Social_Networks_first_assignment

Introduction

This project focuses on the analysis of a bipartite network that represents the Abu Sayyaf Group's involvement in kidnapping events in the Philippines in 2014. The dataset, obtained from the ICON project, links members of the Abu Sayyaf Group with the specific kidnappings they were involved in.

The goal of this analysis is to explore fundamental network properties, such as node connectivity, degree distribution, and overall structure. These properties will allow us to gain insights into the organizational dynamics of the group, which can be reflected in the network's topology.

https://networks.skewed.de/net/kidnappings

Gerdes, L. M., Ringler, K., & Autin, B. (2014). Assessing the Abu Sayyaf Group's Strategic and Learning Capacities. *Studies in Conflict & Terrorism*, *37*(3), 267–293. https://doi.org/10.1080/1057610X.2014.872021

Libraries

```r
library(readr)
library(igraph)
```

```

Attaching package: 'igraph'

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

    decompose, spectrum
```

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

    union
```

```r
library(ggplot2)
library(ggraph)
library(dplyr)
```

```
Attaching package: 'dplyr'

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

    as_data_frame, groups, union

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

    filter, lag

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

    intersect, setdiff, setequal, union
```

Running Code

We load the edges and nodes and create the graph from the data

```r
edges <- read_csv("network/edges.csv")
```

```
Rows: 402 Columns: 2
-- Column specification --------------------------------------------------------
Delimiter: ","
dbl (2): # source, target

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
head(edges, 4)
```

```
# A tibble: 4 x 2
  `# source` target
       <dbl>  <dbl>
1        105     22
2        105     32
3        105     40
4        105     41
```

```r
nodes <- read_csv("network/nodes.csv")
```

```
Rows: 351 Columns: 3
-- Column specification --------------------------------------------------------
Delimiter: ","
chr (2): name, _pos
dbl (1): # index

i Use `spec()` to retrieve the full column specification for this data.
i Specify the column types or set `show_col_types = FALSE` to quiet this message.
```

```r
head(nodes, 4)
```

```
# A tibble: 4 x 3
  `# index` name  `_pos`
      <dbl> <chr> <chr>
1         0 E1    array([ 4.69721231, -2.0472666 ])
2         1 E2    array([ 4.44806661, -2.97223624])
3         2 E3    array([ 2.6345862 , -4.90967325])
4         3 E4    array([ 4.28686718, -2.24611971])
```

```r
# Buiding a graph from data
g <- graph_from_data_frame(d = edges, vertices = nodes, directed = TRUE)
```

Questions

1. What is the number of nodes and links?

Nodes and links are also called vertices and edges. We can check their count with tables or by asking directly for their number:

```r
# links/edges
E(g)
```

```
+ 402/402 edges from 984c373 (vertex names):
 [1] N1 ->E23 N1 ->E33 N1 ->E41 N1 ->E42 N1 ->E44 N2 ->E23 N2 ->E33 N3 ->E6
 [9] N3 ->E9  N3 ->E10 N3 ->E14 N3 ->E19 N3 ->E25 N3 ->E28 N3 ->E31 N3 ->E32
[17] N4 ->E1  N4 ->E30 N4 ->E34 N5 ->E1  N5 ->E2  N5 ->E4  N5 ->E8  N6 ->E8
[25] N6 ->E9  N6 ->E10 N6 ->E11 N7 ->E4  N7 ->E8  N8 ->E8  N9 ->E4  N9 ->E12
[33] N9 ->E30 N9 ->E39 N10->E26 N10->E32 N10->E41 N10->E42 N10->E43 N10->E65
[41] N10->E79 N11->E3  N12->E3  N13->E32 N13->E35 N13->E67 N14->E3  N15->E3
[49] N16->E3  N17->E6  N18->E27 N18->E35 N19->E6  N19->E9  N19->E12 N19->E25
[57] N19->E29 N19->E32 N20->E6  N21->E30 N22->E27 N23->E27 N23->E30 N23->E35
[65] N24->E10 N25->E10 N26->E12 N27->E8  N27->E35 N28->E13 N29->E17 N29->E19
[73] N29->E20 N30->E17 N31->E19 N32->E19 N32->E28 N32->E32 N33->E18 N34->E18
+ ... omitted several edges
```

```r
ecount(g)
```

```
[1] 402
```

```r
# nodes/vertices
V(g)
```

```
+ 351/351 vertices, named, from 984c373:
  [1] E1   E2   E3   E4   E5   E6   E7   E8   E9   E10  E11  E12  E13  E14  E15
 [16] E16  E17  E18  E19  E20  E21  E22  E23  E24  E25  E26  E27  E28  E29  E30
 [31] E31  E32  E33  E34  E35  E36  E37  E38  E39  E40  E41  E42  E43  E44  E45
 [46] E46  E47  E48  E49  E50  E51  E52  E53  E54  E55  E56  E57  E58  E59  E60
 [61] E61  E62  E63  E64  E65  E66  E67  E68  E69  E70  E71  E72  E73  E74  E75
 [76] E76  E77  E78  E79  E80  E81  E82  E83  E84  E85  E86  E87  E88  E89  E90
 [91] E91  E92  E93  E94  E95  E96  E97  E98  E99  E100 E101 E102 E103 E104 E105
[106] N1   N2   N3   N4   N5   N6   N7   N8   N9   N10  N11  N12  N13  N14  N15
[121] N16  N17  N18  N19  N20  N21  N22  N23  N24  N25  N26  N27  N28  N29  N30
[136] N31  N32  N33  N34  N35  N36  N37  N38  N39  N40  N41  N42  N43  N44  N45
+ ... omitted several vertices
```

```r
vcount(g)
```

```
[1] 351
```

Our dataset has 402 links and 351 nodes

2. What is the average degree in the network? And the standard deviation of the degree?

To calculate the average degree and standard deviation of nodes, we can use the degree values obtained from the following code:

```
deg <- degree(g,mode="all")
```

```
mean(deg)
```

```
[1] 2.290598
```

```
sd(deg)
```

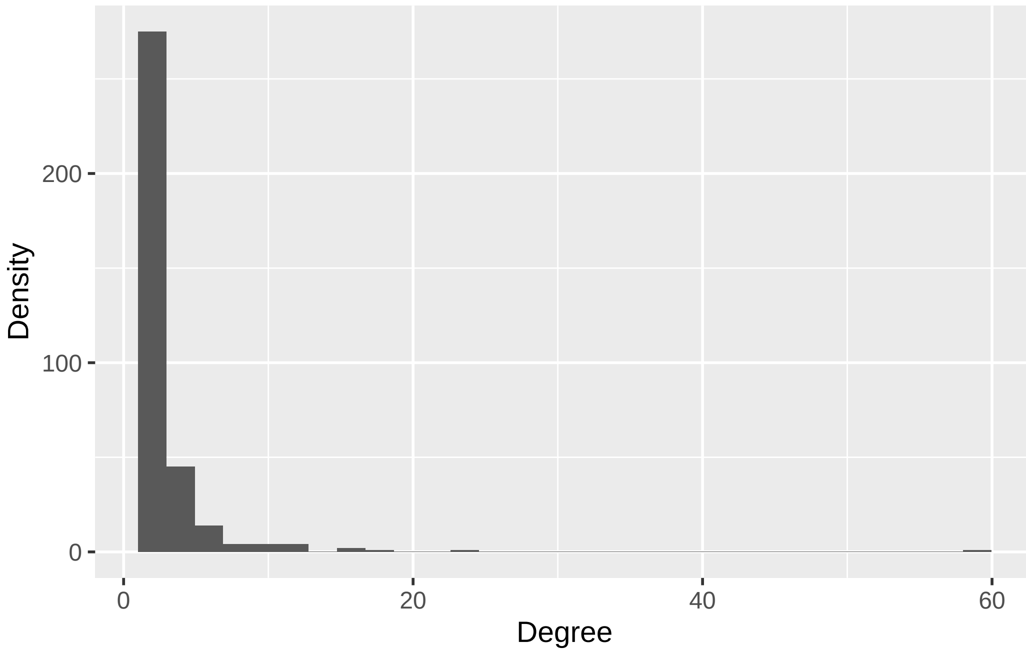
```
[1] 3.885636
```

The average degree of the network is 2.29. This points to a sparse and loosely connected structure, consistent with how clandestine networks often operate: keeping operational cells small to avoid detection. The standard deviation of the degree is 3.89, which is quite high relative to the mean. This suggests a heterogeneous structure: while most nodes are weakly connected, a few nodes are highly connected, likely representing core individuals in the Abu Sayyaf Group or events that involved multiple operatives. This uneven distribution reflects typical features of terrorist or militant group dynamics, where key figures coordinate multiple decentralized actions.

3. Plot the degree distribution in linear-linear scale and in log-log-scale. Does it have a typical connectivity? What is the degree of the most connected node?

```
# linear-linear
ggplot() +
  geom_histogram(aes(x=degree(g,mode="all"))) +
  labs(x="Degree",y="Density")
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```




```
# table of degrees
table_deg <- data.frame(deg)
table_deg %>%
  arrange(desc(deg)) %>%
  head(10)
```

```
     deg
E35   58
E30   23
E27   17
N103  16
E32   15
E65   12
E39   11
E81   11
N127  11
E84   10
```

```
# max degree node
max_degree_node <- which.max(deg)
deg[max_degree_node]
```

```
E35
 58
```

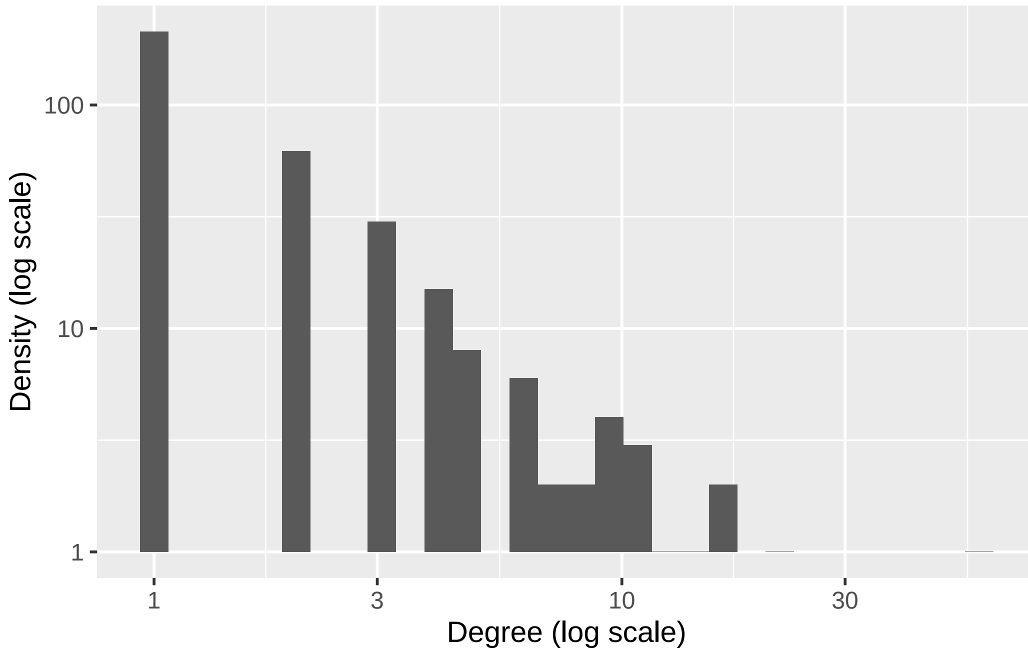
The histogram illustrates the distribution of node degrees within the network, showing how many connections each node has. From the distribution, it's clear that the vast majority of nodes have a very low degree, typically around 1 or 2 connections: most nodes are minimally connected, indicating a sparse network structure. However, the histogram also reveals a long right tail, meaning that while most nodes are weakly connected, a small number of nodes have a significantly higher degree. The most connected node has 58 connections, with a in comparison way lower second most connected node with 23 connections. Still, in comparison to one or two connections that is quite high. The following eight most connected nodes are around the 17 to 10 connections marks. This pattern suggests a right-skewed or heavy-tailed distribution, where a few highly connected nodes coexist with many low-degree nodes, which would indicate a power-law distribution.

```r
# log-log
ggplot() +
  geom_histogram(aes(x=degree(g, mode="all"))) +
  scale_x_log10() +
  scale_y_log10() +
  labs(x="Degree (log scale)", y="Density (log scale)")
```

```
`stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
Warning in scale_y_log10(): log-10 transformation introduced infinite values.
```

```
Warning: Removed 15 rows containing missing values or values outside the scale range
(`geom_bar()`).
```



While with the linear-linear histogram we can observe the skewed distribution of node degrees and the presence of a few highly connected hubs, with the log-log plot we get a better view into the scaling behavior of the network. This plot is created simply by transforming both axes with a common logarithm.

In this plot, the degree distribution appears to align roughly along a straight line in the log-log space. This pattern supports the idea that the degree distribution follows a power-law and hence the network may be scale-free, where a small number of nodes dominate the connectivity, and the probability of a node having a high degree decreases polynomially rather than exponentially.

The few nodes that are highly connected can play a crucial role in the flow of information, connectivity, and resilience of the network. Their presence may also influence the network's vulnerability as if one of them fails or is removed, it could significantly disrupt the system.

4. What is the clustering coefficient (transitivity) in the network?

```r
transitivity(g)
```

```
[1] 0
```

The global clustering coefficient (transitivity) of the network is 0, which means that there are no closed triplets, i.e. there are no triangles where three nodes are all directly connected to each other. This suggests that the network is not clustered at all, and that nodes tend to connect in a more tree-like or chain-like structure rather than forming triangleness with tightly knit groups. Such a result indicates a network with no local cohesion, where neighbors of a node are actually never connected to each other.

This result is significant in the context of Abu Sayyaf operations: while limiting collaboration, enhances the group's resilience against infiltration as each actor is only connected to a "superior" agent.

5. What is the assortativity (degree) in the network?

```r
assortativity_degree(g)
```

```
[1] -0.2336613
```

The assortativity coefficient based on node degree is -0.23, indicating a moderate negative assortativity in the network, where nodes with a high degree tend to connect to nodes with low degree, rather than to other hubs. It seems that in this network the highly connected nods serve as bridges that link many less connected nodes.

Disassortative mixing by degree is normally a pattern found in fictional or artificial networks, which makes it very interesting that it formed here naturally. It reflects a centralized structure where a few key figures repeatedly appear across many separate events, while most events involve only a small number of actors. Hence, for a criminal or militant network it would imply a strategic effort to decentralize risk: central figures maintain influence across various operations without necessarily linking collaborators directly to each other, which may reduce traceability and limit network exposure in case of capture.

6. Using the Louvain method, does the network have a community structure? If so, what is its modularity?

To investigate the structure of the network, we used both the Leiden and Louvain methods.

To use the Louvain algorithm, it is important that we transform our directed network into an undirected network, as it is only designed for undirected networks. We also use the Leiden method as it similar to the Louvain algorithm but works faster (which is not necessary here with our small network but it is still interesting to compare.)

```r
set.seed(123)
g_undirected <- as.undirected(g, mode = "collapse")
```

```
Warning: `as.undirected()` was deprecated in igraph 2.1.0.
i Please use `as_undirected()` instead.
```

```r
# Louvain
louv <- cluster_louvain(g_undirected)

# Leiden
leid <- cluster_leiden(g_undirected)

# Community counts
length(sizes(louv)) # length(louv) also works
```

```
[1] 36
```

```r
length(sizes(leid))
```

```
[1] 351
```

```r
# Modularity
modularity(g_undirected, membership(louv))
```

```
[1] 0.7910448
```

```r
modularity(g_undirected, membership(leid))
```

```
[1] -0.01102386
```

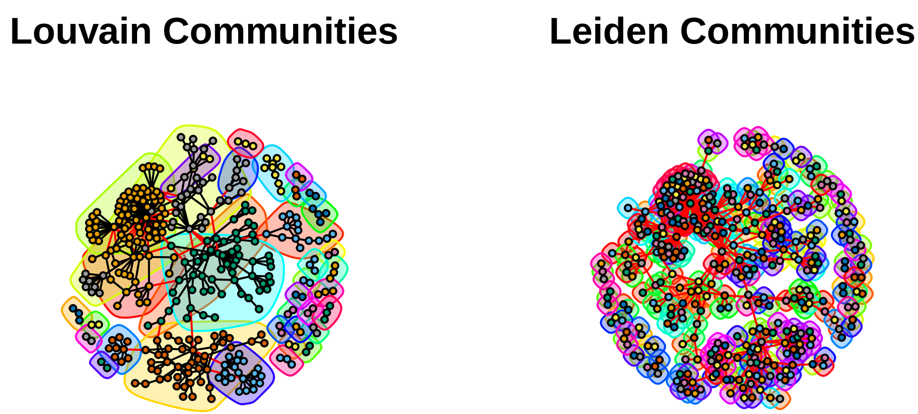
The Louvain method identified ca. 36 communities with a high modularity score of 0.791, indicating a strong community structure. The high modularity suggests that the network contains nodes that are more densely connected within communities than between them, i.e. more densely connected internally than externally, which can be interpreted as meaningful substructures.

The Leiden algorithm, on the other hand, detected 351 communities, corresponding to the total number of nodes in the network. The modularity score of -0.011 indicates that this partition is worse than random, and no significant structural groupings were identified.

Overall, we conclude that the Louvain method seems better suited for detecting structural patterns in the Abu Sayyaf network. To further illustrate the differences between the two methods we visualize them.

```r
par(mfrow = c(1, 2))
plot(louv, g_undirected,
     vertex.label = "", vertex.size = 5, main = "Louvain Communities")

plot(leid, g_undirected,
     vertex.label = "", vertex.size = 5, main = "Leiden Communities")
```



The Louvain method clearly identifies larger and cohesive community clusters. These groups are visibly well-defined, with nodes that are tightly connected internally and more loosely connected to other communities. In contrast, the Leiden algorithm divides the network into many small, fragmented communities, often consisting of just one or two nodes.

We can see where the modularity scores come from, as modularity is really a summary statistic of the network's clustering quality, where the goodness of the partitions are checked. The higher the modularity, the better the description of the network communities are. Hence, the Louvain graph is consistent with a high modularity score, with the negative modularity of Leiden supports the earlier conclusion: Louvain seems more appropriate for revealing community structure in this network.

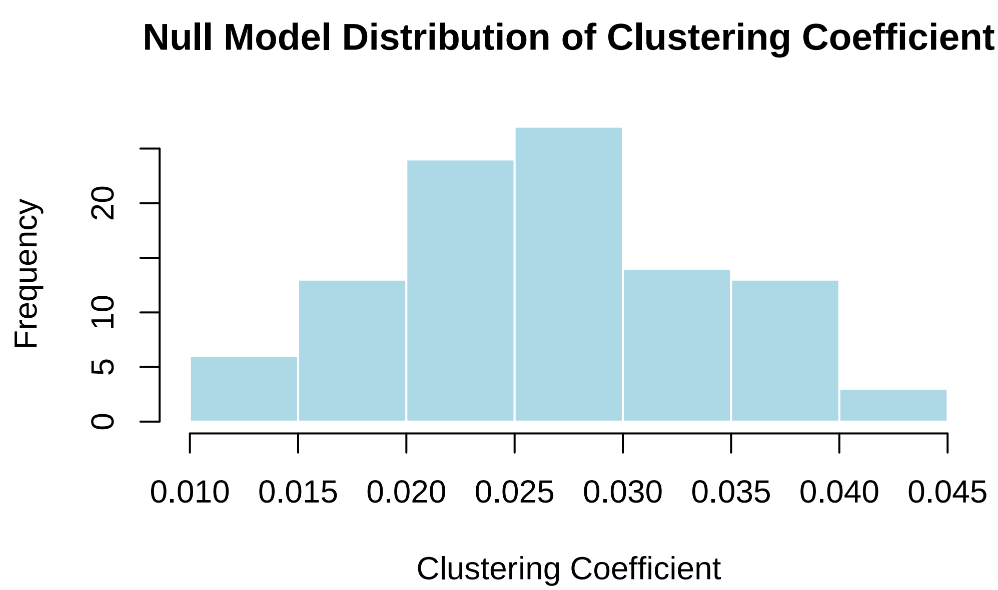
7. Test that the clustering coefficient in the network cannot be statistically explain by a configuration model in which the nodes have the same degree distribution as the original.

```
# Original clustering coefficient (global transitivity)
observed_transitivity <- transitivity(g)

# 100 null models with same degree distribution
null_transitivities <- replicate(100, {
  transitivity(sample_degseq(degree(g), method = "simple"))
})
```

```
Warning: The `method` argument of `sample_degseq()` must be configuration instead of
simple as of igraph 2.1.0.
```

```
# Plot of clustering coefficients distribution from null models
hist(null_transitivities,
     main = "Null Model Distribution of Clustering Coefficient",
     xlab = "Clustering Coefficient", col = "lightblue", border = "white")
abline(v = observed_transitivity, col = "red", lwd = 2)
```



```r
# Empirical p-value
mean(null_transitivities >= observed_transitivity)
```

```
[1] 1
```

The empirical p-value is 1, indicating that none of the 100 networks in the configuration model had as low a clustering coefficient as the original, i.e. the actually observed clustering is not higher than expected by chance. This means that the lack of clustering in the Abu Sayyaf network is not a statistical error of its degree distribution, but rather a structurally significant feature that can be fully explained by degree sequence alone.

Such a pattern suggests intentional decentralization, perhaps to reduce the risk of network detection by closely linked groups. It provides support for the notion of a subdivided cell structure typical of underground organizations.

8. Visualize the neighborhood of the node with the largest centrality (closeness)

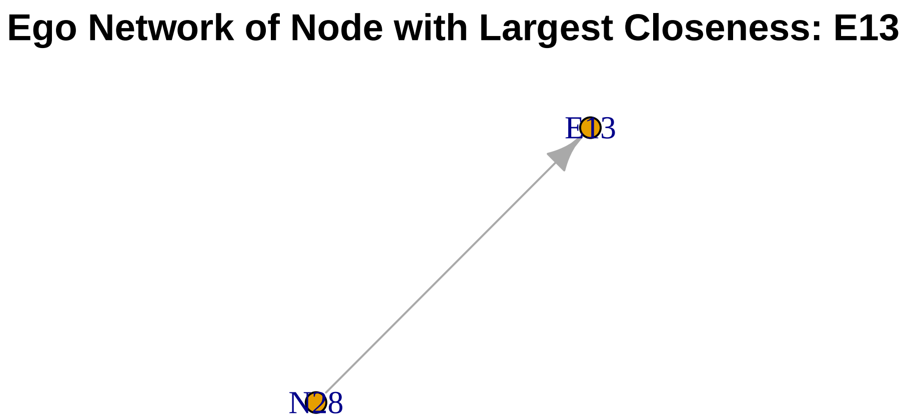
```r
# Closeness centrality
closeness_vals <- closeness(g, mode = "all")

# Node with largest closeness centrality
top_node <- names(which.max(closeness_vals))
top_node
```

```
[1] "E13"
```

```r
# Ego-graph (1 step neighborhood)
ego_graph <- make_ego_graph(g, order = 1, nodes = top_node)[[1]]

# Plot
plot(ego_graph,
     main = paste("Ego Network of Node with Largest Closeness:", top_node))
```



The ego network of node E13, which has the highest Closeness Centrality in the network, shows a very loose structure as it only connects to one other node. Closeness centraility indicates how "close" a node is to all others in terms of the shortest paths. It measures global position, not local connectivity, hence a node does not require many direct links but that it simply sits at the shortest distance from the rest.

In bipartite networks, event nodes such as E13 achieve high closeness as they act as bridges that connect distant parts of the network. In this case, E13 probably plays an important role as it connects two large parts of the graph via indirect paths. Its centrality does not come from its local connectivity, but from its strategic position in the network. It may serve as an important event that connects actors that would otherwise have no connection.

We can check this with a betweenness centraility, which counts the number of shortest paths that pass through a node:

```
betweenness(g_undirected)["E13"]
```

```
E13
  0
```

Suprisingly, we find that E13 has a betweenness centrality of 0, meaning no shortest paths pass through it, which completely undermines our initial theory. :/ The explanation for this could be that E13 connects indirectly to distant nodes, but isn't between other nodes in terms of shortest paths. It's close to them (short paths to many nodes), but not in between them.

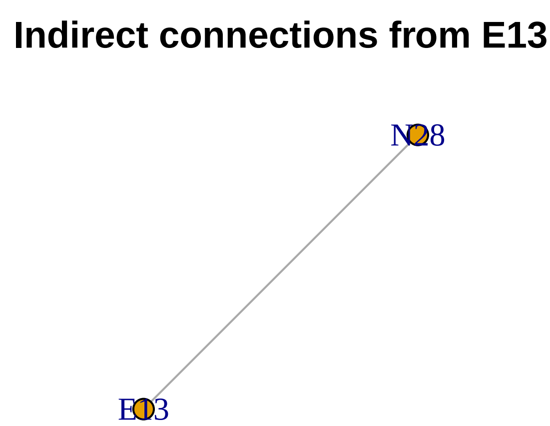
We can try to visualise this.

```
paths <- shortest_paths(g_undirected,
                        from = "E13", to = V(g_undirected),
                        output = "vpath")
```

```
Warning in shortest_paths(g_undirected, from = "E13", to = V(g_undirected), :
At vendor/cigraph/src/paths/unweighted.c:444 : Couldn't reach some vertices.
```

```
subg <- induced_subgraph(g_undirected, vids = unique(unlist(paths$vpath)))

plot(subg,
     main = "Indirect connections from E13")
```



However, this result raises suspicion about the structure of the overall network. If E13 truly had indirect connections to many distant nodes, we would expect the visualised subgraph to include more than one neighbour but the graph actually reveals just a single connection. E13 may not actually be part of a well-connected structure, but instead exists in a small, disconnected fragment of the graph. Hence it might be one of the lower-level agents of the pirate network.

To understand what is actually going on, we examine closeness centrality scores across all nodes to see whether this pattern is isolated or part of a broader issue in the network structure.

```
btw <- betweenness(g_undirected, weights=E(g_undirected)$value)
cls <- closeness(g_undirected)

table <- data.frame(deg, btw, cls)

table %>%
  arrange(desc(cls))
```

```
     deg         btw          cls
E13    1     0.00000 1.0000000000
E22    1     0.00000 1.0000000000
E38    1     0.00000 1.0000000000
E40    1     0.00000 1.0000000000
E45    1     0.00000 1.0000000000
E48    1     0.00000 1.0000000000
E49    1     0.00000 1.0000000000
E50    1     0.00000 1.0000000000
E70    1     0.00000 1.0000000000
E76    1     0.00000 1.0000000000
E80    1     0.00000 1.0000000000
E82    1     0.00000 1.0000000000
E88    1     0.00000 1.0000000000
E92    1     0.00000 1.0000000000
E93    1     0.00000 1.0000000000
E95    1     0.00000 1.0000000000
N28    1     0.00000 1.0000000000
N35    1     0.00000 1.0000000000
N107   1     0.00000 1.0000000000
N108   1     0.00000 1.0000000000
N114   1     0.00000 1.0000000000
N115   1     0.00000 1.0000000000
N118   1     0.00000 1.0000000000
N140   1     0.00000 1.0000000000
N148   1     0.00000 1.0000000000
N155   1     0.00000 1.0000000000
N171   1     0.00000 1.0000000000
N172   1     0.00000 1.0000000000
N186   1     0.00000 1.0000000000
N196   1     0.00000 1.0000000000
N214   1     0.00000 1.0000000000
N245   1     0.00000 1.0000000000
E18    2     1.00000 0.5000000000
```

E51	2	1.00000	0.5000000000
E52	2	1.00000	0.5000000000
E96	2	1.00000	0.5000000000
E97	2	1.00000	0.5000000000
E99	2	1.00000	0.5000000000
E103	2	1.00000	0.5000000000
E46	3	3.00000	0.3333333333
N33	1	0.00000	0.3333333333
N34	1	0.00000	0.3333333333
N119	1	0.00000	0.3333333333
N120	1	0.00000	0.3333333333
N121	1	0.00000	0.3333333333
N122	1	0.00000	0.3333333333
N183	1	0.00000	0.3333333333
N204	1	0.00000	0.3333333333
N205	1	0.00000	0.3333333333
N206	1	0.00000	0.3333333333
N207	1	0.00000	0.3333333333
N208	1	0.00000	0.3333333333
N210	1	0.00000	0.3333333333
N246	1	0.00000	0.3333333333
N109	1	0.00000	0.2000000000
N110	1	0.00000	0.2000000000
N111	1	0.00000	0.2000000000
E98	5	9.00000	0.0833333333
E100	5	9.00000	0.0833333333
N199	3	12.20000	0.0769230769
N200	2	0.20000	0.0588235294
N201	2	0.20000	0.0588235294
N202	2	0.20000	0.0588235294
N203	2	0.20000	0.0588235294
E56	2	7.00000	0.0555555556
N124	1	0.00000	0.0400000000
E35	58	22230.21187	0.0009337068
N41	4	16044.95794	0.0008880995
N43	3	1793.78694	0.0008347245
N92	4	2072.26478	0.0008347245
N13	3	3299.28633	0.0008319468
E65	12	6195.73556	0.0008312552
N82	2	1091.28633	0.0008210181
N230	2	1091.28633	0.0008210181
E104	4	14906.56491	0.0008006405
E32	15	10400.70417	0.0007942812

N10	7	4567.45121	0.0007886435
N130	3	1416.28908	0.0007886435
N48	3	1251.98003	0.0007849294
N23	3	535.14131	0.0007704160
N36	3	535.14131	0.0007704160
N212	3	581.32335	0.0007668712
N44	2	336.12926	0.0007610350
N50	2	336.12926	0.0007610350
N94	2	336.12926	0.0007610350
N229	2	336.12926	0.0007610350
N27	2	1589.97712	0.0007541478
N85	3	427.33080	0.0007541478
N18	2	134.48382	0.0007496252
N73	2	134.48382	0.0007496252
N75	2	134.48382	0.0007496252
N76	2	134.48382	0.0007496252
N77	2	134.48382	0.0007496252
N78	2	134.48382	0.0007496252
N79	2	134.48382	0.0007496252
N97	2	197.94687	0.0007496252
N174	2	197.94687	0.0007496252
N83	2	208.90782	0.0007440476
N180	2	208.90782	0.0007440476
N38	2	843.00000	0.0007418398
N39	1	0.00000	0.0007385524
N40	1	0.00000	0.0007385524
N51	1	0.00000	0.0007385524
N65	1	0.00000	0.0007385524
N69	1	0.00000	0.0007385524
N139	1	0.00000	0.0007385524
N182	1	0.00000	0.0007385524
N198	1	0.00000	0.0007385524
N215	1	0.00000	0.0007385524
N216	1	0.00000	0.0007385524
N217	1	0.00000	0.0007385524
N218	1	0.00000	0.0007385524
N219	1	0.00000	0.0007385524
N220	1	0.00000	0.0007385524
N221	1	0.00000	0.0007385524
N222	1	0.00000	0.0007385524
N223	1	0.00000	0.0007385524
N224	1	0.00000	0.0007385524
N225	1	0.00000	0.0007385524

N226	1	0.00000	0.0007385524
N227	1	0.00000	0.0007385524
N228	1	0.00000	0.0007385524
N231	1	0.00000	0.0007385524
N232	1	0.00000	0.0007385524
N233	1	0.00000	0.0007385524
N234	1	0.00000	0.0007385524
N235	1	0.00000	0.0007385524
N236	1	0.00000	0.0007385524
N238	1	0.00000	0.0007385524
E41	6	1987.12700	0.0007304602
N103	16	15732.85521	0.0007267442
E69	3	815.81649	0.0007230658
E42	5	1275.97277	0.0007199424
E27	17	2033.31229	0.0007127584
N3	9	5336.41390	0.0007062147
E30	23	4006.84545	0.0007037298
N19	6	3818.73711	0.0006973501
E39	11	1563.80320	0.0006958942
N37	6	1623.93140	0.0006849315
N66	4	852.92620	0.0006839945
N113	5	1188.10447	0.0006821282
E67	8	1960.00000	0.0006798097
E47	1	0.00000	0.0006752194
N68	2	315.76079	0.0006747638
N32	3	701.53844	0.0006729475
N131	1	0.00000	0.0006729475
N132	1	0.00000	0.0006729475
N133	1	0.00000	0.0006729475
N134	1	0.00000	0.0006729475
N135	1	0.00000	0.0006729475
N136	1	0.00000	0.0006729475
N197	1	0.00000	0.0006729475
E8	6	1713.87963	0.0006653360
N141	3	1165.98154	0.0006631300
N9	4	1221.46151	0.0006553080
N117	2	283.00000	0.0006535948
N211	1	0.00000	0.0006527415
E79	4	1402.00000	0.0006489293
N213	1	0.00000	0.0006485084
N239	1	0.00000	0.0006485084
N240	1	0.00000	0.0006485084
E26	2	283.00000	0.0006455778

E43	1	0.00000	0.0006447453
E34	6	469.16340	0.0006381621
E74	8	1651.63083	0.0006341154
E78	4	1946.15279	0.0006277464
E55	2	105.74370	0.0006269592
E19	4	2065.58054	0.0006261741
E71	5	1405.70151	0.0006261741
E75	4	1109.15279	0.0006253909
E63	3	826.15279	0.0006246096
N55	2	64.52824	0.0006203474
N72	2	64.52824	0.0006203474
E6	5	4360.69047	0.0006199628
N1	5	1686.18981	0.0006180470
E54	6	1474.02149	0.0006169031
E81	11	2971.04296	0.0006169031
E102	2	2182.45070	0.0006161429
E9	3	372.71911	0.0006153846
E57	3	631.02149	0.0006146281
E83	3	565.00000	0.0006146281
E61	2	348.02149	0.0006138735
E66	2	348.02149	0.0006138735
E12	3	922.90113	0.0006116208
N42	2	47.24722	0.0006097561
N6	4	678.47408	0.0006082725
E25	2	43.69047	0.0006079027
N86	1	0.00000	0.0006053269
N96	1	0.00000	0.0006053269
E85	4	846.00000	0.0006049607
E91	2	283.00000	0.0006035003
E86	1	0.00000	0.0006027728
E89	1	0.00000	0.0006027728
E90	1	0.00000	0.0006027728
N154	1	0.00000	0.0006002401
N4	3	315.48357	0.0005988024
E10	4	743.36832	0.0005984440
N116	1	0.00000	0.0005980861
N22	1	0.00000	0.0005931198
N62	1	0.00000	0.0005931198
N70	1	0.00000	0.0005931198
N74	1	0.00000	0.0005931198
E31	3	565.00000	0.0005899705
E28	2	8.97495	0.0005892752
E14	1	0.00000	0.0005885815

N21	1	0.00000	0.0005868545
N45	1	0.00000	0.0005868545
N46	1	0.00000	0.0005868545
N47	1	0.00000	0.0005868545
N52	1	0.00000	0.0005868545
N53	1	0.00000	0.0005868545
N54	1	0.00000	0.0005868545
N93	1	0.00000	0.0005868545
N95	1	0.00000	0.0005868545
N150	1	0.00000	0.0005868545
N84	2	20.47610	0.0005854801
E29	4	846.00000	0.0005844535
N71	1	0.00000	0.0005813953
N237	1	0.00000	0.0005813953
N243	1	0.00000	0.0005813953
N29	3	1775.43509	0.0005800464
N5	4	394.48534	0.0005780347
N7	2	31.29553	0.0005747126
E64	2	283.00000	0.0005743825
E60	1	0.00000	0.0005737235
E37	1	0.00000	0.0005730659
E58	1	0.00000	0.0005717553
E59	1	0.00000	0.0005717553
E68	1	0.00000	0.0005717553
N87	1	0.00000	0.0005701254
N88	1	0.00000	0.0005701254
N89	1	0.00000	0.0005701254
N90	1	0.00000	0.0005701254
N91	9	2426.11250	0.0005701254
N143	1	0.00000	0.0005701254
N144	1	0.00000	0.0005701254
N209	1	0.00000	0.0005701254
E4	3	167.46790	0.0005608525
N8	1	0.00000	0.0005599104
N242	1	0.00000	0.0005599104
N244	3	2545.05815	0.0005567929
E101	1	0.00000	0.0005515720
N127	11	3266.61847	0.0005506608
N156	3	565.00000	0.0005494505
N160	1	0.00000	0.0005482456
N161	1	0.00000	0.0005482456
N63	1	0.00000	0.0005458515
E17	3	1527.05413	0.0005414185

N81	1	0.00000	0.0005405405
N67	1	0.00000	0.0005376344
N98	1	0.00000	0.0005376344
N142	1	0.00000	0.0005376344
N146	1	0.00000	0.0005376344
N152	1	0.00000	0.0005376344
N125	2	843.00000	0.0005347594
N145	3	3526.00000	0.0005347594
N128	2	46.72849	0.0005336179
N187	2	313.79589	0.0005336179
N31	1	0.00000	0.0005319149
N151	1	0.00000	0.0005319149
N153	1	0.00000	0.0005313496
N17	1	0.00000	0.0005274262
N20	1	0.00000	0.0005274262
E23	3	423.50000	0.0005271481
E1	2	57.77533	0.0005265929
E33	2	140.50000	0.0005265929
E44	2	283.00000	0.0005265929
N49	1	0.00000	0.0005252101
N137	1	0.00000	0.0005252101
N162	1	0.00000	0.0005252101
N163	1	0.00000	0.0005252101
N164	1	0.00000	0.0005252101
N165	1	0.00000	0.0005252101
N166	1	0.00000	0.0005252101
N167	1	0.00000	0.0005252101
N168	1	0.00000	0.0005252101
N169	1	0.00000	0.0005252101
N170	1	0.00000	0.0005252101
N179	1	0.00000	0.0005252101
N181	1	0.00000	0.0005252101
E53	3	1036.38255	0.0005238345
N138	1	0.00000	0.0005235602
N173	1	0.00000	0.0005235602
N175	1	0.00000	0.0005235602
N26	1	0.00000	0.0005213764
E11	1	0.00000	0.0005189414
E21	2	395.99088	0.0005178664
N176	1	0.00000	0.0005165289
N177	1	0.00000	0.0005165289
N178	1	0.00000	0.0005165289
N185	1	0.00000	0.0005154639

N24	1	0.00000	0.0005117707
N25	1	0.00000	0.0005117707
N123	3	1253.71671	0.0005081301
N60	1	0.00000	0.0005055612
N61	1	0.00000	0.0005055612
E72	5	656.40051	0.0005032713
N56	1	0.00000	0.0005015045
N57	1	0.00000	0.0005015045
N58	1	0.00000	0.0005015045
E20	1	0.00000	0.0004982561
E2	1	0.00000	0.0004967710
N112	1	0.00000	0.0004940711
E36	2	564.00000	0.0004918839
E62	1	0.00000	0.0004909180
E84	10	2264.75293	0.0004852014
E73	9	2236.00000	0.0004847310
E94	2	283.00000	0.0004768717
E87	1	0.00000	0.0004755112
E105	1	0.00000	0.0004755112
N30	1	0.00000	0.0004694836
E3	7	2221.00000	0.0004692633
E7	2	816.00000	0.0004670715
E77	3	565.00000	0.0004653327
N2	2	1.00000	0.0004591368
N129	1	0.00000	0.0004587156
N64	1	0.00000	0.0004582951
N147	1	0.00000	0.0004405286
N149	1	0.00000	0.0004405286
N80	2	283.00000	0.0004321521
N188	1	0.00000	0.0004266212
N189	1	0.00000	0.0004266212
N190	1	0.00000	0.0004266212
N191	1	0.00000	0.0004266212
N192	1	0.00000	0.0004266212
N193	1	0.00000	0.0004266212
N194	1	0.00000	0.0004266212
N195	1	0.00000	0.0004266212
N101	1	0.00000	0.0004262575
N102	1	0.00000	0.0004262575
N104	1	0.00000	0.0004262575
N105	1	0.00000	0.0004262575
N106	1	0.00000	0.0004262575
N157	1	0.00000	0.0004262575

```
N158    1      0.00000 0.0004262575
N159    1      0.00000 0.0004262575
N184    1      0.00000 0.0004201681
N59     4   1402.00000 0.0004163197
N11     1      0.00000 0.0004142502
N12     1      0.00000 0.0004142502
N14     1      0.00000 0.0004142502
N15     1      0.00000 0.0004142502
N16     1      0.00000 0.0004142502
N126    1      0.00000 0.0004111842
N241    1      0.00000 0.0004111842
E24     1      0.00000 0.0003850597
E15     2    843.00000 0.0003732736
E5      1      0.00000 0.0003724395
N99     2    564.00000 0.0003380663
E16     2    283.00000 0.0003087373
N100    1      0.00000 0.0002839296
```

The first 32 nodes have a closeness centrality of exactly 1, despite having only one connection. This counterintuitive result points towards the network appearing to be disconnected and fragmented, as these nodes are only connected to one other node in small components. Since no other nodes are reachable from them, their sum of distances is technically 1, resulting in a closeness score of 1 under igraph's calculation.

To avoid this problem, we can try to work with largest connected component, which is typically done with disconnected networks.

```r
cc <- components(g_undirected)
head(cc$csize)
```

```
[1] 285   2   3   2   2   2
```

```r
sort(cc$csize, decreasing = TRUE)
```

```
 [1] 285   9   4   3   3   3   3   3   3   3   2   2   2   2   2   2   2   2   2
[20]   2   2   2   2   2   2   2
```

```r
head(cc$no)
```

```
[1] 26
```

There seems to be 26 components, and the largest one has a size of 285, followed by sizes of 9, 4, 3 and 2. This shows how a significant portion of the network is fragmented into many small, disconnected subgraphs. Hence, we have to focus on the largest connected component (Giant Component) for meaningful structural analysis.

```r
gGC <- decompose(g_undirected)[[1]]

vcount(gGC)
```

```
[1] 285
```

```r
ecount(gGC)
```

```
[1] 357
```

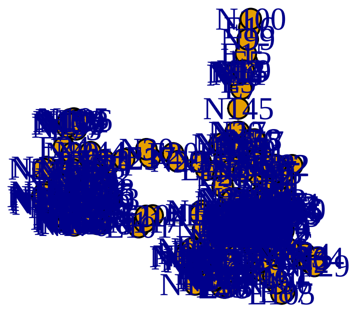
```r
diameter(g)
```

```
[1] 1
```

```r
diameter(gGC)
```

```
[1] 18
```

```r
plot(gGC)
```



This subgraph has 285 nodes and 457 links. When we compare the diameter (largest distance) of our total network (1) with the diameter of our Giant Component (18), we can see that the overall network appears trivially connected due to many isolated pairs of nodes, while the Giant Component reveals the true structural depth and path diversity of the core network.

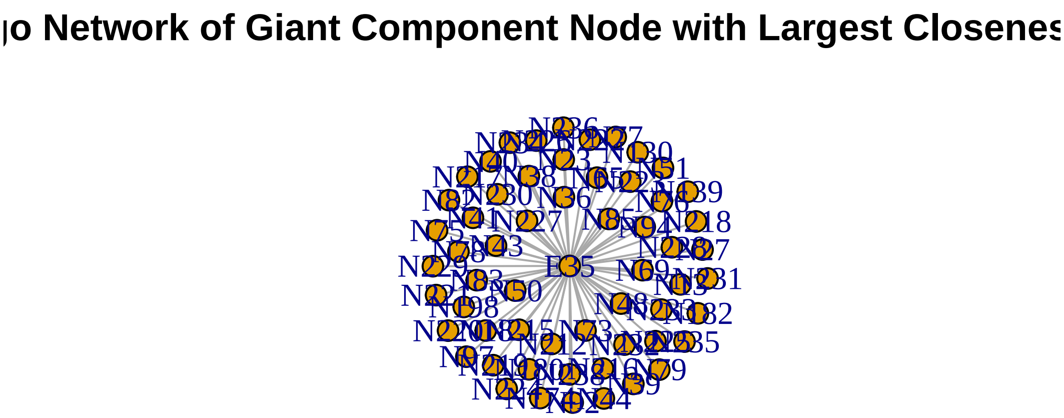
We can now recalculate and visualise the neighborhood of the node with the largest closeness.

```
closeness_vals_GC <- closeness(gGC, mode = "all")
top_node_GC <- names(which.max(closeness_vals_GC))
top_node_GC
```

```
[1] "E35"
```

```
ego_graph_GC <- make_ego_graph(gGC, order = 1, nodes = top_node_GC)[[1]]

plot(ego_graph_GC,
     main = paste("Ego Network of Giant Component Node with Largest Closeness:", top_node_GC)
```



The ego network of the node with the highest closeness centrality in the Giant Component shows a strong hub structure. The central node is directly connected to a large number of other nodes, which are themselves not directly connected to each other. This star-shaped pattern shows that the central node is at a very short distance from many others, resulting in a high closeness score.

In comparison to E13, this node's centrality is structurally meaningful. It plays a key role in connecting a large portion of the network and minimising the average distance to all other nodes in the Giant Component. In conclusion, this shows that closeness centrality is best interpreted for our network within the largest connected component, where the network's true structure and connectivity are visible.

Conclusion

The analysis reveals that the Abu Sayyaf Group's network is fragmented, decentralized, and strategically structured. A few key nodes concentrate connectivity, while the majority remain peripheral and isolated. This setup, characterized by low local clustering and clearly defined communities, reflects an organization designed to operate securely and remain resilient against detection or disruption. The emerging structure is not random, but rather aligned with an operational model that prioritizes cell isolation and tactical centralization.

Overall, the network demonstrates an internal logic aimed at maintaining operational efficiency under conditions of secrecy.