Generalized blockmodeling

Vladimir Batagelj
University of Ljubljana
Faculty of Mathematics and Physics
Jadranska 19
1 000 Ljubljana, Slovenia
E-mail: vladimir.batagelj@uni-lj.si

Anuška Ferligoj
University of Ljubljana
Faculty of Social Sciences
Kardeljeva pl. 5
1 000 Ljubljana, Slovenia
E-mail: anuska.ferligoj@uni-lj.si

Patrick Doreian
University of Pittsburgh
Department of Sociology
PA 15260
Pittsburgh, USA
E-mail: pitpat+@pitt.edu

Keywords: clustering, blockmodeling, social network, pre-specified blockmodel, local optimization.

Edited by:

Received: **Revised:** **Accepted:**

The goal of blockmodeling is to reduce a large, potentially incoherent network to a smaller comprehensible structure that can be interpreted more readily. In the paper we present an overview of basic ideas and developments in this area.

1 Basic Notions

1.1 Network

Let $E = \{X_1, X_2, \ldots, X_n\}$ be a finite set of *units*. The units are related by binary *relations* $R_t \subseteq E \times E, \quad t = 1, \ldots, r, r \geq 1$ which determine a *network*

$$\mathcal{N} = (E, R_1, R_2, \ldots, R_r)$$

In the following we restrict our discussion to a single relation R described by a corresponding binary matrix $\mathbf{R} = [r_{ij}]_{n \times n}$ where

$$r_{ij} = \begin{cases} 1 & X_i R X_j \\ 0 & \text{otherwise} \end{cases}$$

In some applications r_{ij} can be a nonnegative real number expressing the strength of the relation R between units X_i and X_j.

1.1.1 Example: Student Government

In Table 1 and Figure 1 the Student Government network is presented. It consists of communication interactions among twelve members and advisors of the Student Government at the University in Ljubljana (Hlebec, 1993). The results of the measurement are not real interactions among actors but cognition about communication interactions. Data were collected with face to face interviews, conducted in May 1992.

Communication flow among actors was identified by the following question:

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

Table 1: Student Government matrix

		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	0	1	1	0	0	1	0	0	0	0	0
p.minister	2	0	0	0	0	0	0	0	1	0	0	0
minister 2	3	1	1	0	1	0	1	1	1	0	0	0
minister 3	4	0	0	0	0	0	0	1	1	0	0	0
minister 4	5	0	1	0	1	0	1	1	1	0	0	0
minister 5	6	0	1	0	1	1	0	1	1	0	0	0
minister 6	7	0	0	0	1	0	0	0	1	1	0	1
minister 7	8	0	1	0	1	0	0	1	0	0	0	1
adviser 1	9	0	0	0	1	0	0	1	1	0	0	1
adviser 2	10	1	0	1	1	1	0	0	0	0	0	0
adviser 3	11	0	0	0	0	0	1	0	1	1	0	0

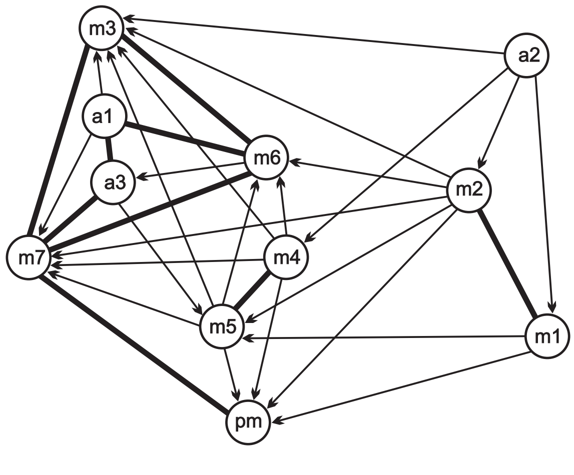


Figure 1: Network graph: Student Government – discussion, recall

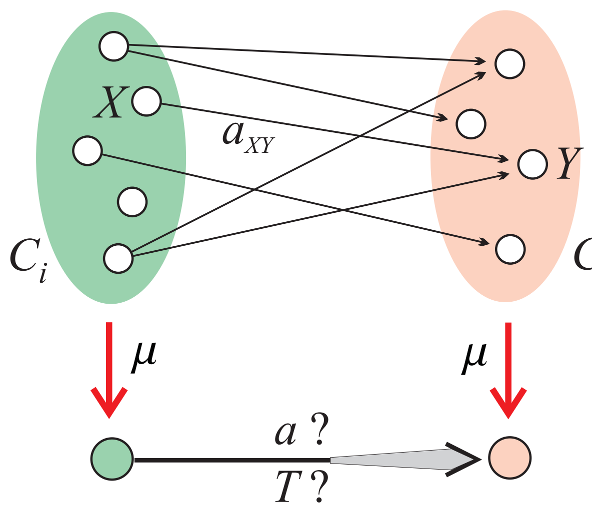


Figure 2: Blockmodeling scheme.

The content of the communication flow was limited to the matters of the Student Government. The time frame was also defined: the question was referred to the six months period. One respondent refused to cooperate in the experiment. As he was not considered in the analysis, the network consists of eleven actors.

1.2 Cluster and Clustering

One of the main procedural goals of blockmodeling is to identify, in a given network, *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*

$$\mathcal{C} = \{C_1, C_2, \ldots, C_k\}$$

which is a partition of the set E: $\bigcup_i C_i = E$ and $i \neq j \Rightarrow C_i \cap C_j = \emptyset$. Each partition determines an equivalence relation (and vice versa).

1.3 Block

A clustering $\mathcal{C}$ 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.

1.4 Blockmodel and Blockmodeling

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

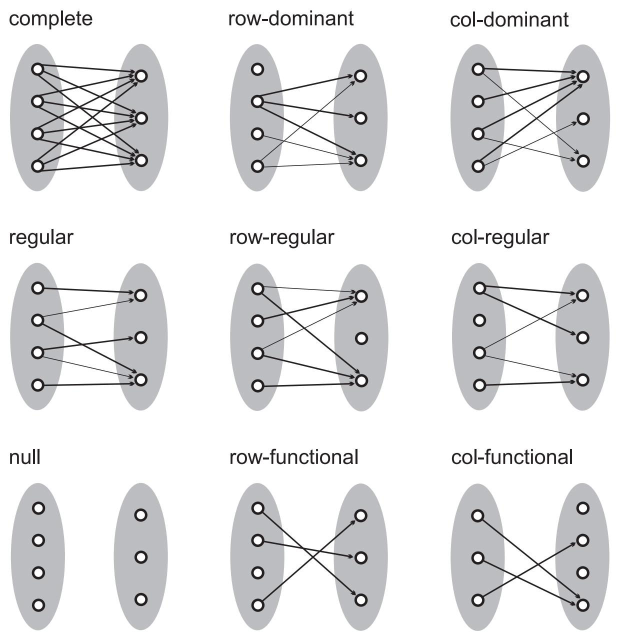


Figure 3: Types of connection between two sets; the left set is the ego-set.

are equivalent, according to some *meaningful* definition of equivalence.

A *blockmodel* consists of structures obtained by identifying all units from the same cluster of the clustering $\mathcal{C}$. For an exact definition of a blockmodel (see Figure 2) 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 Figure 3. A block is *symmetric* if

$$\forall (X,Y) \in C_i \times C_j : (X\,R\,Y \Leftrightarrow Y\,R\,X)$$

Note that for nondiagonal blocks this condition involves a pair of blocks $R(C_i, C_j)$ and $R(C_j, C_i)$.

The reduced graph can be presented by relational matrix, called also *image matrix* (see Table 2).

A clustering and the induced blockmodel of the Student Government is presented in Figure 4.

2 Blockmodeling - Formalization

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

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

Therefore

$$\mathcal{C}(\mu) = \{C(t) : t \in U\}$$

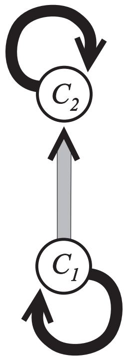
is a partition (clustering) of the set of units E.

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

Table 2: Block types and matrices.

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
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

	C_1	C_2
C_1	complete	regular
C_2	null	complete



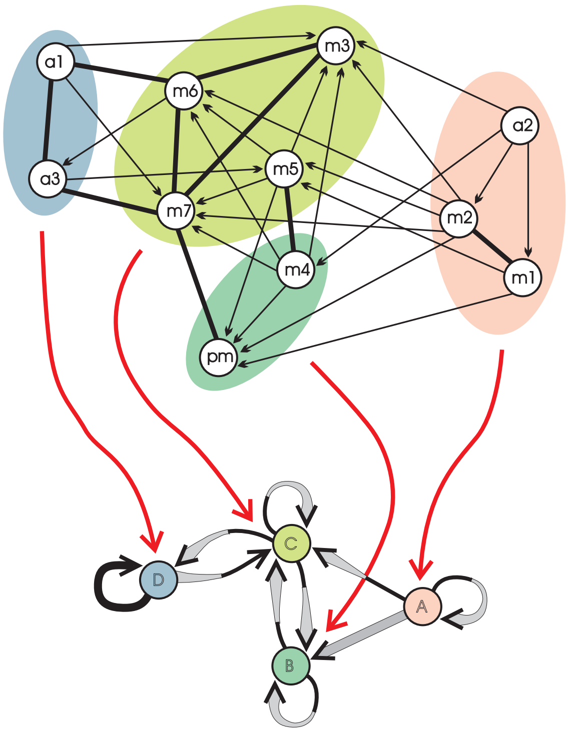


Figure 4: Blockmodeling example.

- U is a set of *positions* (types of units);
- $K \subseteq U \times U$ is a set of *connections*;
- $\mathcal{T}$ is a set of predicates used to describe the types of connections between different clusters in a network. We assume that nul $\in \mathcal{T}$.
- a mapping $\pi : K \to \mathcal{T} \setminus \{\text{nul}\}$ 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 $\mu : E \to U$ determines a blockmodel $\mathcal{M}$ of network $\mathcal{N}$ iff it satisfies the conditions:

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

and

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

2.1 Equivalences

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

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

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

For a compatible equivalence $\approx$ the mapping $\mu : X \mapsto [X]$ determines a blockmodel with $U = E/\approx$.

3 Optimization

3.1 Criterion Function

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: determine the clustering $\mathcal{C}^*$ for which

$$P(\mathcal{C}^*) = \min_{\mathcal{C} \in \Phi} P(\mathcal{C})$$

where $\mathcal{C}$ is a clustering of a given *set of units* E, Φ is the set of all feasible clusterings and $P : \Phi \to \mathbb{R}$ the *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 set of types of connection $\mathcal{T}$ we can introduce the set of *ideal blocks* for a given type $T \in \mathcal{T}$ by

$$\mathcal{B}(C_i, C_j; T) = \{B \subseteq C_i \times C_j : T(B)\}$$

Using Table 3 we can efficiently test whether the block $R(C_i, C_j)$ is of the type T; and define the *deviation* $\delta(C_i, C_j; T)$ of a block $R(C_i, C_j)$ from the nearest ideal block. For example

$$\delta(C_i, C_j; \text{reg}) = |C_i| \cdot (|C_j| - c_j) + |C_j| \cdot (|C_i| - r_i)$$

where c_j is the number of non-zero columns, and r_i is the number of non-zero rows in the block $R(C_i, C_j)$. For details see (Batagelj 1997).

Table 3: Characterizations of Types of Blocks.

null	nul	all 0*
complete	com	all 1*
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*
regular	reg	1-covered rows and 1-covered columns
non-null	one	$\exists$ at least one 1

* except may be diagonal

For the proposed types all deviations are *sensitive*

$$\delta(C_i, C_j; T) = 0 \Leftrightarrow T(R(C_i, C_j)).$$

Therefore a block $R(C_i, C_j)$ is of a type T exactly when the corresponding deviation $\delta(C_i, C_j; T)$ is 0. In the deviation δ we can also incorporate values of lines ν.

Based on deviation $\delta(C_i, C_j; T)$ we introduce the *block-error* $\varepsilon(C_i, C_j; T)$ of $R(C_i, C_j)$ for type T. An example of block-error is

$$\varepsilon(C_i, C_j; T) = w(T)\delta(C_i, C_j; T)$$

where $w(T) > 0$ is a weight of type T.

We extend the block-error to the set of feasible types $\mathcal{T}$ by defining

$$\varepsilon(C_i, C_j; \mathcal{T}) = \min_{T \in \mathcal{T}} \varepsilon(C_i, C_j; T)$$

and

$$\pi(\mu(C_i), \mu(C_j)) = \text{argmin}_{T \in \mathcal{T}} \varepsilon(C_i, C_j; T)$$

To make π well-defined, we order (priorities) the set $\mathcal{T}$ and select the first type from $\mathcal{T}$ which minimizes ε. We combine block-errors into a *total error* – blockmodeling *criterion function*

$$P(\mathcal{C}(\mu); \mathcal{T}) = \sum_{(t,w) \in U \times U} \varepsilon(C(t), C(w); \mathcal{T}).$$

For criterion function P we have

$$P(\mathcal{C}(\mu)) = 0 \Leftrightarrow \mu \text{ is an exact blockmodeling}$$

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*.

3.2 Local Optimization

For solving the blockmodeling problem we use a local optimization procedure (relocation algorithm):

Determine the initial clustering $\mathcal{C}$;
repeat:
  if in the neighborhood of the current clustering $\mathcal{C}$ there exists a clustering $\mathcal{C}'$ such that $P(\mathcal{C}') < P(\mathcal{C})$
  then move to clustering $\mathcal{C}'$.

The neighborhood in this local optimization procedure is determined by the following two transformations:

- *moving* a unit X_k from cluster C_p to cluster C_q (*transition*);

- *interchanging* units X_u and X_v from different clusters C_p and C_q (*transposition*).

3.3 Benefits from Optimization Approach

- *ordinary / inductive blockmodeling*: Given a network $\mathcal{N}$ and set of types of connection $\mathcal{T}$, determine $\mathcal{M}$, i.e., μ, π and α;

- *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 $\mathcal{N}$, a model $\mathcal{M}$, and blockmodeling μ, compute the corresponding criterion function;

- *model fitting / deductive blockmodeling*: Given a network $\mathcal{N}$, set of types $\mathcal{T}$, and a model $\mathcal{M}$, 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; ...

4 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 reduced model is formulated, and partitions are established by minimizing the criterion function. The pre-specified blockmodeling is supported by the program MODEL 2 (Batagelj, 1996).

As an example of pre-specified blockmodel we present in Figure 5 a symmetric acyclic blockmodel of Student Government. The obtained clustering in 4 clusters is almost exact – acyclic model with symmetric clusters. The only error is produced by the arc $(a3, m5)$.

5 Final Remarks

The current, local optimization based, programs for generalized blockmodeling can deal only with networks with at

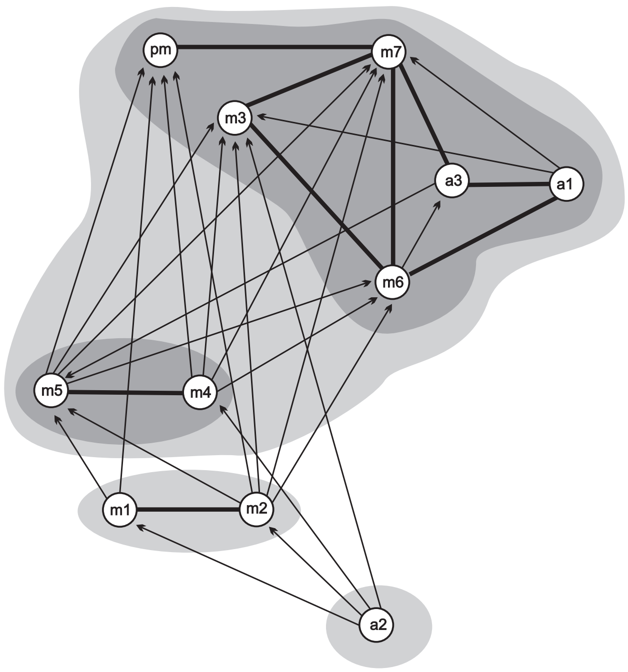


Figure 5: Symmetric acyclic blockmodel of Student Government.

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).

Another interesting problem is the development of blockmodeling of valued networks.

MODEL 2 and related programs and data can be obtained from

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

Acknowledgment: This work was supported by the Ministry of Science and Technology of Slovenia, Project J1-8532.

References

[1] Batagelj, V. (1991): STRAN – STRucture ANalysis. Manual, Ljubljana.

[2] Batagelj, V. (1997): Notes on Blockmodeling. *Social Networks*, **19**, 143-155. Also in: *Abstracts and Short Versions of Papers*, 3rd European Conference on Social Network Analysis, München, 1993: DJI, 1-9.

[3] Batagelj, V., P. Doreian, and A. Ferligoj (1992): An optimizational approach to regular equivalence. *Social Networks*, 14:121–135.

[4] Batagelj, V., A. Ferligoj, and P. Doreian (1992): Direct and indirect methods for structural equivalence. *Social Networks*, 14:63–90.

[5] Batagelj, V., Ferligoj, A., and Doreian, P. (1998): Fitting Pre-Specified Blockmodels, in *Data Science, Classification, and Related Methods*, Eds., C. Hayashi, N. Ohsumi, K. Yajima, Y. Tanaka, H. H. Bock, and Y. Baba, Springer-Verlag, Tokyo, p.p. 199-206.

[6] Borgatti, S.P. and M.G. Everett (1989): The class of all regular equivalences: Algebraic structure and computation. *Social Networks*, 11:65–88.

[7] Doreian, P., V. Batagelj, and A. Ferligoj (1994): Partitioning Networks on Generalized Concepts of Equivalence. *Journal of Mathematical Sociology*, 19/1:1–27.

[8] Doreian, P., V. Batagelj, and A. Ferligoj (1998): Symmetric-Acyclic Decompositions of Networks. To appear in *Journal of Classification*.

[9] Doreian, P. and A. Mrvar (1996) A Partitioning Approach to Structural Balance. *Social Networks* 18:149–168.

[10] Faust, K. (1988): Comparison of methods for positional analysis: Structural and general equivalences. *Social Networks*, 10:313–341.

[11] A. Ferligoj, V. Batagelj, and Doreian, P. (1994): On Connecting Network Analysis and Cluster Analysis. In *Contributions to Mathematical Psychology, Psychometrics, and Methodology* (G.H. Fischer, D. Laming Eds.), New York: Springer.

[12] Lorrain, F. and H.C. White (1971): Structural equivalence of individuals in social networks. *Journal of Mathematical Sociology*, 1:49–80.

[13] Hlebec, V. (1993): Recall versus recognition: Comparison of two alternative procedures for collecting social network data. Developments in Statistics and Methodology. (A. Ferligoj and A. Kramberger, editors) Metodološki zvezki 9, Ljubljana: FDV, 121-128.

[14] White, D.R. and K.P. Reitz (1983): Graph and semigroup homomorphisms on networks of relations. *Social Networks*, 5:193–234.