Pattern Recognition 111 (2021) 107641

Contents lists available at ScienceDirect

Pattern Recognition

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

On finite mixture modeling and model-based clustering of directed weighted multilayer networks

Volodymyr Melnykov[a,*], Shuchismita Sarkar[b], Yana Melnykov[a]

[a] The University of Alabama, Tuscaloosa, AL 35487, USA
[b] Bowling Green State University, Bowling Green, OH 43402, USA

ARTICLE INFO

Article history:
Received 12 May 2019
Revised 29 October 2019
Accepted 6 September 2020
Available online 18 September 2020

Keywords:
Model-based clustering
Directed network
Weighted network
Multilayer network
MCMC

ABSTRACT

A novel approach relying on the notion of mixture models is proposed for modeling and clustering directed weighted networks. The developed methodology can be used in a variety of settings including multilayer networks. Computational issues associated with the developed procedure are effectively addressed by the use of MCMC techniques. The utility of the methodology is illustrated on a set of experiments as well as applications to real-life data containing export trade amounts for European countries.

1. Introduction

The use of networks describing pairwise relationships between objects is very popular in social, organizational, and economic studies. The objects under consideration are typically called nodes or actors and connections between them are known as edges or dyads. The types of edges can be generally divided into directional and nondirectional. The latter assumes that the relationship between the nodes does not involve any directional information. On the contrary, there is an abundance of networks such as email, web, or trading ones that take into account the direction of edges. If the magnitude of characteristics represented by pairwise connections varies from pair to pair, there is a common practice to assign weights to the edges yielding so-called weighted networks. The number of emails sent between two peers or the amount of export between two countries can serve as examples of such weights. Moreover, it is not uncommon to observe multiple relationships between the nodes. Such networks with multivariate edges can be represented by multigraphs and are often referred to as multilayer networks. There is a wide range of literature discussing the methodology devoted to the network analysis. One of the earliest studies relying on the fundamental properties of random graphs can be found in [25]. For an extensive overview of various existing techniques, we refer the reader to [48] and [61].

The objective of cluster analysis is to partition objects in such a way that objects within groups, generally referred to as clusters, have similar characteristics but groups are relatively distinct. There is a large segment of network literature devoted to clustering nodes. While there are several definitions of network clusters, the most common one assumes that a cluster is the set of nodes connected with each other in a similar way. Separate clusters, also known in this context as blocks or communities, exhibit between-cluster connections in a way different from within-cluster connections to ensure that blocks are properly defined. In the most traditional case of undirected networks, the absence or presence of a certain edge between two nodes is coded by a binary indicator. A deterministic clustering algorithm employing this information was discussed by White et al. [63]. An extension of this idea to probabilistic settings was considered by Fienberg and Wasserman [26] as well as [32] and yield the stochastic block model [33] that became immensely popular for the analysis of relational data. Further developments of this model were proposed by Wang and Wong [60] who considered it in the directed graph framework as well as [54] who focused on unobserved or latent block structure. The use of exponential family distributions for modeling block components was discussed by [27,53,55,62]. Some recent studies proposed methods allowing an infinite number of blocks [34], mixed membership labels [3], hierarchical structure [15], edge-weighted networks [1], and use of random graphs for clustering [64,66]. An excellent review of the variants of stochastic block model can be found in [28]. These papers primarily con-

* Corresponding author.
E-mail address: vmelnykov@ua.edu (V. Melnykov).

https://doi.org/10.1016/j.patcog.2020.107641
0031-3203/

sider unilayer binary undirected graphs which limits their application beyond this framework.

As noted by Kivelä et al. [36], "most real and engineered systems include multiple subsystems and layers of connectivity, and developing a deep understanding of multilayer systems necessitates generalizing "traditional" network theory." In a multilayer network, nodes are connected to each other by multivariate relationships. An extensive survey of multilayer networks can be found in [36]. Some approaches to the problem rely on tensor decomposition [23,37]. The application of exponential random graph models with the focus on bilayer networks was investigated in [59]. The hidden structure of multilayer, weighted, time-varying networks was studied in [50]. However, the traditional approach to partitioning multilayer networks relies on converting multivariate edges to one-dimensional representations by finding an appropriate univariate projection [20], aggregating unilayer information [6], or by flattening [7]. Just one of several problems related to such a strategy is that some layers might be more informative than the others. This concern gives rise to the development of methods that aim to alleviate this issue by assessing the importance of variables [22]. In another useful survey of clustering attributed graphs by Bothorel et al. [11], the authors acknowledge that just few developments have been proposed for multilayer graphs. They mention an interesting work by Boden et al. [10] who consider subspace clustering. The method, however, has a limitation that it aims at finding a set of nodes forming a cluster in each layer separately. The reduction of dimensionality can be achieved in various ways but important information is likely to be lost no matter what strategy is chosen [36]. Moreover, as stated in [36], "an important theme that has developed in the literature is the importance of multiplexity-induced correlations (and its analogs in multilayer structures more generally), which can have important ramifications for dynamical processes on networks." The methodology proposed in this paper aims at modeling network layers jointly and thus allows clustering multilayer directed weighted networks more effectively. Another notable approach in this direction is proposed by Barbillon et al. [5], where an extension of a stochastic block model to multiplex networks has been studied and the dependence between the layers is preserved. The considered framework, however, concerns unweighted networks and hence is quite different from the scope of the problem investigated in this paper.

The application of finite mixture models to cluster analysis, also known as model-based clustering [4], shows remarkable modeling flexibility and ability to produce excellent clustering results. Finite mixture model is a convex combination of several probability distributions, also known as mixture components. The distribution can assume any functional form including discrete [12] and continuous, among which multivariate Gaussian density [67,68] is one of the most popular in literature. Sometimes, data are recorded in matrix form and matrix normal distribution [19] is an efficient tool for modeling such data. For estimating the parameters of the associated mixture component, expectation-maximization (EM) algorithm [21,40] is usually employed. If the number of mixture components, commonly referred to as a mixture order, is unknown, the most popular ways of estimating it are based on either Bayesian information criterion (BIC) [52] or integrated completed likelihood (ICL) criterion [8]. A recent notable work by Côme and Latouche [16] is focused on detecting the correct number of blocks in a stochastic block model by considering exact integrated complete likelihood. In this paper, the authors have used a BIC-based approach known for being consistent in detecting the true mixture order given that the model specification is correct [17,35].

The traditional mixture modeling framework assumes independent identically distributed observations and the nature of network data violates this assumption. Another issue is related to the size of network data. In a directed graph with n nodes there are $n(n-1)/2$ possible pairs of edges (if loops are not allowed). High cardinality along with the interdependence of the data pose strong challenges for obtaining accurate model parameter estimates in a computationally efficient way. There have been a number of papers aiming at incorporating mixture models into network analysis. The application of mixtures to modeling edges was discussed by Nowicki and Snijders [49] as well as [18]. The parameter estimation issue was discussed by Zanghi et al. [65] who explored the use of stochastic approximation EM algorithm as well as variational methods [2,30,38] for network mixtures. Both approaches proved to be useful for modeling web networks and assigning membership labels to new nodes based on the information collected from the previously encountered nodes. Vu et al. [58] developed a procedure relying on the variational generalized EM algorithm, where the minorization-maximization algorithm was employed at the expectation step.

In this paper, we consider clustering directed weighted networks, with a special focus on the multilayer setting. It is important to distinguish two different types of weighted networks. Suppose an airline connection network is considered with nodes representing airports and edges reflecting time required to get from one airport to another with a direct flight. Clearly, not all airports are connected directly and hence some edges do not exist. On the contrary, consider a correspondence communication network for a large corporation. The nodes represent corporation departments and edges stand for the number of emails sent between them in each direction. In this case, no correspondence in some directions does not imply that the edge does not exist but rather suggests that a zero value is associated with the edge. These two scenarios are fundamentally different as the latter case presents a complete graph with all edges present. In this paper, we develop a general framework for cluster analysis in the complete graph setting, although the case with non-existing edges is briefly discussed as well. As network cluster analysis can become computationally prohibitive even for a rather low number of nodes, we propose an MCMC-based approach employing the Metropolis-Hastings algorithm [31,46]. The real-life data set that illustrates the developed methodology involves pairwise export trades for 39 European countries. The data include overall trade amounts as well as exports organized by specific categories.

In Section 2, we discuss the methodology for clustering directed weighted networks. In Section 3, we illustrate the effectiveness of the proposed procedure on synthetic data. Section 4 provides the analysis of the trades data set in unilayer and multilayer cases. Finally, the paper is concluded with a discussion in Section 5.

2. Methodology

2.1. Framework for clustering directed weighted network

Although there are several formulations of a cluster in the network analysis framework, in this paper a cluster is defined as a group of nodes that exhibit similar characteristics of connections between them. More specifically, it is assumed that the weights of connecting edges within a cluster follow the same distribution. Between-cluster connections must be sufficiently different from within-cluster ones to ensure the proposed definition of a cluster. This implies that between-cluster connections follow another distribution. The proposed formulation resembles that of the stochastic block model.

Suppose there is a sample of n nodes taken from some heterogeneous directed weighted network population with complete graph structure and K subpopulations. Here n is considered to be known and fixed. Nodes originating from the kth subpopulation form a cluster in the observed sample. Let π_k represent the proportion of nodes that belong to the kth subpopulation. Then, the

probability of observing a directed edge that connects a node in subpopulation k to a node in subpopulation k' is $\pi_k\pi_{k'}$. As the complete graph assumption is used, the total number of edge pairs is $n(n-1)/2$. Let $\boldsymbol{y}_{ij} = (y_{ij}, y_{ji})^\top$ be a bivariate vector representing the weights associated with the pair of edges between nodes i and j, where $i < j$. It is immediate to see that different orderings of nodes produce inequivalent data sets as the order of elements in vectors $\boldsymbol{y}_{ij}$ depends on the order of nodes. For instance, consider two alternative orderings of three nodes: $\mathcal{O}_1 = \{1, 2, 3\}$ and $\mathcal{O}_2 = \{1, 3, 2\}$. It follows that under $\mathcal{O}_j$, the data are given by vectors $\boldsymbol{y}_{12}^{(j)}$, $\boldsymbol{y}_{13}^{(j)}$, and $\boldsymbol{y}_{23}^{(j)}$, where $j = 1, 2$. It is immediate to see that $\boldsymbol{y}_{12}^{(2)} = \boldsymbol{y}_{13}^{(1)}$, $\boldsymbol{y}_{13}^{(2)} = \boldsymbol{y}_{12}^{(1)}$, and $\boldsymbol{y}_{23}^{(2)} = \boldsymbol{E}\boldsymbol{y}_{23}^{(1)}$, where $\boldsymbol{E} = \text{antidiag}\{1, 1\}$ is the so-called exchange matrix containing ones on the antidiagonal and zeros elsewhere. The difference between the two data representations is in the order of edges between the nodes 2 and 3. Thus, the proposed methodology must be invariant to particular ways of ordering nodes.

Let $\boldsymbol{s} = (s_1, \ldots, s_n)^\top \in \mathcal{S}$ represent a specific sequence of node cluster membership labels $s_i \in \{1, \ldots, K\}$, where $\mathcal{S}$ is the space of possible orderings of node memberships. Although the objective of cluster analysis is to partition nodes, a mixture model will be applied to the network edges. Let the sequence of edge membership labels corresponding to $\boldsymbol{s}$ be denoted as $\boldsymbol{e}_{\boldsymbol{s}}$. Observations $\boldsymbol{y}_{ij}$ within a sample are not independent and the traditional formulation of mixture models in *iid* case is not applicable. Instead, we focus on the joint distribution of the data set $\underline{\boldsymbol{y}} = (\boldsymbol{y}_{ij})_{i<j}$ that is given by

$$g(\underline{\boldsymbol{y}}; \boldsymbol{\Theta}) = \sum_{\boldsymbol{s}\in\mathcal{S}} \mathbb{P}\text{r}(\boldsymbol{e}_{\boldsymbol{s}}; \boldsymbol{\Theta}) f_{\underline{\boldsymbol{y}}}(\underline{\boldsymbol{y}}|\boldsymbol{e}_{\boldsymbol{s}}; \boldsymbol{\Theta}), \tag{2.1}$$

where $\boldsymbol{\Theta}$ is the set of model parameters. Assuming conditional independence of edges given the membership labels, Eq. (2.1) can be formulated as

$$g(\underline{\boldsymbol{y}}; \boldsymbol{\Theta}) = \sum_{\boldsymbol{s}\in\mathcal{S}} \prod_{k=1}^{K}\prod_{k'=1}^{K}\prod_{i=1}^{n-1}\prod_{j=i+1}^{n} \left[\pi_k\pi_{k'} f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})\right]^{I(s_i=k, s_j=k')}, \tag{2.2}$$

where $f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})$ is the conditional distribution of $\boldsymbol{y}_{ij}$ with parameter vector $\boldsymbol{\theta}_{kk'}$. The proposed formulation assumes K components with parameters $\boldsymbol{\theta}_{kk}$ for modeling within-cluster edges and $K(K-1)$ components with parameters $\boldsymbol{\theta}_{kk'}$, $k \neq k'$, for modeling between-cluster directed edges. It can be noticed that if $\boldsymbol{y}_{ij} \sim f(\boldsymbol{y}; \boldsymbol{\theta}_{k'k})$, it follows that $\boldsymbol{E}\boldsymbol{y}_{ij} \sim f(\boldsymbol{y}; \boldsymbol{\theta}_{kk'})$. The specific functional form of the relationship between $\boldsymbol{\theta}_{kk'}$ and $\boldsymbol{\theta}_{k'k}$ depends on the form of the distribution f. It can be shown that the distribution in (2.2) satisfies the desired invariance property.

Result 2.1. Let $\underline{\boldsymbol{y}}$ and $\tilde{\underline{\boldsymbol{y}}}$ be two data sets obtained under two different orderings of nodes. Then, $g(\underline{\boldsymbol{y}}; \boldsymbol{\Theta}) = g(\tilde{\underline{\boldsymbol{y}}}; \boldsymbol{\Theta})$.

The proof of Result 2.1 is provided in Appendix A.

The direct maximization of $g(\underline{\boldsymbol{y}}; \boldsymbol{\Theta})$ is not an easy task and thus the EM algorithm [21] is employed as follows below. Suppose $\boldsymbol{z}$ is the missing vector of node membership labels. Then, the complete-data likelihood function can be written as

$$\mathcal{L}_c(\boldsymbol{\Theta}; \underline{\boldsymbol{y}}) = \prod_{\boldsymbol{s}\in\mathcal{S}} \left[\prod_{k=1}^{K}\prod_{k'=1}^{K}\prod_{i=1}^{n-1}\prod_{j=i+1}^{n} \left[\pi_k\pi_{k'} f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})\right]^{I(s_i=k, s_j=k')}\right]^{I(\boldsymbol{z}=\boldsymbol{s})}.$$

The corresponding Q-function, *i.e.*, conditional expectation of the complete-data log likelihood function given $\underline{\boldsymbol{y}}$ is provided by

$$Q(\boldsymbol{\Theta}; \dot{\boldsymbol{\Theta}}, \underline{\boldsymbol{y}}) = \sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{k=1}^{K}\sum_{k'=1}^{K}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n} I(s_i = k, s_j = k') \log\left[\pi_k\pi_{k'} f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})\right], \tag{2.3}$$

where $\ddot{\tau}_{\boldsymbol{s}} = \mathbb{P}\text{r}(\boldsymbol{z} = \boldsymbol{s}|\underline{\boldsymbol{y}}; \dot{\boldsymbol{\Theta}})$ is the posterior probability of sequence $\boldsymbol{s}$. Two dots on top of $\tau_{\boldsymbol{s}}$ denote the current iteration of the EM algorithm and one dot on top of $\boldsymbol{\Theta}$ refers to the previous iteration. At the expectation step of the EM algorithm, this probability can be estimated by the following expression:

$$\ddot{\tau}_{\boldsymbol{s}} = \frac{\prod_{k=1}^{K}\prod_{k'=1}^{K}\prod_{i=1}^{n-1}\prod_{j=i+1}^{n}\left[\dot{\pi}_k\dot{\pi}_{k'} f(\boldsymbol{y}_{ij}; \dot{\boldsymbol{\theta}}_{kk'})\right]^{I(s_i=k, s_j=k')}}{\sum_{\boldsymbol{s}'\in\mathcal{S}}\prod_{k=1}^{K}\prod_{k'=1}^{K}\prod_{i=1}^{n-1}\prod_{j=i+1}^{n}\left[\dot{\pi}_k\dot{\pi}_{k'} f(\boldsymbol{y}_{ij}; \dot{\boldsymbol{\theta}}_{kk'})\right]^{I(s'_i=k, s'_j=k')}}. \tag{2.4}$$

The maximization step involves updating prior probabilities according to

$$\ddot{\pi}_k = \frac{K\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[I(s_i = k) + I(s_j = k)\right]}{n(n-1)}. \tag{2.5}$$

The specific form of expressions for updating parameters $\boldsymbol{\theta}_{kk'}$ depends on the functional form of distribution $f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})$. The cases of normal distributions for univariate and multivariate directed edges are considered in Sections 2.2 and 2.3, respectively. Oftentimes, weights associated with edges are discrete. For example, communication networks can contain information about the number of emails sent within all pairs of nodes. In this case, Poisson distribution can serve as a viable option.

Also, although beyond the scope of this paper, it is possible to relax the complete-graph setting by introducing a hierarchical model assuming that every edge is observed with some component-specific probability. In this scenario, suppose $\alpha_{kk'}$ represents the probability of observing a directed edge connecting nodes that belong to components k and k', respectively. Hence, bidirected edges are observed with probability $\alpha_{kk'}\alpha_{k'k}$ and the probability associated with a non-existing connection between the two nodes is given by $(1-\alpha_{kk'})(1-\alpha_{k'k})$. When just one of the two directed edges is present, the corresponding probability is given by $\alpha_{kk'}(1-\alpha_{k'k})$ or $(1-\alpha_{kk'})\alpha_{k'k}$. In the case of a one-way connection, a univariate representation of $\boldsymbol{y}_{ij}$ should be considered. In other words, we employ the joint distribution $f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})$ with probability $\alpha_{kk'}\alpha_{k'k}$, point mass with probability $(1-\alpha_{kk'})(1-\alpha_{k'k})$, marginal distribution $\int f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})dy_{ji}$ with probability $\alpha_{kk'}(1-\alpha_{k'k})$, and the other marginal $\int f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})dy_{ij}$ with probability $(1-\alpha_{kk'})\alpha_{k'k}$. It can also be noted that the treatment of zero-inflated edges can be conducted similarly to that in the case of non-existent edges. A small pilot study investigating the performance of the proposed methodology in the absence of some edges is considered in Appendix B.

2.2. Multivariate normal distribution for modeling edges

The framework discussed in Section 2.1 is rather flexible. The choice of appropriate distribution $f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})$ should be dictated by a specific application. In this section, we focus on the case of normality as one of the most fundamental and popular in the theory of statistics.

Let $f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})$ be bivariate normal probability density function $\phi(\boldsymbol{y}_{ij}; \boldsymbol{\mu}_{kk'}, \boldsymbol{\Omega}_{kk'})$. If both nodes belong to the kth cluster, it follows that $k = k'$ and the directed edges between the nodes are exchangeable, *i.e.*, both $\boldsymbol{y}_{ij}$ and $\boldsymbol{E}\boldsymbol{y}_{ij}$ follow $\mathcal{N}(\boldsymbol{\mu}_{kk}, \boldsymbol{\Omega}_{kk})$, where $\boldsymbol{\mu}_{kk} \equiv \mu_{kk}\boldsymbol{1}$ and $\boldsymbol{\Omega}_{kk} \equiv \sigma^2_{kk}(\boldsymbol{I} + \rho_{kk}\boldsymbol{E})$. Here, μ_{kk}, σ^2_{kk}, and ρ_{kk} represent mean, variance, and correlation parameters corresponding to the edges in the kth cluster. Also, $\boldsymbol{1} = (1, 1)^\top$, $\boldsymbol{I} = \text{diag}\{1, 1\}$ is a diagonal matrix, and $\boldsymbol{E} = \text{antidiag}\{1, 1\}$ is an exchange matrix.

Taking derivatives of the Q-function in Eq. (2.3) with respect to parmaters μ_{kk}, σ^2_{kk}, and ρ_{kk} yields the following expressions for the maximization step of the EM algorithm:

$$\ddot{\mu}_{kk} = \frac{\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} I(s_i=k,s_j=k)\boldsymbol{y}_{ij}^{\top}(\boldsymbol{I}-\dot{\rho}_{kk}\boldsymbol{E})\boldsymbol{1}}{2(1-\dot{\rho}_{kk})\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} I(s_i=k,s_j=k)}, \tag{2.6}$$

$$\ddot{\sigma}^2_{kk} = \frac{\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} I(s_i=k,s_j=k)(\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})^{\top}(\boldsymbol{I}-\dot{\rho}_{kk}\boldsymbol{E})(\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})}{2(1-\dot{\rho}^2_{kk})\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} I(s_i=k,s_j=k)}. \tag{2.7}$$

The closed form expression is not available for the correlation parameter ρ_{kk}, but it can be readily estimated by solving cubic equation

$$\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n} I(s_i=k,s_j=k)\Big[2\ddot{\sigma}^2_{kk}\rho^3_{kk} - (\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})^{\top}\boldsymbol{E}(\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})\rho^2_{kk}$$
$$-2\big(\ddot{\sigma}^2_{kk} - (\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})^{\top}(\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})\big)\rho_{kk} - (\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})^{\top}\boldsymbol{E}(\boldsymbol{y}_{ij}-\ddot{\mu}_{kk}\boldsymbol{1})\Big] = 0, \tag{2.8}$$

subject to the restriction $|\rho_{kk}| < 1$.

Now, consider the case with the pair of nodes representing different clusters, *i.e.*, $k \neq k'$. Taking derivatives of the Q-function with respect to $\boldsymbol{\mu}_{kk'}$ and $\boldsymbol{\Omega}_{kk'}$ leads to the following expressions:

$$\ddot{\boldsymbol{\mu}}_{kk'} = \frac{\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\Big[I(s_i=k,s_j=k')\boldsymbol{y}_{ij} + I(s_i=k',s_j=k)\boldsymbol{E}\boldsymbol{y}_{ij}\Big]}{\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\Big[I(s_i=k,s_j=k') + I(s_i=k',s_j=k)\Big]}, \tag{2.9}$$

$$\ddot{\boldsymbol{\Omega}}_{kk'} = \frac{\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\Big[I(s_i=k,s_j=k')(\boldsymbol{y}_{ij}-\ddot{\boldsymbol{\mu}}_{kk'})(\boldsymbol{y}_{ij}-\ddot{\boldsymbol{\mu}}_{kk'})^{\top} + I(s_i=k',s_j=k)(\boldsymbol{E}\boldsymbol{y}_{ij}-\ddot{\boldsymbol{\mu}}_{kk'})(\boldsymbol{E}\boldsymbol{y}_{ij}-\ddot{\boldsymbol{\mu}}_{kk'})^{\top}\Big]}{\sum_{\boldsymbol{s}\in\mathcal{S}} \ddot{\tau}_{\boldsymbol{s}} \sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\Big[I(s_i=k,s_j=k') + I(s_i=k',s_j=k)\Big]}. \tag{2.10}$$

The considered model has a relatively low number of parameters, $2.5K^2 + 1.5K - 1$, but provides a rather substantial modeling flexibility. It is also worth mentioning that the parameters of the proposed model can be easily estimated at the M-step based on Eqs. (2.6)–(2.10).

2.3. Matrix normal distribution for modeling multilayer networks

The multilayer network setting occurs when several variables characterize between-node relationships. Let p represent the edge dimensionality. As we mentioned in the Introduction, the existing literature treats such situations by considering lower-dimensional representations with univariate edges. Alternatively, network layers can be treated individually and then an ensemble solution needs to be found by agglomerating p network partitions. Algorithm 1 presents one possible way of agglomeration based on the ideas presented by Michael and Melnykov [47] in the mixture modeling initialization context. From now on, we refer to this method as *naive* approach.

Algorithm 1 Naive approach.

1: For a fixed value of mixture order K, apply the procedure discussed in Section 2.2 to every network layer $r = 1, \ldots, p$ for obtaining p partitions $\boldsymbol{C}_r = (C_{r1}, \ldots, C_{rn})$, where $C_{ri} \in \{1, \ldots, K\}$ and $i = 1, \ldots, n$.
2: Construct a symmetric matrix $\boldsymbol{A}_r = (A_r(i,j))_{n\times n}$ with $A_r(i,j) = A_r(j,i) = I(C_{ri} = C_{rj})$.
3: Define the proximity matrix $\boldsymbol{D} = 1 - \frac{1}{p}\sum_{r=1}^{p}\boldsymbol{A}_r$.
4: Using $\boldsymbol{D}$, perform hierarchical clustering to obtain the final partition vector $\boldsymbol{C} = (C_1, \ldots, C_n)$ with $C_i \in \{1, \ldots, K\}$.

At the first step, the proposed algorithm finds a partition $\boldsymbol{C}_r$ for each variable $r = 1, \ldots, p$ based on the procedure described in Section 2.2 for pre-specified K. Next, p symmetric matrices $\boldsymbol{A}_r$ reflecting the agreement in cluster assignments for every pair of nodes are constructed. Then, the proximity matrix $\boldsymbol{D}$ is proposed to measure the average disagreement in the pair assignments across all network layers. Finally, hierarchical clustering is performed on matrix $\boldsymbol{D}$ to obtain the final partition.

However, the multivariate analysis of multilayer networks is more reasonable as it does not suffer from the loss of information in the course of the individual analysis of layers. Let the pair of multivariate edges be written as matrix $\boldsymbol{Y}_{ij} = (\boldsymbol{y}_{ij1}, \ldots, \boldsymbol{y}_{ijp})$, where each $\boldsymbol{y}_{ijr} = (y_{ijr1}, y_{ijr2})^{\top}$, $r = 1, \ldots, p$, represents the pair of edges between nodes i and j corresponding to the rth characteristic. The distribution of $\boldsymbol{Y}_{ij}$ can be chosen to be matrix normal [19,24,56], with the pdf given by

$$\Phi(\boldsymbol{Y};\boldsymbol{M},\boldsymbol{\Omega},\boldsymbol{\Sigma}) = \frac{(2\pi)^{-\frac{pd}{2}}}{|\boldsymbol{\Sigma}|^{\frac{d}{2}}|\boldsymbol{\Omega}|^{\frac{p}{2}}}\exp\left\{-\frac{1}{2}\mathrm{tr}\big\{\boldsymbol{\Omega}^{-1}(\boldsymbol{Y}-\boldsymbol{M})\boldsymbol{\Sigma}^{-1}(\boldsymbol{Y}-\boldsymbol{M})^{\top}\big\}\right\},$$

where $\boldsymbol{M}$ is the $d \times p$ mean matrix and $\boldsymbol{\Omega}$ and $\boldsymbol{\Sigma}$ are $d \times d$ row and $p \times p$ column covariance parameters, respectively. It is easy to see that matrix vectorization leads to a multivariate normal distribution, *i.e.*, if $\boldsymbol{Y} \sim m\mathcal{N}_{d\times p}(\boldsymbol{M},\boldsymbol{\Omega},\boldsymbol{\Sigma})$, it can be equivalently written as $\mathrm{vec}(\boldsymbol{Y}) \sim \mathcal{N}_{dp}(\mathrm{vec}(\boldsymbol{M}), \boldsymbol{\Sigma}\otimes\boldsymbol{\Omega})$. The presence of the Kronecker product in the covariance matrix introduces a minor non-identifiability issue as $\boldsymbol{\Sigma}\otimes\boldsymbol{\Omega} = (c\boldsymbol{\Sigma})\otimes(c^{-1}\boldsymbol{\Omega})$ for any $c \in R^+$. A simple constraint, such as $|\boldsymbol{\Omega}| = 1$ used in this paper, resolves this drawback effectively.

In the context of finite mixture models, a matrix normal mixture was first introduced by [57]. Other notable recent developments in the matrix mixture modeling framework can be found in [29,45,51]. These papers primarily focus on introducing matrix components capable of modeling skewness.

In our framework, $f(\boldsymbol{Y}_{ij};\boldsymbol{\theta}_{kk'}) \equiv \Phi(\boldsymbol{Y}_{ij};\boldsymbol{M}_{kk'},\boldsymbol{\Omega}_{kk'},\boldsymbol{\Sigma}_{kk'})$ and $d=2$. If there is a reason to assume skewness along the layers, that can be readily addressed by following the strategy proposed by Melnykov and Zhu [45]. As in Section 2.2, two cases of node memberships need to be considered. If both nodes belong to cluster k, the mean matrix $\boldsymbol{M}_{kk}$ can be specified as $\boldsymbol{M}_{kk}=\mathbf{1}\boldsymbol{\mu}_{kk}^{\top}$, where $\boldsymbol{\mu}_{kk}=(\mu_{kk1},\ldots,\mu_{kkp})^{\top}$. The covariance matrix $\boldsymbol{\Omega}_{kk}$ is given in the form $\boldsymbol{\Omega}_{kk}=\sigma_{kk}^2(\boldsymbol{I}+\rho_{kk}\boldsymbol{E})$. Incorporating constraint $|\boldsymbol{\Omega}_{kk}|=1$ leads to $\boldsymbol{\Omega}_{kk}=(1-\rho_{kk}^2)^{-\frac{1}{2}}(\boldsymbol{I}+\rho_{kk}\boldsymbol{E})$ and consequently $\boldsymbol{\Omega}_{kk}^{-1}=(1-\rho_{kk}^2)^{-\frac{1}{2}}(\boldsymbol{I}-\rho_{kk}\boldsymbol{E})$. Taking derivatives with respect to parameters $\boldsymbol{\mu}_{kk}$, ρ_{kk}, and $\boldsymbol{\Sigma}_{kk}$ yields the following expressions:

$$\ddot{\boldsymbol{\mu}}_{kk}=\frac{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}I(s_i=k,s_j=k)\boldsymbol{Y}_{ij}^{\top}(\boldsymbol{I}-\dot{\rho}_{kk}\boldsymbol{E})\mathbf{1}}{2(1-\dot{\rho}_{kk})\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}I(s_i=k,s_j=k)}, \tag{2.11}$$

$$\ddot{\rho}_{kk}=\frac{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}I(s_i=k,s_j=k)\mathrm{tr}\left\{\boldsymbol{E}(\boldsymbol{Y}_{ij}-\mathbf{1}\ddot{\boldsymbol{\mu}}_{kk}^{\top})\dot{\boldsymbol{\Sigma}}_{kk}^{-1}(\boldsymbol{Y}_{ij}-\mathbf{1}\ddot{\boldsymbol{\mu}}_{kk}^{\top})^{\top}\right\}}{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}I(s_i=k,s_j=k)\mathrm{tr}\left\{(\boldsymbol{Y}_{ij}-\mathbf{1}\ddot{\boldsymbol{\mu}}_{kk}^{\top})\ddot{\boldsymbol{\Sigma}}_{kk}^{-1}(\boldsymbol{Y}_{ij}-\mathbf{1}\ddot{\boldsymbol{\mu}}_{kk}^{\top})^{\top}\right\}}, \tag{2.12}$$

$$\ddot{\boldsymbol{\Sigma}}_{kk}=\frac{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}I(s_i=k,s_j=k)(\boldsymbol{Y}_{ij}-\mathbf{1}\ddot{\boldsymbol{\mu}}_{kk}^{\top})(\boldsymbol{I}-\ddot{\rho}_{kk}\boldsymbol{E})(\boldsymbol{Y}_{ij}-\mathbf{1}\ddot{\boldsymbol{\mu}}_{kk}^{\top})^{\top}}{2\sqrt{1-\ddot{\rho}_{kk}^2}\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}I(s_i=k,s_j=k)}. \tag{2.13}$$

It can be noticed that due to the specific choice of a restriction on matrix $\boldsymbol{\Omega}_{kk}$, the correlation parameter ρ_{kk} can be estimated in the closed form.

The second case assumes that two nodes belong to clusters k and k' such that $k\neq k'$. Taking derivatives of the Q-function with respect to mean matrix $\boldsymbol{M}_{kk'}$ and covariance matrix $\boldsymbol{\Sigma}_{kk'}$ yields the following expressions:

$$\ddot{\boldsymbol{M}}_{kk'}=\frac{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[I(s_i=k,s_j=k')\boldsymbol{Y}_{ij}+I(s_i=k',s_j=k)\boldsymbol{E}\boldsymbol{Y}_{ij}\right]}{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[I(s_i=k,s_j=k')+I(s_i=k',s_j=k)\right]}, \tag{2.14}$$

$$\ddot{\boldsymbol{\Sigma}}_{kk'}=\frac{\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[I(s_i=k,s_j=k')(\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})^{\top}\dot{\boldsymbol{\Omega}}_{kk'}^{-1}(\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})+I(s_i=k',s_j=k)(\boldsymbol{E}\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})^{\top}\dot{\boldsymbol{\Omega}}_{kk'}^{-1}(\boldsymbol{E}\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})\right]}{2\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[I(s_i=k,s_j=k')+I(s_i=k',s_j=k)\right]}. \tag{2.15}$$

A theorem discussed in [14] can be employed for estimating $\boldsymbol{\Omega}_{kk'}$. The theorem states that if $\boldsymbol{W}$ is a symmetric positive definite matrix, there exists a $d\times d$ symmetric matrix $\boldsymbol{\Omega}$ with $|\boldsymbol{\Omega}|=1$ that minimizes $\mathrm{tr}(\boldsymbol{W}\boldsymbol{\Omega}^{-1})$ and it is given by $\boldsymbol{\Omega}=\boldsymbol{W}/|\boldsymbol{W}|^{1/d}$. This leads to the expression

$$\ddot{\boldsymbol{\Omega}}_{kk'}=\frac{\ddot{\boldsymbol{W}}_{kk'}}{\sqrt{|\ddot{\boldsymbol{W}}_{kk'}|}}, \tag{2.16}$$

where $\ddot{\boldsymbol{W}}_{kk'}$ is a scatter matrix that can be calculated as

$$\ddot{\boldsymbol{W}}_{kk'}=\sum_{\boldsymbol{s}\in\mathcal{S}}\ddot{\tau}_{\boldsymbol{s}}\sum_{i=1}^{n-1}\sum_{j=i+1}^{n}\left[I(s_i=k,s_j=k')(\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})\ddot{\boldsymbol{\Sigma}}_{kk'}^{-1}(\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})^{\top}+I(s_i=k',s_j=k)(\boldsymbol{E}\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})\ddot{\boldsymbol{\Sigma}}_{kk'}^{-1}(\boldsymbol{E}\boldsymbol{Y}_{ij}-\ddot{\boldsymbol{M}}_{kk'})^{\top}\right]. \tag{2.17}$$

Closed-form expressions provided in Eqs. (2.11)–(2.17) constitute the maximization step of the EM algorithm taking into account the presence of multivariate edges in directed weighted networks.

2.4. Calculation of posterior probabilities at the expectation step

The expectation step presented in Eq. (2.4) involves calculating $|\mathcal{S}| = K^n$ posterior probabilities. This quantity can be prohibitively large in many real-life situations. We propose employing an MCMC scheme that allows resolving this issue effectively. The Metropolis-Hastings algorithm [31,46] is one of the most well known MCMC-based procedures that can be seen as a rejection sampling technique. Some potential problems associated with this algorithm include relatively slow convergence if the algorithm is trapped in the proximity of a local mode and difficulties related to the choice of efficient proposals. There exist several extensions of the Metropolis-Hastings algorithm that aim at tackling these challenges. In particular, the multiple try Metropolis method [39] employs multiple proposals and weights associated with them to calculate the acceptance probability. This weight-based selection approach has certain computational advantages as discussed by Calderhead [13], Martino [41], Martino and Read [42]. Although the multiple try Metropolis generally allows choosing more efficient proposals, it also requires some calibration, *e.g.*, deciding on the number of proposals to be considered. In this section we outline a variant of the Metropolis-Hastings algorithm similar to the one employed in [44]. Special attention is paid to the choice of a promising proposal point.

Let $\boldsymbol{z} = (z_1, \ldots, z_n)^\top$ represent the current sequence of node memberships. First, choose $\nu \in \{1, \ldots, n\}$ with probability $1/n$. The current membership of this node is z_ν. Next, form the set $\tilde{\mathcal{Z}}_\nu^-$ consisting of alternative memberships for the νth node, *i.e.*, $\tilde{\mathcal{Z}}_\nu^- = \{1, \ldots, K\} \setminus z_\nu$. For every element $\tilde{z}_\nu \in \tilde{\mathcal{Z}}_\nu^-$, form a partition proposal $\tilde{\boldsymbol{z}} = (z_1, \ldots, z_{\nu-1}, \tilde{z}_\nu, z_{\nu+1}, \ldots, z_n)^\top$ and calculate the corresponding within-cluster sum of squares for edge weights, $W_{\tilde{z}}$. Out of $K-1$ available proposals, choose $\tilde{\boldsymbol{z}}$ with probability $W_{\tilde{z}_\nu}^{-1} / \sum_{\omega \in \tilde{\mathcal{Z}}_\nu^-} W_\omega^{-1}$. It can be seen that partitions that have smaller within-cluster sums of squares have a higher chance of occurrence. Vector $\tilde{\boldsymbol{z}}$ is the current membership proposal with the corresponding transition probability given by $\Pr(\boldsymbol{z} \to \tilde{\boldsymbol{z}}) = n^{-1} W_{\tilde{z}_\nu}^{-1} / \sum_{\omega \in \tilde{\mathcal{Z}}_\nu^-} W_\omega^{-1}$. Following similar logic, the reverse transition probability $\Pr(\tilde{\boldsymbol{z}} \to \boldsymbol{z})$ can be obtained as $\Pr(\tilde{\boldsymbol{z}} \to \boldsymbol{z}) = n^{-1} W_{z_\nu}^{-1} / \sum_{\omega \in \check{\mathcal{Z}}_\nu^-} W_\omega^{-1}$, where $\check{\mathcal{Z}}_\nu^-$ is defined as $\check{\mathcal{Z}}_\nu^- = \{1, \ldots, K\} \setminus \tilde{z}_\nu$ and $\check{\boldsymbol{z}} = (z_1, \ldots, z_{\nu-1}, \check{z}_\nu, z_{\nu+1}, \ldots, z_n)^\top$ for all $\check{z} \in \check{\mathcal{Z}}_\nu^-$. Finally, one realization U following Uniform (0, 1) is obtained. The proposal $\tilde{\boldsymbol{z}}$ is accepted if $\alpha > U$, where α is given by

$$\alpha = \min\left\{1, \frac{\Pr(\tilde{\mathbf{z}} \to \mathbf{z})\pi\left(\tilde{\mathbf{z}}|\underline{\mathbf{y}}; \boldsymbol{\Theta}\right)}{\Pr(\mathbf{z} \to \tilde{\mathbf{z}})\pi\left(\mathbf{z}|\underline{\mathbf{y}}; \boldsymbol{\Theta}\right)}\right\}$$
$$= \min\left\{1, \frac{W_{\tilde{z}_\nu}\left(\sum_{\omega \in \tilde{\mathcal{Z}}_\nu^-} W_\omega^{-1}\right) f_{\underline{\mathbf{y}}}\left(\underline{\mathbf{y}}|\mathbf{e}_{\tilde{\mathbf{z}}}; \boldsymbol{\Theta}\right)}{W_{z_\nu}\left(\sum_{\omega \in \check{\mathcal{Z}}_\nu^-} W_\omega^{-1}\right) f_{\underline{\mathbf{y}}}\left(\underline{\mathbf{y}}|\mathbf{e}_{\mathbf{z}}; \boldsymbol{\Theta}\right)}\right\}. \quad (2.18)$$

In this paper, the burning time and number of MCMC realizations for the proposed scheme are chosen to be $10n$ and $100n$, respectively. As shown in Section 3, these values ensure good performance of the suggested procedure in all considered settings. The MCMC-based calculation of posterior probabilities is feasible even for a high number of nodes because the vast majority of possible sequences have near-zero probabilities of occurrence and just few of them will be accepted as proposals in the course of the MCMC-based algorithm. Thus, the number of posterior probabilities that need to be calculated in practice can be reduced dramatically. Appendix C presents a small simulation experiment demonstrating the effectiveness of the proposed strategy.

2.5. Implementation issues

The problem of unbounded likelihood is well-documented in finite mixture modeling literature [43]. If the number of observations associated with a component is insufficient, a singular or nearly-singular covarince matrix estimate can be observed. This leads to the unbounded likelihood or so-called spurious solutions. Gaussian mixture models with unrestricted covariances are known to suffer from this issue. Model proposed in Eq. (2.2) might face similar issues depending on the choice of component distributions and particular parameterization. Both special cases considered in Sections 2.2 and 2.3 are subject to the unbounded likelihood in the proposed formulation. While various strategies for alleviating the issue have been discussed in the related literature (see, *e.g.*, [43]), we employ the approach relying on finding the best local maximizer that does not exhibit the behavior consistent with the presence of spurious solutions.

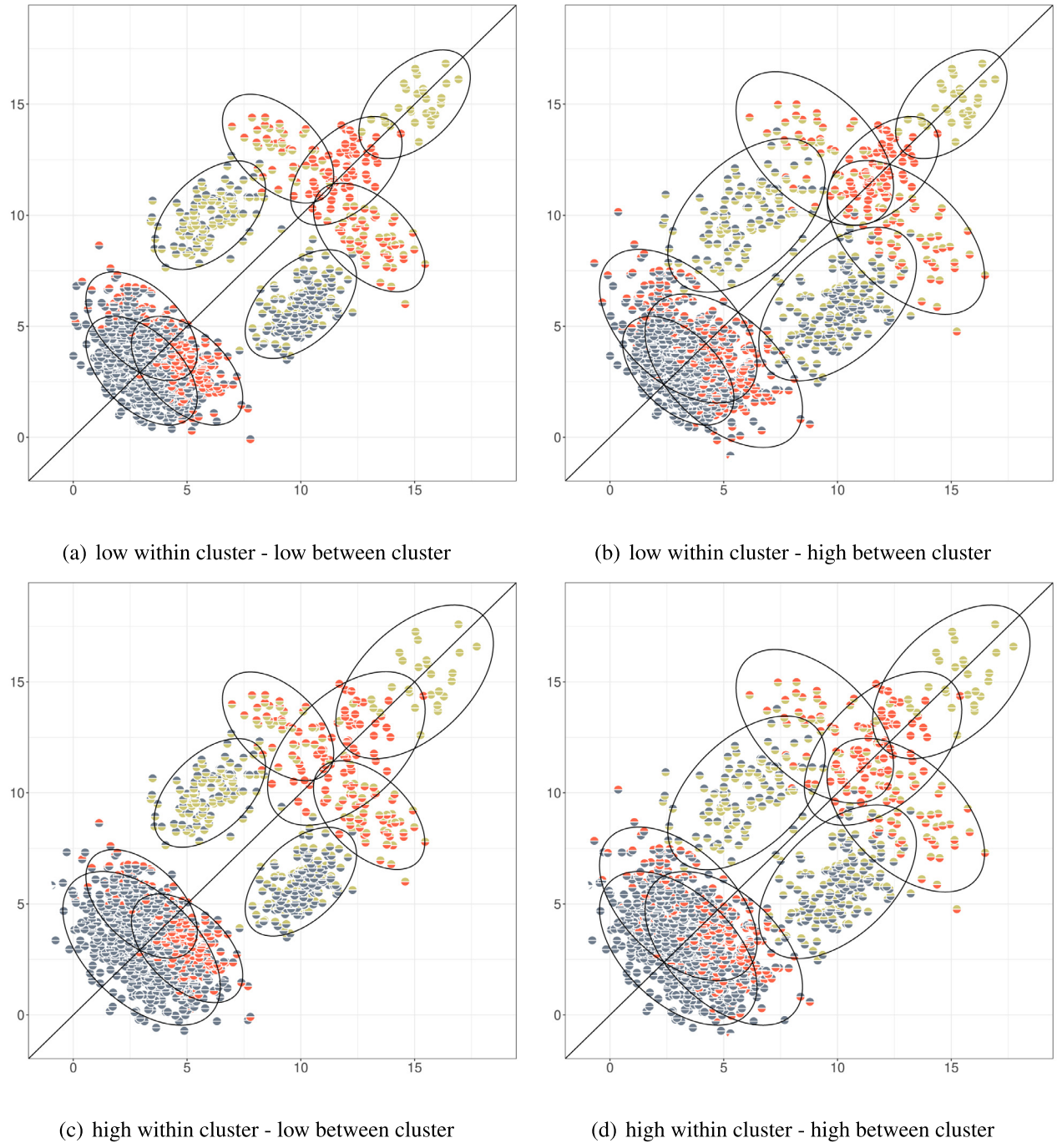
Initialization is another important problem that is well-documented in the literature on mixture modeling. The starting point of the EM algorithm specifies the best local maximizer that will be found in the course of the algorithm. Numerous initialization strategies are proposed. In this work, we employed the one based on random starts and known as *emEM* [9]. Starting from several randomly selected points, short EM algorithms are run for some pre-specified number of iterations. Then, the main EM algorithm is initialized by the most promising short EM run as measured by the achieved likelihood value. It is worth mentioning that the effect of poor initialization of our procedure is not as severe as usual due to the presence of burning time for the MCMC procedure incorporated at the expectation step.

3. Experimental evaluation

3.1. Simulation study on networks with different complexity levels

In this section, we aim at evaluating the performance of the developed methodology in various clustering complexity settings. Fig. 1 illustrates four settings considered in this study. Each of the four plots provided represents a simulated data set. The blue, red, and khaki colors of points illustrate cluster memberships, with every point representing a specific directed edge. Therefore, within-cluster edges are shown in one color, while between-cluster edges are given in two colors. In accordance with the parameterization discussed in Section 2.2, the ellipses on the diagonal illustrate distributions of within-cluster edges and the ellipses that present reflections over the diagonal show distributions of between-cluster edges. Four different scenarios are considered in Fig. 1. The simplest is presented in plot (a) and is characterized by relatively low variability in distributions of edges. Plots (b) and (c) present two cases with covariance matrices associated with between-clutser and within-cluster edges being respectively multiplied by factor 2. Plot (d) illustrates the most difficult case when all covariance matrices from the simplest scenarion are multiplied by 2. While plots shown in Fig. 1 can serve as a helpful visualization tool, it is worth reminding that the data are highly dependent and thus the traditional interpretation of scatter plots is not entirely applicable in our setting. In fact, even a considerable overlap between some components does not necessarily present a difficult clustering case. For instance, the blue component exhibits a rather substantial overlap with the component modeling edges between blue and red clusters. However, excellent clustering results can be obtained here as reflected by the output presented in Table 1.

In this study, we consider sample sizes $n = 50, 100, 200$. In each case, 100 data sets are simulated with 50%, 30%, and 20% of nodes belonging to the blue, red, and khaki clusters, respectively. The parameters of the employed models can be found in the Supplement. As we can see from Table 1 presenting sample mean and standard deviations (in parentheses) of ARI values, reasonable results can be obtained even in case $n = 50$. As expected, the cases with



(a) low within cluster - low between cluster

(b) low within cluster - high between cluster

(c) high within cluster - low between cluster

(d) high within cluster - high between cluster

Fig. 1. Illustrative examples with varying network complexity as discussed in Section 3.1.

Table 1
Performance of the proposed procedure in four settings as discussed in Section 3.1. Low - low represents the case with low between-cluster and low within-cluster edge variability, low - high denotes the case with low within-cluster and high between-cluster edge variability, *etc.* Sample means and standard deviations (shown in parentheses) of ARI values are presented.

# nodes	low - low	low - high	high - low	high - high
50	0.953 (0.133)	0.880 (0.194)	0.933 (0.158)	0.903 (0.205)
100	0.993 (0.040)	0.962 (0.121)	0.981 (0.092)	0.969 (0.142)
200	1.000 (0.000)	1.000 (0.000)	1.000 (0.000)	0.957 (0.148)

the lowest and highest variability are the easiest and most difficult, respectively. The performance of the procedure improves considerably along with the sample size increase. In the case of $n = 200$, perfect solutions can be found in all cases except for the most difficult one. In the latter case, the best partition was found in 92 out of 100 cases. As expected, the procedure demonstrates improving performance along with the sample size increase.

We have also performed a small experiment assuming unknown mixture order in the "high within cluster - high between cluster" setting. There were 100 data sets with 50 nodes simulated from the corresponding three-component mixture. The correct order was detected 69 times. For the remaining 31 data sets, the model underestimated the mixture order and identified two components. Indeed, this is not surprising due to considerable variability in data and relatively low sample size.

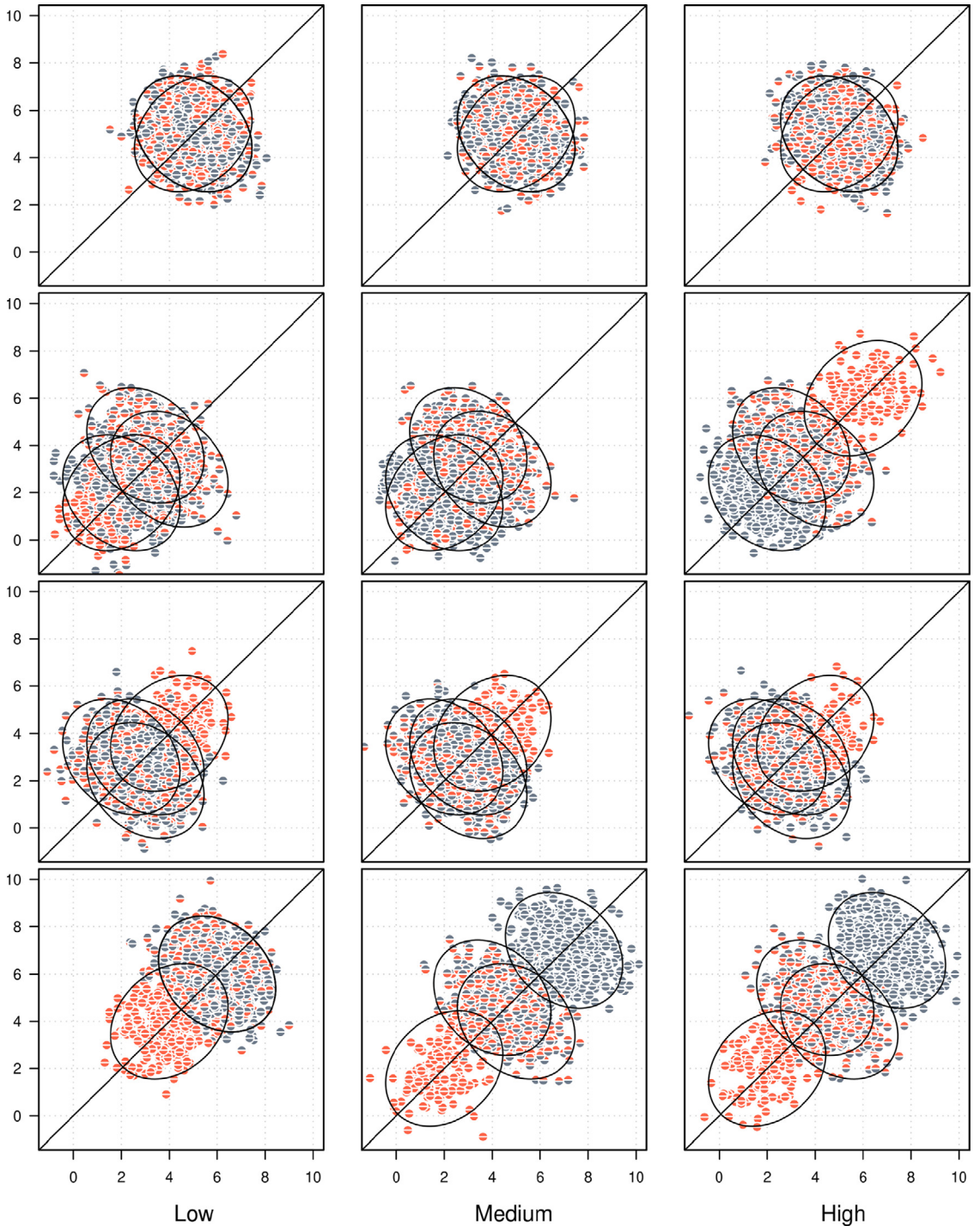


Fig. 2. Illustration of experiments outlined in Section 3.2. The vertical panels represent situations with various degree of separation.

3.2. Comparison between the naive and matrix-variate approaches to clustering multilayer network

This section provides a brief simulation study comparing the naive and matrix-variate approaches and demonstrating the advantages of the latter one. For the purpose of illustration, two-cluster settings of varying complexity levels have been considered for a multilayer network with between-node relationships characterized by four variables. In each setting, 100 data sets consisting of 50 nodes have been simulated. Panels (a)–(c) presented in Fig. 2 reflect the three complexity levels under consideration. Each row exhibits one of the variables in the multilayer relationship. Per description in Section 3.1, the blue and red colors represent true cluster memberships. The distributions of within-cluster and between-cluster edges are captured by the ellipses located along the diagonal and ones symmetrically reflected over it, respectively. In a two-cluster case, there are two ellipses representing within-cluster and two more displaying between-cluster relationships. As can be seen from Fig. 2, the four variables exhibit different level of importance for cluster analysis. While the third variable presents some minor separation in the two clusters, the first one is not helpful for the task of discrimination in all settings considered. The varying level

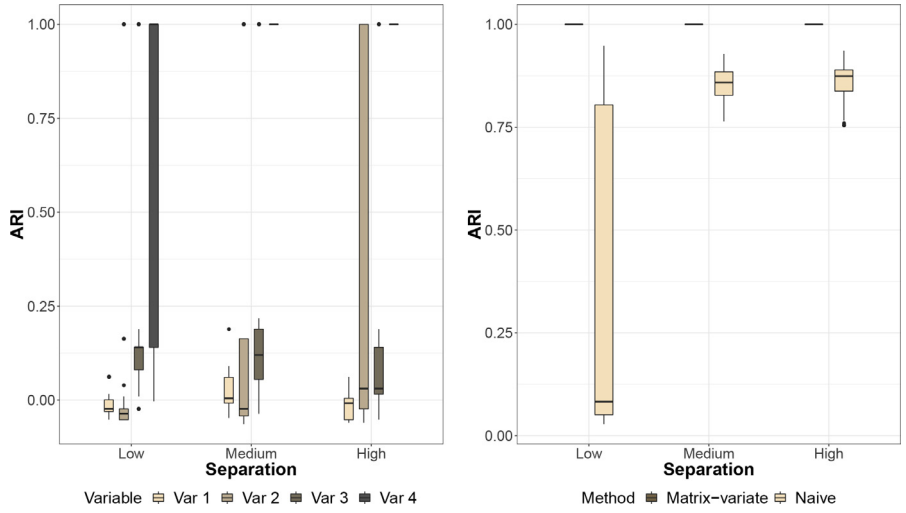


Fig. 3. (a) Performance of the unilayer approach for each variable; (b) performance comparison between the matrix-variate and naive approaches.

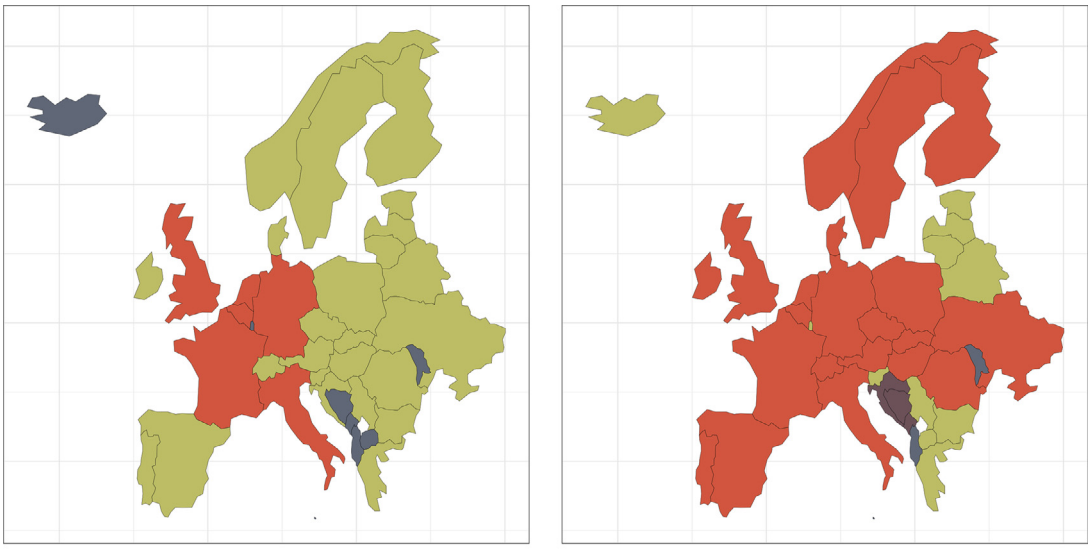
of overlap in the second and fourth variables defines the degree of clustering complexity associated with each setting. In particular, setting (c) is the easiest for clustering due to the distinct separation in second and fourth variables. On the contrary, setting (a) presents the most challenging situation. Setting (b) can be seen as the intermediate setting with just the fourth variable contributing to separation.

Boxplots presented in Fig. 3(a) display the performance of the methodology discussed in Section 2.2 applied to each of the four network layers. It can be clearly seen that in the case of low separation, none of the variables are able to identify the underlying cluster structure. In the medium separation scenario, the fourth layer is capable of finding the correct partition. In the last setting, one would expect the extra separation in the second layer to help in retrieving the true cluster structure. However, for the red component the inter-cluster and intra-cluster correlations are in the opposite direction which results in poor classification. Fig. 3(b) presents the comparison of the naive and matrix-variate approaches in the above-mentioned cases. As expected, the performance of the naive method improves along with the increase in the between cluster separation. However, the matrix-variate approach is able to recover true cluster memberships in all three cases considered. This brief simulation study highlights the importance of analyzing the network layers jointly.

4. Applications to European trade data

In this section, we apply the proposed methodology to cluster European countries based on country-to-country trade amounts observed in 2014. The data are publicly available from the *World Integrated Trade Solution* web-site (https://wits.worldbank.org). Transcontinental countries are excluded from the study. Also, information for Monaco, Kosovo, and Liechtenstein is not available. As a result, the data set contains 39 countries in total. In the first part of the study, we apply the methodology discussed in Section 2.2 to between-country total trade volumes. Fig. 4(a) illustrates the optimal clustering solution obtained. There are three clusters in total with sizes: 6, 9, and 24. The first group represented by the red color contains the countries with very high trade amounts between themselves (cluster mean $52,769.863 mln). United Kingdom, France, Germany, Italy, Belgium, and Netherlands fall into this cluster. The group with nine countries that is shown in the grey color includes Iceland, Andorra, Luxembourg, Malta, Bosnia and Herzegovina, Montenegro, Albania, Macedonia, and Moldova. This cluster is characterized by very low trades between themselves (mean export $14.355 mln) and rather low trades with other countries. The rest of European countries form a major group consisting of 24 nodes and are illustrated by the khaki color. The corresponding mean export within this group is $1,145.939 mln. The trades between countries in red and khaki clusters are at rather high rate, with mean exports from red to khaki and backward from khaki to red being at the levels of $7,144.463 mln and $6,859.876 mln, respectively. The obtained clustering result is rather reasonable. The countries constituting the grey cluster do not trade in high amounts either due to the size of a country, its remoteness, or some other reason such as a recent internal or external conflict. The countries that fall into the red cluster form the core of the European economy.

In the second part of our study instead of considering total trades, we focus on three (out of 21 available) related trade categories: *Capital Goods, Consumer Goods*, and *Intermediate Goods*. *Capital Goods* are generally defined as final goods that are meant to be used in the process of production. *Intermediate Goods* are known as goods that are intended to be consumed in the production process. On the contrary, *Consumer Goods* are those goods that are not made for the future production but intended to be used by consumers. As we focus on these three trade categories, we cluster the countries based on directed trivariate edges. Thus, the methodology discussed in Section 2.3 is employed now. Fig. 4(b) presents the obtained four-cluster solution that is proposed by our methodology. We can observe a rather dramatic change compared with plot (a). In particular, there are many more countries that fall into the red cluster. There are 21 of them in total. The mean trades associated with *Capital, Consumer*, and *Intermediate Goods* in this cluster are $2,493.202 mln, $3,689.111 mln, and $2,242.728 mln, respectively. The grey cluster of low trading countries shrinks to just four: Andorra, Malta, Albania, and Moldova. The corresponding mean trades are as follows : $0.006 mln, $11.994 mln, and $0.003 mln. Relatively high trades associated with *Consumer Goods* and very low trades in other categories make this group of countries different from the others. There is a magenta cluster consisting of three neighboring countries: Croatia, Bosnia and Herzegovina, and Montenegro. Their mean exports are given by $36.573 mln, $256.449 mln, and $131.409 mln. Finally, the remaining 11 countries fall into the khaki cluster with mediocre trade amounts: $31.906 mln, $112.327 mln, and $58.474 mln. The cluster consists of Iceland, Luxembourg, and countries from Eastern Europe that are not as large as Poland, Romania, or Ukraine. While the means

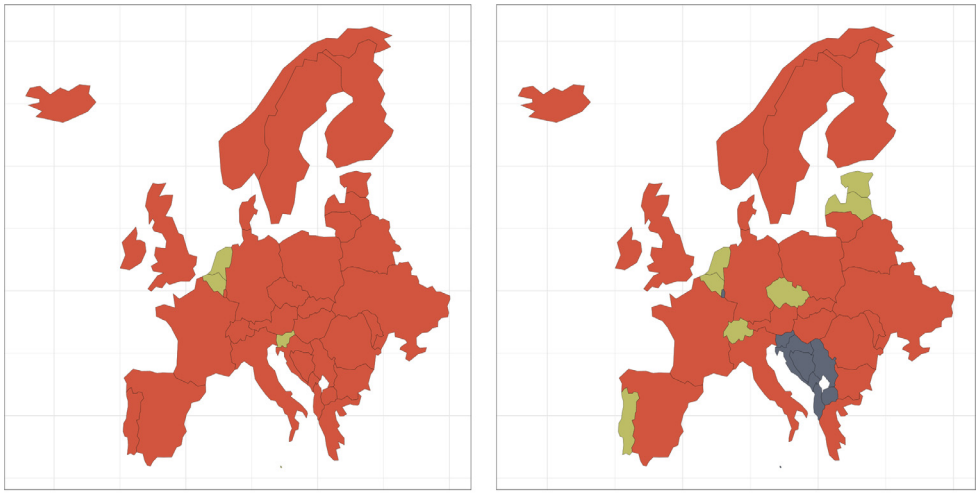


(a) Solution based on total trades (b) Solution based on three trade categories

Fig. 4. Clustering solutions for European trade data considered in Section 4 based on methodologies discussed in (a) Section 2.2 and (b) Section 2.3.

Table 2
Comparison between the matrix-variate approach, naive approach, and unilayer approach for each trade category for the European trade data considered in Section 4.

Matrix-variate approach	Naive approach				Unilayer: *Capital Goods*				Unilayer: *Consumer Goods*				Unilayer: *Intermediate goods*			
	Red	Khaki	Grey	Magenta	Red	Khaki	Grey	Magenta	Red	Khaki	Grey	Magenta	Red	Khaki	Grey	Magenta
Red	12	9	0	0	12	9	0	0	16	0	5	0	9	9	0	3
Khaki	0	5	0	6	0	6	3	2	5	3	0	3	0	11	0	0
Grey	0	0	4	0	0	0	4	0	0	2	0	2	0	0	4	0
Magenta	0	0	1	2	0	0	1	2	0	0	0	3	0	2	1	0



(a) Solution based on total trades (b) Solution based on three trade categories

Fig. 5. Clustering solutions for the relative share of European trade data considered in Section 4 based on methodologies discussed in (a) Section 2.2 and (b) Section 2.3.

of khaki and magenta clusters are similar for *Capital Goods*, they differ more than two times for *Consumer* and *Intermediate Goods*. One can observe that detected clusters are meaningful as they have clear economic or geographic interpretations.

Table 2 presents a comparison between the matrix-variate approach, the naive approach described in Algorithm 1, and unilayer approach applied on each trade category. The aggregated assignment for the naive method is obtained by combining the 4-cluster solutions attained from each trade category. There is 59% agreement between the solutions provided by the matrix-variate method and naive one. The naive approach agrees with the individual layers 90%, 67% and 62% times, respectively. This implies that the clustering solution presented by the naive approach is mostly dominated by the class assignment obtained from the analysis of *Capital Goods*. The agreement between the individual layers varies between 51% and 64%. The substantial mismatch between the so-

lutions provided by the analysis of each individual layer reinforces the need of employing the matrix-variate approach that relies on modeling layers jointly.

The final analysis considered in this section considers another interesting problem suggested by one of anonymous reviewers. The aim of the study is to partition the countries based on their relative share of trades. We mimic the analysis based on the trade magnitude and use the proportion of trade volume relative to the total trade in the corresponding category and direction. Fig. 5 represents the clustering solution obtained for unilayer and multilayer cases, respectively. The obtained clustering solution in this case is quite different from the partitioning obtained based on the trade volume. In the unilayer case, two clusters are detected. The small khaki cluster consists of 4 countries only. The result obtained in the multilayer setting is more interesting. There are three clusters detected in that case. The grey cluster consists of neighboring countries in the Adriatic peninsula. The khaki cluster mainly represents smaller western European countries. The rest of the countries form the big red cluster. The means of relative trades for this clustering result can be found in the Supplement.

5. Discussion

A novel mixture modeling approach is proposed for clustering nodes in directed weighted networks. The proposed procedure relies on the notion of finite mixture model, is rather flexible, and can be readily employed in various settings. In particular, its application is especially appealing in the case of multilayer networks. A traditional approach to analyzing networks with multivariate edges is based on agglomerating the information from the layers and projecting it to the univariate space. Indeed, such a projection is likely to lead to the loss of important information that can result in drawing partial or misleading conclusions. On the contrary, the proposed procedure employs matrix-variate distributions that allow modeling the dependence between the layers explicitly. Another well-known limitation of many network clustering algorithms is their incapability to operate in situations with large or even moderate network sizes. Due to the proposed MCMC technique, the methodology considered in this paper can be applied to moderate sample size cases. However, its application to larger networks is a subject of future work. Other important directions of future work include studying the application of distributions alternative to the normal one. In particular, a normal distribution cannot provide an adequate fit in the case of the heavy presence of zero-inflated data, *i.e.*, networks with zero-weighted edges. One more interesting extension of the developed procedure is clustering dynamic networks. In this framework, edge weights vary over time exhibiting time-related behavior. Grouping nodes based on observed temporal patterns is of interest in many dynamic network applications. A related extension to process monitoring and anomaly detection is another interesting angle of looking at the analysis of temporal networks. Such extensions can be accommodated by means of employing $2 \times p \times T$-dimensional tensor distributions, where T represents the number of time points. This model formulation provides an effective way of modeling the time-related dependence with some traditional time series covariance matrix structures. Finally, modeling networks with completely or partially (*i.e.*, for some layers) missing edges is another important direction of future work that has been briefly addressed in this paper. In this setting, alternative definitions of clusters can also be investigated. In other words, the notion of cluster can be formulated not based on just the connection strength, *i.e.*, the edge weight, but also the presence of such a connection. To summarize, there is a considerable number of interesting and important extensions of the work presented in this paper that are beyond the scope of this manuscript but need to be addressed in the future.

Declaration of Competing Interest

None.

Appendix A. Proof of Result 2.1

Let $\boldsymbol{s} = (s_1, \ldots, s_n)^\top$ represent the sequence of node memberships and $\tilde{\boldsymbol{s}} = (\tilde{s}_1, \ldots, \tilde{s}_n)^\top$ be another sequence of node memberships under some alternative order of nodes. Also, let γ_i map the position of the ith node in the alternative ordering to the position of the same node in the original ordering. It can be noticed that $\tilde{s}_i = s_{\gamma_i}$ as well as $\tilde{\boldsymbol{y}}_{ij} = \boldsymbol{y}_{\gamma_i\gamma_j} \sim f(\boldsymbol{y}; \boldsymbol{\theta}_{kk'})$ for $\gamma_i < \gamma_j$. On the other hand, if $\gamma_i > \gamma_j$, it follows that $\tilde{\boldsymbol{y}}_{ij} = \boldsymbol{E}\boldsymbol{y}_{\gamma_j\gamma_i} \sim f(\boldsymbol{y}; \boldsymbol{\theta}_{kk'})$ and $\boldsymbol{y}_{\gamma_j\gamma_i} \sim f(\boldsymbol{y}; \boldsymbol{\theta}_{k'k})$. Then, we obtain

$$
\begin{aligned}
g(\underline{\tilde{\boldsymbol{y}}}; \boldsymbol{\Theta}) &= \sum_{\boldsymbol{s}\in\mathcal{S}} \prod_{k=1}^{K} \prod_{k'=1}^{K} \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \left[\pi_k \pi_{k'} f(\tilde{\boldsymbol{y}}_{ij}; \boldsymbol{\theta}_{kk'})\right]^{I(\tilde{s}_i=k, \tilde{s}_j=k')} \\
&= \sum_{\boldsymbol{s}\in\mathcal{S}} \prod_{k=1}^{K} \prod_{k'=1}^{K} \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \Big[\pi_k \pi_{k'} f(\boldsymbol{y}_{\gamma_i\gamma_j}; \boldsymbol{\theta}_{kk'})^{I(\gamma_i<\gamma_j)} \\
&\quad f(\boldsymbol{E}\boldsymbol{y}_{\gamma_j\gamma_i}; \boldsymbol{\theta}_{kk'})^{I(\gamma_i>\gamma_j)}\Big]^{I(s_{\gamma_i}=k, s_{\gamma_j}=k')} \\
&= \sum_{\boldsymbol{s}\in\mathcal{S}} \prod_{k=1}^{K} \prod_{k'=1}^{K} \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \Big[\pi_k \pi_{k'} f(\boldsymbol{y}_{\gamma_i\gamma_j}; \boldsymbol{\theta}_{kk'})^{I(\gamma_i<\gamma_j)} \\
&\quad f(\boldsymbol{y}_{\gamma_j\gamma_i}; \boldsymbol{\theta}_{k'k})^{I(\gamma_i>\gamma_j)}\Big]^{I(s_{\gamma_i}=k, s_{\gamma_j}=k')} \\
&= \sum_{\boldsymbol{s}\in\mathcal{S}} \prod_{k=1}^{K} \prod_{k'=1}^{K} \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \left[\pi_k \pi_{k'} f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})^{I(i<j)} f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})^{I(i<j)}\right]^{I(s_i=k, s_j=k')} \\
&= \sum_{\boldsymbol{s}\in\mathcal{S}} \prod_{k=1}^{K} \prod_{k'=1}^{K} \prod_{i=1}^{n-1} \prod_{j=i+1}^{n} \left[\pi_k \pi_{k'} f(\boldsymbol{y}_{ij}; \boldsymbol{\theta}_{kk'})\right]^{I(s_i=k, s_j=k')} = g(\underline{\boldsymbol{y}}; \boldsymbol{\Theta})
\end{aligned}
$$

and this completes the proof.

Appendix B. Simulation study on networks with zero-inflated or missing edges

A small simulation study is considered in this section to demonstrate the model performance on networks with zero-inflated or missing edges. The treatment of both cases can be very similar. This is an example of a situation, where the assumption of the complete graph can be relaxed. For this experiment, a three-component network with 50 nodes is considered. The mixing proportions of the components are chosen to be 0.5, 0.3, and 0.2, respectively. For the simplicity of illustration, we assume that all components have the same probability of missing connection and a directed edges must be missing in pairs. The experiment is performed in four settings. The first setting corresponds to a complete graph, *i.e.*, no edges are missing in that case. In the other settings, 10%, 25%, and 50% of edges are missing, respectively. Table 3 displays the mean and standard deviations of ARI values under the considered settings. As expected, the procedure shows good performance but the classification agreement deteriorates noticeably as the proportion of missing edges increases.

Table 3
Performance of the proposed procedure in the four settings discussed in Appendix B.

complete graph	10% missing edges	25% missing edges	50% missing edges
0.883 (0.211)	0.742 (0.276)	0.588 (0.313)	0.348 (0.312)

Appendix C. Study of the MCMC-based procedure

Another simulation study is considered in this section to evaluate the performance of the proposed MCMC-based method for estimating posterior probabilities τ_s. As the exact procedure considering all possible sequences becomes computantially infeasible even for a low number of nodes, in this experiment we let $n = 10$ and $K = 3$. We simulate 100 sets of data, initialize them identically, and run the exact procedure as well as the EM algorithm with MCMC-based calculations at the expectation step. The objective of this experiment is to assess the accuracy of the proposed procedure as reflected by the proximity of maximized log-likelihood values obtained with the exact and MCMC-based procedures. The results of this experiment can be found in Table 4. Sample mean and standard deviations (in parentheses) are reported for log-likelihood as well as adjusted Rand index values (ARI) for both methods. As we can see, the statistics for log-likelihood values are very similar, with those corresponding to the exact procedure being just marginally better. The produced partitions match in all 100 cases as reflected by identical ARI values presented in Table 4. This small experiment suggests that the MCMC-based calculations of posterior probabilities can be successfully employed in our framework.

Table 4
Comparison of exact and MCMC-based methods. Sample mean and standard deviations (in parentheses) of $\log \mathcal{L}$ and ARI values are reported.

Exact		MCMC-based	
$\log \mathcal{L}$	ARI	$\log \mathcal{L}$	ARI
−185.08 (7.040)	1.000 (0.000)	−185.10 (7.051)	1.000 (0.000)

Supplementary material

Supplementary material associated with this article can be found, in the online version, at doi:10.1016/j.patcog.2020.107641.

References

[1] C. Aicher, A.Z. Jacobs, A. Clauset, Learning latent block structure in weighted networks, J. Complex Netw. 3 (2) (2014) 221–248.
[2] E. Airoldi, D. Blei, E. Xing, S. Fienberg, A latent mixed membership model for relational data, in: Proceedings of the 3rd International Workshop on Link Discovery, ACM, 2005, pp. 82–89.
[3] E.M. Airoldi, D.M. Blei, S.E. Fienberg, E.P. Xing, Mixed membership stochastic blockmodels, J. Mach. Learn. Res. 9 (Sep) (2008) 1981–2014.
[4] J.D. Banfield, A.E. Raftery, Model-based Gaussian and non-Gaussian clustering, Biometrics 49(3) (1993) 803–821.
[5] P. Barbillon, S. Donnet, E. Lazega, A. Bar-Hen, Stochastic block models for multiplex networks: an application to a multilevel network of researchers, J. R. Stat. Soc. Ser A 180 (1) (2017) 295–314.
[6] M. Barigozzi, G. Fagiolo, G. Mangioni, Identifying the community structure of the international-trade multi-network, Physica A 390 (11) (2011) 2051–2066.
[7] M. Berlingerio, M. Coscia, F. Giannotti, A. Monreale, D. Pedreschi, Foundations of multidimensional network analysis, in: 2011 International Conference on Advances in Social Networks Analysis and Mining, IEEE, 2011, pp. 485–489.
[8] C. Biernacki, G. Celeux, G. Govaert, Assessing a mixture model for clustering with the integrated completed likelihood, IEEE Trans. Pattern Anal. Mach. Intell. 22 (7) (2000) 719–725.
[9] C. Biernacki, G. Celeux, G. Govaert, Choosing starting values for the EM algorithm for getting the highest likelihood in multivariate Gaussian mixture models, Comput. Stat. Data Anal. 413 (2003) 561–575.
[10] B. Boden, S. Günnemann, H. Hoffmann, T. Seidl, Mining coherent subgraphs in multi-layer graphs with edge labels, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2012, pp. 1258–1266.
[11] C. Bothorel, J.D. Cruz, M. Magnani, B. Micenkova, Clustering attributed graphs: models, measures and methods, Netw. Sci. 3 (3) (2015) 408–444.
[12] N. Bouguila, W. ElGuebaly, Discrete data clustering using finite mixture models, Pattern Recognit. 42 (1) (2009) 33–42.
[13] B. Calderhead, A general construction for parallelizing Metropolis-Hastings algorithms, Proc. Natl. Acad. Sci. 111 (49) (2014) 17408–17413.
[14] G. Celeux, Govaert, Gaussian parsimonious clustering models, Comput. Stat. Data Anal. 28 (1995) 781–793.
[15] A. Clauset, C. Moore, M.E. Newman, Hierarchical structure and the prediction of missing links in networks, Nature 453 (7191) (2008) 98–101.
[16] E. Côme, P. Latouche, Model selection and clustering in stochastic block models based on the exact integrated complete data likelihood, Stat. Model. 15 (6) (2015) 564–589.
[17] A. Dasgupta, A.E. Raftery, Detecting features in spatial point processes with clutter via model-based clustering, J. Am. Stat. Assoc. 93 (441) (1998) 294–302.
[18] J.-J. Daudin, F. Picard, S. Robin, A mixture model for random graphs, Stat. Comput. 18 (2) (2008) 173–183.
[19] A.P. Dawid, Some matrix-variate distribution theory: notational considerations and a bayesian application, Biometrika 68 (1) (1981) 265–274.
[20] M. De Domenico, A. Solé-Ribalta, E. Cozzo, M. Kivelä, Y. Moreno, M.A. Porter, S. Gómez, A. Arenas, Mathematical formulation of multilayer networks, Phys. Rev. X 3 (4) (2013).
[21] A.P. Dempster, N.M. Laird, D.B. Rubin, Maximum likelihood for incomplete data via the EM algorithm (with discussion), J. R. Stat. Soc. Ser. B 39 (1977) 1–38.
[22] X. Dong, P. Frossard, P. Vandergheynst, N. Nefedov, Clustering with multi-layer graphs: a spectral perspective, IEEE Trans. Signal Process. 60 (11) (2012) 5820–5831.
[23] D.M. Dunlavy, T.G. Kolda, W.P. Kegelmeyer, Multilinear algebra for analyzing data with multiple linkages, in: Graph Algorithms in the Language of Linear Algebra, SIAM, 2011, pp. 85–114.
[24] P. Dutilleul, The MLE algorithm for the matrix normal distribution, J. Stat. Comput. Simul. 64 (2) (1999) 105–123.
[25] P. Erdös, A. Rényi, On random graphs i, Publicationes Math. Debrecen 6 (1959) 290–297.
[26] S.E. Fienberg, S.S. Wasserman, Categorical data analysis of single sociometric relations, Sociol. Methodol. 12 (1981) 156–192.
[27] O. Frank, D. Strauss, Markov graphs, J. Am. Stat. Assoc. 81 (395) (1986) 832–842.
[28] T. Funke, T. Becker, Stochastic block models: a comparison of variants and inference methods, PLoS ONE 14 (4) (2019) 1–40.
[29] M.P. Gallaugher, P.D. McNicholas, Finite mixtures of skewed matrix variate distributions, Pattern Recognit. 80 (2018) 83–93.
[30] G. Govaert, M. Nadif, An em algorithm for the block mixture model, IEEE Trans. Pattern Anal. Mach.Intell. 4 (2005) 643–647.
[31] W.K. Hastings, Monte Carlo Sampling Methods Using Markov Chains and Their Applications, Oxford University Press, 1970.
[32] P.W. Holland, K.B. Laskey, S. Leinhardt, Stochastic blockmodels: first steps, Soc Netw. 5 (2) (1983) 109–137.
[33] B. Karrer, M.E. Newman, Stochastic blockmodels and community structure in networks, Phys. Rev. E 83 (1) (2011).
[34] C. Kemp, J.B. Tenenbaum, T.L. Griffiths, T. Yamada, N. Ueda, Learning systems of concepts with an infinite relational model, in: AAAI'06 Proceedings of the 21st National Conference on Artificial Intelligence, vol. 1, 2006, pp. 381–388.
[35] C. Keribin, Consistent estimation of the order of finite mixture models, Sankhyā: Indian J. Stat. 62 (2000) 49–66.
[36] M. Kivelä, A. Arenas, M. Barthelemy, J.P. Gleeson, Y. Moreno, M.A. Porter, Multilayer networks, J. Complex Netw. 2 (3) (2014) 203–271.
[37] T.G. Kolda, B.W. Bader, J.P. Kenny, Higher-order web link analysis using multilinear algebra, in: 5th IEEE International Conference on Data Mining (ICDM'05), IEEE, 2005, pp. 8–pp.
[38] C. Liu, H.-C. Li, K. Fu, F. Zhang, M. Datcu, W.J. Emery, Bayesian estimation of generalized gamma mixture model based on variational em algorithm, Pattern Recognit. 87 (2019) 269–284.
[39] J.S. Liu, F. Liang, W.H. Wong, The multiple-try method and local optimization in metropolis sampling, J. Am. Stat. Assoc. 95 (2012) 121–134.
[40] J. Ma, S. Fu, On the correct convergence of the EM algorithm for gaussian mixtures, Pattern Recognit. 38 (12) (2005) 2602–2611.
[41] L. Martino, A review of multiple try MCMC algorithms for signal processing, Digit Signal Process. 75 (2018) 134–152.
[42] L. Martino, J. Read, On the flexibility of the design of multiple try metropolis schemes, Comput. Stat. 28 (6) (2013) 2797–2823.
[43] V. Melnykov, R. Maitra, Finite mixture models and model-based clustering, Stat. Surv. 4 (2010) 80–116, doi:10.1214/09-SS053.
[44] V. Melnykov, R. Maitra, D. Nettleton, Accounting for spot matching uncertainty in the analysis of proteomics data from two-dimensional gel electrophoresis, Sankhya B 73 (1) (2011) 123.
[45] V. Melnykov, X. Zhu, On model-based clustering of skewed matrix data, J. Multivar Anal. 167 (2018) 181–194.
[46] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equation of state calculations by fast computing machines, J. Chem. Phys. 21 (6) (1953) 1087–1092.
[47] S. Michael, V. Melnykov, An effective strategy for initializing the EM algorithm in finite mixture models, Adv. Data Anal. Classif. 10 (4) (2016) 564–583.
[48] M.E. Newman, Networks: An Introduction, Oxford University Press, 2010.
[49] K. Nowicki, T.A.B. Snijders, Estimation and prediction for stochastic blockstructures, J. Am. Stat. Assoc. 96 (455) (2001) 1077–1087.
[50] T.P. Peixoto, Inferring the mesoscale structure of layered, edge-valued, and time-varying networks, Phys. Rev. E 92 (4) (2015).
[51] S. Sarkar, X. Zhu, V. Melnykov, S. Ingrassia, On parsimonious models for modeling matrix data, Comput. Stat. Data Anal. 142 (2020).
[52] G. Schwarz, Estimating the dimensions of a model, Ann. Stat. 6 (1978) 461–464.

[53] M. Schweinberger, M.S. Handcock, Local dependence in random graph models: characterization, properties and statistical inference, J. Am. Stat. Assoc. 77 (3) (2015) 647–676.
[54] T.A. Snijders, K. Nowicki, Estimation and prediction for stochastic blockmodels for graphs with latent block structure, J. Classif. 14 (1) (1997) 75–100.
[55] T.A. Snijders, P.E. Pattison, G.L. Robins, M.S. Handcock, New specifications for exponential random graph models, Sociol. Methodol. 36 (1) (2006) 99–153.
[56] M.S. Srivastava, T. Rosen, D. Rosen, Models with a Kronecker product covariance structure: estimation and testing, Math. Methods Stat. 17 (4) (2008) 357–370.
[57] C. Viroli, Finite mixtures of matrix normal distributions for classifying three-way data, Stat. Comput. 21 (4) (2011) 511–522.
[58] D.Q. Vu, D.R. Hunter, M. Schweinberger, Model-based clustering of large networks, Ann. Appl. Stat. 7 (2) (2013) 1010–1039.
[59] P. Wang, G. Robins, P. Pattison, E. Lazega, Exponential random graph models for multilevel networks, Soc. Netw. 35 (1) (2013) 96–115.
[60] Y.J. Wang, G.Y. Wong, Stochastic blockmodels for directed graphs, J. Am. Stat. Assoc. 82 (397) (1987) 8–19.
[61] S. Wasserman, K. Faust, Social Network Analysis: Methods and Applications, Cambridge University Press, 1994.
[62] S. Wasserman, P. Pattison, Logit models and logistic regressions for social networks: i. An introduction to Markov graphs and p^*, Psychometrika 61 (3) (1996) 401–425.
[63] H.C. White, S.A. Boorman, R.L. Breiger, Social structure from multiple networks. I. Blockmodels of roles and positions, Am. J. Sociol. 81 (4) (1976) 730–780.
[64] H. Zanghi, C. Ambroise, V. Miele, Fast online graph clustering via Erdős–Rényi mixture, Pattern Recognit. 41 (12) (2008) 3592–3599.
[65] H. Zanghi, F. Picard, V. Miele, C. Ambroise, Strategies for online inference of model-based clustering in large and growing networks, Ann.Appl. Stat. 4 (2) (2010) 687–714.
[66] H. Zanghi, S. Volant, C. Ambroise, Clustering based on random graph model embedding vertex features, Pattern Recognit. Lett. 31 (9) (2010) 830–836.
[67] B. Zhang, C. Zhang, X. Yi, Competitive em algorithm for finite mixture models, Pattern Recognit. 37 (1) (2004) 131–144.
[68] J. Zhang, T. Chen, J. Hu, On the relationship between gaussian stochastic blockmodels and label propagation algorithms, J. Stat. Mech. 2015 (3) (2015) 1–21.

Volodymyr Melnykov received his Ph.D. degree in Statistics from Iowa State University in 2009. He also received a M.S. degree in Applied Statistics from Bowling Green State University. He is currently a Professor at the University of Alabama. He also serves on the Board of Directors of Classification Society of North America. His main research interests include model based clustering methods, clustering high-dimensional objects, and data visualization.

Shuchismita Sarkar received her Ph.D. degree in Applied Statistics at the University of Alabama in 2019, She also has a M.S. in Applied Statistics and Informatics from Indian Institute of Technology, Bombay and a M.S. degree in Applied Statistics from Western Michigan University. She has 7+ years of work experience in credit card analytics industry. She is currently an Assistant Professor of Applied Statistics and Operation Research in Bowling Green State University. Her research interests lies in machine learning with a primary focus on model-based clustering.

Yana Melnykov is an Assistant Professor of Applied Statistics at the University of Alabama. She received her Ph.D. degree from the University of Alabama in 2017. She also holds a M.S. degree in Statistics from North Dakota State University. Her areas of research include finite mixture modeling, change point processes, and anomaly detection.