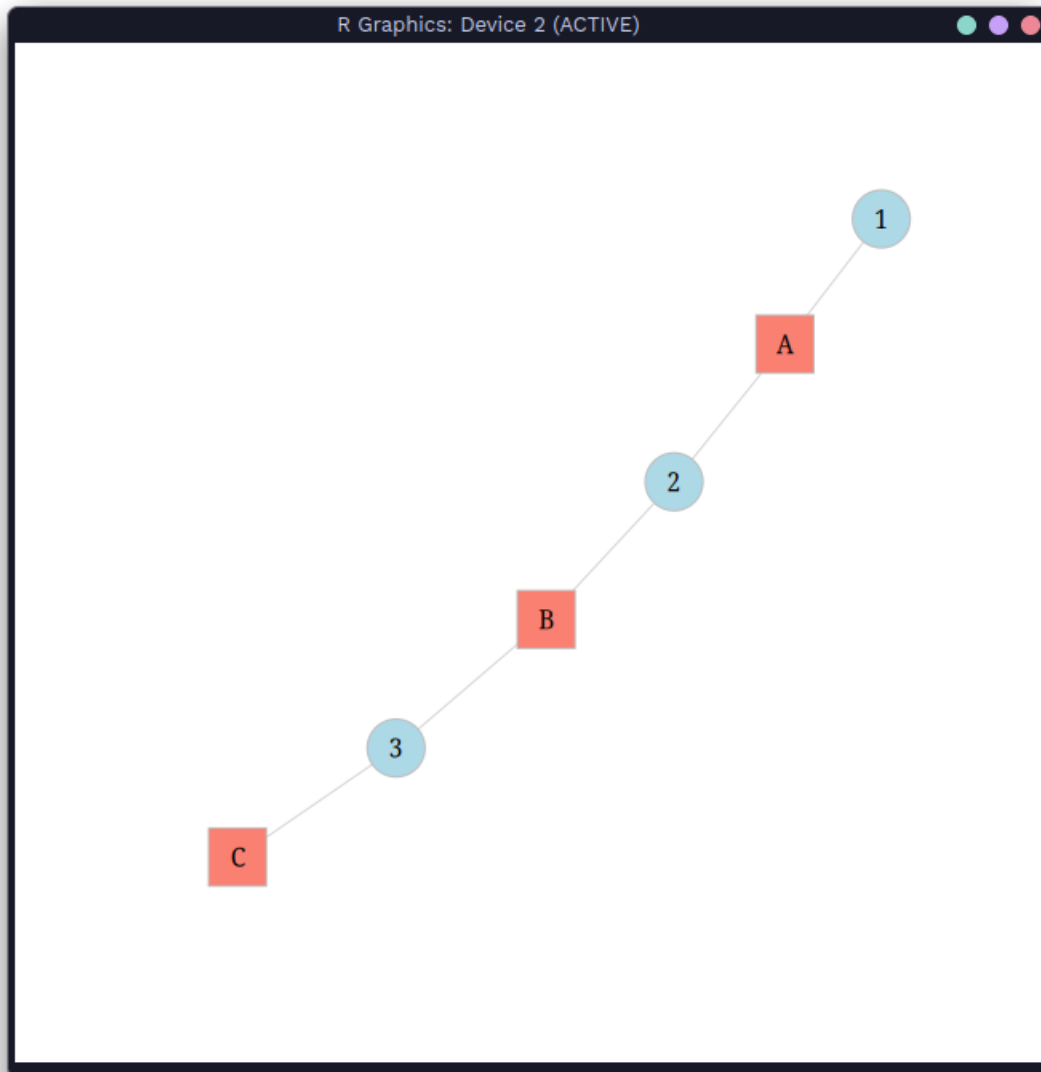


```
> plot(g, vertex.label.cex = 0.8, vertex.label.color = "black")  
> V(g)$color <- ifelse(V(g)$type, "lightblue", "salmon")  
> V(g)$shape <- ifelse(V(g)$type, "circle", "square")  
> E(g)$color <- "lightgray"  
> plot(g, vertex.label.cex = 0.8, vertex.label.color = "black")
```

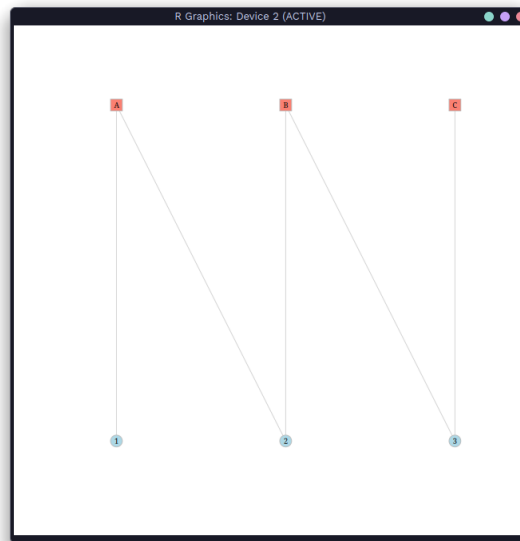


```
> V(g)$label.color <- "black"  
> V(g)$label.cex <- 1  
> V(g)$frame.color <- "gray"
```

```
> V(g)$size <- 18  
> plot(g, layout = layout_with_graphopt)
```



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

```
Warning message:
`graph.data.frame()` was deprecated in igraph 2.0.0.
i Please use `graph_from_data_frame()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> plot(g)
> bipartite.mapping(g)
$res
[1] TRUE

$type
  A    B    C    1    2    3
FALSE FALSE FALSE  TRUE  TRUE  TRUE

Warning message:
`bipartite.mapping()` was deprecated in igraph 2.0.0.
i Please use `bipartite_mapping()` instead.
This warning is displayed once every 8 hours.
Call `lifecycle::last_lifecycle_warnings()` to see where this warning was generated.
> 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)
```



HSNC UNIVERSITY, MUMBAI
KISHINCHAND CHELLARAM COLLEGE
M. Sc Computer Science Semester 3 (2024 – 25)



Practical No. 9

Aim: Hamming distance.

Name: Vishal Waikar

Roll No.: KFPMSCCS021

Date: 05/10/24

Sign:

Code:

```
> library(e1071)
> x <- c(0, 0, 0, 0)
> y <- c(0, 1, 0, 1)
> z <- c(1, 0, 1, 1)
> w <- c(0, 1, 1, 1)
> hamming.distance(x, y)
> hamming.distance(y, z)
> hamming.distance(y, w)
> hamming.distance(z, w)
> hamming.distance(x, w)
> hamming.distance(x, z)
```

```
~ : R — Konsole

** R
** inst
** byte-compile and prepare package for lazy loading
** help
*** installing help indices
** building package indices
** installing vignettes
** testing if installed package can be loaded from temporary location
** checking absolute paths in shared objects and dynamic libraries
** testing if installed package can be loaded from final location
** testing if installed package keeps a record of temporary installation path
* DONE (e1071)

The downloaded source packages are in
  '/tmp/RtmpHecAD9/downloaded_packages'
> library(e1071)
> x <- c(0, 0, 0, 0)
> y <- c(0, 1, 0, 1)
> z <- c(1, 0, 1, 1)
> w <- c(0, 1, 1, 1)
> hamming.distance(x, y)
[1] 2
> hamming.distance(y, z)
[1] 3
> hamming.distance(y, w)
[1] 1
> hamming.distance(z, w)
[1] 2
> hamming.distance(x, w)
[1] 3
> hamming.distance(x, z)
[1] 3
>
```