Methods of Network Analysis

Clustering and Blockmodeling

4. Blockmodeling

Vladimir Batagelj
University of Ljubljana, Slovenia

University of Konstanz, Algorithms and Data Structures

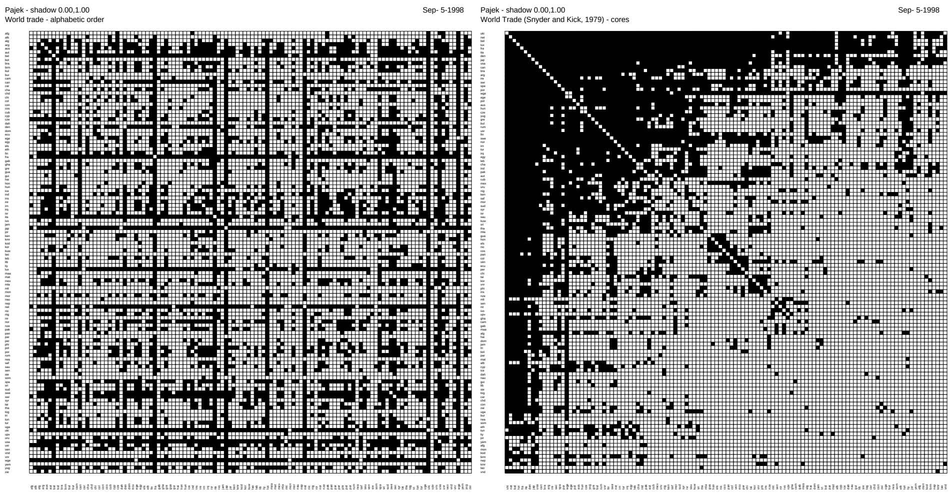
June 13, 2002, 14-16h, room F 426

Blockmodeling

- matrix rearrangement view on blockmodeling
- blockmodeling and clustering
  structural and regular equivalence
- generalized blockmodeling
- prespecified blockmodels
- 2-mode blockmodeling

Matrix rearrangement view on blockmodeling

Snyder & Kick's World trade network / $n = 118, m = 514$



Alphabetic order of countries and ordering based on the core decomposition

Ordering the matrix

There are several ways how to rearrange a given matrix – determine an *ordering* or *permutation* of its rows and columns – to get some insight into its structure:

- ordering by degree;
- ordering by connected components;
- ordering by core number, connected components inside core levels, and degree;
- ordering according to a hierarchical clustering and some other property.

There exists also some special procedures to determine the ordering. We shall describe them on 2-mode networks.

A *2-mode network* is a network (V, L, w) in which the vertices are partitioned into two sets V_1 and V_2 such that every link connects a vertex from V_1 to a vertex from V_2 (bipartite graph).

Seriation

Given a 2-mode network $\mathbf{N} = (\mathbf{U}, \mathbf{W}, R, w)$, $R \subseteq \mathbf{U} \times \mathbf{W}$, $w : R \to \mathbb{R}_0^+$.
Let $\rho : \mathbf{U} \to 1..n$, $\sigma : \mathbf{W} \to 1..m$ be bijections ('permutations').

For a row-unit $\mathrm{X} \in \mathbf{U}$ we define its row-sum

$$r(\mathrm{X}) = \sum_{\mathrm{Y} \in R(\mathrm{X})} w(\mathrm{X}, \mathrm{Y})$$

and for col-unit $\mathrm{Y} \in \mathbf{W}$ its col-sum

$$c(\mathrm{Y}) = \sum_{\mathrm{X} \in R^{-1}(\mathrm{Y})} w(\mathrm{X}, \mathrm{Y})$$

and finally we define the row-weight $p(\mathrm{X})$ and the col-weight $q(\mathrm{Y})$

$$p(\mathrm{X}) = \frac{1}{r(\mathrm{X})} \sum_{\mathrm{Y} \in R(\mathrm{X})} \sigma(\mathrm{Y}) w(\mathrm{X}, \mathrm{Y}) \qquad q(\mathrm{Y}) = \frac{1}{c(\mathrm{Y})} \sum_{\mathrm{X} \in R^{-1}(\mathrm{Y})} \rho(\mathrm{X}) w(\mathrm{X}, \mathrm{Y})$$

If $r(\mathrm{X}) = 0$ then $p(\mathrm{X}) = 0$; if $c(\mathrm{X}) = 0$ then $q(\mathrm{X}) = 0$.

. . . seriation

The *seriation* algorithm:

determine (random) σ ;
repeat
  compute row-weights $p(\mathrm{X})$, $\mathrm{X} \in \mathbf{U}$; $\rho := sort(\mathbf{U}, p)$;
  compute col-weights $q(\mathrm{Y})$, $\mathrm{Y} \in \mathbf{W}$; $\sigma := sort(\mathbf{W}, q)$;
until orderings stabilize (or max-steps reached)

In the case of 1-mode network we set $\rho = \sigma$ and use as a unit-weight $p(\mathrm{X}) + q(\mathrm{X})$.

Clumping

The idea of clumping approach is to arrange the units in such an order that a measure of 'clumpiness' is maximized. The ordering is built sequentially by greedy adding a new unit at the best place in the current ordering represented by a list containing k units and two 'sentinel' units $[\mathrm{X}_0 = \mathbf{0}, \mathrm{X}_1, \mathrm{X}_2, \ldots, \mathrm{X}_k, \mathbf{0} = \mathrm{X}_{k+1}]$. Inserting a row-unit X in this list after the element X_i produces the row-'clumpiness'

$$Q(i) = \sum_{\mathrm{Y} \in R(\mathrm{X}) \cap (R(\mathrm{X}_i) \cup R(\mathrm{X}_{i+1}))} w(\mathrm{X}, \mathrm{Y})(w(\mathrm{X}_i, \mathrm{Y}) + w(\mathrm{X}_{i+1}, \mathrm{Y}))$$

If $(\mathrm{X}, \mathrm{Y}) \notin R$ we set $w(\mathrm{X}, \mathrm{Y}) = 0$. The *clumping* algorithm

select (random) $\mathrm{X} \in \mathbf{U}$; $\mathbf{S} := \mathbf{U} \setminus \{\mathrm{X}\}$; $order := [\mathbf{0}, \mathrm{X}, \mathbf{0}]$; $k := 1$;
while $\mathbf{S} \neq \emptyset$ **do begin**
  select $\mathrm{X} \in \mathbf{S}$; $\mathbf{S} := \mathbf{S} \setminus \{\mathrm{X}\}$;
  for $i := 0$ **to** k **do** compute $Q(i)$;
  $j := \mathrm{argmax}_i\, Q(i)$; insert X in the $order$ after X_j; $k := k + 1$
end;

. . . clumping

In the same way (transpose the network) we can order also the columns.

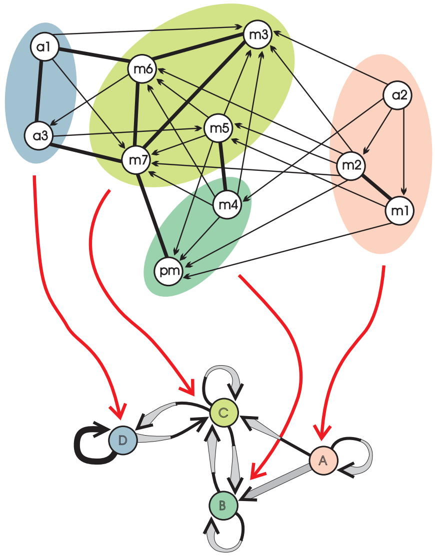
In the case of 1-mode network we use the combined criterion – sum of row- and col-clumpiness.

Untested idea 1: subtract the squares of values with no matching value.

Untested idea 2: for large networks it would be better to design the algorithm around the sets $R(\mathrm{X}_i) \cup R(\mathrm{X}_{i+1})$.

Blockmodeling as a clustering problem

The goal of *blockmodeling* is to reduce a large, potentially incoherent network to a smaller comprehensible structure that can be interpreted more readily. Blockmodeling, as an empirical procedure, is based on the idea that units in a network can be grouped according to the extent to which they are equivalent, according to some *meaningful* definition of equivalence.



Cluster, clustering, blocks

One of the main procedural goals of blockmodeling is to identify, in a given network $\mathbf{N} = (\mathbf{U}, R)$, $R \subseteq \mathbf{U} \times \mathbf{U}$, *clusters* (classes) of units that share structural characteristics defined in terms of R. The units within a cluster have the same or similar connection patterns to other units. They form a *clustering* $\mathbf{C} = \{C_1, C_2, \ldots, C_k\}$ which is a *partition* of the set $\mathbf{U}$. Each partition determines an equivalence relation (and vice versa). Let us denote by $\sim$ the relation determined by partition $\mathbf{C}$.

A clustering $\mathbf{C}$ partitions also the relation R into *blocks*

$$R(C_i, C_j) = R \cap C_i \times C_j$$

Each such block consists of units belonging to clusters C_i and C_j and all arcs leading from cluster C_i to cluster C_j. If $i = j$, a block $R(C_i, C_i)$ is called a *diagonal* block.

Structural and regular equivalence

Regardless of the definition of equivalence used, there are two basic approaches to the equivalence of units in a given network (compare Faust, 1988):

- the equivalent units have the same connection pattern to the **same** neighbors;
- the equivalent units have the same or similar connection pattern to (possibly) **different** neighbors.

The first type of equivalence is formalized by the notion of structural equivalence and the second by the notion of regular equivalence with the latter a generalization of the former.

Structural equivalence

Units are equivalent if they are connected to the rest of the network in *identical* ways (Lorrain and White, 1971). Such units are said to be *structurally equivalent*.

The units X and Y are *structurally equivalent*, we write $\mathrm{X} \equiv \mathrm{Y}$, iff the permutation (transposition) $\pi = (\mathrm{X}\ \mathrm{Y})$ is an automorphism of the relation R (Borgatti and Everett, 1992).

In other words, X and Y are structurally equivalent iff:

s1.	$\mathrm{X}R\mathrm{Y} \Leftrightarrow \mathrm{Y}R\mathrm{X}$	s3.	$\forall \mathrm{Z} \in \mathbf{U} \setminus \{\mathrm{X}, \mathrm{Y}\} : (\mathrm{X}R\mathrm{Z} \Leftrightarrow \mathrm{Y}R\mathrm{Z})$
s2.	$\mathrm{X}R\mathrm{X} \Leftrightarrow \mathrm{Y}R\mathrm{Y}$	s4.	$\forall \mathrm{Z} \in \mathbf{U} \setminus \{\mathrm{X}, \mathrm{Y}\} : (\mathrm{Z}R\mathrm{X} \Leftrightarrow \mathrm{Z}R\mathrm{Y})$

The blocks for structural equivalence are null or complete with variations on diagonal in diagonal blocks.

Regular equivalence

Integral to all attempts to generalize structural equivalence is the idea that units are equivalent if they link in equivalent ways to other units that are also equivalent.

White and Reitz (1983): The equivalence relation $\approx$ on $\mathbf{U}$ is a *regular equivalence* on network $\mathbf{N} = (\mathbf{U}, R)$ if and only if for all $\mathrm{X}, \mathrm{Y}, \mathrm{Z} \in \mathbf{U}$, $\mathrm{X} \approx \mathrm{Y}$ implies both

R1. $\mathrm{X}R\mathrm{Z} \Rightarrow \exists \mathrm{W} \in \mathbf{U} : (\mathrm{Y}R\mathrm{W} \wedge \mathrm{W} \approx \mathrm{Z})$

R2. $\mathrm{Z}R\mathrm{X} \Rightarrow \exists \mathrm{W} \in \mathbf{U} : (\mathrm{W}R\mathrm{Y} \wedge \mathrm{W} \approx \mathrm{Z})$

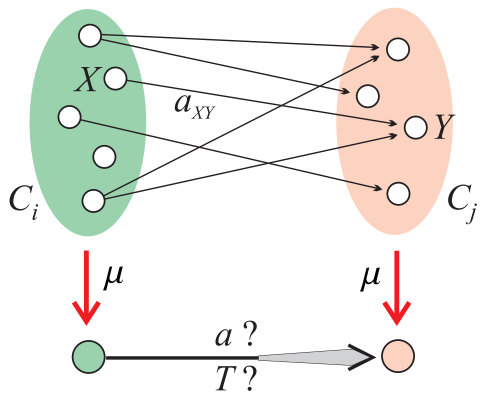
Another view of regular equivalence is based on colorings (Everett, Borgatti 1996).

Theorem 4.1 (Batagelj, Doreian, Ferligoj, 1992) *Let $\mathbf{C} = \{C_i\}$ be a partition corresponding to a regular equivalence $\approx$ on the network $\mathbf{N} = (\mathbf{U}, R)$. Then each block $R(C_u, C_v)$ is either null or it has the property that there is at least one 1 in each of its rows and in each of its columns. Conversely, if for a given clustering $\mathbf{C}$, each block has this property then the corresponding equivalence relation is a regular equivalence.*

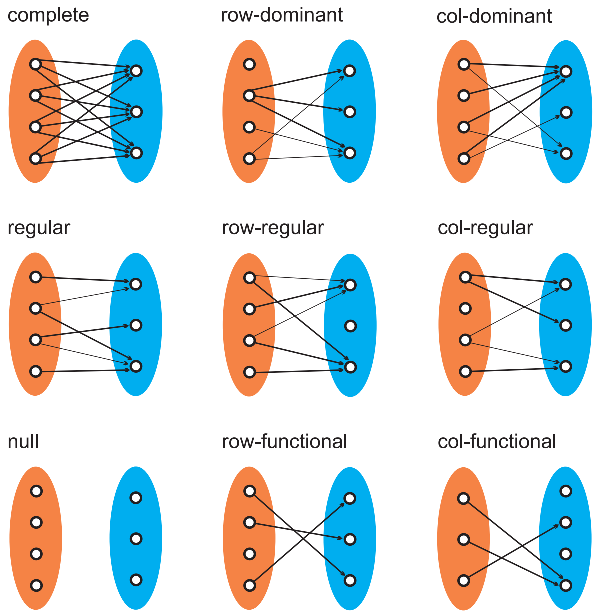
The blocks for regular equivalence are null or 1-covered blocks.

Generalized Blockmodeling

A *blockmodel* consists of structures obtained by identifying all units from the same cluster of the clustering $\mathbf{C}$. For an exact definition of a blockmodel we have to be precise also about which blocks produce an arc in the *reduced graph* and which do not, and of what *type*. Some types of connections are presented in the figure on the next slide. The reduced graph can be represented by relational matrix, called also *image matrix*.

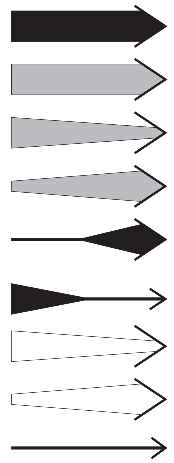


Block Types



Characterizations of Types of Blocks

null	nul	all 0 *	
complete	com	all 1 *	
regular	reg	1-covered rows and columns	
row-regular	rre	each row is 1-covered	
col-regular	cre	each column is 1 -covered	
row-dominant	rdo	$\exists$ all 1 row *	
col-dominant	cdo	$\exists$ all 1 column *	
row-functional	rfn	$\exists!$ one 1 in each row	
col-functional	cfn	$\exists!$ one 1 in each column	
non-null	one	$\exists$ at least one 1	

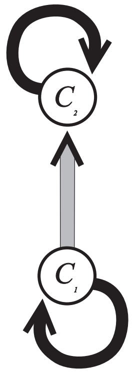


* except this may be diagonal

A block is *symmetric* iff $\forall \mathrm{X}, \mathrm{Y} \in C_i \times C_j : (\mathrm{X}R\mathrm{Y} \Leftrightarrow \mathrm{Y}R\mathrm{X})$.

Block Types and Matrices

$$\begin{array}{cccc|cccc}
1 & 1 & 1 & 1 & 1 & 1 & 0 & 0 \\
1 & 1 & 1 & 1 & 0 & 1 & 0 & 1 \\
1 & 1 & 1 & 1 & 0 & 0 & 1 & 0 \\
1 & 1 & 1 & 1 & 1 & 0 & 0 & 0 \\
\hline
0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 0 & 1 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 0 & 1 \\
0 & 0 & 0 & 0 & 1 & 1 & 1 & 0
\end{array}$$



	C_1	C_2
C_1	complete	regular
C_2	null	complete

Formalization of blockmodeling

Let V be a set of positions or images of clusters of units. Let $\mu : \mathbf{U} \to V$ denote a mapping which maps each unit to its position. The cluster of units $C(t)$ with the same position $t \in V$ is

$$C(t) = \mu^{-1}(t) = \{\mathrm{X} \in \mathbf{U} : \mu(\mathrm{X}) = t\}$$

Therefore

$$\mathbf{C}(\mu) = \{C(t) : t \in V\}$$

is a partition (clustering) of the set of units $\mathbf{U}$.

Blockmodel

A *blockmodel* is an ordered sextuple $\mathcal{M} = (V, K, \mathcal{T}, Q, \pi, \alpha)$ where:

- V is a set of *types of units* (images or representatives of classes);
- $K \subseteq V \times V$ is a set of *connections*;
- $\mathcal{T}$ is a set of predicates used to describe the *types of connections* between different classes (clusters, groups, types of units) in a network. We assume that $\mathrm{nul} \in \mathcal{T}$. A mapping $\pi : K \to \mathcal{T} \setminus \{\mathrm{nul}\}$ assigns predicates to connections;
- Q is a set of *averaging rules*. A mapping $\alpha : K \to Q$ determines rules for computing values of connections.

A (surjective) mapping $\mu : \mathbf{U} \to V$ determines a blockmodel $\mathcal{M}$ of network $\mathbf{N}$ iff it satisfies the conditions:

$$\forall (t, w) \in K : \pi(t, w)(C(t), C(w))$$

and

$$\forall (t, w) \in V \times V \setminus K : \mathrm{nul}(C(t), C(w)).$$

Equivalences

Let $\sim$ be an equivalence relation over $\mathbf{U}$ and $[\mathrm{X}] = \{\mathrm{Y} \in \mathbf{U} : \mathrm{X} \approx \mathrm{Y}\}$. We say that $\sim$ is *compatible* with $\mathcal{T}$ over a network $\mathbf{N}$ iff

$$\forall \mathrm{X}, \mathrm{Y} \in \mathbf{U} \exists T \in \mathcal{T} : T([\mathrm{X}], [\mathrm{Y}]).$$

It is easy to verify that the notion of compatibility for $\mathcal{T} = \{\mathrm{nul}, \mathrm{reg}\}$ reduces to the usual definition of regular equivalence (White and Reitz 1983). Similarly, compatibility for $\mathcal{T} = \{\mathrm{nul}, \mathrm{com}\}$ reduces to structural equivalence (Lorrain and White 1971).

For a compatible equivalence $\sim$ the mapping $\mu \colon \mathrm{X} \mapsto [\mathrm{X}]$ determines a blockmodel with $V = \mathbf{U}/\sim$.

The problem of establishing a partition of units in a network in terms of a selected type of equivalence is a special case of **clustering problem** that can be formulated as an optimization problem.

Criterion function

One of the possible ways of constructing a criterion function that directly reflects the considered equivalence is to measure the fit of a clustering to an ideal one with perfect relations within each cluster and between clusters according to the considered equivalence.

Given a clustering $\mathbf{C} = \{C_1, C_2, \ldots, C_k\}$, let $\mathcal{B}(C_u, C_v)$ denote the set of all ideal blocks corresponding to block $R(C_u, C_v)$. Then the global error of clustering $\mathbf{C}$ can be expressed as

$$P(\mathbf{C}) = \sum_{C_u, C_v \in \mathbf{C}} \min_{B \in \mathcal{B}(C_u, C_v)} d(R(C_u, C_v), B)$$

where the term $d(R(C_u, C_v), B)$ measures the difference (error) between the block $R(C_u, C_v)$ and the ideal block B. d is constructed on the basis of characterizations of types of blocks. The function d has to be compatible with the selected type of equivalence.

For example, for structural equivalence, the term $d(R(C_u, C_v), B)$ can be expressed, for non-diagonal blocks, as

$$d(R(C_u, C_v), B) = \sum_{\mathrm{X} \in C_u, \mathrm{Y} \in C_v} |r_{\mathrm{X\,Y}} - b_{\mathrm{X\,Y}}|.$$

where $r_{\mathrm{X\,Y}}$ is the observed tie and $b_{\mathrm{X\,Y}}$ is the corresponding value in an ideal block. This criterion function counts the number of 1s in erstwhile null blocks and the number of 0s in otherwise complete blocks. These two types of inconsistencies can be weighted differently.

Determining the block error, we also determine the type of the best fitting ideal block (the types are ordered).

The criterion function $P(\mathbf{C})$ is *sensitive* iff $P(\mathbf{C}) = 0 \Leftrightarrow \mu$ (determined by $\mathbf{C}$) is an exact blockmodeling. For all presented block types sensitive criterion functions can be constructed (Batagelj, 1997).

The obtained optimization problem can be solved by local optimization. Once a partitioning μ and types of connection π are determined, we can also compute the values of connections by using averaging rules.

Benefits from Optimization Approach

- *ordinary / inductive blockmodeling*: Given a network $\mathbf{N}$ and set of types of connection $\mathcal{T}$, determine the model $\mathcal{M}$;
- *evaluation of the quality of a model, comparing different models, analyzing the evolution of a network* (Sampson data, Doreian and Mrvar 1996): Given a network $\mathbf{N}$, a model $\mathcal{M}$, and blockmodeling μ, compute the corresponding criterion function;
- *model fitting / deductive blockmodeling*: Given a network $\mathbf{N}$, set of types $\mathcal{T}$, and a family of models, determine μ which minimizes the criterion function (Batagelj, Ferligoj, Doreian, 1998).
- we can fit the network to a partial model and analyze the residual afterward;
- we can also introduce different constraints on the model, for example: units X and Y are of the same type; or, types of units X and Y are not connected; ...

Pre-Specified Blockmodels

The pre-specified blockmodeling starts with a blockmodel specified, in terms of substance, ***prior to an analysis***. Given a network, a set of ideal blocks is selected, a family of reduced models is formulated, and partitions are established by minimizing the criterion function.

The basic types of models are:

$$\begin{array}{|c|c|}\hline * & * \\ \hline * & 0 \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline * & 0 \\ \hline * & * \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline * & 0 \\ \hline 0 & * \\ \hline\end{array} \qquad \begin{array}{|c|c|}\hline 0 & * \\ \hline * & 0 \\ \hline\end{array}$$

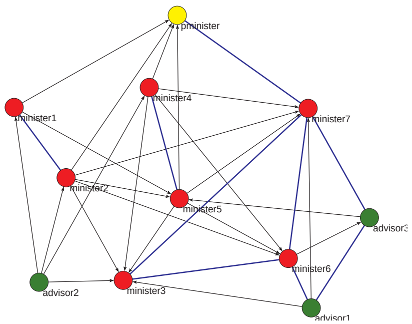
center - periphery | hierarchy | clustering | bipartition

The Student Government at the University of Ljubljana in 1992

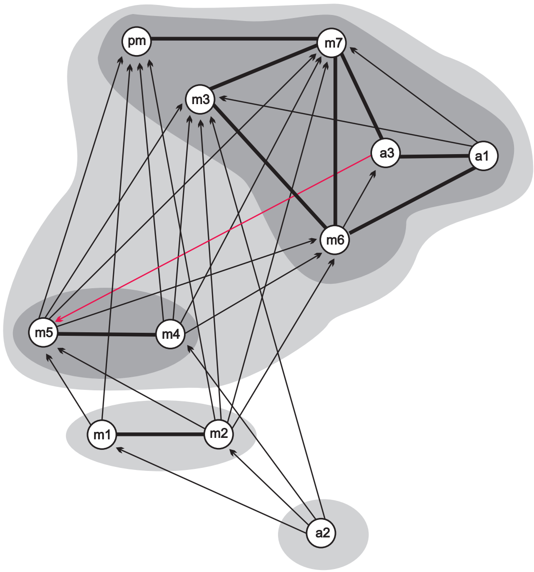
The relation is determined by the following question (Hlebec, 1993):

> Of the members and advisors of the Student Government, whom do you (most often) talk with (about the matters of the Student Government)?

		m 1	p 2	m 3	m 4	m 5	m 6	m 7	m 8	a 9	a 10	a 11
minister 1	1	·	1	1	·	·	1	·	·	·	·	·
p.minister	2	·	·	·	·	·	·	·	1	·	·	·
minister 2	3	1	1	·	1	·	1	1	1	·	·	·
minister 3	4	·	·	·	·	·	·	1	1	·	·	·
minister 4	5	·	1	·	1	·	1	1	1	·	·	·
minister 5	6	·	1	·	1	1	·	1	1	·	·	·
minister 6	7	·	·	·	1	·	·	·	1	1	·	1
minister 7	8	·	1	·	1	·	·	1	·	·	·	1
adviser 1	9	·	·	·	1	·	·	1	1	·	·	1
adviser 2	10	1	·	1	1	1	·	·	·	·	·	·
adviser 3	11	·	·	·	·	·	1	·	1	1	·	·



A Symmetric Acyclic Blockmodel of Student Government

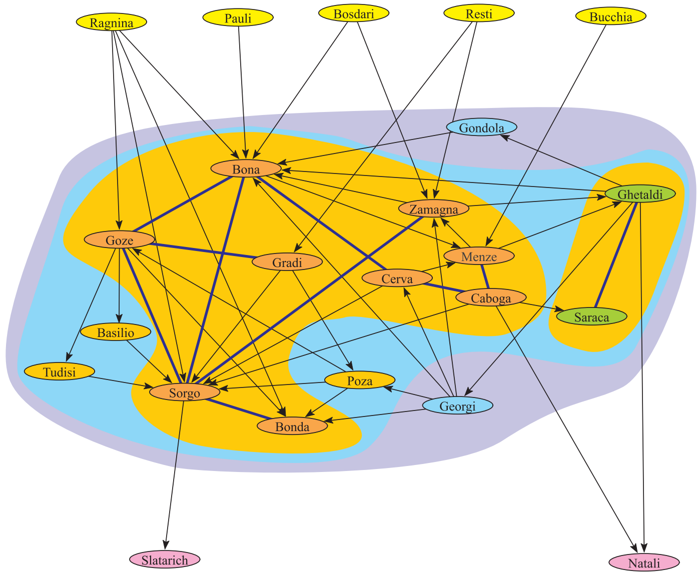


The obtained clustering in 4 clusters is almost exact. The only error is produced by the arc $(a3, m5)$.

Ragusan Noble Families Marriage Network, 18th and 19th Century

		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23
Basilio	1	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	1	·	·
Bona	2	·	·	·	·	·	·	1	·	·	·	2	·	2	·	·	·	·	·	·	·	1	·	·
Bonda	3	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	2	·	·
Bosdari	4	·	1	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	1
Bucchia	5	·	·	·	·	·	·	·	·	·	·	·	·	1	·	·	·	·	·	·	·	·	·	·
Caboga	6	·	·	·	·	·	·	1	·	·	·	·	·	1	1	·	·	·	·	1	·	1	·	·
Cerva	7	·	1	·	·	·	1	·	·	·	·	·	·	1	·	·	·	·	·	·	·	1	·	·
Georgi	8	·	1	2	·	·	·	1	·	·	·	·	·	·	·	·	4	·	·	·	·	·	·	1
Ghetaldi	9	·	1	·	·	·	·	·	1	·	1	·	·	·	1	·	·	·	·	1	·	·	·	·
Gondola	10	·	1	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
Goze	11	1	2	1	·	·	·	·	·	·	·	2	2	·	·	·	·	·	·	·	·	2	1	·
Gradi	12	·	·	·	·	·	·	·	·	·	·	1	·	·	·	·	1	·	·	·	·	3	·	·
Menze	13	·	·	·	·	·	1	·	·	1	·	·	·	·	·	·	·	·	·	·	·	·	·	1
Natali	14	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
Pauli	15	·	1	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
Poza	16	·	·	2	·	·	·	·	·	·	·	1	·	·	·	·	1	·	·	·	·	1	·	·
Ragnina	17	·	1	1	·	·	·	·	·	·	·	1	·	·	·	·	·	·	·	·	·	1	·	·
Resti	18	·	·	·	·	·	·	·	·	·	·	·	1	·	·	·	·	·	·	·	·	·	·	1
Saraca	19	·	·	·	·	·	·	·	·	1	·	·	·	·	·	·	·	·	·	·	·	·	·	·
Slatarich	20	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·
Sorgo	21	·	2	1	·	·	·	·	·	·	·	1	·	·	·	·	·	·	·	·	1	1	·	1
Tudisi	22	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	·	1	·	·
Zamagna	23	·	1	·	·	·	·	·	·	2	·	·	·	·	·	·	·	·	·	·	·	1	·	·

A Symmetric-Acyclic Decomposition of the Ragusan Families Network



Blockmodeling in 2-mode networks

We already presented some ways of rearranging 2-mode network matrices at the beginning of this lecture.

It is also possible to formulate this goal as a generalized blockmodeling problem where the solutions consist of two partitions — row-partition and column-partition.

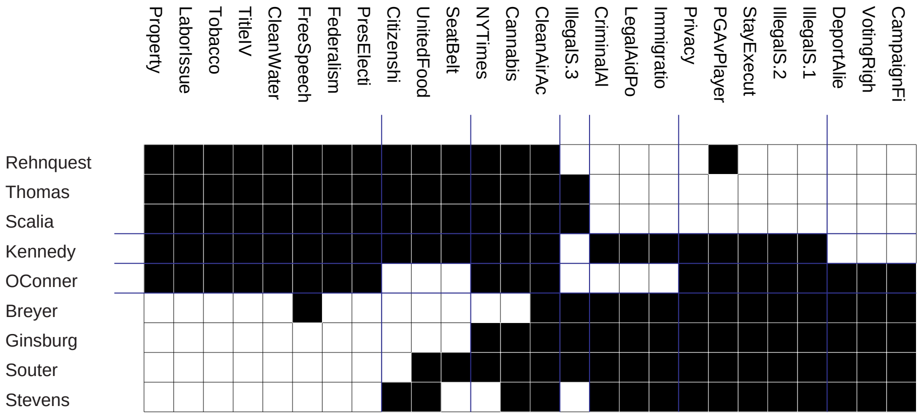
Supreme Court Voting for Twenty-Six Important Decisions

Issue	Label	Br	Gi	So	St	OC	Ke	Re	Sc	Th
Presidential Election	PE	-	-	-	-	+	+	+	+	+
Criminal Law Cases										
Illegal Search 1	CL1	+	+	+	+	+	+	-	-	-
Illegal Search 2	CL2	+	+	+	+	+	+	-	-	-
Illegal Search 3	CL3	+	+	+	-	-	-	-	+	+
Seat Belts	CL4	-	-	+	-	-	+	+	+	+
Stay of Execution	CL5	+	+	+	+	+	+	-	-	-
Federal Authority Cases										
Federalism	FA1	-	-	-	-	+	+	+	+	+
Clean Air Action	FA2	+	+	+	+	+	+	+	+	+
Clean Water	FA3	-	-	-	-	+	+	+	+	+
Cannabis for Health	FA4	0	+	+	+	+	+	+	+	+
United Foods	FA5	-	-	+	+	-	+	+	+	+
NY Times Copyrights	FA6	-	+	+	-	+	+	+	+	+
Civil Rights Cases										
Voting Rights	CR1	+	+	+	+	+	-	-	-	-
Title VI Disabilities	CR2	-	-	-	-	+	+	+	+	+
PGA v. Handicapped Player	CR3	+	+	+	+	+	+	+	-	-
Immigration Law Cases										
Immigration Jurisdiction	Im1	+	+	+	+	-	+	-	-	-
Deporting Criminal Aliens	Im2	+	+	+	+	+	-	-	-	-
Detaining Criminal Aliens	Im3	+	+	+	+	-	+	-	-	-
Citizenship	Im4	-	-	-	+	-	+	+	+	+
Speech and Press Cases										
Legal Aid for Poor	SP1	+	+	+	+	-	+	-	-	-
Privacy	SP2	+	+	+	+	+	+	-	-	-
Free Speech	SP3	+	-	-	-	+	+	+	+	+
Campaign Finance	SP4	+	+	+	+	+	-	-	-	-
Tobacco Ads	SP5	-	-	-	-	+	+	+	+	+
Labor and Property Rights Cases										
Labor Rights	LPR1	-	-	-	-	+	+	+	+	+
Property Rights	LPR2	-	-	-	-	+	+	+	+	+

The Supreme Court Justices and their 'votes' on a set of 26 "important decisions" made during the 2000-2001 term, Doreian and Fujimoto (2002).

The Justices (in the order in which they joined the Supreme Court) are: Rehnquist (1972), Stevens (1975), O'Conner (1981), Scalia (1982), Kennedy (1988), Souter (1990), Ginsburg (1993) and Breyer (1994).

. . . Supreme Court Voting / a (4,7) partition



upper – conservative / lower – liberal

Final Remarks

The current, local optimization based, programs for generalized blockmodeling can deal only with networks with at most some hundreds of units. What to do with larger networks is an open question. For some specialized problems also procedures for (very) large networks can be developed (Doreian, Batagelj, Ferligoj, 1998; Batagelj, Zaveršnik, 2002).

Another interesting problem is the development of *blockmodeling of valued networks* or more general *relational data analysis* (Batagelj, Ferligoj, 2000).

Most of described procedures are implemented in Pajek – program for analysis and visualization of large networks. It is freely available, for noncommercial use, at:

http://vlado.fmf.uni-lj.si/pub/networks/pajek/

The current version of these lectures is available at:

http://vlado.fmf.uni-lj.si/pub/networks/doc/

The author's e-mail address is: **vladimir.batagelj@uni-lj.si**