



Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

Experiment No. 6
Social Network Analysis using R (for example: Community Detection Algorithm)
Date of Performance: 05/10/2023
Date of Submission: 12/10/2023



Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

Aim: Social Network Analysis using R (for example: Community Detection Algorithm)

Theory:

Online social platforms have enabled people around the world to interact with each other and build relationships with others they share common interests with. This can be observed in real life — naturally, we tend to develop and maintain relationships with others that are similar to us. People with similar interests tend to gravitate towards each other and become associated in communities — clusters or groups of people that share similar traits with each other. Since people tend to cluster with others similar to them, we can use community detection to identify users with a high number of degrees (connections) and see how far their reach can travel in the network.

- User Data Extraction — Since we are only interested in user data, we will only extract the following variables:
- User_id — Yelp user ID; this is needed to make nodes and edges
- Name — user's first name
- Review count — the number of reviews user has written
- Yelping since — date user joined Yelp
- Friends — a list containing all of the user's friends by user_id
- Fans — number of fans user has
- Elite — number of years the user has Elite status
- Average stars — user's average rating of all reviews written

CODE:

```
library(igraph)
gizvan <- function(G) { c=
decompose. graph (G) l =
length(c)
v <= vector() while(l==1){
x <-E(G)
```



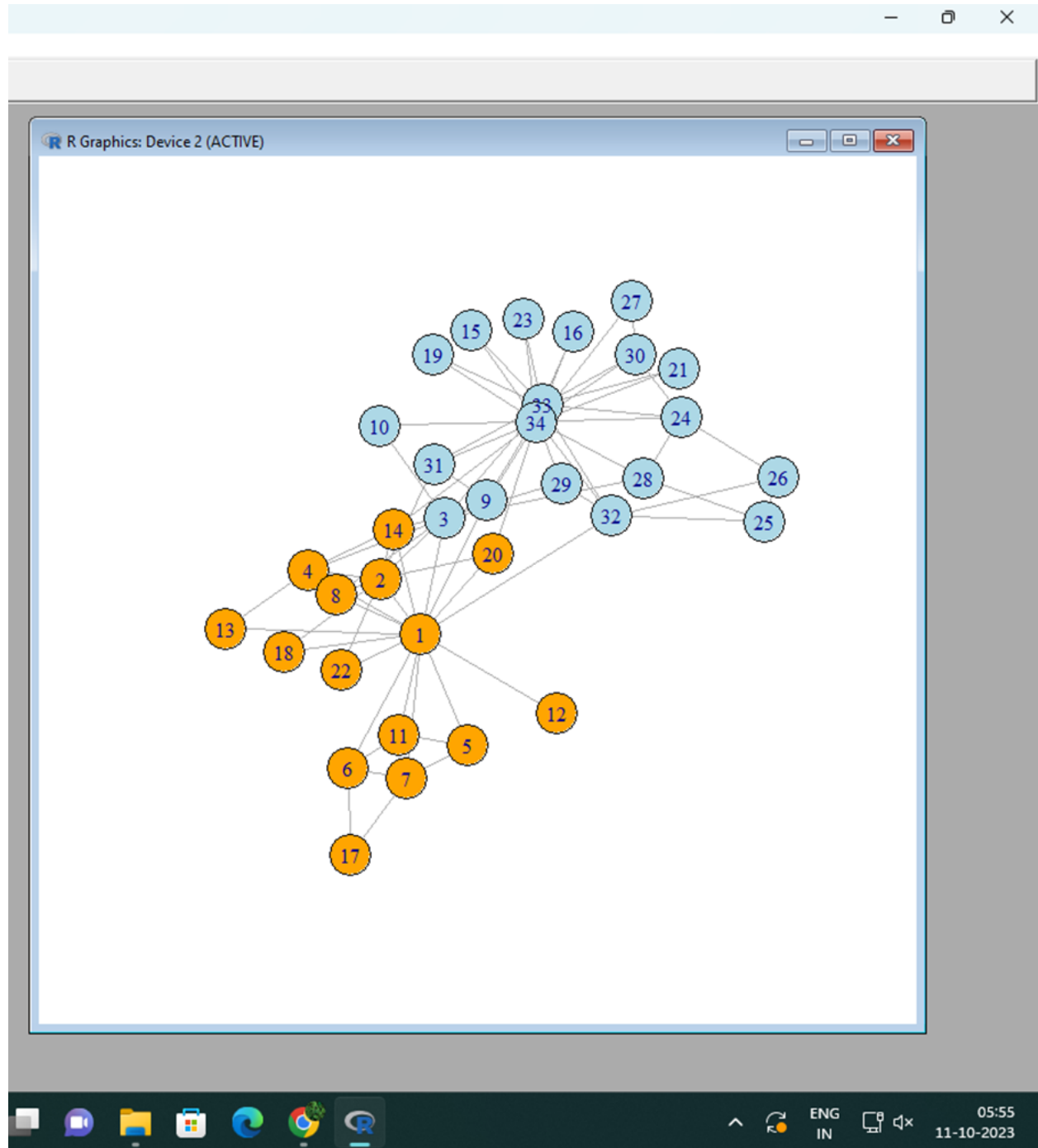
Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

```
y <- edge_betweenness (G) z
<- which.max(y)
edge <- x[z]
a <- ends (G,z[1]) [1]
b <- ends(G,z[1]) [2] v
<- c(v,a,b)
G <- delete_edges (G, edge) c =
decompose.graph (G)
l= length(c)
}
if(l==2)(
paths <- shortest.paths (G) for(i
in 1:length(V(G)))(
if (paths[a, i] !=Inf) {
V(G) [i]$color = "lightblue"
}
else{
V(G) [i]$color = "orange"
}
}
G <- G + edge(v)
plot(G)
}
return(c)
}
g <- read.graph("C:/Users/admin/Desktop/CommunityDetection/karate.gml",format = "gml")
plot(g)
c <- girvan(g)
```



Output:





Vidyavardhini's College of Engineering & Technology

Department of Computer Engineering

Conclusion:

This experiment focused on Social Network Analysis using R, specifically implementing a Community Detection Algorithm on a social network graph. The algorithm extracts user data and relationships from a social platform, such as Yelp, and uses the igraph library in R to detect communities within the network. The Girvan-Newman algorithm was applied to find communities by iteratively removing edges with the highest betweenness centrality. The result is a visualization of the network with color-coded communities, providing insights into how users cluster based on their interactions and similarities.