Mathematical foundations of networks: Using graphs to represent social relations

Mark S. Handcock

handcock@ucla.edu

Statistical Analysis of Networks
Based on Notes by Peter D. Hoff
October 3, 2019

Fully observed binary network

The vast majority of network analysis tools have been developed for fully observed binary networks.

Binary network: Relational data consisting of a single dichotomous relation, typically taken indicate the presence or absence of a relationship.

Fully observed network: The relationship between each pair of individuals is observed to be either present or absent.

In such cases, the network data can be represented

- as a binary sociomatrix, or
- as a graph.

Nodes and edges

Formally, a graph consists of

- a set of nodes $\mathcal{N} = \{1, \ldots, n\}$;
- a set of edges or lines between nodes $\mathcal{E} = \{e_1, \ldots, e_m\}$

The graph is denoted $\mathcal{G} = (\mathcal{N}, \mathcal{E})$.

Each edge $e \in \mathcal{E}$ is expressed in terms of the pair of nodes the line connects.

Undirected graph:
The edges have no direction, and the edge $\{i, j\}$ is the same as the edge $\{j, i\}$:

$$\{i, j\} = \{j, i\}$$

i.e. each edge is an **unordered pair** of nodes.

Directed graph:
The edges have direction, and the edge (i, j) is not the same as the edge (j, i):

$$(i, j) \neq (j, i)$$

i.e. each edge is an **ordered pair** of nodes.

Example of an undirected graph

$$\mathcal{N} = \{1, 2, 3, 4, 5\}$$
$$\mathcal{E} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{3, 5\}, \{4, 5\}\}$$

Exercise: Draw this graph

Example of a directed graph

$$\mathcal{N} = \{1, 2, 3, 4, 5\}$$
$$\mathcal{E} = \{(1, 3), (2, 3), (2, 4), (2, 5), (3, 1), (3, 5), (4, 5), (5, 4)\}$$

Exercise: Draw this graph

Some graph terminology

For an undirected graph $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$

- **adjacent** : nodes i and j are adjacent if $\{i, j\} \in \mathcal{E}$
- **incident** : node i is incident with edge e if $e = \{i, j\}$ for some $j \in \mathcal{N}$.
- **empty** : the graph is empty if $\mathcal{E} = \emptyset$, i.e. there are no edges.
- **complete** : the graph is complete if

  $$\mathcal{E} = \{\{i, j\} : i \in \mathcal{N}, j \in \mathcal{N}, i \neq j\},$$

  that is, all possible edges are present.

Similar definitions are used for directed graphs.

Exercise: Identify some adjacent nodes and incident node-edge pairs for the previous two example graphs.

Subgraphs

A graph $\mathcal{G}_s = (\mathcal{N}_s, \mathcal{E}_s)$ is a **subgraph** of $\mathcal{G} = (\mathcal{N}, \mathcal{E})$ if

- $\mathcal{N}_s \subset \mathcal{N}$
- $\mathcal{E}_s \subset \mathcal{E}$ *and* all edges in $\mathcal{E}_s$ are between nodes in $\mathcal{N}_s$.

Examples:

$$\mathcal{N} = \{1, 2, 3, 4, 5\}$$
$$\mathcal{E} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}, \{2, 5\}, \{3, 5\}, \{4, 5\}\}$$

$$\mathcal{N}_{s_1} = \{1, 2, 3, 4\}$$
$$\mathcal{E}_{s_1} = \{\{1, 2\}, \{1, 3\}, \{2, 3\}, \{2, 4\}\} \quad \text{(generated by nodes 1,2,3 and 4)}$$

$$\mathcal{N}_{s_1} = \{1, 2, 3, 4\}$$
$$\mathcal{E}_{s_1} = \{\{1, 2\}, \{1, 3\}, \{2, 4\}\} \quad (\text{generated by edges } \{1, 2\}, \{1, 3\}, \{2, 4\})$$

Node-generated subgraph

Let $\mathcal{N}_s \subset \mathcal{N}$.

The **subgraph generated by** $\mathcal{N}_s$ is the subgraph $\mathcal{G}_s = (\mathcal{N}_s, \mathcal{E}_s)$ where $\mathcal{E}_s$ includes all edges in $\mathcal{E}$ between nodes in $\mathcal{N}_s$.

Mathematically,

$$\mathcal{E}_s = \mathcal{E} \cap \{\{i, j\} : i \in \mathcal{N}_s, j \in \mathcal{N}_s\}.$$

Node generated subgraphs are useful:

- often such a subgraph is of scientific interest;
- often there is missing data for some nodes, and so we might focus on the subgraph generated by nodes with no missing data.
- often we want to identify **cohesive subgroups** of nodes, that is, subsets of nodes with a dense node-generated subgraphs.

Dyads and triads

Some simple but useful subgraphs are dyads and triads:

Dyad: A dyad is a subgraph generated by a single pair $i, j \in \mathcal{N}$:

- In an undirected graph, the possible states of the dyad are given by either $\mathcal{E}_s = \{i, j\}$ or $\mathcal{E}_s = \emptyset$, the empty or complete graphs.
- In a directed graph, the four possible states of the dyad are given by

$$\begin{array}{cc} \mathcal{E}_s = \emptyset & \mathcal{E}_s = \{(i, j)\} \\ \mathcal{E}_s = \{(j, i)\} & \mathcal{E}_s = \{(i, j), (j, i)\} \end{array}$$

Dyads and triads

A triad is a subgraph generated by a triple of nodes.
For an undirected graph, a triad can be in one of $2^3 = 8$ possible states.

Exercise: Draw the eight possible states.

Note that

- 3 of the 1-edge triad states are equivalent, or **isomorphic**,
- 3 of the 2-edge triad states are isomorphic.

Isomorphic: Two graphs $\mathcal{G}$ and $\mathcal{G}'$ are isomorphic if

- "there is a 1-1 mapping from nodes of $\mathcal{G}$ to the nodes of $\mathcal{G}'$ that preserves the adjacency of nodes."
- or equivalently, $\mathcal{G}'$ can be obtained by relabeling the nodes of $\mathcal{G}$.

Edge-generated subgraph

Let $\mathcal{E}_s \subset \mathcal{E}$.

The **subgraph generated by** $\mathcal{E}_s$ is the subgraph $\mathcal{G}_s = (\mathcal{N}_s, \mathcal{E}_s)$ where $\mathcal{N}_s$ includes all nodes in $\mathcal{N}$ incident with an edge from $\mathcal{E}_s$.

These subgraphs may arise in certain types of network sampling schemes, for example, network event data:

- international conflicts: conflicts are recorded along with the aggressor and target countries.
- transactional data: transactional events are recorded, along with the participating parties.

Edge-generated subgraphs may be misrepresentative of the underlying graph:

$$\{i, j\} \subset \mathcal{N}_s, (i, j) \in \mathcal{E} \not\Rightarrow (i, j) \in \mathcal{E}_s$$

These subgraphs are used less frequently than node-generated subgraphs.

Edge lists

Graphs are often stored on a computer in terms of their edge set, or **edge list**.

The edge list completely represents the graph unless there are **isolated nodes**:

Isolated node or **isolate**: A node that is not adjacent to any other node.

Examples:

In the presence of isolates, the graph can be represented with the edge list and a list of isolates.

Graph-theoretic terms

- **graph**, **node set**, **edge set**, **edge list**
- **undirected graph**, **directed graph**
- **adjacent**, **incident**, **empty**, **complete**
- **subgraph**, **generated subgraph**, **dyad**, **triad**
- **isomorphic**
- **isolate**

Contrast to matrix representations

Recall the matrix representation of a relational dataset: Let

- $\mathcal{N} = \{1, \ldots, n\}$
- $y_{i,j}$ = the (possibly directed) relationship from node i to node j
- $\mathbf{Y}$ be the $n \times n$ matrix with entries $\{y_{i,j} : i = 1, \ldots, n, j \in 1, \ldots, n\}$.

The diagonal entries of $\mathbf{Y}$ are not defined, or "not available."

$$\mathbf{Y} = \begin{pmatrix} na & y_{1,2} & y_{1,3} & y_{1,4} & y_{1,5} & y_{1,6} \\ y_{2,1} & na & y_{2,3} & y_{2,4} & y_{2,5} & y_{2,6} \\ y_{3,1} & y_{3,2} & na & y_{3,4} & y_{3,5} & y_{3,6} \\ y_{4,1} & y_{4,2} & y_{4,3} & na & y_{4,5} & y_{4,6} \\ y_{5,1} & y_{5,2} & y_{5,3} & y_{5,4} & na & y_{5,6} \\ y_{6,1} & y_{6,2} & y_{6,3} & y_{6,4} & y_{6,5} & na \end{pmatrix}$$

Adjacency matrices

Suppose we have dichotomous (presence/absence) relationship measured between pairs of nodes in a node set $\mathcal{N} = \{1, \ldots, n\}$.

As discussed, such relational data can be expressed as a graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$.

The data can also be represented by an $n \times n$ matrix $\mathbf{Y} = \{y_{i,j} : i, j \in \mathcal{N}, i \neq j\}$, where

$$y_{i,j} = \begin{cases} 1 & \text{if } (i,j) \in \mathcal{E} \\ 0 & \text{if } (i,j) \notin \mathcal{E} \end{cases}$$

This matrix is called the **adjacency matrix** of the graph $\mathcal{G} = (\mathcal{N}, \mathcal{E})$.

- The adjacency matrix of every graph is a square, binary matrix with an undefined diagonal.
- Every square, binary matrix with an undefined diagonal corresponds to a graph.

Graphs and matrices

For an *undirected* binary relation, $\{i, j\} = \{j, i\}$ and so $y_{i,j} = y_{j,i}$ by design.

- the representing graph is an undirected graph;
- the representing adjacency matrix is symmetric.

For a *directed* binary relation, $(i, j) \neq (j, i)$ and it is possible that $y_{i,j} \neq y_{j,i}$.

- the representing graph is a directed graph;
- the representing adjacency matrix is possibly asymmetric.

Exercise: Draw the directed graph represented by the following matrix:

$$\mathbf{Y} = \begin{pmatrix} na & 0 & 1 & 1 & 0 & 1 \\ 1 & na & 1 & 0 & 0 & 1 \\ 0 & 0 & na & 1 & 0 & 1 \\ 0 & 0 & 1 & na & 0 & 1 \\ 1 & 0 & 1 & 1 & na & 1 \\ 0 & 0 & 1 & 1 & 0 & na \end{pmatrix}$$

Advantages of sociomatrices

Recall, any relational variable measured on a nodeset can be represented by a **sociomatrix**:

Sociomatrix: An square matrix with undefined diagonal entries.

Clearly a sociomatrix can represent a wider variety of relational data than a graph or adjacency matrix.

$$\mathbf{Y}_1 = \begin{pmatrix} na & 1 & 0 & 1 & 0 \\ 0 & na & 0 & 1 & 0 \\ 0 & 1 & na & 0 & 0 \\ 0 & 0 & 0 & na & 0 \\ 0 & 0 & 0 & 1 & na \end{pmatrix} \quad \mathbf{Y}_2 = \begin{pmatrix} na & 2.1 & na & 0.0 & 0.1 \\ 0.0 & na & 4.1 & 0.0 & na \\ 2.1 & 2.9 & na & 0.0 & 1.2 \\ 0.0 & 0.0 & na & na & 5.4 \\ na & 2.1 & 4.1 & 0.0 & na \end{pmatrix}$$

The sociomatrix on the left can alternatively be expressed as a graph.

The sociomatrix on the right cannot:

- the value of the relation is not dichotomous;
- the value of the relation is not measured for all pairs.

no measured relation $\not\Rightarrow$ no relation

Sociomatrices

Advantages of sociomatrices:

- can represent valued (non-dichotomous) relations;
- can indicate missing data.

Graph representations can do neither of these things, yet people will nevertheless try to shoehorn incomplete, non-dichotomous relational data into a graphical representation:

$$\mathbf{Y} = \begin{pmatrix} na & 2.1 & na & 0.0 & 0.1 \\ 0.0 & na & 4.1 & 0.0 & na \\ 2.1 & 2.9 & na & 0.0 & 1.2 \\ 0.0 & 0.0 & na & na & 5.4 \\ na & 2.1 & 4.1 & 0.0 & na \end{pmatrix} \Rightarrow \tilde{\mathbf{Y}} = \begin{pmatrix} na & 1 & 0 & 0 & 1 \\ 0 & na & 1 & 0 & 0 \\ 1 & 1 & na & 0 & 1 \\ 0 & 0 & 0 & na & 1 \\ 0 & 1 & 1 & 0 & na \end{pmatrix}$$

The sociomatrix on the right is representable as a graph, but

- coarsens the data (throws away information)
- misrepresents uncertainty in the missing values.

Compression

Disadvantage of sociomatrices:

- are an inefficient representation for sparse networks.

Consider an $n \times n$ binary sociomatrix $\mathbf{Y}$ that is $p\%$ 1's, where p is close to zero.

- the size of the matrix grows quadratically in n
- the number of "1"s in the matrix grows linearly with n.

For such matrices, an edge list provides a much more compact representation:

$$\mathbf{Y} = \begin{pmatrix} na & 1 & 0 & 0 & 1 \\ 0 & na & 1 & 0 & 0 \\ 1 & 0 & na & 0 & 1 \\ 0 & 0 & 0 & na & 1 \\ 0 & 0 & 1 & 0 & na \end{pmatrix}$$

$$\mathcal{E} = \{(1,2),(1,5),(2,3),(3,1),(3,5),(4,5),(5,3)\}$$

The advantage of $\mathcal{E}$ over $\mathbf{Y}$ increases as n increases, if p remains fixed.

Weighted edges

Often the relational variable is either zero or some arbitrary non-zero value.

- communication networks:

$$y_{i,j} = \text{number of emails sent from } i \text{ to } j$$
$$y_{i,j} \in \{0, 1, 2, \ldots\}$$

- conflict networks:

$$y_{i,j} = \text{ military relationship between } i \text{ and } j$$
$$y_{i,j} \in \{-1, 0, 1\}$$

In both cases, $y_{i,j} = 0$ for the vast majority of i, j-pairs.
In such cases, a **weighted edge list** can be more efficient than a sociomatrix.

Weighted edges

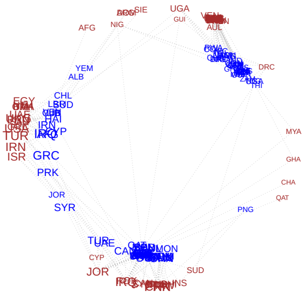
$$\mathbf{Y} = \begin{pmatrix} na & 8 & 0 & 0 & 2 \\ 0 & na & 1 & 0 & 0 \\ 7 & 0 & na & 0 & 4 \\ 0 & 0 & 0 & na & 1 \\ 0 & 0 & 13 & 0 & na \end{pmatrix} \quad \mathcal{E} = \begin{pmatrix} 1 & 2 & 8 \\ 1 & 5 & 2 \\ 2 & 3 & 1 \\ 3 & 1 & 7 \\ 3 & 5 & 4 \\ 4 & 5 & 1 \\ 5 & 3 & 13 \end{pmatrix}$$

Compression:

- $\mathbf{Y}$ is $n \times n$
- $\mathcal{E}$ is $m \times 3$, where m is the number of non-zero relationships.

Variables:

- country population
- country polity
- number of militarized disputes between country pairs
- amount of trade between country pairs
- geographic distance between country pairs
- number of shared IGOs between country pairs



In what ways can we represent these data?

Example: International conflict, 1990-2000

Nodal variables: population, gdp and polity.

```
conflict90s$nodevars[1:10,]

##        pop    gdp polity
## AFG 24.78  19.29  -4.64
## ALB  3.28   8.95   3.82
## ALG 27.89 133.61  -3.91
## ANG 10.66  15.38  -2.55
## ARG 34.77 352.38   7.18
## AUL 18.10 408.06  10.00
## AUS  7.99 170.76  10.00
## BAH  0.57   7.45  -9.27
## BEL 10.12 215.01  10.00
## BEN  5.49   6.03   5.45
```

Nodal variables can be stored as an $n \times p$ matrix $\mathbf{X}$:

- n is the number of nodes;
- p is the number of nodal variables.

Example: International conflict, 1990-2000

Dyadic variables: conflict, imports, shared IGOs, distance.

Conflict:

	AFG	ALB	ALG	ANG	ARG	AUL	AUS
AFG	na	0	0	0	0	0	0
ALB	0	na	0	0	0	0	0
ALG	0	0	na	0	0	0	0
ANG	0	0	0	na	0	0	0
ARG	0	0	0	0	na	0	0
AUL	0	0	0	0	0	na	0
AUS	0	0	0	0	0	0	na

	CHN	DRC	IRN	IRQ	ISR	JOR	PRK
CHN	na	0	0	0	0	0	0
IRN	0	0	0	6	0	0	0
IRQ	0	0	0	0	1	0	0
JOR	0	0	0	1	0	0	0
PRK	3	0	0	0	0	0	0
TUR	0	0	2	6	0	0	0
USA	0	0	1	7	0	0	2

These data may be stored as an **asymmetric sociomatrix** or more compactly as a **weighted, directed edgelist**.

Example: International conflict, 1990-2000

Dyadic variables: conflict, imports, shared IGOs, distance.

Imports:

	AFG	ALB	ALG	ANG	ARG	AUL	AUS
AFG	na	0	0	0	0	0	0
ALB	0	na	0	0	0	0	0.01
ALG	0	0	na	0.01	0.06	0.03	0.13
ANG	0	0	0	na	0.02	0	0
ARG	0	0	0.01	0.01	na	0.09	0.07
AUL	0	0	0	0	0.06	na	0.23
AUS	0	0	0.22	0	0.02	0.03	na
⋮							

These data may be stored as an **asymmetric sociomatrix** or more compactly as a **weighted, directed edgelist**.

Example: International conflict, 1990-2000

```
imports<-(conflict90s$dyadvars)[,,2]
imports[1:7,1:7]

##     AFG ALB  ALG  ANG  ARG  AUL  AUS
## AFG   0   0 0.00 0.00 0.00 0.00 0.00
## ALB   0   0 0.00 0.00 0.00 0.00 0.01
## ALG   0   0 0.00 0.01 0.06 0.03 0.13
## ANG   0   0 0.00 0.00 0.02 0.00 0.00
## ARG   0   0 0.01 0.01 0.00 0.09 0.07
## AUL   0   0 0.00 0.00 0.06 0.00 0.23
## AUS   0   0 0.22 0.00 0.02 0.03 0.00

sm2el(imports[1:7,1:7])

##     row col    w
## ALB   2   7 0.01
## ALG   3   4 0.01
## ALG   3   5 0.06
## ALG   3   6 0.03
## ALG   3   7 0.13
## ANG   4   5 0.02
## ARG   5   3 0.01
## ARG   5   4 0.01
## ARG   5   6 0.09
## ARG   5   7 0.07
## AUL   6   5 0.06
## AUL   6   7 0.23
## AUS   7   3 0.22
## AUS   7   5 0.02
## AUS   7   6 0.03
```

Example: International conflict, 1990-2000

Dyadic variables: conflict, imports, shared IGOs, distance.

Distance:

	AFG	ALB	ALG	ANG	ARG	AUL	AUS
AFG	na	4.33	5.86	7.59	15.27	11.35	4.56
ALB	4.33	na	1.54	5.61	11.61	15.6	0.81
ALG	5.86	1.54	na	5.18	10.17	16.97	1.68
ANG	7.59	5.61	5.18	na	7.78	13.26	6.35
ARG	15.27	11.61	10.17	7.78	na	11.72	11.82
AUL	11.35	15.6	16.97	13.26	11.72	na	15.91
AUS	4.56	0.81	1.68	6.35	11.82	15.91	na
⋮							

These data may be stored as a **symmetric sociomatrix** or as a **weighted, undirected edgelist**.