Conceptualisations of network position

Outline

Conceptualising similarity of social positions

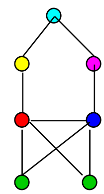
- Structural equivalence
- Blocks and block models
- Automorphic equivalence
- Regular equivalence
- Outdegree and indegree equivalence
- Block types
- Generalised blockmodelling
- *A priore* blockmodels
- Density matrices

Introduction to example: Doreian's (1988) political network

Structural equivalence

The two green nodes are connected to exactly the same alters and are said to be ***structurally equivalent***

They hold identical ***positions*** in the network



Formally, nodes a and b are ***structurally equivalent*** if, whenever (a,x) is an edge in G then so is (b,x), and conversely $(x \neq a, b)$

We can allocate nodes that are structurally equivalent to a structural equivalence ***class***, or ***block*** – represented by colour in the diagram above.

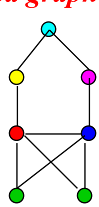
Blockmodel

For a structural equivalence relation:

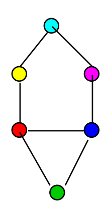
If a node in block x is connected to a node in block y, then ***every*** node in block x is connected to ***every*** node in block y

In this case we define an edge (x,y) between block x and block y, and the result is a ***reduced graph*** or ***blockmodel*** for the network

e.g.

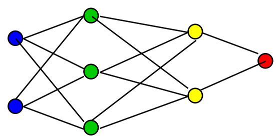


graph

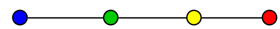


blockmodel

Another example



graph



blockmodel

00111000
00111000
11000110
11000110
11000110
00111001
00111001
00000110

adjacency matrix for graph

0100
1010
0101
0010

adjacency matrix for blockmodel

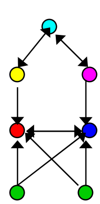
Structural equivalence for directed graphs

In a ***directed graph***:

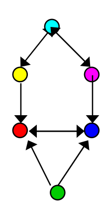
nodes *a* and *b* are ***structurally equivalent*** if,

(a) whenever (*a*,*x*) is an ***arc*** in *G* then so is (*b*,*x*), and conversely, ***and***

(b) whenever (*x*,*a*) is an ***arc*** in *G* then so is (*x*,*b*), and conversely



directed graph



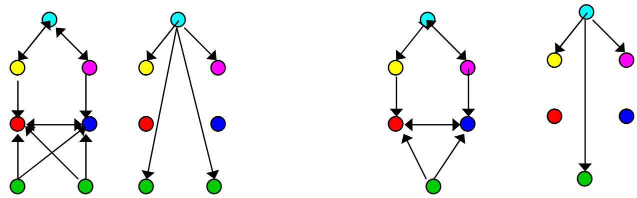
blockmodel

Structural equivalence for multiple graphs

The extension to ***multiple graphs*** is straightforward:

Nodes are ***structurally equivalent*** in a multiple graph or network if they are structurally equivalent in ***each*** of the constituent graphs

e.g.,



graph A graph B blockmodel

Some possible blockmodels

Factions

1 0 0
0 1 0
0 0 1

Hierarchy

1 0 0
1 1 0
1 1 1

Group-brokerage

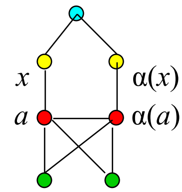
1 1 1
1 1 0
1 0 1

Core-periphery

1 1
1 0

Automorphic equivalence

A ***permutation*** α of the node set V of a graph is a re-labelling of the nodes: we write α(a) for the node to which the label a is attached by the relabelling



A permutation α is an ***automorphism*** of the graph G if, whenever (a,x) is an edge in G, then (α(a),α(x)) is an edge in G, and conversely.

Nodes a and b are ***automorphically equivalent*** if b = α(a) for some automorphism α

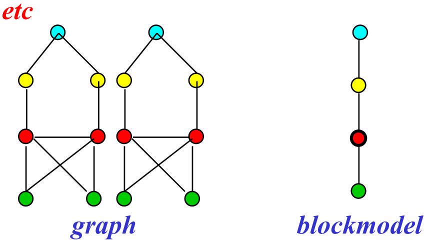
Example: nodes with the same colour in the graph above are automorphically equivalent

Why is automorphic equivalence interesting?

Automorphically equivalent nodes have the same position in a network in a more abstract sense than structurally equivalent nodes: they are not connected to the exact same nodes, but to nodes that play analogous roles in the network

a potential representation for roles as we currently understand them – leader, principal, broker, loner, clown, etc

Note: if nodes are structually equivalent, then they are also automorphically equivalent, but the converse does not hold



Generalisation to directed and multiple graphs is straightforward, as for structural equivalence

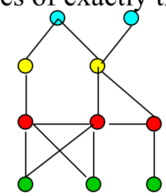
Note: thick edge on a node indicates a ***loop*** (***self-tie***)

Regular equivalence

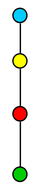
Two nodes *a* and *b* are ***regularly equivalent*** if

(a) whenever (*a*,*x*) is an edge in *G*, then there is some node *y* that is regularly equivalent to *x* for which (*b*,*y*) is an edge in *G*.

In other words, if regularly equivalent nodes have the same colour, then each node in a class of regularly equivalent nodes is connected to other nodes of exactly the same set of colours (e.g red to yellow, green)



graph



(regular equivalence) blockmodel

Regular equivalence for directed and multiple graphs

Two nodes *a* and *b* in a **directed** graph *G* are ***regularly equivalent*** if

(a) whenever (*a*,*x*) is an **arc** in *G*, then there is some node *y* that is regularly equivalent to *x* for which (*b*,*y*) is an **arc** in *G*; **and**

whenever (*x*,*a*) is an **arc** in *G*, then there is some node *y* that is regularly equivalent to *x* for which (*y*,*b*) is an **arc** in *G*.

For **multiple** graphs and directed graphs, two nodes are ***regularly equivalent*** if they are regularly equivalent in each of the constituent graphs.

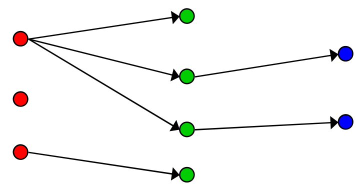
Note: if two nodes are automorphically equivalent, they are also regularly equivalent, so:

Structural equivalence
  ⇒ Automorphic equivalence
    ⇒ Regular equivalence

increasingly general form of positional equivalence

© Department of Psychology, University of Melbourne

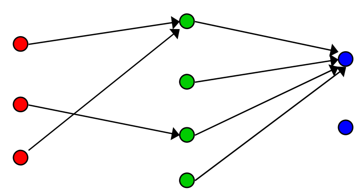
Indegree equivalence in directed graphs



Two nodes *a* and *b* in a **directed** graph *G* are ***indegree equivalent*** if, whenever (x,a) is an arc in *G*, then there is some node *y* that is **indegree equivalent** to *x* for which (y,b) is an arc in *G*.

Each node of a given colour has incoming ties from nodes of the same colour(s): every ***green*** (***blue***) node receives a tie from some ***red*** (***green***) node

Outdegree equivalence in directed graphs



Two nodes *a* and *b* in a **directed** graph *G* are ***outdegree equivalent*** if, whenever (a,x) is an arc in *G*, then there is some node *y* that is **outdegree equivalent** to *x* for which (b,y) is an arc in *G*.

Regular equivalence $\Rightarrow$ { Outdegree equivalence ***and*** Indegree equivalence

Regular, indegree and outdegree equivalence: adjacency matrices

Regular equivalence	*Indegree equivalence*	*Outdegree equivalence*
000110000	000001100	000100000
000101000	000100000	000000100
000000100	000011000	000100100
110000010	001000010	010000001
100000001	000000010	101000001
010000011	110000001	100000011
001000010	111000000	011000001
000101100	000010100	000100100
000011000	000101100	000000100

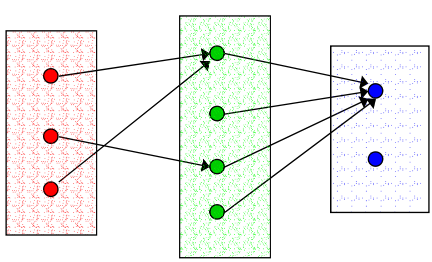
For ***nonzero*** submatrices, there is:

a 1 in every row and column	***a 1 in every column***	***a 1 in every row***
regular	***column-regular***	***row-regular***

Why are these forms of equivalence of interest?

All of these forms of equivalence and several others (eg Pattison, 1993) have the property that:

there is a path at the block level (assuming a block-to-block tie whenever there is at least one node-level tie) if and only if there is at least one path at the node level



Submatrix patterns in generalised blockmodelling (Doreian et al, 2005)

Different forms of equivalence are associated with different non-zero submatrix patterns and hence different ways of deriving inter-block relations from the relations among constituent nodes

Submatrix types (Doreian, Batagelj, and Ferligoj, 2005):

Type	*Pattern*
null	**all 0**
complete	**all 1**
regular	**1 in every row and column**
row-regular (outdegree)	**1 in every row**
column-regular (indegree)	**1 in every column**
row-dominant	**a row has all 1's**
column-dominant	**a column has all 1's**

Blockmodels and generalised bloclmodels

A ***blockmodel*** represents the *relations* among social positions, and comprises:

- **an assignment of nodes to blocks, or positions (classes of equivalent nodes), and**
- **a specification of relations between blocks**

Different forms of equivalence are associated with different requirements for submatrix patterns of inter-block relations

The most common practical applications to date have involved structural equivalence (vast majority) and regular equivalence ("niche" applications) but software availability has played an important part.

In a ***generalised blockmodel*** (Doreian et al., 2004, 2005), each submatrix may correspond to a different form of equivalence

© Department of Psychology, University of Melbourne

A priore blockmodel

Occasionally, blocks are determined *a priore* according to some measured or hypothesised role, e.g.,

- partner or associate in a law firm
- level of position classification in an organisation
- girl or boy in a classroom

In this case, the ***density matrix*** is often of interest: a matrix reporting the proportion of possible ties within or between categories that are actually observed

Advice relation

		partner	associate
e.g.	**partner**	0.50	0.05
	associate	0.60	0.20

An example: Doreian's (1988) political network

Actors: 14 most prominent actors of a midwestern county

Ties: mutual private and public political support (consensus judgments of the local newspaper's reporters covering political affairs)

County Council comprises 7 members, elected for 4-year terms

County Auditor and Sheriff are elected positions

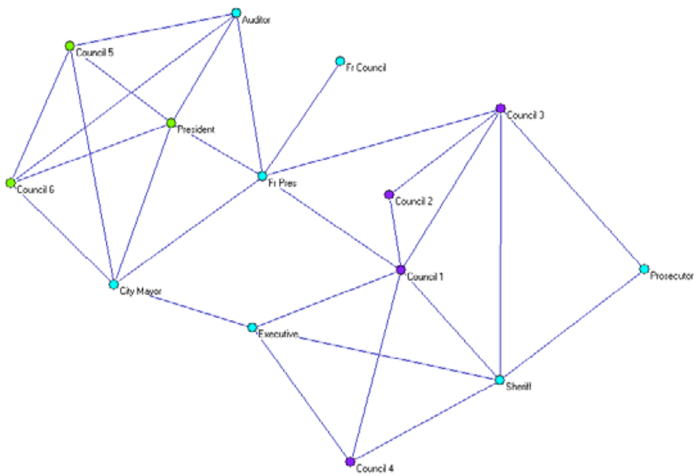
Later, a vote on a contentious issue (construction of a new jail) was taken: of interest was the relationship between councillors' votes and the network structure

The vote:

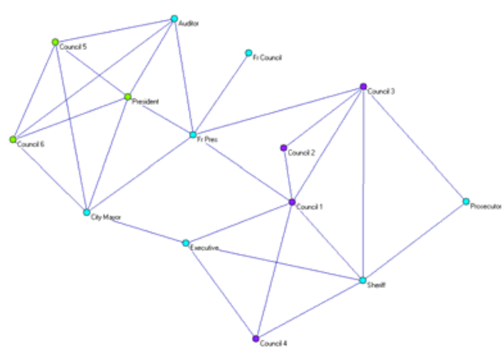
For: Council 1, Council 2, Council 3, Council 4

Against: President, Council 5, Council 6

Doreian (1988) political network
colour codes vote (aqua = non-voting)



Structural equivalence



Note that council members 5 and 6 are structurally equivalent

References

Doreian, P., Batagelj, V., & Ferligoj, A. (2004). *Generalized blockmodeling*. Cambridge: Cambridge University Press.

Doreian, P., Batagelj, V., & Ferligoj, A. (2005). Positional analyses of sociometric data. Pp. 77-97 in P. Carrington, J. Scott & S. Wasserman (Eds.) *Models and Methods in Social Network Analysis*. New York: Cambridge University Press.

Lorrain, F., & White, H. C. (1971). Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, *1*, 49-80.

Pattison, P. (1993). *Algebraic models for social networks*. New York: Cambridge University Press.

White, H.C., Boorman, S., & Breiger, R. L. (1976). Social structure from multiple networks, I. Blockmodels of roles and positions. *American Journal of Sociology*, *81*,730-780.

White, D. R., & Reitz, K. (1983). Graph and semigroup homomorphisms on networks of relations. *Social Networks*, *5*, 193-234.