

Social Network Statistics

Jordan Robinson

2025-11-25

```
# --- Load required libraries ---
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(tidyverse)

## Warning: package 'dplyr' was built under R version 4.3.3

## Warning: package 'stringr' was built under R version 4.3.3

## -- Attaching core tidyverse packages ----- tidyverse 2.0.0 --
## v dplyr     1.1.4      v readr     2.1.4
## vforcats   1.0.0      v stringr   1.5.1
## v ggplot2   3.4.4      v tibble    3.2.1
## v lubridate 1.9.3      v tidyverse 1.3.0
## v purrr     1.0.2

## -- Conflicts ----- tidyverse_conflicts() --
## x lubridate::%--%()    masks igraph::%--%()
## x dplyr::as_data_frame() masks tibble::as_data_frame(), igraph::as_data_frame()
## x purrr::compose()     masks igraph::compose()
## x tidyverse::crossing() masks igraph::crossing()
## x dplyr::filter()      masks stats::filter()
## x dplyr::lag()         masks stats::lag()
## x purrr::simplify()    masks igraph::simplify()
## i Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts to become errors
```

```

# --- Load data ---
edges <- read_csv("gunViolenceEdges.csv") %>%
  filter(!is.na(to)) %>%
  select(-`...1`)

## New names:
## Rows: 2276 Columns: 3
## -- Column specification
## ----- Delimiter: ","
## (2): from, to dbl (1): ...1
## 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.
## * ` ` -> `...1`

nodeData <- read_csv("gunViolenceCleanData.csv")

## New names:
## Rows: 2442 Columns: 10
## -- Column specification
## ----- Delimiter: ","
## (7): MasterID, ArrestID, WEAPON_CATEGORY, CrimeType, CrimeCategory, Arre... dbl
## (3): ...1, INCI_ID, WARD
## 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.
## * ` ` -> `...1`

nodes <- nodeData %>%
  group_by(MasterID) %>%
  summarize(
    numIncidents = length(unique(INCI_ID)),
    arrested = any(!is.na(ArrestID)),
    ward = unique(WARD)[1],
    arrestDate = ifelse(any(!is.na(ArrestID)), ArrestDate[1], NA)
  )

# --- Build directed graph ---
g <- graph_from_data_frame(d = edges,
                           vertices = nodes,
                           directed = TRUE)

# --- Basic network metrics ---
num_nodes <- vcount(g)
num_edges <- ecount(g)
avg_degree <- mean(degree(g))
density_val <- edge_density(g)

# --- Convert to undirected for some measures ---
g_undirected <- as.undirected(g, mode = "collapse")

## Warning: `as.undirected()` was deprecated in igraph 2.1.0.
## i Please use `as_undirected()` instead.
## This warning is displayed once every 8 hours.

```

```

## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

# --- Triangles and clustering ---
num_triangles <- sum(count_triangles(g_undirected)) / 3
avg_clustering <- transitivity(g_undirected, type = "average")

# --- Connected components ---
components_info <- components(g_undirected)
num_components <- components_info$no

# --- Diameter of largest component ---
largest_comp <- induced_subgraph(g_undirected,
                                   which(components_info$membership == which.max(components_info$csize)))
diameter_val <- diameter(largest_comp)
avg_path_length <- mean_distance(largest_comp, directed = FALSE)

# --- Combine global metrics ---
network_metrics <- tibble(
  Metric = c("Number of Nodes",
             "Number of Edges",
             "Average Degree",
             "Density",
             "Number of Triangles",
             "Average Clustering Coefficient",
             "Number of Connected Components",
             "Diameter (Largest Component)"),
  Value = c(num_nodes,
            num_edges,
            avg_degree,
            density_val,
            num_triangles,
            avg_clustering,
            num_components,
            diameter_val)
)
)

# --- Centrality measures for each node ---
node_centrality <- tibble(
  accountID = V(g)$accountID,
  username = V(g)$username,
  degree_centrality = degree(g, mode = "all", normalized = TRUE),
  betweenness_centrality = betweenness(g, normalized = TRUE),
  closeness_centrality = closeness(g, normalized = TRUE),
  eigenvector_centrality = eigen_centrality(g, directed = TRUE, scale = TRUE)$vector
)

## Warning: The 'scale' argument of 'eigen_centrality()' is deprecated as of igraph 2.1.1.
## i eigen_centrality() will always behave as if scale=TRUE were used.
## This warning is displayed once every 8 hours.
## Call `lifecycle::last_lifecycle_warnings()` to see where this warning was
## generated.

```

```
plot(  
  g,  
  vertex.size = log(degree(g) + 1) * 5,  
  vertex.color = ifelse(V(g)$arrested, "red", "skyblue"),  
  vertex.label = NA,  
  main = "Gun Violence Involvement Network"  
)
```

Gun Violence Involvement Network

