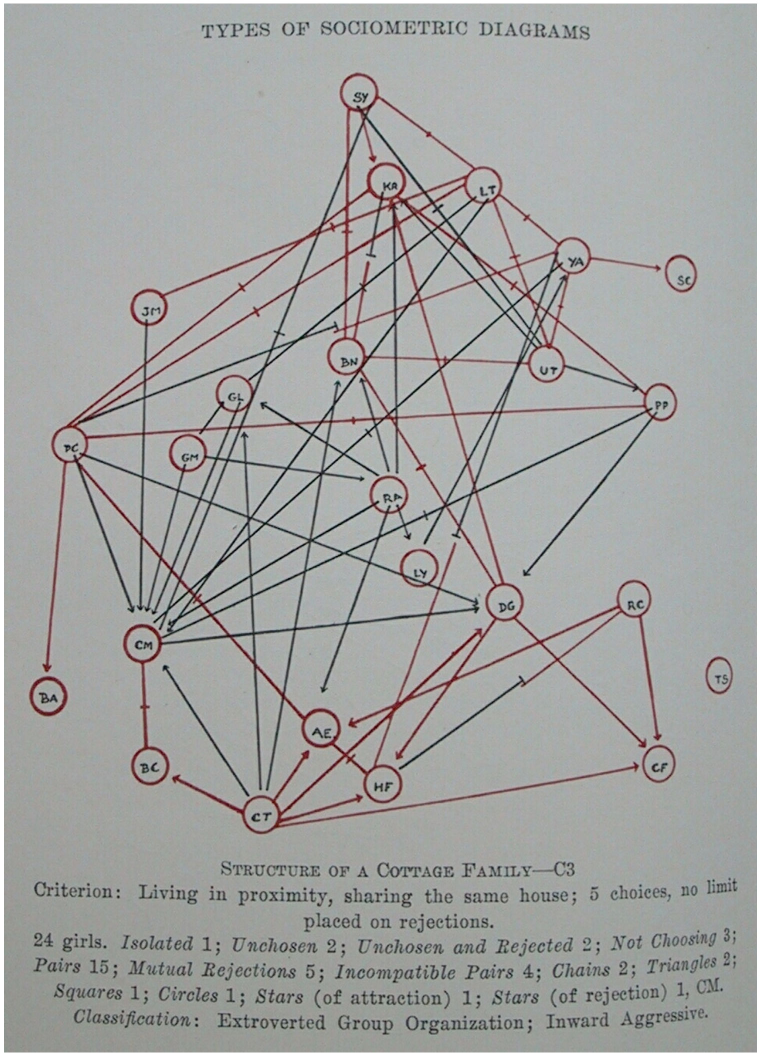


Course on Social Network Analysis

Blockmodeling

Anuška Ferligoj,
Vladimir Batagelj,
Andrej Mrvar

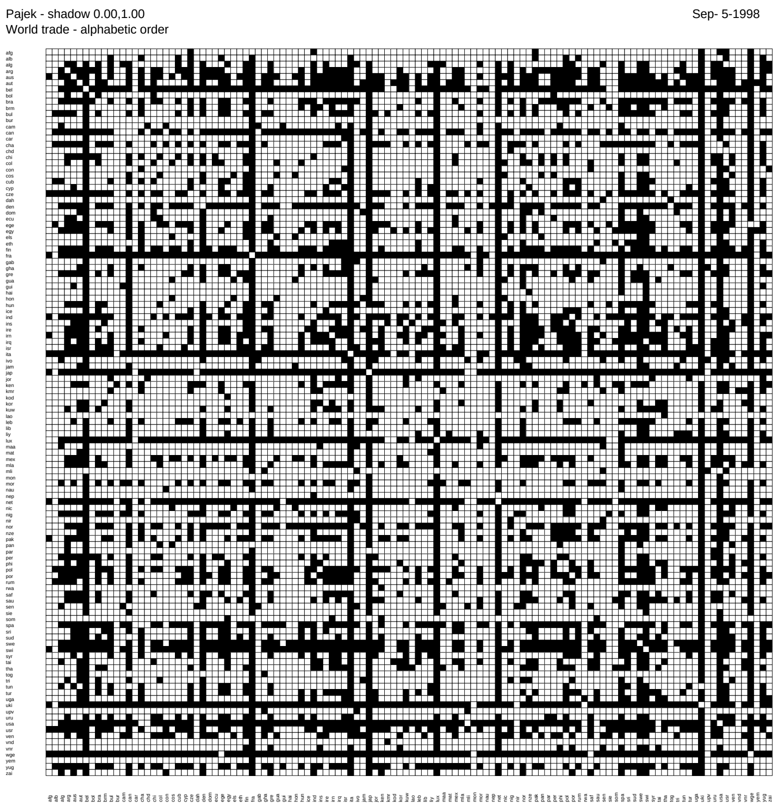
University of Ljubljana

Slovenia

Padova, April 10-11, 2003

Outline

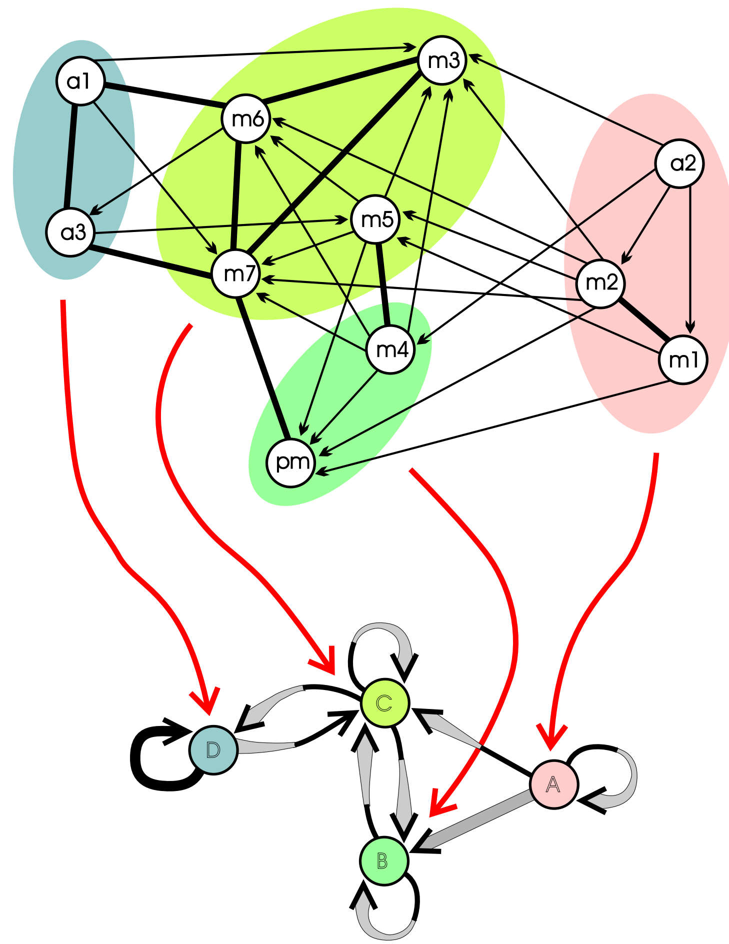
Introduction to Blockmodeling



Alphabetic order of countries (left) and rearrangement (right)

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.

Equivalences

Regardless of the definition of equivalence used, there are two basic approaches to the equivalence of units in a given network (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 (Lorrain and White, 1971) and the second by the notion of *regular* equivalence (White and Reitz, 1983) with the latter a generalization of the former.

Structural Equivalence

Units are *structurally equivalent* if they are connected to the rest of the network in identical ways. From the definition it follows that there are four possible ideal blocks:

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	1

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

0	1	1	1
1	0	1	1
1	1	0	1
1	1	1	0

Regular Equivalence

Intuitively, two units are *regularly equivalent* if they are equally connected to equivalent others.

Batagelj, Doreian, and Ferligoj (1992) proved that regular equivalence produces two types of blocks: null blocks and 1-covered blocks:

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

1	0	1	0	0
0	0	1	0	1
0	1	0	0	0
1	0	1	1	0

Generalized equivalence / Block Types

complete (X × Y)

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	1	1	1

row-dominant (X × Y)

0	1	0	0	0
1	1	1	1	1
0	0	0	0	0
0	0	0	1	0

col-dominant (X × Y)

0	0	1	0	0
0	0	1	1	0
1	1	1	0	0
0	0	1	0	1

regular (X × Y)

0	1	0	0	0
1	0	1	1	0
0	0	1	0	1
1	1	0	0	0

row-regular (X × Y)

0	1	0	0	0
0	1	1	0	0
1	0	1	0	0
0	1	0	0	1

col-regular (X × Y)

0	1	0	1	0
1	0	1	0	0
1	1	0	1	1
0	0	0	0	0

null (X × Y)

0	0	0	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

row-functional (X × Y)

0	0	0	1	0
0	0	1	0	0
1	0	0	0	0
0	0	0	1	0

col-functional (X × Y)

1	0	0	0
0	1	0	0
0	0	1	0
0	0	0	0
0	0	0	1

Establishing Blockmodels

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 (Φ, P) as follows:

Determine the clustering $\mathbf{C}^\star \in \Phi$ for which

$$P(\mathbf{C}^\star) = \min_{\mathbf{C} \in \Phi} P(\mathbf{C})$$

where Φ is the set of *feasible clusterings* and P is a *criterion function*.

Since the set of units $\mathbf{U}$ is finite, the set of feasible clusterings is also finite. Therefore the set $\mathrm{Min}(\Phi, P)$ of all solutions of the problem (optimal clusterings) is not empty.

Criterion function

Criterion functions can be constructed

- *indirectly* as a function of a compatible (dis)similarity measure between pairs of units, or
- *directly* as a function measuring the fit of a clustering to an ideal one with perfect relations within each cluster and between clusters according to the considered types of connections (equivalence).

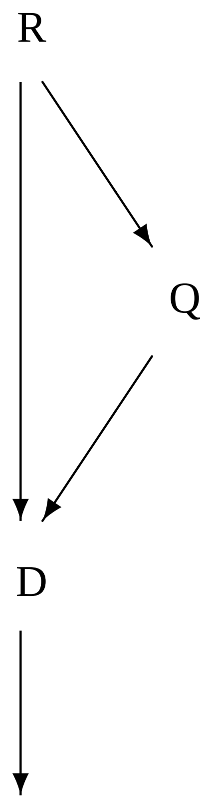
Indirect Approach

RELATION

DESCRIPTIONS OF UNITS

DISSIMILARITY MATRIX

STANDARD CLUSTERING ALGORITHMS



original relation

path matrix

triads

orbits

hierarchical algorithms,

relocation algorithm, leader algorithm

Dissimilarities

The dissimilarity measure d is *compatible* with a considered equivalence $\sim$ if for each pair of units holds

$$X_i \sim X_j \Leftrightarrow d(X_i, X_j) = 0$$

Not all dissimilarity measures typically used are compatible with structural equivalence. For example, the *corrected Euclidean-like dissimilarity*

$$d(X_i, X_j) =$$

$$= \sqrt{(r_{ii} - r_{jj})^2 + (r_{ij} - r_{ji})^2 + \sum_{\substack{s=1 \\ s \neq i,j}}^{n} ((r_{is} - r_{js})^2 + (r_{si} - r_{sj})^2)}$$

is compatible with structural equivalence.

The indirect clustering approach does not seem suitable for establishing clusterings in terms of regular equivalence since there is no evident way how to construct a compatible (dis)similarity measure.

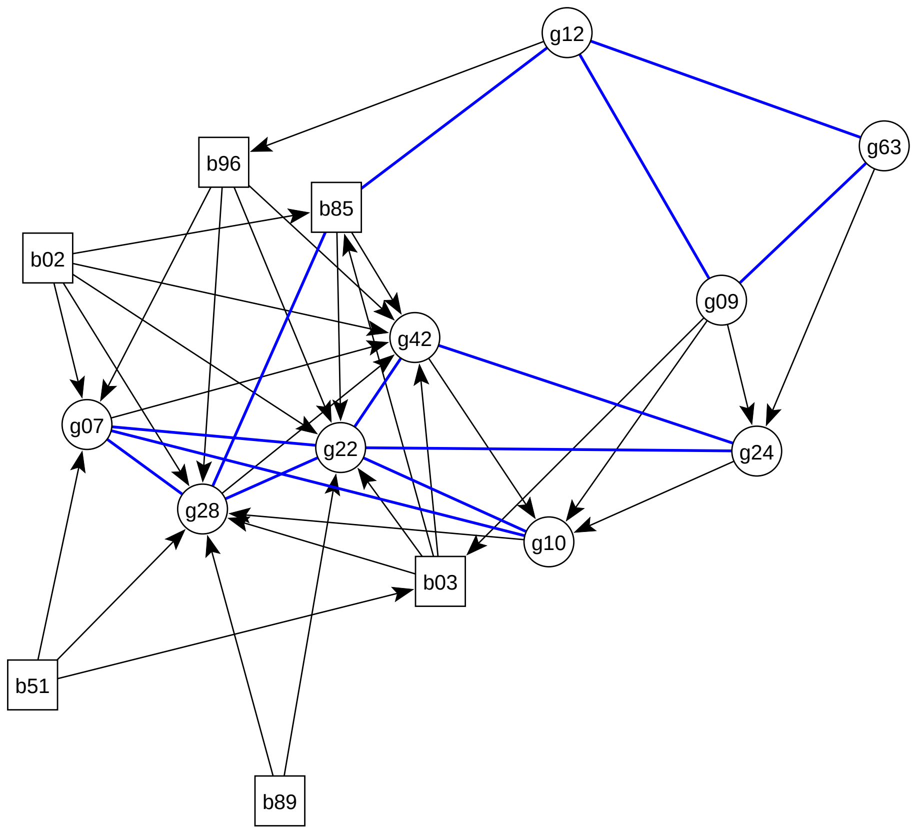
Example

The analyzed network consists of social support exchange relation among fifteen students of the Social Science Informatics fourth year class (2002/2003) at the Faculty of Social Sciences, University of Ljubljana. Interviews were conducted in October 2002.

Support relation among students was identified by the following question:

> Introduction: You have done several exams since you are in the second class now. Students usually borrow studying material from their colleagues.
>
> Enumerate (list) the names of your colleagues that you have most often borrowed studying material from. (The number of listed persons is not limited.)

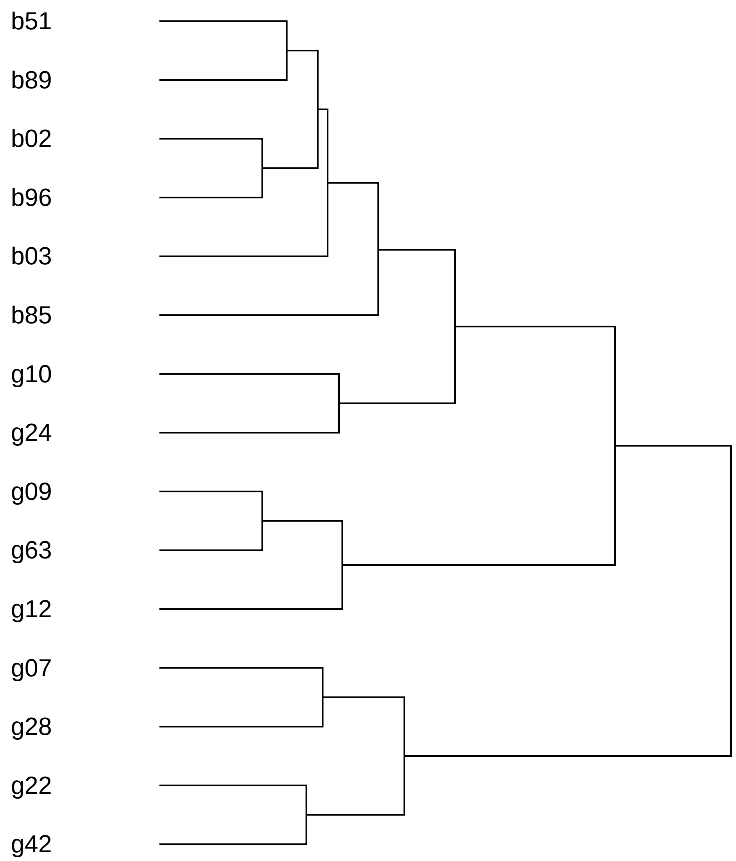
Class network



`class.net`

Vertices represent students in the class; circles – girls, squares – boys. Opposite pairs of arcs are replaced by edges.

Indirect approach



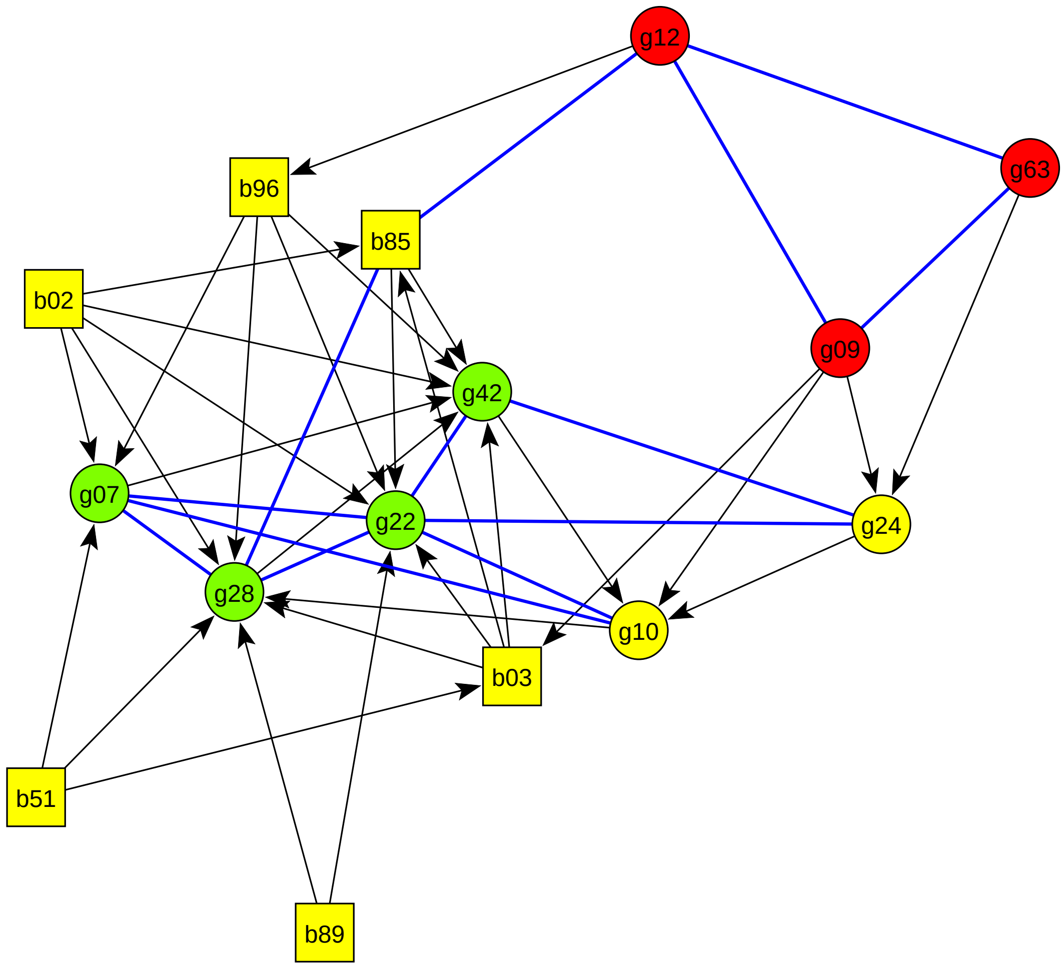
Using *Corrected Euclidean-like dissimilarity* and *Ward clustering method* we obtain the following dendrogram.

From it we can determine the number of clusters: 'Natural' clusterings correspond to clear 'jumps' in the dendrogram.

If we select 3 clusters we get the partition **C**.

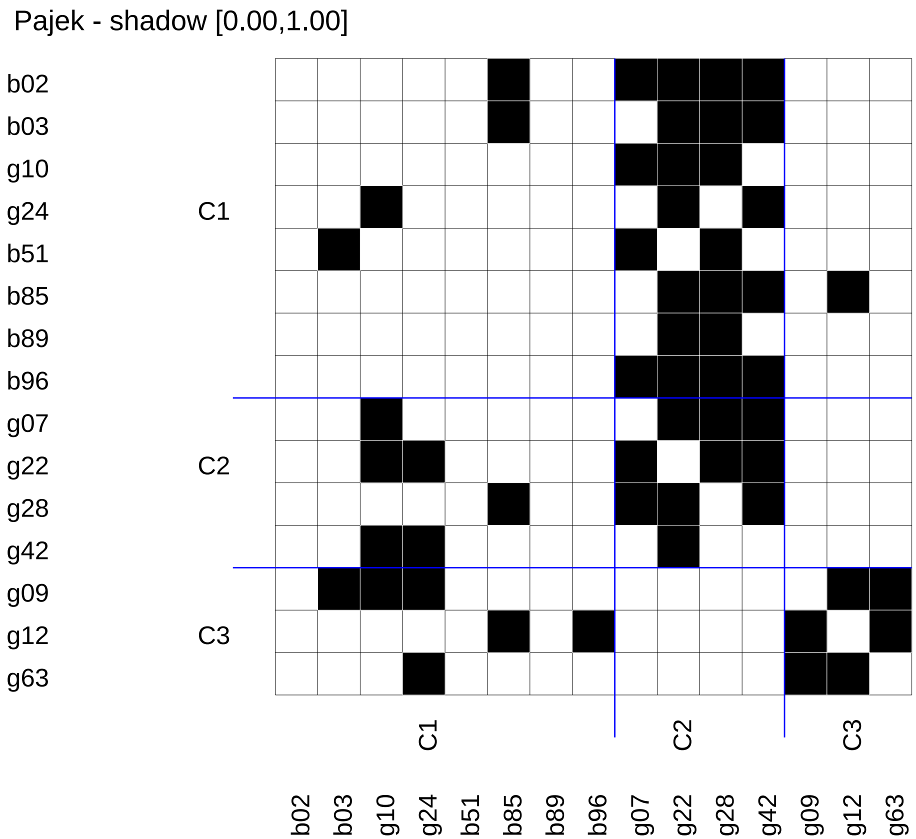
$$\mathbf{C} = \{\{b51, b89, b02, b96, b03, b85, g10, g24\}, \{g09, g63, g12\}, \{g07, g28, g22, g42\}\}$$

Partition in 3 clusters



On the picture, vertices in the same cluster are of the same color.

Matrix



The partition can be used also to reorder rows and columns of the matrix representing the network. Clusters are divided using blue vertical and horizontal lines.

Direct Approach

The second possibility for solving the blockmodeling problem is to construct an appropriate criterion function directly and then use a local optimization algorithm to obtain a ‘good’ clustering solution.

Criterion function $P(\mathbf{C})$ has to be *sensitive* to considered equivalence:

$$P(\mathbf{C}) = 0 \Leftrightarrow \mathbf{C} \text{ defines considered equivalence.}$$

A 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. The function d has to be compatible with the selected type of equivalence.

Local Optimization

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

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

Pre-specified blockmodeling

In the previous slides the inductive approaches for establishing blockmodels for a set of social relations defined over a set of units were discussed. Some form of equivalence is specified and clusterings are sought that are consistent with a specified equivalence.

Another view of blockmodeling is deductive in the sense of starting with a blockmodel that is specified in terms of substance prior to an analysis.

In this case given a network, set of types of ideal blocks, and a reduced model, a solution (a clustering) can be determined which minimizes the criterion function.

Some Blockmodel Types

1. *Cohesive subgroups* with intraposition ties and no ties between positions:

1	0	0
0	1	0
0	0	1

2. A *center-periphery* model with one *core* position (center) which is internally cohesive and connected with all other positions. The other positions (periphery) are all connected to the core position and are not connected to each other:

1	1	1
1	0	0
1	0	0

... Some Blockmodel Types

3. A *centralized* model which, a special case of the center-periphery model, where all ties are from the core position or toward it:

1	1	1
0	0	0
0	0	0

or

1	0	0
1	0	0
1	0	0

4. A *hierarchical* model with the positions on a single path:

0	1	0
0	0	1
0	0	0

or

0	0	0
1	0	0
0	1	0

. . . Some Blockmodel Types

5. A *transitivity* model similar to hierarchical model with permitted ties from lower positions to all higher positions:

0	1	1
0	0	1
0	0	0

or

0	0	0
1	0	0
1	1	0

Prespecified blockmodeling example

We expect that center-periphery model exists in the network: some students having good studying material, some not.

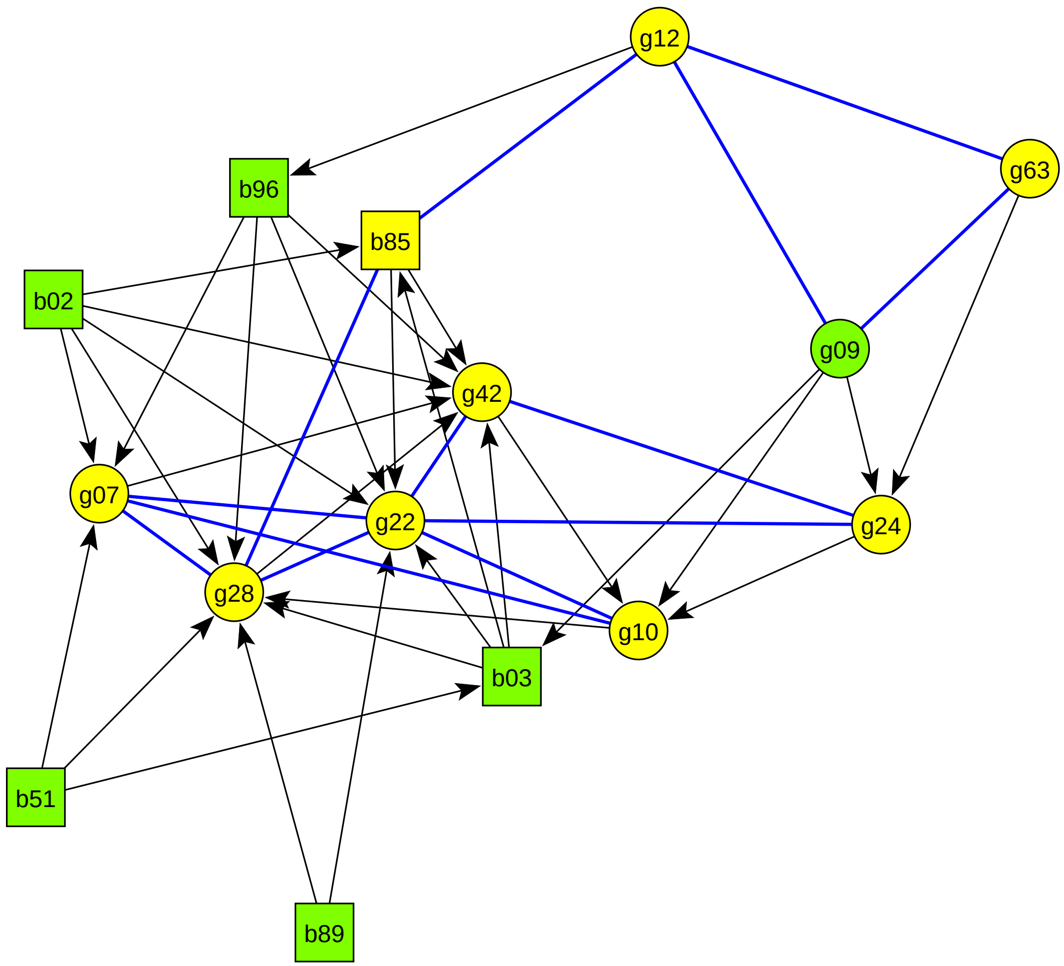
Prespecified blockmodel: (com/complete, reg/regular, -/null block)

	1	2
1	[com reg]	-
2	[com reg]	-

Using local optimization we get the partition:

$$\mathbf{C} = \{\{b02, b03, b51, b85, b89, b96, g09\}, \{g07, g10, g12, g22, g24, g28, g42, g63\}\}$$

2 clusters solution



Model

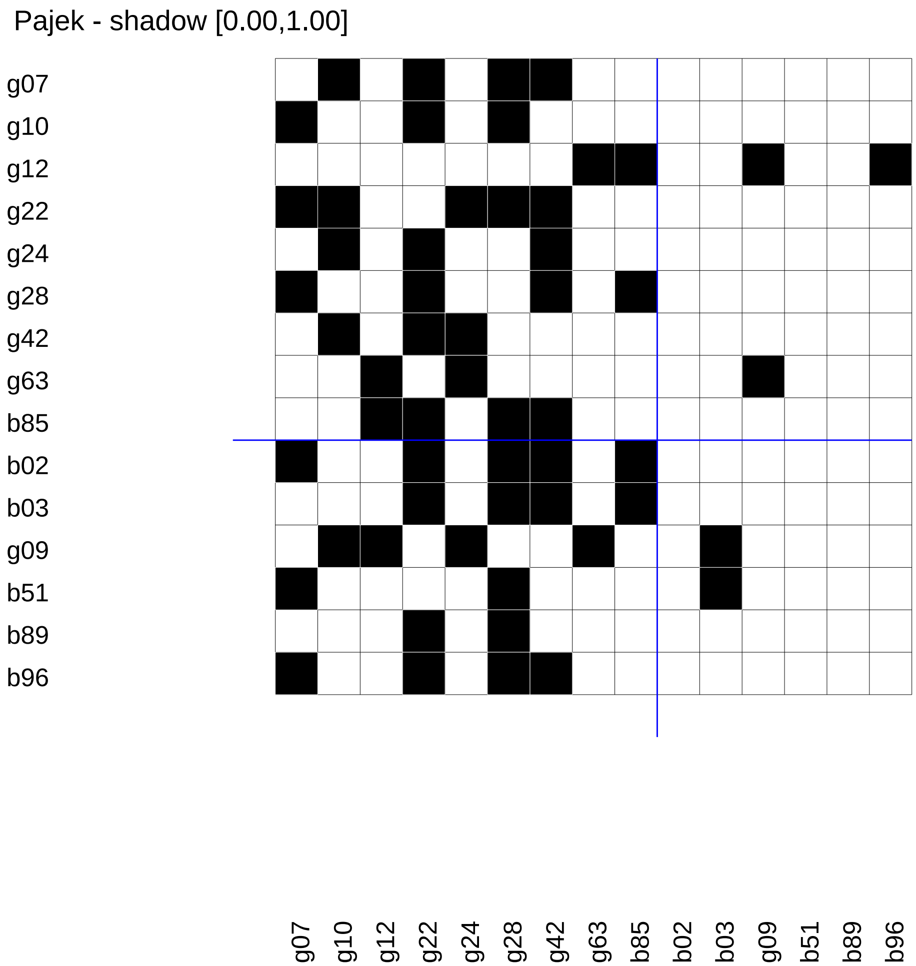


Image and Error Matrices:

	1	2
1	reg	-
2	reg	-

	1	2
1	0	3
2	0	2

Total error = 5

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; . . .

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).

The generalized blockmodeling is 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/