Social Networks 40 (2015) 1–16

Contents lists available at ScienceDirect

Social Networks

journal homepage: www.elsevier.com/locate/socnet

Simultaneous-direct blockmodeling for multiple relations in Pajek

Matthew Dabkowski[a,*], Ronald Breiger[b], Ferenc Szidarovszky[c]

[a] Department of Systems and Industrial Engineering, University of Arizona, Tucson, United States
[b] School of Sociology, University of Arizona, Tucson, United States
[c] Reliasoft Corporation, 1450 South Eastside Loop, Tucson, United States

ARTICLE INFO

Keywords:
Blockmodeling
Multiple relations
Social position
Pajek

ABSTRACT

The foundational research on blockmodeling focused on theorizing and identifying social roles and positions across multiple networks (White et al., 1976). Generalized blockmodeling provided a breakthrough in theory and research by permitting ideal block types that implement a wider class of role equivalence within a network (Doreian et al., 2005). Notwithstanding these successes and related progress that we discuss, a direct approach for the blockmodeling of multiple relations remains an open problem in the generalized blockmodeling literature (Doreian, 2006). With this in mind, we propose a simple and novel means of formulating and fitting generalized blockmodels for multiple relations. We make use of existing capabilities of the open-source network analysis software Pajek (Batagelj and Mrvar, 2011; Mrvar and Batagelj, 2013). In particular, by constructing an appropriate augmented adjacency matrix and carefully crafted constraints and penalties, Pajek's criterion function can be simultaneously minimized over multiple relations. This technique is first described in detail using a hypothetical friendship network, and then its value is reinforced through reanalysis of a classic, real world example.

Published by Elsevier B.V.

1. Introduction and background

Consider a social network consisting of $i = 1, \ldots, n$ actors and $j = 1, \ldots, m$ relations. Moreover, assume the tie between any two actors on a given relation (a one mode network) is dichotomous, where 1 and 0 indicate the presence and absence of a tie respectively. As such, our social network can be represented as m $(n \times n)$ binary, not necessarily symmetric, sociomatrices. From this relational data on individual actors across multiple relations, we are interested in testing a hypothesized structure of social positions. Known as a blockmodel, this structure partitions (or clusters) actors into non-overlapping subsets (positions) which are linked by a pattern of ties (Wasserman and Faust, 1994: p. 395). More formally, given $k = 1, \ldots, l$ positions, a blockmodel consists of m $(l \times l)$ image matrices, where the image matrix for relation j defines the social structure between positions on relation j.

Based on the above, blockmodeling multiple relations is the task of (1) developing image matrices for m relations and (2) assigning actors to l positions. Ideally, the distribution of individual ties matches the blockmodel; however, reality is rarely this kind. In particular, the blockmodel may be misspecified, and/or the data may contain errors. Moreover, even if the blockmodel is reasonably accurate and the data relatively error free, the task of optimally permuting actors into positions that transcend relations is daunting, as blockmodeling even a single relation is NP hard (Chan et al., 2013). Technically, this means that, in the worst case, blockmodeling problems are not solvable in polynomial time. Practically, this implies that finding solutions is inherently slow, and the situation becomes progressively, explosively worse as the number of actors increases.

Given this reality, it is not surprising that blockmodeling multiple relations (especially generalized blockmodeling) remains an open problem (Batagelj et al., 2004: p. 466; Doreian et al., 2005: pp. 356–357; Doreian, 2006). Nonetheless, recent work by Brusco, Doreian, Steinley, and Satornino makes significant strides in blockmodeling multiple relations (2013). In particular, by applying multiobjective tabu search, their algorithm generates an approximate Pareto set of locally optimal blockmodels, which are subsequently evaluated to settle on a "best" solution(s). That said, their code is written in FORTRAN 90 and, therefore, inaccessible to a vast majority of social scientists.

* Corresponding author at: Department of Systems and Industrial Engineering, University of Arizona, 1127 E. James E. Rogers Way, Room 111, P.O. Box 210020, Tucson, AZ 85721-0020, United States. Tel.: +1 520 333 3462; fax: +1 520 621 6555; mobile: +1 520 248 6130.

E-mail addresses: mfd1@email.arizona.edu, matthew.dabkowski@gmail.com (M. Dabkowski), breiger@email.arizona.edu (R. Breiger), szidar@sie.arizona.edu (F. Szidarovszky).

http://dx.doi.org/10.1016/j.socnet.2014.06.003
0378-8733/Published by Elsevier B.V.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s16	s17	s18	s19	s20
s1	0	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0
s2	0	0	0	1	0	1	1	0	0	1	0	0	0	0	1	0	0	0	1	0
s3	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	1
s4	0	1	0	0	0	1	1	0	0	1	0	0	0	0	1	0	0	0	1	0
s5	1	0	0	0	0	0	0	0	0	0	0	1	1	0	0	0	1	0	0	0
s6	0	1	0	1	0	0	1	0	0	1	0	0	0	0	1	0	0	0	1	0
s7	0	1	0	1	0	1	0	0	0	1	0	0	0	0	1	0	0	0	1	0
s8	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	1	0	1	0	0
s9	0	0	1	0	0	0	0	0	0	0	0	0	0	1	0	0	0	0	0	1
s10	0	1	0	1	0	1	1	0	0	0	0	0	0	0	1	0	0	0	1	0
s11	0	0	0	0	0	0	0	1	0	0	0	0	0	0	0	1	0	1	0	0
s12	1	0	0	0	1	0	0	0	0	0	0	0	1	0	0	0	1	0	0	0
s13	1	0	0	0	1	0	0	0	0	0	0	1	0	0	0	0	1	0	0	0
s14	0	0	1	0	0	0	0	0	1	0	0	0	0	0	0	0	0	0	0	1
s15	0	1	0	1	0	1	1	0	0	1	0	0	0	0	0	0	0	0	1	0
s16	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	0	0	1	0	0
s17	1	0	0	0	1	0	0	0	0	0	0	1	1	0	0	0	0	0	0	0
s18	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	0	0	0
s19	0	1	0	1	0	1	1	0	0	1	0	0	0	0	1	0	0	0	0	0
s20	0	0	1	0	0	0	0	0	1	0	0	0	0	1	0	0	0	0	0	0

Fig. 1. Hypothetical friendship network for a classroom of 20 third-graders.

	B1	B2	B3	B4
B1	1	0	0	0
B2	0	1	0	0
B3	0	0	1	0
B4	0	0	0	1

Fig. 2. Ideal blockmodel for our hypothetical friendship network.

The primary contribution of this paper is to present an alternative, simplified approach to blockmodeling multiple relations using the existing capabilities of the network analysis software Pajek. In particular, by constructing an appropriate augmented adjacency matrix and carefully crafted constraints and penalties, Pajek's criterion function can be simultaneously minimized over multiple relations. Furthermore, while this approach can be narrowly categorized as deterministic, one-mode, unsigned, and confirmatory (using the taxonomy of Brusco and Steinley, 2011), Pajek's ability to (1) handle two-mode data, (2) partition signed networks, and (3) perform exploratory fitting suggests future refinements and broader potential.

2. A motivating example

2.1. Blockmodeling a single relation

Imagine a hypothetical classroom of 20 third-graders where students are asked to identify their friends from the class roster. The responses from each of the 20 students are subsequently assembled into the (20×20) sociomatrix seen in Fig. 1 (the "current friend" sociomatrix), and we are interested in testing our suspicion that the class is clustered into small, cohesive subgroups. Furthermore, suppose by observing the students at recess, we suspect there are four clusters of friends. Finally, from the cliquish behavior of the students, we hypothesize that each cluster of friends will have a dense concentration of friendship ties inside the cluster but few friendship ties outside of it. Taken together, the image matrix in Fig. 2 represents our hypothesized blockmodel, and (by design) the rows and columns of the sociomatrix in Fig. 1 can be permuted to produce the equivalent sociomatrix in Fig. 3.

From this rearrangement of the students, our hypothesized structure is confirmed. Specifically, there are four clusters of friends, and the clusters have friendship ties only inside of themselves. Additionally, from Fig. 3, our position or block membership is B_1 = {s1, s5, s12, s13, s17}, B_2 = {s8, s11, s16, s18}, B_3 = {s2, s4, s6, s7, s10, s15, s19}, and B_4 = {s3, s9, s14, s20}.

	s1	s5	s12	s13	s17	s8	s11	s16	s18	s2	s4	s6	s7	s10	s15	s19	s3	s9	s14	s20
s1	0	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s5	1	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s12	1	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s13	1	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s17	1	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s8	0	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0
s11	0	0	0	0	0	1	0	1	1	0	0	0	0	0	0	0	0	0	0	0
s16	0	0	0	0	0	1	1	0	1	0	0	0	0	0	0	0	0	0	0	0
s18	0	0	0	0	0	1	1	1	0	0	0	0	0	0	0	0	0	0	0	0
s2	0	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0
s4	0	0	0	0	0	0	0	0	0	1	0	1	1	1	1	1	0	0	0	0
s6	0	0	0	0	0	0	0	0	0	1	1	0	1	1	1	1	0	0	0	0
s7	0	0	0	0	0	0	0	0	0	1	1	1	0	1	1	1	0	0	0	0
s10	0	0	0	0	0	0	0	0	0	1	1	1	1	0	1	1	0	0	0	0
s15	0	0	0	0	0	0	0	0	0	1	1	1	1	1	0	1	0	0	0	0
s19	0	0	0	0	0	0	0	0	0	1	1	1	1	1	1	0	0	0	0	0
s3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1
s9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	1
s14	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	0	1
s20	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	1	1	0

Fig. 3. Permuted sociomatrix for our hypothetical friendship network.

	s1	s2	s3	s4	s5	s6	s7	s8	s9	s10	s11	s12	s13	s14	s15	s16	s17	s18	s19	s20
s1	0	0	1	0	0	0	0	1	1	0	1	0	0	1	0	1	0	1	0	1
s2	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s3	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0
s4	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s5	0	0	1	0	0	0	0	1	1	0	1	0	0	1	0	1	0	1	0	1
s6	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s7	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s9	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0
s10	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s11	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s12	0	0	1	0	0	0	0	1	1	0	1	0	0	1	0	1	0	1	0	1
s13	0	0	1	0	0	0	0	1	1	0	1	0	0	1	0	1	0	1	0	1
s14	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0
s15	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s16	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s17	0	0	1	0	0	0	0	1	1	0	1	0	0	1	0	1	0	1	0	1
s18	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0
s19	1	0	1	0	1	0	0	1	1	0	1	1	1	1	0	1	1	1	0	1
s20	0	0	0	0	0	0	0	1	0	0	1	0	0	0	0	1	0	1	0	0

Fig. 4. Hypothetical sociomatrix generated in response to the question "Who would you like to become friends with?".

	B1	B2	B3	B4
B1	1	0	0	0
B2	0	1	0	0
B3	0	0	1	0
B4	0	0	0	1

	B1	B2	B3	B4
B1	0	0	0	0
B2	1	0	0	0
B3	1	1	0	0
B4	1	1	1	0

Fig. 5. Ideal blockmodel for our hypothetical friendship network. The image matrices for the "current friends" and "desired friends" networks are given on the left and right respectively, where block popularity decreases from B1 to B4.

2.2. Blockmodeling with multiple relations

Extending our example above, suppose we also asked each student to identify other students they would like to become friends with. Once again, the responses from each of the 20 students are assembled, yielding the (20×20) sociomatrix seen in Fig. 4 (the "desired friend" sociomatrix).

In this case, we might hypothesize that a student would want to become friends with students who are perceived as being more popular than him or herself. Additionally, if we suspect that students of similar popularity are friends with one another, then the image matrices in Fig. 5 describe our hypothesized blockmodel for the two relations.

As before, the rows and columns of the sociomatrix in Fig. 4 can be permuted to produce the equivalent sociomatrix in Fig. 6.