

Trump World Associations

Mencía Gómez and Laura Toro

2025-04-18

Load data

First we open our dataset “**Trump World Associations**”: a network of people and organizations related to Donald J Trump, the President of the United States.

This network refers to his first period as president, specifically, in 2017. We believe it provides insight on the people, entities and organizations connected to Trump, who proved to be a very important figure both then and again now since his reelection. Therefore, we’ve chosen to study this network as a way to get to understand his environment better.

We change names so they don’t give us problems in the next exercises:

```
library(readr)
```

```
## Warning: package 'readr' was built under R version 4.4.3
```

```
df <- read_csv("trumpworld.csv")
```

```
## Rows: 3380 Columns: 6
## -- Column specification -----
## Delimiter: ","
## chr (6): Entity A Type, Entity A, Entity B Type, Entity B, Connection, Sourc...
##
## 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.
```

```
names(df) <- gsub(" ", "", names(df)) #name change
```

```
df
```

```
## # A tibble: 3,380 x 6
##   EntityAType EntityA EntityBType EntityB Connection 'Source(s)'
##   <chr>      <chr>      <chr>      <chr>      <chr>      <chr>
## 1 Organization 4 SHADOW TREE LANE M~ Organizati~ 4 SHAD~ Ownership https://ww~
## 2 Organization 40 WALL DEVELOPMENT ~ Organizati~ 40 WAL~ Ownership https://ww~
## 3 Organization 40 WALL STREET LLC Organizati~ 40 WAL~ Ownership https://ww~
## 4 Organization 40 WALL STREET MEMBE~ Organizati~ 40 WAL~ Ownership https://ww~
## 5 Organization 401 MEZZ VENTURE LLC Organizati~ 401 NO~ Ownership https://ww~
## 6 Organization 401 MEZZ VENTURE LLC Organizati~ TRUMP ~ Ownership https://ww~
## 7 Organization 401 NORTH WABASH VEN~ Organizati~ TRUMP ~ Ownership https://ww~
## 8 Organization 401 NORTH WABASH VEN~ Organizati~ TRUMP ~ Ownership https://ww~
```

```
## 9 Organization 4C INSIGHTS      Person      JEREMY~ Former ad~ https://ww~
## 10 Organization 809 NORTH CANON MEMB~ Organizati~ 809 NO~ Ownership https://ww~
## # i 3,370 more rows
```

Question 1

What is the number of nodes and links?

```
library(igraph)
```

```
## Warning: package 'igraph' was built under R version 4.4.3
```

```
##
```

```
## Adjuntando el paquete: 'igraph'
```

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

```
##
```

```
##      decompose, spectrum
```

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

```
##
```

```
##      union
```

```
edges <- data.frame(from = df$EntityA, to = df$EntityB)
```

```
g <- graph_from_data_frame(edges, directed = FALSE)
```

```
vcount(g) #nodes
```

```
## [1] 2669
```

```
ecount(g) #links
```

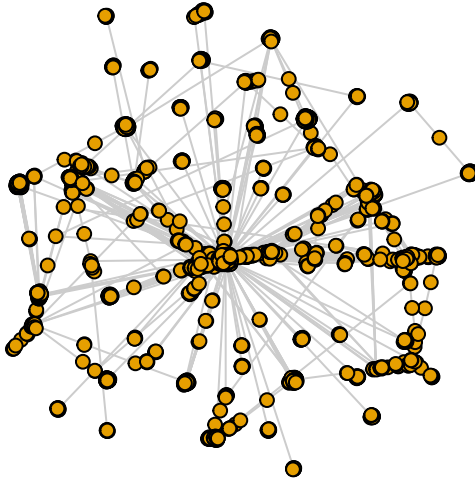
```
## [1] 3380
```

The network is composed of 2669 nodes, representing individuals or organizations associated with Donald Trump. There are 3380 links indicating various types of associations between these entities, such as ownership or shared positions.

We can see it in a graph:

```
plot(g,
      vertex.size = 6,
      vertex.label = NA,
      edge.color = "gray80",
      main = "Trump World Network")
```

Trump World Network



Question 2

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

```
# Degree of nodes
deg <- degree(g, mode = "all")

# Average degree
mean(deg)
```

```
## [1] 2.532784
```

```
# Standard deviation of the degree
sd(deg)
```

```
## [1] 15.75445
```

We see that on average, each node has about 2.53 connections (associations). The standard deviation is quite high (15.75), indicating that there is a large variability in the number of connections between nodes; that is, some nodes must have many more connections than others. This means that most people and organizations have few connections, but some nodes have a much higher number of connections.

Question 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?

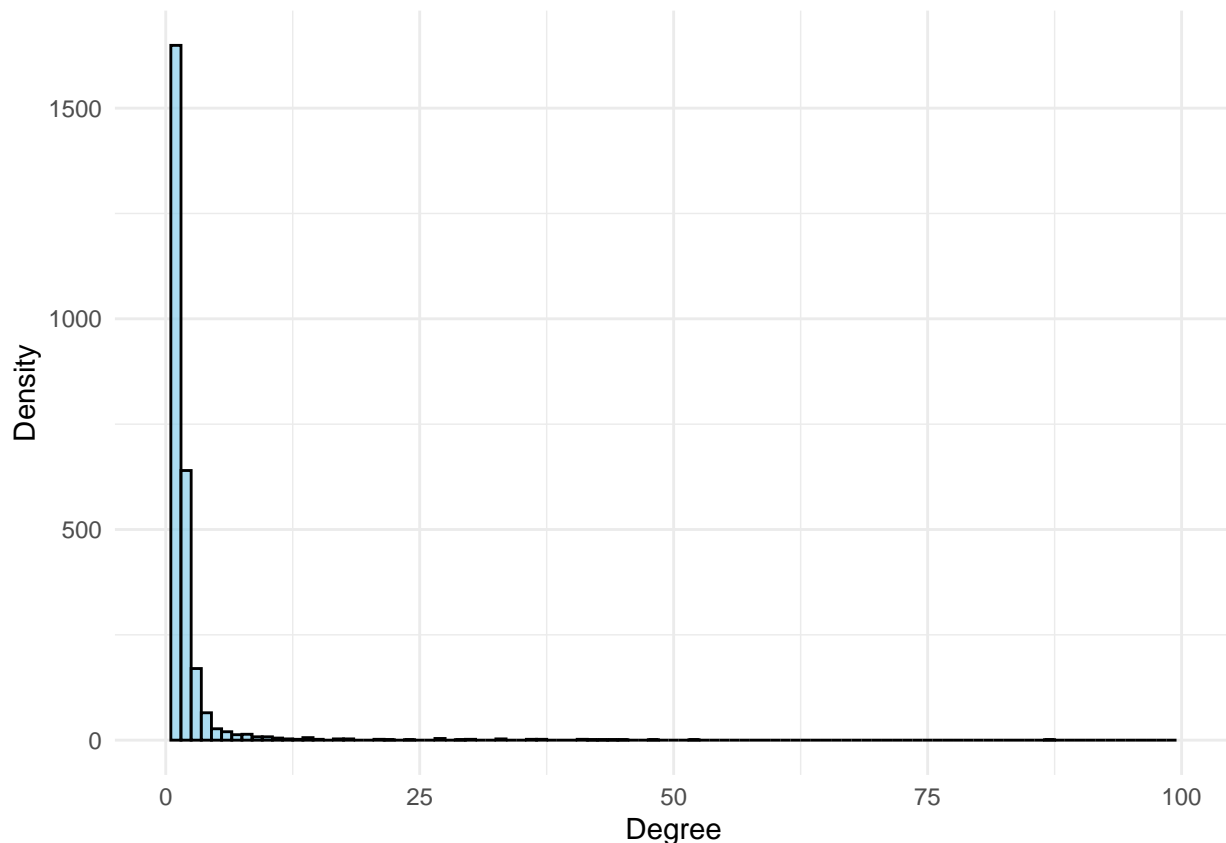
LINEAR SCALE

We limited the graph's boundary to 100 degrees (axis X) because there were nodes with more than 500 connections, but there were only very few (very very few) nodes with these degrees, making it impossible to distinguish the degree distribution of the most common nodes in the graph. This type of adjustment is common when dealing with long-tailed distributions, in which a few nodes are highly connected but do not represent the majority of connections in the network.

```
library (ggplot2)
deg_df <- data.frame(degree = deg)
ggplot(deg_df, aes(x = degree)) +
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7) +
  labs(x = "Degree", y = "Density") +
  theme_minimal() +
  xlim(0, 100)
```

```
## Warning: Removed 4 rows containing non-finite outside the scale range
## ('stat_bin()').
```

```
## Warning: Removed 2 rows containing missing values or values outside the scale range
## ('geom_bar()').
```

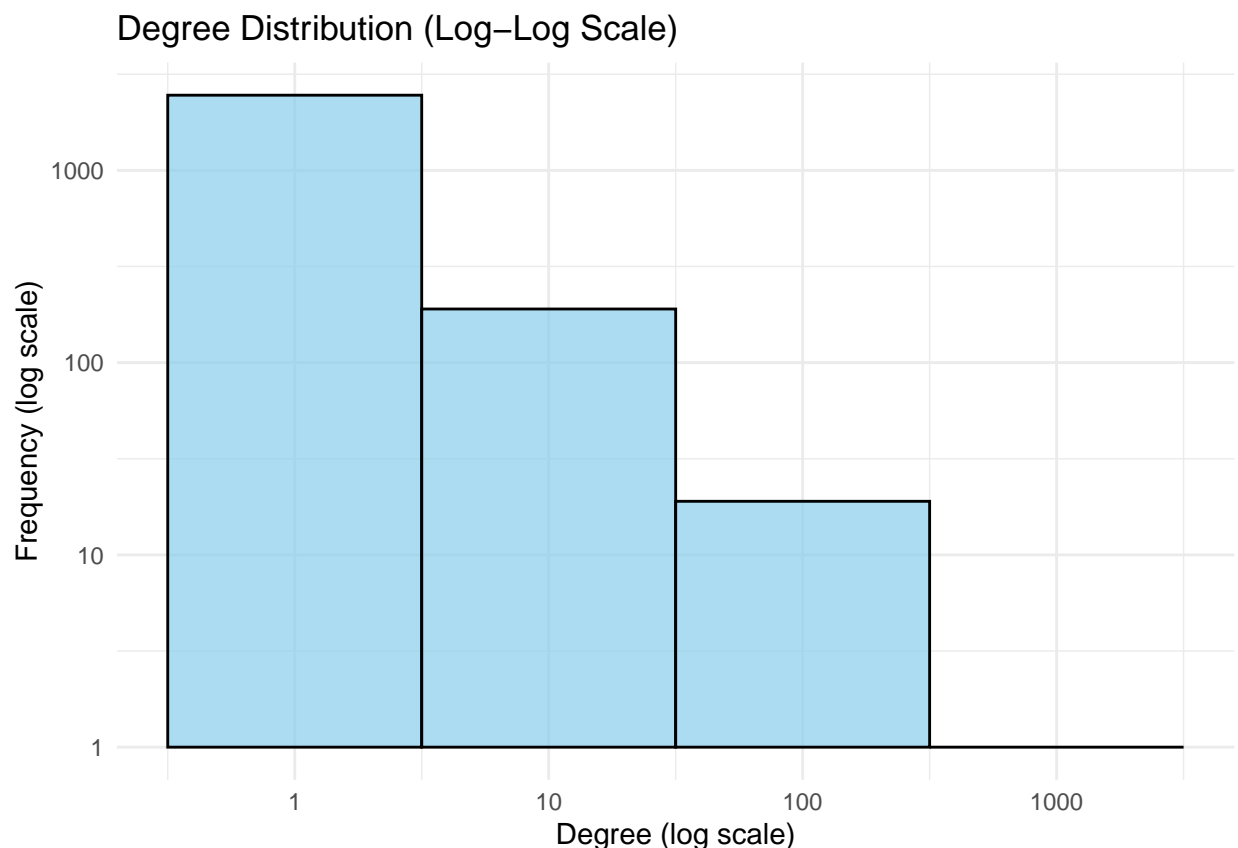


What we observe in the histogram is that in the first column we have lots of values and then drops exponentially, suggesting that in the Trump network there are many nodes (the vast majority) with a low degree (few connections). On the other hand, there are a few nodes with a much higher degree, causing the distribution to drop rapidly after the first values. It's a network with **predominantly low connected nodes and a few highly connected nodes**. It's a characteristic of dispersed or sparse networks with few highly connected nodes, therefore it is a non-homogeneous connectivity.

LOG SCALE

This type of graph is going to be very useful since in the previous one we saw the exponential decline, so we know that the distribution follows a power law, where the frequency of nodes with a higher degree decreases exponentially. Also the fact that we had to limit the X axis to 100 degrees shows that the linear scale was not good enough to represent the nodes distribution and that a logarithmic scale is better:

```
ggplot(deg_df, aes(x = degree)) +  
  geom_histogram(binwidth = 1, color = "black", fill = "skyblue", alpha = 0.7) +  
  scale_x_log10() +  
  scale_y_log10() +  
  labs(x = "Degree (log scale)", y = "Frequency (log scale)") +  
  theme_minimal() +  
  ggtitle("Degree Distribution (Log-Log Scale)")
```



As we mentioned before, and looking at this log-log graph, the linear decay indicates a power-law distribution: the network follows the typical pattern of a scale-free network. That is, a few nodes have many connections, while the majority have few.

Now we calculate the degree of the most connected node:

```
max_deg <- max(deg)
max_deg
```

```
## [1] 757
```

```
which(deg == max_deg)
```

```
## DONALD J. TRUMP
##           211
```

The maximum degree in the network is 757, which means the most connected node has 757 relationships or associations. We see that this node corresponds to President Trump (211 is the position in the degree vector).

Question 4

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

```
transitivity(g)
```

```
## [1] 0.005174213
```

The clustering coefficient of nearly 0 (0.00517) indicates that the network has low transitivity: a node's neighboring nodes are unlikely to be connected to each other. This suggests that the network doesn't form dense local relationships or triangles. In other words, the connections in the network are more sparse and don't tend to form strong clusters.

Question 5

What is the assortativity (degree) in the network?

```
assortativity_degree(g)
```

```
## [1] -0.168972
```

The network has negative assortativity (-0.168), meaning that high-degree nodes tend to connect with low-degree nodes, and vice versa. In other words, the most connected nodes do not seek to connect with other equally connected nodes. It is a disassortative network.

Question 6

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

The Louvain method for detecting communities within the network groups nodes that are more connected to each other than to the rest of the network.

```
louvain <- cluster_louvain(g)
```

Modularity:

```
modularity_value <- modularity(louvain)  
cat("Modularity:", modularity_value, "\n")
```

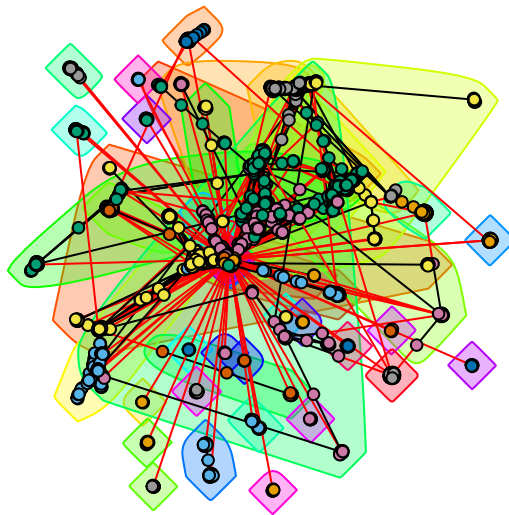
```
## Modularity: 0.8018045
```

```
cat("Number of communities:", length(louvain), "\n")
```

```
## Number of communities: 56
```

Plotting the communities:

```
plot(louvain, g, vertex.label = NA, vertex.size = 5) #For clarity, we don't see the labels, just the no
```



#We can see which members belong to which community

```
membership_df <- data.frame(Node = names(membership(louvain)),  
                             Community = membership(louvain))  
head(membership_df)
```

##		Node Community
## 4	SHADOW TREE LANE MEMBER CORP.	1
## 40	WALL DEVELOPMENT ASSOCIATES LLC	2
## 40	WALL STREET LLC	2
## 40	WALL STREET MEMBER CORP.	2
## 401	MEZZ VENTURE LLC	1
## 401	NORTH WABASH VENTURE LLC	1

The modularity value is 0.8 approximately, which is very high, indicating a **strong community structure**. We have **56 communities in the network**, where nodes are more densely connected. For example, one community could represent a business group, another a political group, another a circle of relatives, etc. We can also compute what community the nodes belong to and see it in a dataframe.

Question 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

To find out if the clustering coefficient we see in the network is real or if it could happen by chance, we calculate the clustering of the real network (already calculated), then we create a random network that has the same number of connections per node, but placed randomly, and finally compare both values.

```
library(ggraph)
```

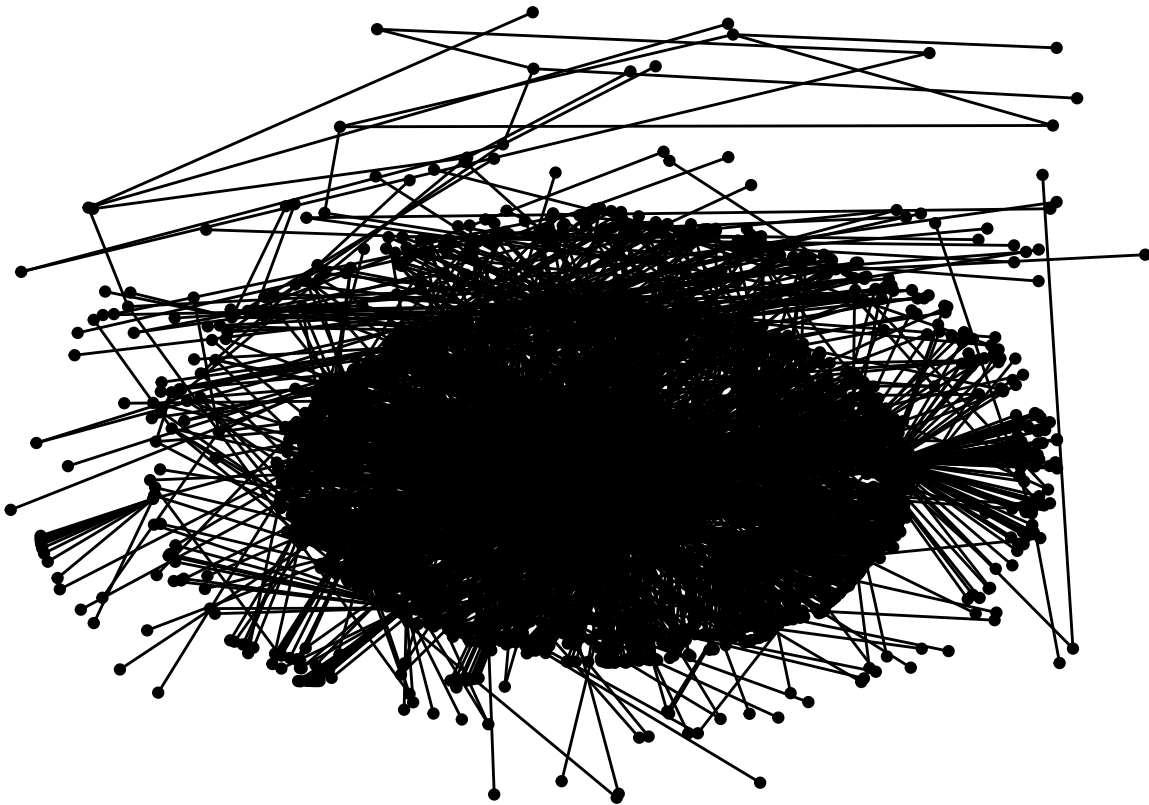
```
## Warning: package 'ggraph' was built under R version 4.4.3
```

```
set.seed(123)
```

```
transitivity(g)
```

```
## [1] 0.005174213
```

```
ggraph(g,layout="kk")+geom_edge_link()+geom_node_point()+theme_void()
```

```
transitivity(sample_degseq(degree(g)))
```

```
## [1] 0.008137921
```

The global clustering of our original network is 0.005, while for our configuration model it's 0.01, meaning **it has a higher clustering than the original**. This suggests that the clustering observed can be explained by the degree distribution alone.

Question 8

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

```
library(igraph)

#We get the node with largest closeness
closeness_centralities <- closeness(g)
node_max_closeness <- which.max(closeness_centralities)
node_max_closeness
```

```
## DONALD J. TRUMP
##                211
```

```
#We get the neighborhood
```

```
neighbors_of_node <- ego(g, order = 1, nodes = node_max_closeness)[[1]]  
neighbors_of_node
```

```
## + 758/2669 vertices, named, from b00c3ee:
```

```
## [1] DONALD J. TRUMP  
## [2] 4 SHADOW TREE LANE MEMBER CORP.  
## [3] 40 WALL DEVELOPMENT ASSOCIATES LLC  
## [4] 40 WALL STREET LLC  
## [5] 40 WALL STREET MEMBER CORP.  
## [6] 401 MEZZ VENTURE LLC  
## [7] 401 NORTH WABASH VENTURE LLC  
## [8] 809 NORTH CANON MEMBER CORPORATION  
## [9] ACE ENTERTAINMENT HOLDINGS INC  
## [10] AJIT PAI  
## + ... omitted several vertices
```

```
#Plot
```

```
subgraph <- induced_subgraph(g, neighbors_of_node)
```

```
V(subgraph)$color <- "skyblue"
```

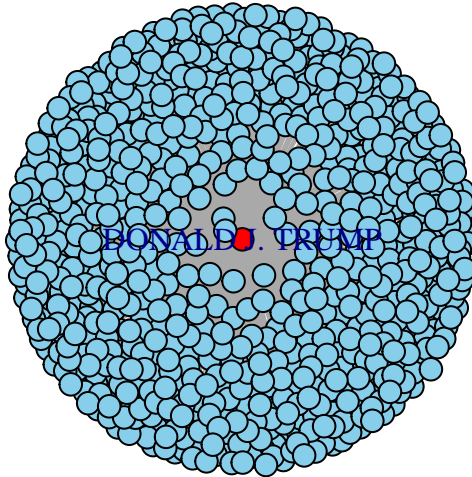
```
V(subgraph)$color[which(V(subgraph)$name == V(g)$name[node_max_closeness])] <- "red"
```

```
V(subgraph)$label <- NA
```

```
V(subgraph)$label[which(V(subgraph)$name == V(g)$name[node_max_closeness])] <- V(g)$name[node_max_closeness]
```

```
plot(subgraph, vertex.size = 10, main = "Neighborhood of node with highest closeness")
```

Neighborhood of node with highest closeness



First we get the node with the highest closeness, which is, unsurprisingly, Donald Trump. Then we get the neighbors of that node and we plot a graph showing us only its neighborhood. Because we have a very large number of variables, seeing all of them in a plot is impossible, so we keep only the label of Donald Trump (the central node) and not the rest.

We can make this plot interactive to better see how the central node is connected (warning: this can only be visualized in the Rmd):

```
library(visNetwork)
```

```
## Warning: package 'visNetwork' was built under R version 4.4.3
```

```
#In the node selection, you have to select "DONALD J. TRUMP"
```

```
visIgraph(g, physics = T) %>%  
  visOptions(highlightNearest = TRUE, nodesIdSelection = TRUE)
```

If we want to know exactly the names of the nodes closest to the node with the greatest closeness, we can modify the code to sort the neighbors according to their proximity to the central node and select the first 10.

```
library(igraph)
```

```
closeness_centralities <- closeness(g)
```

```

# Node with highest closeness
node_max_closeness <- which.max(closeness_centralities)
central_node_name <- V(g)$name[node_max_closeness]
cat("Node with highest closeness:", central_node_name, "\n")

## Node with highest closeness: DONALD J. TRUMP

# Get the neighbors of the node with the highest closeness
neighbors_ids <- neighbors(g, node_max_closeness)
neighbor_names <- V(g)$name[neighbors_ids]

# Sort neighbors by proximity centrality
neighbor_closeness <- closeness(g, vids = neighbors_ids)
neighbor_ordered <- order(neighbor_closeness, decreasing = TRUE)

# Top 10
top_10_neighbors <- neighbor_names[neighbor_ordered][1:10]
cat("Top 10 closest nodes to", central_node_name, "\n")

## Top 10 closest nodes to DONALD J. TRUMP :

print(top_10_neighbors)

```

```

## [1] "JARED KUSHNER"           "DONALD TRUMP JR."
## [3] "IVANKA TRUMP"            "WILBUR ROSS"
## [5] "STEVEN MNUCHIN"          "MAR-A-LAGO CLUB, INC"
## [7] "STEPHEN BANNON"          "THE TRUMP ORGANIZATION, INC."
## [9] "KELLYANNE CONWAY"        "REBEKAH MERCER"

```

We see that **among Trump’s closest nodes in 2017 are:** first Jared Kushner (American businessman, Trump’s advisor in 2017), Ivanka Trump (Donald Trump’s daughter and Jared Kushner’s wife), Steven Mnuchin (US Treasury Secretary), Stephen Bannon (White House chief strategist in 2017) and Jellyanne Conway (Donald Trump’s campaign manager in 2016).

Conclusion

In this analysis, based on Donald Trump’s 2017 network of associations, we can draw several key conclusions. The network is composed of 2,669 nodes (individuals and organizations), with 3,380 links. Most nodes have few links, while a few have many more, suggesting a **logarithmic-like distribution**, where a few nodes are much more connected than the majority.

The connections in the network don’t tend to form dense clusters or triangles, meaning that the network **doesn’t have a strong tendency to form closed relationships** (low transitivity). Furthermore, the **most connected nodes tend to connect with less connected nodes**, reflecting a “disassortative” network.

We also identified 56 communities within the network, with a modularity of 0.8, indicating a **fairly strong community structure**. (Although the network does not have many triangles or dense connections between direct neighbors, it does have groups of nodes that are more connected to each other than to the rest of the network)

Finally, analyzing the node with the highest closeness centrality, which is Donald Trump, we look at his **closest neighbors, such as Jared Kushner, Ivanka Trump, and other key advisors in his inner circle during his presidency.**