Clustering Multivariate Count Data via Dirichlet-Multinomial Network Fusion

Xin Zhao[a], Jingru Zhang[b], and Wei Lin[a,*]

Abstract

Clustering of multivariate count data has widespread applications in areas such as text analysis and microbiome studies. The need to account for overdispersion generally results in a nonconvex loss function, which does not fit into the existing convex clustering framework. Moreover, prior knowledge of a network over the samples, often available from citation or similarity relationships, is not taken into account. We introduce Dirichlet-multinomial network fusion (DMNet) for clustering multivariate count data, which models the samples via Dirichlet-multinomial distributions with individual parameters and employs a weighted group L_1 fusion penalty to pursue homogeneity over a prespecified network. To circumvent the nonconvexity issue, we present two exponential family approximations to the Dirichlet-multinomial distribution, which are amenable to efficient optimization and theoretical analysis. We derive an ADMM algorithm and establish nonasymptotic error bounds for the proposed methods. Our bounds involve a trade-off between the connectivity of the network and its fidelity to the true parameter. The usefulness of our methods is illustrated through simulation studies and two text clustering applications.

Keywords: Convex clustering; Exponential family approximation; Group L_1 fusion; Nonasymptotic error bound; Overdispersion; Text analysis

[a]School of Mathematical Sciences and Center for Statistical Science, Peking University, Beijing 100871, China

[b]Department of Biostatistics, Epidemiology and Informatics, Perelman School of Medicine, University of Pennsylvania, Philadelphia, PA 19104, USA

[*]Corresponding author. E-mail address: weilin@math.pku.edu.cn.

1 Introduction

Cluster analysis is an unsupervised learning task central to various applications. It aims to partition objects into groups such that objects within the same group tend to be more similar than those from different groups. Existing clustering techniques fall into two major classes. One class is by its nature algorithmic and heuristic. This includes linkage-based hierarchical clustering (Ackerman and Ben-David, 2016), which takes a greedy strategy to form the clusters, and K-means clustering (Steinley, 2006), which uses an iterative descent algorithm to find an approximate solution to a combinatorial optimization problem. The other class adopts a probabilistic formulation and performs the clustering on the basis of a probability model, usually a parametric or nonparametric mixture model (Bouveyron et al., 2019). See, for example, Everitt et al. (2011) for an overview.

While clustering of continuous and certain types of discrete data has been extensively studied, clustering of multivariate count data has received little attention. Such data are ubiquitous in applications ranging from genomics and ecology to text analysis and marketing research. For instance, in text analysis, each document can be summarized as frequency counts of words and phrases; in microbiome studies, metagenomic sequencing yields microbial taxa counts for each sample. In these applications, the multinomial proportions are subject to substantial individual variability. As a result, the count data are overdispersed in the sense that the observed count variances are much greater than those predicted by the multinomial distribution with fixed proportions. To account for the overdispersion, the Dirichlet-multinomial (DM) distribution (Mosimann, 1962) has gained wide popularity and has served as a building block in many generative and regression models (Blei, Ng, and Jordan, 2003; Chen and Li, 2013). Combining the DM distribution with the mixture modeling framework leads to Dirichlet-multinomial mixture (DMM) models, which have been developed for text and microbiome clustering problems (Nigam et al., 2000; Holmes, Harris, and Quince, 2012; Anderlucci and Viroli, 2020). As with other mixture models, maximum likelihood estimation for the DMM model via the EM algorithm suffers from the issues of local minima and slow convergence. The computational difficulties deterio-

rate in high dimensions where hundreds or thousands of words or taxa are to be analyzed simultaneously.

Inspired by the success of the fused Lasso (Tibshirani et al., 2005), convex clustering has been proposed as a remedy for the instability of classical clustering methods (Pelckmans et al., 2005; Hocking et al., 2011; Lindsten, Ohlsson, and Ljung, 2011). By using a (group) L_1 fusion penalty, the method encourages the cluster centers to take on only a small number of distinct values. It can be viewed as a convex relaxation of the L_0 problems related to linkage-based hierarchical clustering and K-means clustering. The convex formulation allows for the development of both efficient optimization algorithms (Chi and Lange, 2015) and strong theoretical guarantees (Zhu et al., 2014; Tan and Witten, 2015; Radchenko and Mukherjee, 2017; Chi and Steinerberger, 2019). A similar model-based clustering approach has also been proposed by combining the L_1 fusion penalty with the Gaussian mixture model (Guo et al., 2010). All the above work, however, focuses on Euclidean distance and continuous data. It is not clear how the methodology and theory would extend to discrete, and in particular multivariate count, data.

In addition to the observations, prior knowledge of a network over the samples is often available in practice, which may provide useful information for the clustering problem. For instance, in bibliometrics, citations between two papers would suggest similar topics and hence memberships to the same cluster; in gut microbiome studies, similarities on dietary patterns would indicate that two microbiomes tend to be clustered in the same enterotype (Wu et al., 2011). In principle, such prior network information can be incorporated into the convex clustering framework by appropriately choosing the affinity parameters or weights in the adaptive L_1 fusion penalty (Hocking et al., 2011). Hallac, Leskovec, and Boyd (2015) developed scalable algorithms based on the alternating direction method of multipliers (ADMM) for the network Lasso problem. Chi and Steinerberger (2019) established theoretical guarantees for recovering a partition tree, but required that the edge weights be perfectly specified by the tree. Nevertheless, the prior work did not consider the modeling of multivariate count data or explain the impact of the network structure in general.

To bridge the gap in the literature, we introduce Dirichlet-multinomial network fusion (DMNet) for clustering multivariate count data. The method models the samples via DM distributions with individual parameters and employs a weighted group L_1 fusion penalty to pursue homogeneity over a prespecified network. It can be regarded as a hard version of the DMM model, with deterministic cluster assignments determined by a convex fusion penalty. The DMNet method, however, has the important limitation that, since the log-likelihood function for the DM model is nonconcave, the optimization problem is still nonconvex. To circumvent the nonconvexity issue, we further present two exponential family approximations, one previously proposed by Elkan (2006) and one novel, to the DM distribution. This results in two convex formulations, DMNet+ and DMNet++, which are amenable to efficient optimization and theoretical analysis. We derive an ADMM algorithm for implementing the proposed methods. Moreover, we establish theoretical guarantees for DMNet++ in terms of nonasymptotic error bounds, which shed light on the role of the network structure in the clustering problem.

The rest of the article proceeds as follows. Section 2 introduces our model and methodology. Section 3 describes the algorithms for the proposed methods. Nonasymptotic theory for DMNet++ is presented in Section 4. Simulation studies and two real data applications are given in Sections 5 and 6, respectively. Section 7 concludes the article with some discussion. Proofs of theoretical results are provided in the Appendix.

2 Model and Methodology

2.1 Dirichlet-Multinomial Model and DMNet

Suppose we observe the multivariate counts $\mathbf{y}_i = (y_{i1}, \ldots, y_{ip})^T$ for sample i and let $\mathbf{y} = (\mathbf{y}_1^T, \ldots, \mathbf{y}_n^T)^T$. The Dirichlet-multinomial (DM) model (Mosimann, 1962) assumes that $\mathbf{y}_i$ are multinomial with total counts $N_i = \sum_{j=1}^p y_{ij}$ and cell probabilities $\boldsymbol{\pi}_i$ and that $\boldsymbol{\pi}_i$ are Dirichlet with parameters $\boldsymbol{\alpha}_i = (\alpha_{i1}, \ldots, \alpha_{ip})^T$, where $\alpha_{ij} > 0$; that is,

$$\begin{aligned} \mathbf{y}_i \,|\, \boldsymbol{\pi}_i &\sim \text{Mult}(N_i, \boldsymbol{\pi}_i), \\ \boldsymbol{\pi}_i &\sim \text{Dir}(\boldsymbol{\alpha}_i), \quad i = 1, \ldots, n. \end{aligned} \tag{1}$$

Note that we have overparametrized the DM model with individual parameters $\boldsymbol{\alpha}_i$, so that distinct values of $\boldsymbol{\alpha}_i$ would correspond to different clusters. The joint density of $\mathbf{y}_i$ and $\boldsymbol{\pi}_i$ is given by

$$f(\mathbf{y}_i, \boldsymbol{\pi}_i; \boldsymbol{\alpha}_i) = \frac{N_i!}{\prod_{j=1}^p y_{ij}!} \prod_{j=1}^p \pi_{ij}^{y_{ij}} \times \frac{\Gamma(\alpha_i^+)}{\prod_{j=1}^p \Gamma(\alpha_{ij})} \prod_{j=1}^p \pi_{ij}^{\alpha_{ij}-1},$$

where $\alpha_i^+ = \sum_{j=1}^p \alpha_{ij}$. By integrating out $\boldsymbol{\pi}_i$ over the $(p-1)$-simplex $\mathbb{S}^{p-1}$, we obtain the marginal density

$$\begin{aligned} f(\mathbf{y}_i; \boldsymbol{\alpha}_i) &= \frac{N_i!}{\prod_{j=1}^p y_{ij}!} \frac{\Gamma(\alpha_i^+)}{\prod_{j=1}^p \Gamma(\alpha_{ij})} \int_{\mathbb{S}^{p-1}} \prod_{j=1}^p \pi_{ij}^{y_{ij}+\alpha_{ij}-1} \, d\boldsymbol{\pi}_i \\ &= \frac{N_i!}{\prod_{j=1}^p y_{ij}!} \frac{\Gamma(\alpha_i^+)}{\Gamma\left(N_i + \alpha_i^+\right)} \prod_{j=1}^p \frac{\Gamma(y_{ij} + \alpha_{ij})}{\Gamma(\alpha_{ij})}. \end{aligned} \tag{2}$$

The log-likelihood for the whole dataset is then, up to a constant,

$$\ell(\boldsymbol{\alpha}) = \frac{1}{n} \sum_{i=1}^n \left[\log \Gamma(\alpha_i^+) - \log \Gamma(N_i + \alpha_i^+) + \sum_{j=1}^p \{\log \Gamma(y_{ij} + \alpha_{ij}) - \log \Gamma(\alpha_{ij})\} \right], \tag{3}$$

where $\boldsymbol{\alpha} = (\boldsymbol{\alpha}_1^T, \ldots, \boldsymbol{\alpha}_n^T)^T$.

Furthermore, we assume that the relationships among the samples are specified by a network or weighted graph $G = (V, E, W)$, where each node in $V = \{1, \ldots, n\}$ represents a sample, each edge in $E \subset V \times V$ indicates a tendency for two adjacent nodes to be clustered together, and $W = (w_{ij})_{(i,j)\in E}$ consists of edge weights that measure the strengths of the tendencies. In our text data example, such a graph is given by the citation network over the papers, which provides prior information about the similarities between papers. In order to pursue homogeneity over the network and perform the clustering, we adopt a weighted group L_1 fusion penalty and consider the optimization problem

$$\widehat{\boldsymbol{\alpha}} = \underset{\boldsymbol{\alpha} \in \mathbb{R}_+^{np}}{\arg\min} \left\{ -\ell(\boldsymbol{\alpha}) + \lambda \sum_{(i,j)\in E} w_{ij} \|\log \boldsymbol{\alpha}_i - \log \boldsymbol{\alpha}_j\|_2 \right\},$$

or, after the reparametrization $\boldsymbol{\theta} = \log \boldsymbol{\alpha}$,

$$\widehat{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta} \in \mathbb{R}^{np}}{\arg\min} \left\{ -\widetilde{\ell}(\boldsymbol{\theta}) + \lambda \sum_{(i,j)\in E} w_{ij} \|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|_2 \right\}, \tag{4}$$

where $\widetilde{\ell}(\boldsymbol{\theta}) = \ell(e^{\boldsymbol{\theta}})$ and $\lambda > 0$ is a tuning parameter that controls the trade-off between a good fit to the overparametrized DM model and a good agreement on the parameters

over the network. A proper choice of λ should strike a balance between two extremes: when $\lambda = 0$, each sample forms a different cluster; when $\lambda = \infty$, all samples in the same connected component are clustered together. We call problem (4) the Dirichlet-multinomial network fusion (DMNet).

2.2 The DMNet+ Approximation

Although the DM distribution has been widely used to model text and microbiome data, theoretical understanding of the model has long been lacking. The major obstacle is that the distribution is not in an exponential family, so that many theoretical tools do not apply. As a consequence, the loss function in problem (4) is nonconvex and does not fit into the existing convex clustering framework. To simplify the intractable form of the DM distribution, Elkan (2006) proposed an exponential family approximation which takes the sparsity of text data into account. Note first that the case of $y_{ij} = 0$ does not contribute to the density (2) and we can write

$$f(\mathbf{y}_i; \boldsymbol{\alpha}_i) = \frac{N_i! \Gamma(\alpha_i^+)}{\Gamma\big(N_i + \alpha_i^+\big)} \prod_{j: y_{ij} \geq 1} \frac{\Gamma(y_{ij} + \alpha_{ij})}{y_{ij}! \Gamma(\alpha_{ij})}.$$

In view of the fact that most words have zero or small counts, it is reasonable to assume that $\alpha_{ij} \ll 1$ for most j. For $y_{ij} \geq 1$, using the approximation

$$\frac{\Gamma(y_{ij} + \alpha_{ij})}{\Gamma(\alpha_{ij})} = \prod_{k=0}^{y_{ij}-1} (k + \alpha_{ij}) \sim (y_{ij} - 1)! \alpha_{ij} \tag{5}$$

when α_{ij} is small, we can approximate the density (2) by

$$f^E(\mathbf{y}_i; \boldsymbol{\alpha}_i) = \frac{N_i! \Gamma(\alpha_i^+)}{\Gamma\big(N_i + \alpha_i^+\big)} \prod_{j: y_{ij} \geq 1} \frac{\alpha_{ij}}{y_{ij}}. \tag{6}$$

The resulting approximation to the log-likelihood (3) is

$$\ell^E(\boldsymbol{\alpha}) = \frac{1}{n} \sum_{i=1}^{n} \left\{ \log \Gamma(\alpha_i^+) - \log \Gamma(N_i + \alpha_i^+) + \sum_{j=1}^{p} I(y_{ij} \geq 1) \log \alpha_{ij} \right\}, \tag{7}$$

where $I(\cdot)$ is the indicator function. This corresponds to an exponential family with natural parameter $\boldsymbol{\theta} = \log \boldsymbol{\alpha}$. Reparametrizing and combining with the weighted group L_1 fusion

penalty leads to the optimization problem

$$\widehat{\boldsymbol{\theta}}^E = \underset{\boldsymbol{\theta}\in\mathbb{R}^{np}}{\arg\min}\left\{-\widetilde{\ell}^E(\boldsymbol{\theta}) + \lambda \sum_{(i,j)\in E} w_{ij}\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|_2\right\}. \tag{8}$$

We call (6) the DM+ model and (8) the DMNet+ problem.

Note that (6) is not an exact density because it is not normalized. Nevertheless, the next lemma characterizes how close it is to the density (2). By the approximation (5), if y_{ij} are bounded, then for any $0 < \varepsilon < 1$, there exists $0 < \delta < 1$ such that $|\Gamma(y_{ij} + \alpha_{ij})/\Gamma(\alpha_{ij}) - (y_{ij}-1)!\alpha_{ij}| \le \varepsilon$ whenever $\alpha_{ij} \le \delta$. Define $J_{i1}(\delta) = \{j : y_{ij} \ge 1, \alpha_{ij} \le \delta\}$ and $J_{i2}(\delta) = \{j : y_{ij} \ge 1, \alpha_{ij} > \delta\}$ with cardinalities $|J_{i1}(\delta)| \asymp s_1(p)$ and $|J_{i2}(\delta)| \asymp s_2(p)$.

Lemma 1. *Assume that there exist constants $L, M > 0$ such that*

$$\frac{N_i!\Gamma(\alpha_i^+)}{\Gamma(N_i + \alpha_i^+)} \le L \tag{9}$$

and on $J_{i2}(\delta)$,

$$\frac{\Gamma(y_{ij} + \alpha_{ij})}{y_{ij}!\Gamma(\alpha_{ij})} \le M \tag{10}$$

for all $i = 1, \ldots, n$. Then

$$\max_i |f(\mathbf{y}_i; \boldsymbol{\alpha}_i) - f^E(\mathbf{y}_i; \boldsymbol{\alpha}_i)| \le LM^{s_2(p)}(\varepsilon^{s_1(p)} + 2\delta^{s_1(p)}).$$

Assumptions (9) and (10) are met provided that α_i^+ are not too small and α_{ij} are not too large on $J_{i2}(\delta)$. The approximation bound tends to zero whenever $s_1(p) \gg s_2(p)$. As observed by Elkan (2006), these assumptions are likely to hold for real text data.

2.3 The DMNet++ Approximation

Although the exponential family approximation (6) simplifies the form of the density (2), it is still cumbersome to work with. In particular, it does not factorize over the N_i multinomial samplings, so that the large-N_i behavior of the sampling process is not easy to characterize. Also, the gamma function complicates and slows down the optimization. To resolve these issues, we now derive a new exponential family approximation that is more convenient from both theoretical and computational perspectives.

Denote by $\mathbf{z}_i^{(m)} = (z_{i1}^{(m)}, \ldots, z_{ip}^{(m)})^T$ the one-hot encoding of the mth outcome for sample i, so that $\mathbf{y}_i = \sum_{m=1}^{N_i} \mathbf{z}_i^{(m)}$. The DM model (1) can be rewritten as

$$\mathbf{z}_i^{(m)} \,|\, \boldsymbol{\pi}_i \sim \mathrm{Mult}(1, \boldsymbol{\pi}_i), \quad m = 1, \ldots, N_i,$$
$$\boldsymbol{\pi}_i \sim \mathrm{Dir}(\boldsymbol{\alpha}_i), \quad i = 1, \ldots, n.$$

The random vectors $\mathbf{z}_i^{(m)}$ are dependent because of the latent variables $\boldsymbol{\pi}_i$. Nevertheless, we treat them as independent and find the mean-field approximation

$$\prod_{m=1}^{N_i} f_i^M(\mathbf{z}_i^{(m)}; \boldsymbol{\alpha}_i) = \prod_{m=1}^{N_i} \prod_{j=1}^{p} \left(\frac{\alpha_{ij}}{\alpha_i^+}\right)^{z_{ij}^{(m)}} = \prod_{j=1}^{p} \left(\frac{\alpha_{ij}}{\alpha_i^+}\right)^{y_{ij}},$$

where $f_i^M(\cdot; \boldsymbol{\alpha}_i)$ is the density of $\mathbf{z}_i^{(m)}$ from (2) with $N_i = 1$. This gives the approximating density

$$f^M(\mathbf{y}_i; \boldsymbol{\alpha}_i) = \frac{N_i!}{\prod_{j=1}^p y_{ij}!} \prod_{j=1}^{p} \left(\frac{\alpha_{ij}}{\alpha_i^+}\right)^{y_{ij}} \tag{11}$$

and the log-likelihood

$$\ell^M(\boldsymbol{\alpha}) = \frac{1}{n} \sum_{i=1}^{n} \left(-N_i \log \alpha_i^+ + \sum_{j=1}^{p} y_{ij} \log \alpha_{ij} \right). \tag{12}$$

By reparametrizing and incorporating the penalty, we arrive at the optimization problem

$$\widehat{\boldsymbol{\theta}}^M = \operatorname*{arg\,min}_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} \left\{ -\widetilde{\ell}^M(\boldsymbol{\theta}) + \lambda \sum_{(i,j) \in E} w_{ij} \|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|_2 \right\}. \tag{13}$$

We call (11) the DM++ model and (13) the DMNet++ problem. Comparing the approximations (7) and (12), we see that the latter is much simpler and does not involve the gamma function. A caveat is that under the latter approximation, $\boldsymbol{\alpha}_i$ are identifiable only up to a multiplicative constant, and correspondingly $\boldsymbol{\theta}_i$ are identifiable only up to an additive constant. This issue can be resolved by restricting the parameter space $\boldsymbol{\Theta}$ in a manner to be made precise later.

To illustrate the similarity between the DM model and the DM+ and DM++ approximations, we generated from the DM model $n = 600$ texts in three groups with parameters specified in Section 5.1. The log probabilities of these texts from the DM, DM+, and DM++ models are shown in Figure 1. We see that, while both approximations work sufficiently well, DM++ tends to be a better approximation to the DM model.

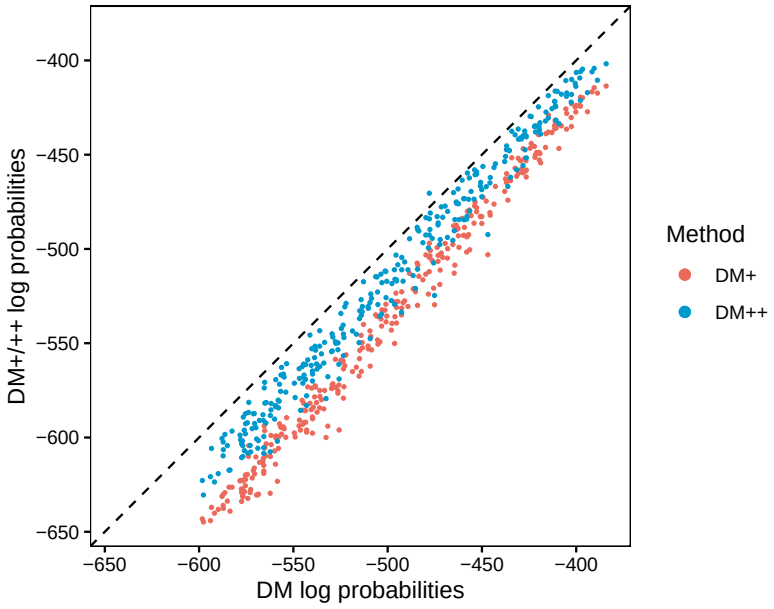


Figure 1: DM versus DM+ and DM++ log probabilities.

3 Optimization

This section details the optimization algorithms, initialization strategy, and tuning parameter selection for the proposed DMNet, DMNet+, and DMNet++ methods.

3.1 ADMM Algorithm

We follow the general framework of Hallac, Leskovec, and Boyd (2015) to derive an ADMM algorithm for solving problems (4), (8), and (13). To decouple the variables from adjacent edges, we introduce auxiliary variables $\boldsymbol{\eta}_{ij}$ and write problem (4) as

$$\begin{aligned} \text{minimize} \quad & -\widetilde{\ell}(\boldsymbol{\theta}) + \lambda \sum_{(i,j)\in E} w_{ij} \|\boldsymbol{\eta}_{ij} - \boldsymbol{\eta}_{ji}\|_2 \\ \text{subject to} \quad & \boldsymbol{\theta}_i = \boldsymbol{\eta}_{ij}, \quad i = 1, \ldots, n, \quad j \in N(i), \end{aligned}$$

where $N(i) = \{j : (i,j) \in E\}$. The augmented Lagrangian in scaled form is

$$\begin{aligned} L_\rho(\boldsymbol{\theta}, \boldsymbol{\eta}, \mathbf{u}) = -\widetilde{\ell}(\boldsymbol{\theta}) + \sum_{(i,j)\in E} \Big\{ & \lambda w_{ij} \|\boldsymbol{\eta}_{ij} - \boldsymbol{\eta}_{ji}\|_2 - \frac{\rho}{2}(\|\mathbf{u}_{ij}\|_2^2 + \|\mathbf{u}_{ji}\|_2^2) \\ & + \frac{\rho}{2}(\|\boldsymbol{\theta}_i - \boldsymbol{\eta}_{ij} + \mathbf{u}_{ij}\|_2^2 + \|\boldsymbol{\theta}_j - \boldsymbol{\eta}_{ji} + \mathbf{u}_{ji}\|_2^2) \Big\}, \end{aligned}$$

where $\mathbf{u}_{ij}$ are the scaled dual variables and $\rho > 0$ is the penalty parameter. The ADMM updates at iteration t then consist of the following steps:

$$\boldsymbol{\theta}_i^{(t+1)} = \arg\min_{\boldsymbol{\theta}_i} \left\{ -\widetilde{\ell}(\boldsymbol{\theta}) + \sum_{j \in N(i)} \frac{\rho}{2} \|\boldsymbol{\theta}_i - \boldsymbol{\eta}_{ij}^{(t)} + \mathbf{u}_{ij}^{(t)}\|_2^2 \right\}, \tag{14}$$

$$\begin{aligned}(\boldsymbol{\eta}_{ij}^{(t+1)}, \boldsymbol{\eta}_{ji}^{(t+1)}) = \arg\min_{\boldsymbol{\eta}_{ij}, \boldsymbol{\eta}_{ji}} \Big\{ & \lambda w_{ij} \|\boldsymbol{\eta}_{ij} - \boldsymbol{\eta}_{ji}\|_2 \\ & + \frac{\rho}{2} (\|\boldsymbol{\theta}_i^{(t+1)} - \boldsymbol{\eta}_{ij} + \mathbf{u}_{ij}^{(t)}\|_2^2 + \|\boldsymbol{\theta}_j^{(t+1)} - \boldsymbol{\eta}_{ji} + \mathbf{u}_{ji}^{(t)}\|_2^2) \Big\},\end{aligned} \tag{15}$$

$$\mathbf{u}_{ij}^{(t+1)} = \mathbf{u}_{ij}^{(t)} + \boldsymbol{\theta}_i^{(t+1)} - \boldsymbol{\eta}_{ij}^{(t+1)}.$$

The subproblem (15) has the closed-form solution (Hallac, Leskovec, and Boyd, 2015)

$$\begin{aligned}\boldsymbol{\eta}_{ij}^{(t+1)} &= r_{ij}(\boldsymbol{\theta}_i^{(t+1)} + \mathbf{u}_{ij}^{(t)}) + (1 - r_{ij})(\boldsymbol{\theta}_j^{(t+1)} + \mathbf{u}_{ji}^{(t)}), \\ \boldsymbol{\eta}_{ji}^{(t+1)} &= (1 - r_{ij})(\boldsymbol{\theta}_i^{(t+1)} + \mathbf{u}_{ij}^{(t)}) + r_{ij}(\boldsymbol{\theta}_j^{(t+1)} + \mathbf{u}_{ji}^{(t)}),\end{aligned}$$

where

$$r_{ij} = \max\left(1 - \frac{\lambda w_{ij}}{\rho \|\boldsymbol{\theta}_i^{(t+1)} + \mathbf{u}_{ij}^{(t)} - (\boldsymbol{\theta}_j^{(t+1)} + \mathbf{u}_{ji}^{(t)})\|_2}, \frac{1}{2}\right).$$

The subproblem (14) can be solved by expectation–maximization (EM), minorization–maximization (MM), or Newton-type methods; see, for example, Zhou and Lange (2010). However, for solving subproblems of ADMM, less accurate yet simpler methods are usually sufficient. Note that the optimality condition for (14) is

$$-\nabla_{\boldsymbol{\theta}_i} \widetilde{\ell}(\boldsymbol{\theta}) + \rho \sum_{j \in N(i)} (\boldsymbol{\theta}_i - \boldsymbol{\eta}_{ij}^{(t)} + \mathbf{u}_{ij}^{(t)}) = \mathbf{0}.$$

Rearranging terms yields the implicit gradient descent equation

$$\boldsymbol{\theta}_i = \frac{1}{\rho |N(i)|} \nabla_{\boldsymbol{\theta}_i} \widetilde{\ell}(\boldsymbol{\theta}) + \frac{1}{|N(i)|} \sum_{j \in N(i)} (\boldsymbol{\eta}_{ij}^{(t)} - \mathbf{u}_{ij}^{(t)}).$$

To solve for $\boldsymbol{\theta}_i$, we apply the fixed point iteration

$$\boldsymbol{\theta}_i^{(t+1)} = \frac{1}{\rho |N(i)|} \nabla_{\boldsymbol{\theta}_i} \widetilde{\ell}(\boldsymbol{\theta}^{(t)}) + \frac{1}{|N(i)|} \sum_{j \in N(i)} (\boldsymbol{\eta}_{ij}^{(t)} - \mathbf{u}_{ij}^{(t)}),$$

which guarantees convergence for sufficiently large ρ. A similar algorithm was derived and analyzed by Yin et al. (2018). Here, for the DMNet problem (4), the gradient is given by

$$\frac{\partial}{\partial \theta_{ij}} \widetilde{\ell}(\boldsymbol{\theta}) = \frac{1}{n} \left\{ \Psi\left(\sum_{k=1}^{p} e^{\theta_{ik}}\right) - \Psi\left(N_i + \sum_{k=1}^{p} e^{\theta_{ik}}\right) + \Psi(y_{ij} + e^{\theta_{ij}}) - \Psi(e^{\theta_{ij}}) \right\} e^{\theta_{ij}}, \tag{16}$$

where $\Psi(\cdot)$ is the digamma function. The resulting ADMM algorithm is summarized in Algorithm 1, where for DMNet+ and DMNet++ the gradient should be replaced by

$$\frac{\partial}{\partial\theta_{ij}}\widetilde{\ell}^{E}(\boldsymbol{\theta}) = \frac{1}{n}\left[\left\{\Psi\left(\sum_{k=1}^{p} e^{\theta_{ik}}\right) - \Psi\left(N_i + \sum_{k=1}^{p} e^{\theta_{ik}}\right)\right\}e^{\theta_{ij}} + I(y_{ij} \geq 1)\right],$$

and

$$\frac{\partial}{\partial\theta_{ij}}\widetilde{\ell}^{M}(\boldsymbol{\theta}) = \frac{1}{n}\left(y_{ij} - \frac{N_i e^{\theta_{ij}}}{\sum_{k=1}^{p} e^{\theta_{ik}}}\right),$$

respectively.

Algorithm 1 ADMM algorithm for DMNet

Initialize $\boldsymbol{\theta}_i$, $\boldsymbol{\eta}_{ij} = \boldsymbol{\theta}_i$, and $\mathbf{u}_{ij} = \mathbf{0}$ for $i = 1, \ldots, n$, $j \in N(i)$

while not converged **do**

Apply the update for a certain number of times:

$\boldsymbol{\theta}_i \leftarrow \frac{1}{\rho|N(i)|}\nabla_{\boldsymbol{\theta}_i}\widetilde{\ell}(\boldsymbol{\theta}) + \frac{1}{|N(i)|}\sum_{j\in N(i)}(\boldsymbol{\eta}_{ij} - \mathbf{u}_{ij})$ with the gradient given by (16)

$r_{ij} \leftarrow \max\left(1 - \frac{\lambda w_{ij}}{\rho\|\boldsymbol{\theta}_i + \mathbf{u}_{ij} - (\boldsymbol{\theta}_j + \mathbf{u}_{ji})\|_2}, \frac{1}{2}\right)$

$\boldsymbol{\eta}_{ij} \leftarrow r_{ij}(\boldsymbol{\theta}_i + \mathbf{u}_{ij}) + (1 - r_{ij})(\boldsymbol{\theta}_j + \mathbf{u}_{ji})$

$\boldsymbol{\eta}_{ji} \leftarrow (1 - r_{ij})(\boldsymbol{\theta}_i + \mathbf{u}_{ij}) + r_{ij}(\boldsymbol{\theta}_j + \mathbf{u}_{ji})$

$\mathbf{u}_{ij} \leftarrow \mathbf{u}_{ij} + \boldsymbol{\theta}_i - \boldsymbol{\eta}_{ij}$

end while

The penalty parameter ρ inversely controls the step size in the optimization process and should be set large enough to ensure the convergence of the ADMM algorithm. Roughly speaking, ρ should be chosen proportional to the nonconvexity of the loss function; see the condition of Theorem 1 in Li and Pong (2016).

3.2 Initialization of DM Parameters

For the nonconvex DMNet problem, it would be critical to choose good initial values for the DM parameters $\boldsymbol{\alpha}_i$ or $\boldsymbol{\theta}_i$. Since each node is similar to its neighbors on the network by assumption, we treat them as a DM population with common parameters $\boldsymbol{\alpha}_i$ and each node as a group with individual parameters $\boldsymbol{\pi}_i$, and follow the idea of Weir and Hill (2002) to derive a moment estimator for $\boldsymbol{\alpha}_i$. To this end, let $N^*(i) = N(i) \cup \{i\}$ and define the group averages $\widehat{\pi}_{ij} = y_{ij}/N_i$ and population averages $\bar{\pi}_{ij} = \sum_{k\in N^*(i)} y_{kj}/N_i^+$, where

$N_i^+ = \sum_{k \in N^*(i)} N_k$. The between-group and within-group sums of squares are expressed as

$$S_{ij} = \frac{1}{|N(i)|} \sum_{k \in N^*(i)} N_i (\widehat{\pi}_{kj} - \bar{\pi}_{ij})^2$$

and

$$T_{ij} = \frac{1}{N_i^+ - |N(i)| - 1} \sum_{k \in N^*(i)} N_k \widehat{\pi}_{kj} (1 - \widehat{\pi}_{kj}),$$

respectively. By direct calculation, we obtain

$$ES_{ij} = \frac{\alpha_{ij}}{\alpha_i^+} \left(1 - \frac{\alpha_{ij}}{\alpha_i^+}\right) (1 - \gamma_i + \widetilde{N}_i \gamma_i),$$
$$ET_{ij} = \frac{\alpha_{ij}}{\alpha_i^+} \left(1 - \frac{\alpha_{ij}}{\alpha_i^+}\right) (1 - \gamma_i),$$

where $\gamma_i = 1/(1 + \alpha_i^+)$ are the overdispersion parameters and

$$\widetilde{N}_i = \frac{1}{|N(i)|} \left(N_i^+ - \frac{\sum_{k \in N^*(i)} N_k^2}{N_i^+} \right).$$

Replacing the expectations by sample quantities, summing over all j, and solving for γ_i yields the moment estimator

$$\widehat{\gamma}_i = \frac{\sum_{j=1}^p S_{ij} - \sum_{j=1}^p T_{ij}}{\sum_{j=1}^p S_{ij} + (\widetilde{N}_i - 1) \sum_{j=1}^p T_{ij}}.$$

The initial values $\boldsymbol{\alpha}_i^{(0)} = (\alpha_{i1}^{(0)}, \ldots, \alpha_{ip}^{(0)})$ are then set to

$$\alpha_{ij}^{(0)} = \frac{1 - \widehat{\gamma}_i}{\widehat{\gamma}_i} \bar{\pi}_{ij}.$$

Alternatively, one can initialize the DMNet problem using the solutions to DMNet+ and DMNet++ problems. Our numerical experience suggests that this may further improve the estimation, but generally not the clustering, performance of DMNet.

3.3 Tuning Parameter Selection

The performance of our proposed methods hinges on the choice of the tuning parameter λ. Tan and Witten (2015) suggested an approach to choosing λ in the denoising setting based on the extended Bayesian information criterion (Chen and Chen, 2008) and an unbiased estimator of the degrees of freedom. It is unclear, however, whether their estimator of

the degrees of freedom can be extended to our model and general graphs. Besides, its computation involves inversion of large matrices and is computationally prohibitive when both n and p are large. Here we propose to choose λ by K-fold cross-validation, which is easy to implement. In the kth split, denote the training set and test set by S_k and T_k, respectively. The cluster membership of each sample in T_k is determined by maximizing the log-likelihood over the distinct values of $\widehat{\boldsymbol{\alpha}}_j$ obtained from S_k. For DMNet, we choose the optimal λ that minimizes the cross-validation error

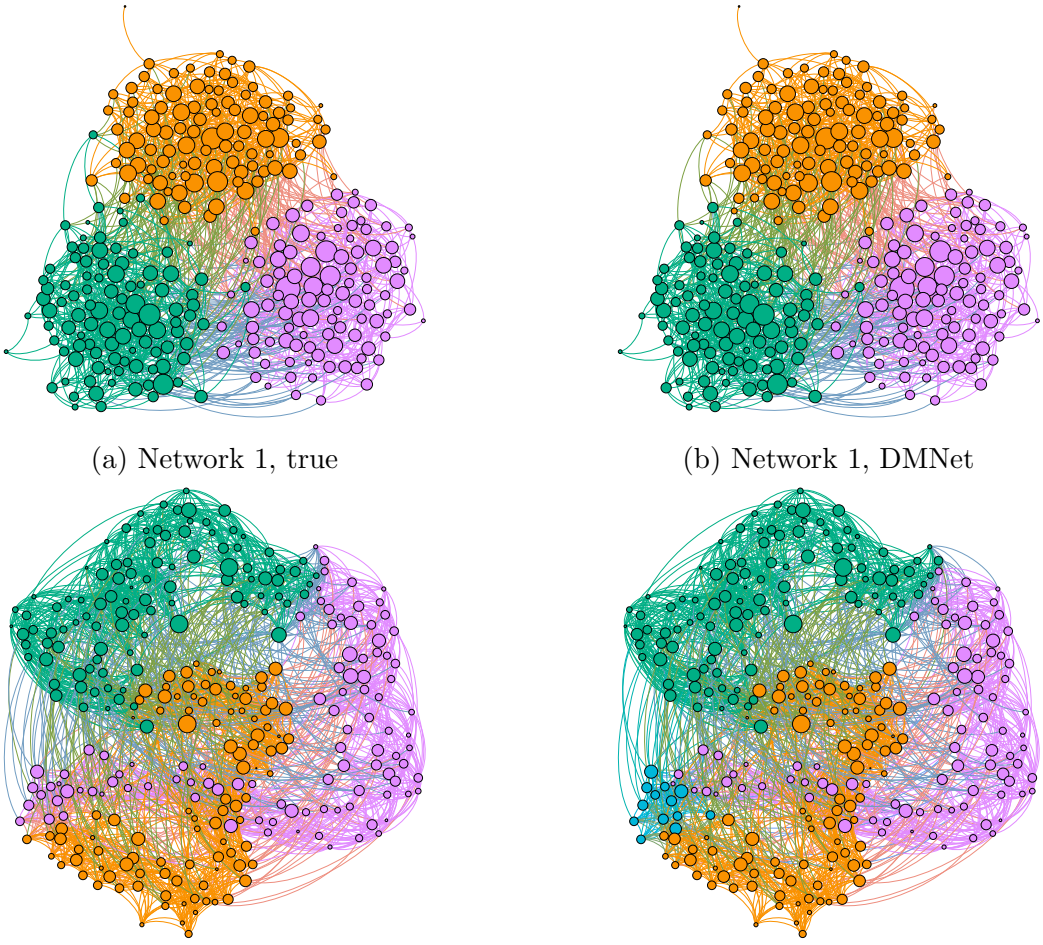
$$\mathrm{CV}(\lambda) = -\frac{1}{K}\sum_{k=1}^{K}\sum_{i\in T_k}\max_{j\in S_k}\log f(\mathbf{y}_i, \widehat{\boldsymbol{\alpha}}_j(\lambda)), \tag{17}$$

where $\widehat{\boldsymbol{\alpha}}_j(\lambda)$ are obtained from S_k with λ fixed and refitted with the predicted cluster memberships to reduce the bias incurred by the penalty. For DMNet+ and DMNet++, f in (17) is replaced by f^E and f^M, respectively.

To demonstrate the performance of our cross-validation procedure, we generated $n = 300$ texts in three groups of equal size with parameters $\boldsymbol{\alpha}^{(1)} = (0.005, 0.01, \ldots, 1)^T$, $\boldsymbol{\alpha}^{(2)} = (1, 0.995, \ldots, 0.005)^T$, and $\boldsymbol{\alpha}^{(3)} = (0.5, 0.495, \ldots, 0.005, 1, 0.995, \ldots, 0.505)^T$. The other settings are as described in Section 5.1. The true cluster memberships and those obtained by DMNet under two network topologies are depicted in Figure 2. The results for DMNet+ and DMNet++ are almost identical to that for DMNet and hence are omitted. From Figure 2, we see that our cross-validation procedure is able to identify the major clusters that are consistent with the true groups, with one more minor cluster for the more difficult setting of Network 2. In practice, it is sometimes convenient to specify the number of clusters. Note that the tuning parameter λ controls the strength of regularization, and increasing λ will result in fewer clusters. From this relationship one can easily choose the appropriate λ that corresponds to a particular number of clusters.

4 Theory

We now develop theoretical guarantees for the proposed DMNet++ method. Our theory is nonasymptotic in nature and allows the dimension p, sample size n, and total counts N_i to grow simultaneously. In the setting of denoising over a complete graph, similar results



(a) Network 1, true

(b) Network 1, DMNet

(c) Network 2, true

(d) Network 2, DMNet

Figure 2: True clusters and clusters obtained by DMNet with cross-validation.

were obtained by Tan and Witten (2015). Our theoretical development is different from theirs in at least two aspects. First, we analyze exponential family models, rather than denoising problems, which are more technically involved. Second, our results allow for a general network topology and arbitrary weights, thereby providing insights into the impact of the network structure on the performance of our method.

Consider the DMNet++ problem (13). Our goal is to estimate the target parameter

$$\boldsymbol{\theta}^* = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}}\{-E\widetilde{\ell}^M(\boldsymbol{\theta})\}. \tag{18}$$

Let $\mathbf{D}$ be the $|E|p \times np$ difference matrix that maps $\boldsymbol{\theta}$ to $w_{ij}(\boldsymbol{\theta}_i - \boldsymbol{\theta}_j)_{(i,j)\in E}$, that is, $\mathbf{D} = \mathbf{D}_0 \otimes \mathbf{I}_p$, where $\mathbf{D}_0$ is the $|E| \times n$ oriented incidence matrix of G that puts w_{ij} in position i and $-w_{ij}$ in position j in the row indexed by $(i,j) \in E$, and $\otimes$ denotes the Kronecker

product. Define $\mathcal{R}_0(\boldsymbol{\zeta}) = \sum_{(i,j)\in E} \|\boldsymbol{\zeta}_{(i,j)}\|_2$ for $\boldsymbol{\zeta} = (\boldsymbol{\zeta}_1^T, \ldots, \boldsymbol{\zeta}_{|E|}^T)^T \in \mathbb{R}^{|E|p}$, so that the weighted L_1 fusion penalty

$$\mathcal{R}(\boldsymbol{\theta}) = \mathcal{R}_0(\mathbf{D}\boldsymbol{\theta}) = \sum_{(i,j)\in E} w_{ij}\|\boldsymbol{\theta}_i - \boldsymbol{\theta}_j\|_2.$$

To resolve the nonidentifiability issue of the DMNet++ model, we assume that both problems (13) and (18) are optimized over some suitably normalized parameter space. Specifically, we impose the following condition.

Condition 1. *The problems* (13) *and* (18) *are optimized over the parameter space* $\boldsymbol{\Theta} = \boldsymbol{\Theta}_1 \times \cdots \times \boldsymbol{\Theta}_n$, *where*

$$\boldsymbol{\Theta}_i \subset \left\{\boldsymbol{\theta}_i \in \mathbb{R}^p : \sum_{j=1}^p \theta_{ij} = 0\right\}.$$

Such a normalization is for theoretical convenience and not essential in practice, since the DMNet++ model serves only as an approximation to the DMNet model. Numerical evidence from our experiments shows that solving problem (13) without the normalization constraint generally performs well in terms of estimation and clustering accuracy. Moreover, we write the approximating density (11) in the canonical form

$$f^M(\mathbf{y}_i; \boldsymbol{\theta}_i) = h(\mathbf{y}_i)\exp\{\langle \boldsymbol{\theta}_i, \mathbf{y}_i\rangle - A_i(\boldsymbol{\theta}_i)\},$$

where $A_i(\boldsymbol{\theta}_i) = N_i \log C(\boldsymbol{\theta}_i)$ and $C(\boldsymbol{\theta}_i) = \sum_{j=1}^p e^{\theta_{ij}}$. It is clear that the Fisher information matrix $\nabla^2 A_i(\boldsymbol{\theta}_i)$ has a zero eigenvalue with corresponding eigenvector $(1/\sqrt{p}, \ldots, 1/\sqrt{p})^T$. The following condition ensures that the positive eigenvalues of $N_i^{-1}\nabla^2 A_i(\boldsymbol{\theta}_i)$ are bounded away from zero.

Condition 2. *There exists a constant* $\kappa > 0$ *such that* $\inf_{\boldsymbol{\theta}_i \in \boldsymbol{\Theta}_i} \lambda_{\min}^+(\nabla^2 \log C(\boldsymbol{\theta}_i)) \geq \kappa$, *where* $\lambda_{\min}^+(\cdot)$ *denotes the smallest positive eigenvalue.*

Finally, we assume that the total counts N_i are of the same order.

Condition 3. *There exist constants* $\underline{c}, \bar{c} > 0$ *such that* $\underline{c}N \leq N_i \leq \bar{c}N$ *for all* $i = 1, \ldots, n$.

We are ready to state our main result, which provides nonasymptotic error bounds for the DMNet++ estimator $\widehat{\boldsymbol{\theta}}^M$.

Theorem 1. *Assume that Conditions 1–3 hold. If*

$$\lambda \geq \frac{2}{n}\sqrt{\frac{\bar{c}Np\log(|E|p)}{\lambda_G}},$$

where $\lambda_G = \lambda_{\min}^{+}(\mathbf{D}^T\mathbf{D})$*, then the estimator* $\widehat{\boldsymbol{\theta}}^M$ *defined in* (13) *satisfies*

$$\frac{1}{n}\|\widehat{\boldsymbol{\theta}}^M - \boldsymbol{\theta}^*\|_2^2 \leq \frac{3\lambda}{2\kappa\underline{c}N}\mathcal{R}(\boldsymbol{\theta}^*) + \frac{\bar{c}}{4\kappa^2\underline{c}^2}\left(\frac{rp}{Nn} + \frac{1}{Nn}\sqrt{rp\log n}\right) \tag{19}$$

with probability at least $1 - 2(|E|p)^{-1} - \exp\{-\min(c_1\log n, c_2\sqrt{rp\log n})\}$*, where* r *is the number of connected components of* G *and* $c_1, c_2 > 0$ *are some constants.*

A few remarks are in order. First, we have chosen a scaling factor of $1/n$ for the error bounds, rather than $1/(np)$ as in Tan and Witten (2015). Note that the scaling factor $1/n$ is necessary because, in the clustering setting, $\boldsymbol{\theta}^*$ contains as many as $O(n)$ identical copies of a few distinct individual parameters. On the other hand, scaling by $1/p$ is not needed in our case, since the growth of p can be compensated by the total count N and will not affect the rate of convergence as long as $p/N = O(1)$.

Moreover, our bounds depend on the structure of the network through the spectral properties of G. Note that $\mathbf{D}^T\mathbf{D} = \mathbf{L} \otimes \mathbf{I}_p$, where $\mathbf{L} = \mathbf{D}_0^T\mathbf{D}_0$ is the Laplacian matrix of $\widetilde{G} = (V, E, \widetilde{W})$ with $\widetilde{W} = (w_{ij}^2)_{(i,j)\in E}$, that is, the matrix with $\sum_{j\in N(i)} w_{ij}^2$ on the diagonal, $-w_{ij}^2$ in the (i,j)th off-diagonal entry for $(i,j) \in E$, and zeros elsewhere. It is well known that $\mathbf{L}$ has a zero eigenvalue with multiplicity r (Godsil and Royle, 2001). The second smallest eigenvalue of $\mathbf{L}$ is known as the algebraic connectivity and plays a key role in many dynamical phenomena such as synchronization in complex networks (Barrat, Barthélemy, and Vespignani, 2008).

Consequently, the bound (19) decomposes into two terms arising from estimating the components of $\boldsymbol{\theta}$ in the range and null spaces of $\mathbf{L}$. The second term includes a factor of r, since the parameters in different connected components are estimated separately. The first term involves λ_G and $\mathcal{R}(\boldsymbol{\theta}^*)$. The spectral gap λ_G coincides with the smallest algebraic connectivity of the connected components of G, which sets the limit for borrowing information over the network. The quantity $\mathcal{R}(\boldsymbol{\theta}^*)$ measures the fidelity of the network to the true parameter $\boldsymbol{\theta}^*$, which reflects the loss due to misspecified network structure. In

the case of an unweighted complete graph as considered by Tan and Witten (2015), we have $\lambda_G = n$ and $\mathcal{R}(\boldsymbol{\theta}^*)$ can be as large as $O(n^2)$, which is clearly suboptimal. Our result reveals a trade-off between two effects: a more connected network will have a smaller r and a larger λ_G, but is likely to have a larger $\mathcal{R}(\boldsymbol{\theta}^*)$. Therefore, it is advisable to increase the connectivity of the network while maintaining the accuracy of most edges.

5 Simulation Studies

In this section, we report on simulation studies to evaluate the numerical performance of our proposed methods and compare them with some commonly used clustering methods.

5.1 Settings

We set the dimension $p = 200$ and generated the samples in three groups of equal size 150 or unequal size $(100, 150, 200)$ with the parameter values

$$\boldsymbol{\alpha}^{(1)} = (0.005, 0.01, \ldots, 1)^T, \quad \boldsymbol{\alpha}^{(2)} = (0.005, 0.01, \ldots, 0.5, 1, 0.995, \ldots, 0.505)^T,$$
$$\boldsymbol{\alpha}^{(3)} = (0.5, 0.495, \ldots, 0.005, 0.505, 0.51, \ldots, 1)^T.$$

The total counts N_i were drawn randomly from 80 to 120. These settings were chosen to reflect the heterogeneity of word frequencies and the typical length of a scientific abstract. We generated the networks in the following two ways:

- Network 1 (Block): Each pair of nodes within the same group were connected with probability 0.08, and those between different groups were connected with probability 0.01.
- Network 2 (Small-world): Following the Watts–Strogatz model (Watts and Strogatz, 1998), we started from a regular ring lattice with degree 10 for each node and rewired each edge with probability 0.1.

We adopted the adaptive weights $w_{ij} = \|\widetilde{\boldsymbol{\theta}}_i - \widetilde{\boldsymbol{\theta}}_j\|_2^{-\gamma}$ with initial estimates $\widetilde{\boldsymbol{\theta}}_i = \log \widehat{\boldsymbol{\pi}}_i = \log(\mathbf{y}_i/N_i)$ and $\gamma = 1$. We repeated each simulation 100 times. The penalty parameter in

the ADMM algorithm was set to $\rho = 1000$ and the tuning parameter λ was selected by fivefold cross-validation.

5.2 Performance Comparisons

We compare our methods with the following eight clustering procedures for Euclidean and network data: the Dirichlet-multinomial mixture model (DMM, Holmes, Harris, and Quince, 2012), a mixture model approach to spectral clustering (Spectral, Di Nuzzo and Ingrassia, 2022), K-means and K-means++ (Arthur and Vassilvitskii, 2007) applied to the proportions, the Louvain method (Blondel et al., 2008) based on modularity optimization and its fast consensus variant (FCLouvain, Tandon et al., 2019), mixture models via the EM algorithm (EM, Newman and Leicht, 2007), and a structural clustering algorithm for networks (SCAN, Xu et al., 2007). Note that the first four methods use only the count data, while the last four use only the network data. We also compare our method with its oracle counterpart, which estimates the DM parameters with the cluster memberships known in advance.

We assess the performance of different methods using five measures. The first two are the L_2 errors for estimating $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$. The other three measures quantify the clustering performance. Purity is the correct classification rate when each cluster is assigned to the major class in that cluster. The other two measures are calculated by comparing pairwise cluster memberships from the actual and predicted partitions. Treating this as a binary classification problem, a positive assigns two data points to the same cluster, while a negative assigns them to different clusters. The Rand index (RI) is the accuracy and the F_1 measure (F_1) is the harmonic mean of precision and recall, defined respectively by

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}, \qquad F_1 = \frac{2\mathrm{TP}}{2\mathrm{TP} + \mathrm{FP} + \mathrm{FN}},$$

where TP, TN, FP, and FN are the numbers of true positives, true negatives, false positives, and false negatives, respectively. See, for example, Manning, Raghavan, and Schütze (2008, sec. 16.3). All three clustering measures range from 0 to 1, and a value closer to 1 indicates a better performance.

The simulation results under Networks 1 and 2 are summarized in Tables 1 and 2, respectively. We observe that DMNet, DMNet+, and DMNet++ have a very close performance across all settings and performance measures, indicating that the last two methods serve as good approximations to the first. In terms of clustering quality, our proposed methods consistently outperform the competing procedures, mostly by a large margin. Among the procedures using only the count or network data, FCLouvain and EM seem to perform better than the others under Network 1, while DMM has some advantages under Network 2, where the network information is weak. Moreover, the count-based methods perform slightly better in the unequal size setting, while the network-based methods perform slightly worse. In all settings, our methods benefit from both the count and network information and, as a result, outperform the competitors. Compared with the oracle, our proposed methods perform quite satisfactorily in estimating both $\boldsymbol{\alpha}$ and $\boldsymbol{\theta}$. The gap in estimation performance tends to diminish as the clustering quality improves. Overall, the simulation results demonstrate the superiority of our methods and underscore the effectiveness of combining count and network data for clustering. In addition, the timing results reported in Table 3 show that DMNet+ and DMNet++ achieve a reduction of about 10% and 20%, respectively, in computation time compared to DMNet. We therefore recommend using DMNet++ as a reliable and efficient approach in practice, especially for large-scale problems.

6 Applications to Text Data

We illustrate our proposed methods by applying them to two real bibliographic datasets, CiteSeer and Cora, from Sen et al. (2008). Both datasets include class labels that allow us to assess the clustering quality of different methods.

6.1 CiteSeer

The CiteSeer dataset consists of 3312 computer science papers grouped into six categories: Agents, Artificial Intelligence (AI), Database (DB), Human–Computer Interaction (HCI),

Table 1: Means and standard errors (in parentheses) of performance measures for various methods based on 100 replications under Network 1.

Cluster size	Method	$\|\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}\|_2$	$\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_2$	Purity	RI	F_1
Equal	DMNet	25.31 (0.74)	112.01 (5.28)	1.00 (0.00)	1.00 (0.00)	0.99 (0.01)
	DMNet+	25.47 (0.83)	112.27 (5.17)	1.00 (0.00)	1.00 (0.00)	0.99 (0.01)
	DMNet++	25.34 (0.81)	112.36 (5.13)	1.00 (0.00)	1.00 (0.00)	0.99 (0.01)
	DMM	35.58 (5.58)	139.13 (40.24)	0.61 (0.21)	0.65 (0.22)	0.67 (0.12)
	Spectral	—	—	0.41 (0.05)	0.51 (0.09)	0.40 (0.08)
	K-means	—	—	0.50 (0.05)	0.62 (0.01)	0.33 (0.04)
	K-means++	—	—	0.48 (0.06)	0.57 (0.05)	0.43 (0.04)
	Louvain	—	—	0.98 (0.07)	0.95 (0.06)	0.92 (0.09)
	FCLouvain	—	—	1.00 (0.00)	0.98 (0.06)	0.96 (0.16)
	EM	—	—	0.97 (0.07)	0.97 (0.05)	0.95 (0.07)
	SCAN	—	—	0.49 (0.02)	0.52 (0.01)	0.45 (0.01)
	Oracle	24.73 (0.84)	106.62 (3.05)	—	—	—
Unequal	DMNet	24.52 (1.34)	116.65 (2.77)	0.99 (0.03)	0.98 (0.02)	0.98 (0.02)
	DMNet+	25.22 (1.37)	112.52 (2.99)	0.99 (0.03)	0.98 (0.02)	0.98 (0.02)
	DMNet++	25.22 (1.37)	111.09 (3.18)	0.99 (0.03)	0.98 (0.02)	0.98 (0.02)
	DMM	32.43 (5.19)	121.93 (36.37)	0.74 (0.14)	0.77 (0.18)	0.78 (0.11)
	Spectral	—	—	0.51 (0.06)	0.52 (0.08)	0.43 (0.09)
	K-means	—	—	0.55 (0.07)	0.62 (0.02)	0.36 (0.05)
	K-means++	—	—	0.56 (0.08)	0.59 (0.06)	0.47 (0.07)
	Louvain	—	—	0.96 (0.08)	0.87 (0.07)	0.79 (0.11)
	FCLouvain	—	—	1.00 (0.00)	0.89 (0.14)	0.73 (0.36)
	EM	—	—	0.89 (0.01)	0.90 (0.01)	0.86 (0.01)
	SCAN	—	—	0.46 (0.01)	0.46 (0.01)	0.43 (0.01)
	Oracle	24.60 (0.93)	110.96 (2.96)	—	—	—

Table 2: Means and standard errors (in parentheses) of performance measures for various methods based on 100 replications under Network 2.

Cluster size	Method	$\|\widehat{\boldsymbol{\alpha}} - \boldsymbol{\alpha}\|_2$	$\|\widehat{\boldsymbol{\theta}} - \boldsymbol{\theta}\|_2$	Purity	RI	F_1
Equal	DMNet	36.59 (2.09)	144.66 (23.04)	0.67 (0.01)	0.74 (0.02)	0.69 (0.03)
	DMNet+	35.80 (2.03)	143.53 (23.76)	0.67 (0.01)	0.74 (0.02)	0.69 (0.03)
	DMNet++	35.88 (2.06)	142.99 (22.92)	0.67 (0.01)	0.74 (0.02)	0.69 (0.03)
	DMM	35.58 (5.58)	139.13 (40.24)	0.61 (0.21)	0.65 (0.22)	0.67 (0.12)
	Spectral	—	—	0.41 (0.05)	0.51 (0.09)	0.40 (0.08)
	K-means	—	—	0.50 (0.05)	0.62 (0.01)	0.33 (0.04)
	K-means++	—	—	0.48 (0.06)	0.57 (0.05)	0.43 (0.04)
	Louvain	—	—	0.74 (0.13)	0.72 (0.10)	0.61 (0.07)
	FCLouvain	—	—	0.96 (0.02)	0.70 (0.01)	0.18 (0.06)
	EM	—	—	0.53 (0.07)	0.62 (0.04)	0.43 (0.06)
	SCAN	—	—	0.77 (0.03)	0.66 (0.01)	0.30 (0.03)
	Oracle	24.73 (0.84)	106.62 (3.05)	—	—	—
Unequal	DMNet	33.49 (1.91)	112.58 (20.84)	0.82 (0.06)	0.81 (0.04)	0.74 (0.07)
	DMNet+	33.83 (1.89)	112.28 (21.40)	0.82 (0.06)	0.81 (0.04)	0.74 (0.07)
	DMNet++	33.81 (1.81)	111.96 (20.60)	0.82 (0.06)	0.81 (0.04)	0.74 (0.07)
	DMM	32.43 (5.19)	121.93 (36.37)	0.74 (0.14)	0.77 (0.18)	0.78 (0.11)
	Spectral	—	—	0.51 (0.06)	0.52 (0.08)	0.43 (0.09)
	K-means	—	—	0.55 (0.07)	0.62 (0.02)	0.36 (0.05)
	K-means++	—	—	0.56 (0.08)	0.59 (0.06)	0.47 (0.07)
	Louvain	—	—	0.74 (0.12)	0.69 (0.10)	0.58 (0.08)
	FCLouvain	—	—	0.96 (0.02)	0.68 (0.01)	0.18 (0.05)
	EM	—	—	0.54 (0.07)	0.60 (0.04)	0.43 (0.06)
	SCAN	—	—	0.72 (0.02)	0.61 (0.01)	0.24 (0.02)
	Oracle	24.60 (0.93)	110.96 (2.96)	—	—	—

Table 3: Means and standard errors (in parentheses) of run times (in seconds, excluding cross-validation) for the proposed methods based on 100 replications.

Network	Method	Equal size	Unequal size
1	DMNet	74.38 (1.04)	81.53 (2.06)
	DMNet+	68.43 (1.18)	68.10 (1.94)
	DMNet++	62.27 (1.21)	63.61 (1.96)
2	DMNet	100.76 (2.10)	103.12 (2.57)
	DMNet+	90.13 (2.64)	95.70 (2.94)
	DMNet++	82.35 (2.70)	87.36 (2.78)

Information Retrieval (IR), and Machine Learning (ML). Also included is a citation network of 4732 edges. After removing self-loops, duplicated edges, and citing or cited papers not present in the corpus, and isolated papers, we were left with 3264 papers and 4536 edges. The vocabulary has a total of 3703 unique words, out of which we focus on the 445 words with frequency not too low (appearing in fewer than 50 papers) or too high (appearing in more than 200 papers). The Hopkins statistic (Hopkins and Skellam, 1954) for the normalized count data is 0.87, suggesting a fairly strong clustering tendency. To apply our methods, we chose the adaptive weights as in the simulations with $\gamma = 3$ and the optimal λ by fivefold cross-validation.

The results for the proposed and competing methods evaluated by three clustering performance measures are shown in Table 4. Additionally, we include in our comparisons the natural approach that combines word embedding with the network lasso. Specifically, we use Word2Vec (Mikolov et al., 2013) to convert individual words into vectors of 100 dimensions and represent a text by the average of all word vectors in that text. We then apply the network lasso to the text vectors. We see that our methods compare favorably with the others and are close to the best in terms of all three measures. The Louvain and FCLouvain methods have high purity and RI values because they divided the dataset into many small clusters, with the largest sizes being only 114 and 24, respectively. This inevitably yields a very low F_1 measure and inferior clustering quality. The Word2Vec method has a reasonable performance in terms of all three measures, but still performs worse than our methods. The advantages of our methods over word embedding are mainly due to

Table 4: Performance measures for various methods on real data.

	CiteSeer			Cora		
Method	Purity	RI	F_1	Purity	RI	F_1
DMNet	0.71	0.79	0.32	0.75	0.81	0.36
DMNet+	0.71	0.79	0.32	0.75	0.81	0.36
DMNet++	0.71	0.79	0.32	0.75	0.81	0.36
DMM	0.00	0.30	0.21	0.00	0.30	0.30
Spectral	0.11	0.35	0.33	0.00	0.30	0.30
K-means	0.46	0.71	0.33	0.34	0.64	0.25
K-means++	0.35	0.56	0.34	0.31	0.21	0.30
Louvain	0.84	0.82	0.02	0.90	0.82	0.02
FCLouvain	0.85	0.82	0.01	0.90	0.82	0.01
EM	0.27	0.57	0.27	0.36	0.63	0.28
SCAN	0.39	0.63	0.23	0.46	0.67	0.21
Word2Vec	0.61	0.66	0.21	—	—	—

the following reasons: (1) word embedding aims to capture word–word associations, which are essential for context-based text prediction and generation tasks but are less important for text clustering purposes; (2) word embedding achieves dimensionality reduction through low-dimensional vector representations, which may incur more information loss and be more sensitive to tuning parameter selection.

To gain insight into how the network structure affects the clustering performance, we visualize the ground truth network and the network obtained by DMNet in Figure 3. Nodes of the latter network are labeled by assigning each cluster to the major class in that cluster. The results for DMNet+ and DMNet++ are almost identical to that for DMNet and hence not shown. The group adjacency matrix of the network, which counts the numbers of edges within and between groups, along with some summary statistics is given in Table 5, with the convention that within-group edges are counted twice. From Figure 3, it is apparent that the Agents and IR groups are identified most accurately. Indeed, the largest Agents and IR clusters respectively contain 675 and 898 papers, among which 412 and 477 are correctly labeled and account for 69.8% and 71.6% of the true Agents and IR groups. This is reasonable in view of the fact these two groups have the highest or close to highest

Table 5: Group adjacency matrix, percentages of within-group edges, and average degrees for the CiteSeer network.

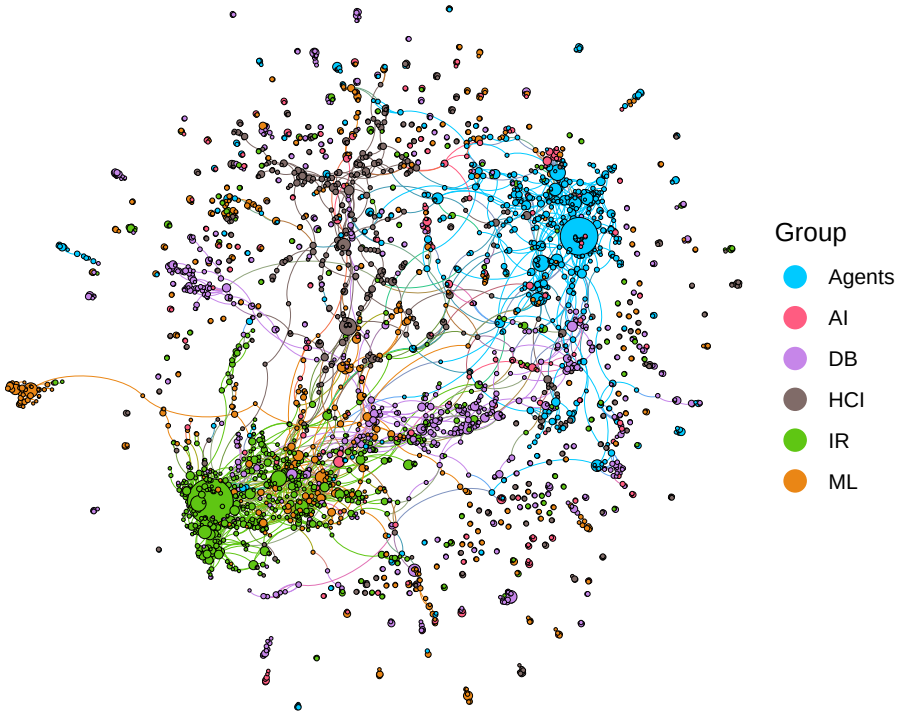
Group	Agents	AI	DB	HCI	IR	ML	Percent.	Degree
Agents	1378	93	50	86	47	79	79.5	2.94
AI	93	190	64	16	43	108	37.0	2.14
DB	50	64	1256	34	180	60	76.4	2.41
HCI	86	16	34	882	64	28	79.5	2.21
IR	47	43	180	64	2082	238	78.4	3.98
ML	79	108	60	28	238	904	63.8	2.42

average degrees and percentages of within-group edges as shown in Table 5. The third largest cluster is labeled as HCI, which consists of 190 papers. Among these, 173 overlap with the true HCI group, accounting for only 34.5% of the latter, which is not surprising since the HCI group has the second lowest average degree. The DB and ML groups are hardly separated from the IR group, largely owing to their close proximity to the latter in addition to their relatively low average degrees and percentages of within-group edges. The AI group has strong interactions with the Agents and ML groups and shows no clear clustering tendency. As a consequence, no major clusters are found for the AI group.

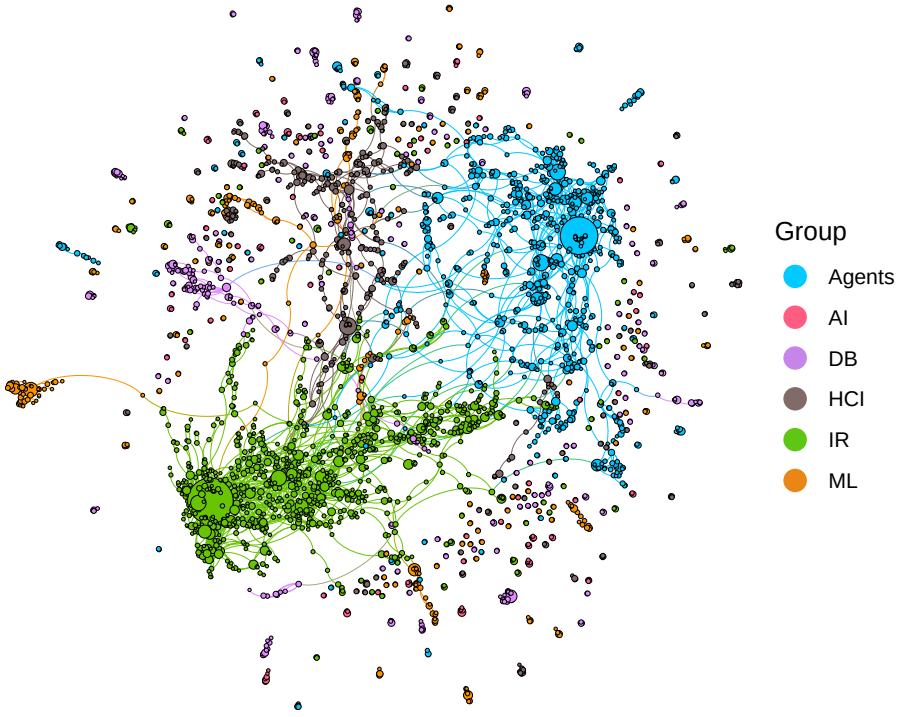
6.2 Cora

The Cora dataset consists of 2708 machine learning papers grouped into seven categories: Case Based (CB), Genetic Algorithms (GA), Neural Networks (NN), Probabilistic Methods (PM), Reinforcement Learning (ReL), Rule Learning (RuL), and Theory. The citation network includes 5429 edges, out of which 5278 were retained after the preprocessing. From the vocabulary of 1433 unique words, we removed those appearing in fewer than 30 and more than 250 papers, resulting in 409 words. The Hopkins statistic is 0.93, suggesting a rather strong clustering tendency. The same settings of adaptive weights and tuning parameters as for the CiteSeer dataset were adopted.

The results for different methods in terms of three clustering performance measures are shown in Table 4. We see that our methods achieve the highest or close to highest RI and



(a)



(b)

Figure 3: Analysis of the CiteSeer dataset: (a) ground truth network; (b) network obtained by DMNet.

Table 6: Group adjacency matrix, percentages of within-group edges, and average degrees for the Cora network.

Group	CB	GA	NN	PM	ReL	RuL	Theory	Percent.	Degree
CB	834	30	54	19	28	46	75	76.8	3.64
GA	30	1654	53	2	62	2	23	90.6	4.37
NN	54	53	2350	137	67	16	161	82.8	3.47
PM	19	2	137	1320	20	6	88	82.9	3.74
ReL	28	62	67	20	818	2	32	79.5	4.74
RuL	46	2	16	6	2	506	80	76.9	3.66
Theory	75	23	161	88	32	80	1068	69.9	4.35

F_1 values. The Louvain and FCLouvain methods yield a higher purity, but at the price of very low RI and F_1 values. The ground truth network and the network obtained by DMNet are depicted in Figure 4, while the group adjacency matrix and summary statistics of the network are shown in Table 6. As seen from Figure 4, the GA and ReL groups are identified remarkably well, both of which have a high average degree and low interactions with other groups. Indeed, the GA and ReL groups respectively include 429 and 220 papers, among which 373 and 165 are correctly labeled and account for 89.2% and 76.0% of the true GA and ReL groups. The two largest clusters are labeled as Theory and NN, which contain 637 and 453 papers, respectively. However, they range too widely and cover also a substantial part of the RuL, CB, and PM groups because of their strong interactions with the latter groups. In summary, these results showcase the effectiveness of our methods and corroborate our theoretical findings about the impact of the network structure.

7 Discussion

We have developed a clustering framework for effectively combining nodewise multivariate count data and prior network information. Our theoretical and numerical results provide insights into the performance gain of our approach and the critical role of the network structure. Our framework may be extended in several ways. First, the DM model can be replaced by a more flexible family of models such as the logistic normal multinomial and

(a)

(b)

Figure 4: Analysis of the Cora dataset: (a) ground truth network; (b) network obtained by DMNet.

zero-inflated models to account for a general correlation structure and excess zeros (Zhang and Lin, 2019; Tang and Chen, 2019). Second, in the situation where a single network provides inaccurate or insufficient information, one can in principle incorporate multiple networks to further boost the performance. In our text data example, it would be desirable to exploit coauthorship and citation networks of authors. Finally, dimension reduction and variable selection techniques may be employed to reduce the number of features and upweight contributions from representative words.

Acknowledgements

This work was supported by Beijing Natural Science Foundation grant Z190001, National Natural Science Foundation of China grants 12171012, 92046021, and 12026606, and Pazhou Lab.

A Proofs

A.1 Proof of Lemma 1

The density (2) and its approximation (6) can be written

$$f(\mathbf{y}_i;\boldsymbol{\alpha}_i) = \frac{N_i!\Gamma(\alpha_i^+)}{\Gamma(N_i+\alpha_i^+)} \prod_{j\in J_{i1}(\delta)} \frac{\Gamma(y_{ij}+\alpha_{ij})}{y_{ij}!\Gamma(\alpha_{ij})} \prod_{j\in J_{i2}(\delta)} \frac{\Gamma(y_{ij}+\alpha_{ij})}{y_{ij}!\Gamma(\alpha_{ij})} \equiv L_i g_{i1} g_{i2},$$

$$f^E(\mathbf{y}_i;\boldsymbol{\alpha}_i) = \frac{N_i!\Gamma(\alpha_i^+)}{\Gamma(N_i+\alpha_i^+)} \prod_{j\in J_{i1}(\delta)} \frac{\alpha_{ij}}{y_{ij}} \prod_{j\in J_{i2}(\delta)} \frac{\alpha_{ij}}{y_{ij}} \equiv L_i g_{i1}^E g_{i2}^E.$$

Note that on $J_{i1}(\delta)$,

$$\left|\frac{\Gamma(y_{ij}+\alpha_{ij})}{y_{ij}!\Gamma(\alpha_{ij})} - \frac{\alpha_{ij}}{y_{ij}}\right| \le \varepsilon, \qquad \frac{\alpha_{ij}}{y_{ij}} \le \delta.$$

By the assumptions, it follows that

$$\begin{aligned}
|f(\mathbf{y}_i;\boldsymbol{\alpha}_i) - f^E(\mathbf{y}_i;\boldsymbol{\alpha}_i)| &\le L_i(g_{i2}|g_{i1} - g_{i1}^E| + g_{i1}^E|g_{i2} - g_{i2}^E|) \\
&\le L(\varepsilon^{|J_{i1}(\delta)|} M^{|J_{i2}(\delta)|} + 2\delta^{|J_{i1}(\delta)|} M^{|J_{i2}(\delta)|}) \\
&= LM^{s_2(p)}(\varepsilon^{s_1(p)} + 2\delta^{s_1(p)})
\end{aligned}$$

for all $i = 1, \ldots, n$, which proves the bound.

A.2 Proof of Theorem 1

Before proving Theorem 1, we set up some notation to decompose $\boldsymbol{\theta}$ into two components, of which only one is penalized. Similar decompositions were also used by Liu, Yuan, and Ye (2013) and Tan and Witten (2015). Note that $|E| \geq n - r$ and $\operatorname{rank}(\mathbf{D}_0) = n - r$. Let $\mathbf{D} = \mathbf{U}\boldsymbol{\Sigma}\mathbf{V}_{(1)}^T$ be the singular value decomposition of $\mathbf{D}$, where $\mathbf{U} \in \mathbb{R}^{|E|p \times (n-r)p}$, $\mathbf{V}_{(1)} \in \mathbb{R}^{np \times (n-r)p}$, and $\boldsymbol{\Sigma} \in \mathbb{R}^{(n-r)p \times (n-r)p}$ is a diagonal matrix of positive singular values. Augment the matrix $\mathbf{V}_{(1)}$ with $\mathbf{V}_{(2)} \in \mathbb{R}^{np \times rp}$ such that $\mathbf{V} = (\mathbf{V}_{(1)}, \mathbf{V}_{(2)})$ is orthogonal. Let $\mathbf{C} = \mathbf{U}\boldsymbol{\Sigma}$, $\boldsymbol{\theta}_{(1)} = \mathbf{V}_{(1)}^T\boldsymbol{\theta}$, and $\boldsymbol{\theta}_{(2)} = \mathbf{V}_{(2)}^T\boldsymbol{\theta}$, so that $\boldsymbol{\theta} = \mathbf{V}_{(1)}\boldsymbol{\theta}_{(1)} + \mathbf{V}_{(2)}\boldsymbol{\theta}_{(2)}$ and

$$\mathcal{R}(\boldsymbol{\theta}) = \mathcal{R}_0(\mathbf{D}\boldsymbol{\theta}) = \mathcal{R}_0(\mathbf{C}\boldsymbol{\theta}_{(1)}),$$

for all $\boldsymbol{\theta} \in \mathbb{R}^{np}$. Let $\mathbf{C}^+ = \boldsymbol{\Sigma}^{-1}\mathbf{U}^T$ be the Moore–Penrose pseudoinverse of $\mathbf{C}$, so that $\mathbf{C}^+\mathbf{C} = \mathbf{I}$. Denote by $\mathbf{C}_{(i,j)}$ (resp. $\mathbf{C}^+_{(i,j)}$) the submatrix of $\mathbf{C}$ (resp. $\mathbf{C}^+$) with the p rows (resp. columns) indexed by $(i, j) \in E$.

The following lemma provides the tail bounds needed for the proof of Theorem 1.

Lemma 2. *The score function $\nabla\widetilde{\ell}^M(\boldsymbol{\theta})$ at $\boldsymbol{\theta}^*$ satisfies*

$$P\left(\max_{(i,j)\in E} \|(\mathbf{C}^+_{(i,j)})^T\mathbf{V}_{(1)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2 \geq \frac{1}{n}\sqrt{\frac{\bar{c}Np\log(|E|p)}{\lambda_G}}\right) \leq \frac{2}{|E|p} \tag{20}$$

and

$$P\left(\|\mathbf{V}_{(2)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2^2 \geq \frac{\bar{c}N}{4n^2}(rp + \sqrt{rp\log n})\right) \leq \exp\{-\min(c_1\log n, c_2\sqrt{rp\log n})\} \tag{21}$$

for some constants $c_1, c_2 > 0$.

Proof. The score functions take the form

$$\nabla_{\boldsymbol{\theta}_i}\widetilde{\ell}^M(\boldsymbol{\theta}^*) = \frac{1}{n}\sum_{m=1}^{N_i}(\mathbf{z}_i^{(m)} - \boldsymbol{\pi}_i^*), \tag{22}$$

where $\boldsymbol{\pi}_i^* = e^{\boldsymbol{\theta}_i^*}/C(e^{\boldsymbol{\theta}^*})$. Note that

$$\begin{aligned}\max_{(i,j)\in E} \|(\mathbf{C}^+_{(i,j)})^T\mathbf{V}_{(1)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2 &\leq \sqrt{p}\max_{(i,j)\in E} \|(\mathbf{C}^+_{(i,j)})^T\mathbf{V}_{(1)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_\infty \\ &= \sqrt{p}\|(\mathbf{C}^+)^T\mathbf{V}_{(1)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_\infty. \end{aligned} \tag{23}$$

In view of (22), $n(\partial/\partial\theta_{ij})\widetilde{\ell}^M(\boldsymbol{\theta}^*)$ is sub-Gaussian with mean zero and variance at most $\bar{c}N/4$. Combining with the fact that $\|(\mathbf{C}^+)^T\mathbf{V}_{(1)}^T\|_2^2 = \|\mathbf{U}\boldsymbol{\Sigma}^{-1}\mathbf{V}_{(1)}^T\|_2^2 = 1/\lambda_G$, we see that each component of $n(\mathbf{C}^+)^T\mathbf{V}_{(1)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)$ is sub-Gaussian with mean zero and variance at most $\bar{c}N/(4\lambda_G)$. Thus, by the sub-Gaussian tail bound and the union bound,

$$P(n\|(\mathbf{C}^+)^T\mathbf{V}_{(1)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_\infty \geq t) \leq 2|E|p\exp\left(-\frac{2\lambda_G t^2}{\bar{c}N}\right)$$

for all $t > 0$. Choosing $t = \sqrt{\bar{c}N\log(|E|p)/\lambda_G}$ and using (23) yields the bound (20).

Turning to the bound (21), since $\mathbf{V}_{(2)}\mathbf{V}_{(2)}^T$ is a projection matrix of rank rp, we have $\|\mathbf{V}_{(2)}\mathbf{V}_{(2)}^T\|_2 = 1$ and $\|\mathbf{V}_{(2)}\mathbf{V}_{(2)}^T\|_F^2 = \mathrm{tr}(\mathbf{V}_{(2)}\mathbf{V}_{(2)}^T) = rp$. By the Hanson–Wright inequality (Rudelson and Vershynin, 2013), there exist constants $c_1, c_2 > 0$ such that

$$P(n^2\|\mathbf{V}_{(2)}^T\nabla\widetilde{\ell}(\boldsymbol{\theta}^*)\|_2^2 \geq t + \sigma^2 rp) \leq \exp\left\{-\min\left(\frac{c_1 t^2}{\sigma^4 rp}, \frac{c_2 t}{\sigma^2}\right)\right\}$$

for all $t > 0$, where $\sigma^2 = \bar{c}N/4$. Choosing $t = \sigma^2\sqrt{rp\log n}$ yields the bound (21).

Proof of Theorem 1. Let $\widehat{\boldsymbol{\Delta}} = (\widehat{\boldsymbol{\Delta}}_1^T, \ldots, \widehat{\boldsymbol{\Delta}}_n^T)^T = \widehat{\boldsymbol{\theta}}^M - \boldsymbol{\theta}^*$. By a Taylor expansion and Condition 2, the Bregman divergence associated with the loss function $-\widetilde{\ell}^M(\boldsymbol{\theta})$ between $\widehat{\boldsymbol{\theta}}^M$ and $\boldsymbol{\theta}^*$ satisfies

$$\begin{aligned} B(\widehat{\boldsymbol{\theta}}^M, \boldsymbol{\theta}^*) &\equiv -\widetilde{\ell}^M(\widehat{\boldsymbol{\theta}}) + \widetilde{\ell}^M(\boldsymbol{\theta}^*) + \langle\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*), \widehat{\boldsymbol{\Delta}}\rangle = -\widehat{\boldsymbol{\Delta}}^T\nabla^2\widetilde{\ell}^M(\bar{\boldsymbol{\theta}})\widehat{\boldsymbol{\Delta}} \\ &= \frac{1}{n}\sum_{i=1}^n \widehat{\boldsymbol{\Delta}}_i^T\nabla^2 A_i(\bar{\boldsymbol{\theta}}_i)\widehat{\boldsymbol{\Delta}}_i \geq \frac{\kappa}{n}\sum_{i=1}^n N_i\|\widehat{\boldsymbol{\Delta}}_i\|_2^2 \geq \frac{\kappa\underline{c}N}{n}\|\widehat{\boldsymbol{\Delta}}\|_2^2, \end{aligned} \tag{24}$$

where $\bar{\boldsymbol{\theta}} = (\bar{\boldsymbol{\theta}}_1^T, \ldots, \bar{\boldsymbol{\theta}}_n^T)^T$ is some point between $\widehat{\boldsymbol{\theta}}^M$ and $\boldsymbol{\theta}^*$. On the other hand, by the optimality of $\widehat{\boldsymbol{\theta}}^M$,

$$B(\widehat{\boldsymbol{\theta}}^M, \boldsymbol{\theta}^*) \leq \lambda\mathcal{R}(\boldsymbol{\theta}^*) - \lambda\mathcal{R}(\widehat{\boldsymbol{\theta}}^M) + |\langle\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*), \widehat{\boldsymbol{\Delta}}\rangle|. \tag{25}$$

Also, by the optimality of $\widehat{\boldsymbol{\theta}}^M = \mathbf{V}_{(1)}\widehat{\boldsymbol{\theta}}_{(1)}^M + \mathbf{V}_{(2)}\widehat{\boldsymbol{\theta}}_{(2)}^M$ and the mean value theorem,

$$\mathbf{0} = \mathbf{V}_{(2)}^T\nabla\widetilde{\ell}^M(\widehat{\boldsymbol{\theta}}^M) = \mathbf{V}_{(2)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*) + \mathbf{V}_{(2)}^T\nabla^2\widetilde{\ell}^M(\widetilde{\boldsymbol{\theta}})\mathbf{V}_{(2)}\widehat{\boldsymbol{\Delta}}_{(2)},$$

where $\widetilde{\boldsymbol{\theta}}$ is some point between $((\boldsymbol{\theta}_{(1)}^*)^T, (\widehat{\boldsymbol{\theta}}_{(2)}^M)^T)^T$ and $\boldsymbol{\theta}^*$. Rearranging and using Condition 2 as in (24) gives

$$\|\mathbf{V}_{(2)}^T\nabla\widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2^2 = \|\mathbf{V}_{(2)}^T\nabla^2\widetilde{\ell}^M(\widetilde{\boldsymbol{\theta}})\mathbf{V}_{(2)}\widehat{\boldsymbol{\Delta}}_{(2)}\|_2^2 \geq \frac{(\kappa\underline{c}N)^2}{n^2}\|\widehat{\boldsymbol{\Delta}}_{(2)}\|_2^2. \tag{26}$$

Now we write the inner product in (25) as

$$\begin{aligned}\langle \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*), \widehat{\boldsymbol{\Delta}} \rangle &= \langle \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*), \mathbf{V}_{(1)} \widehat{\boldsymbol{\Delta}}_{(1)} + \mathbf{V}_{(2)} \widehat{\boldsymbol{\Delta}}_{(2)} \rangle \\ &= \langle \mathbf{V}_{(1)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*), \widehat{\boldsymbol{\Delta}}_{(1)} \rangle + \langle \mathbf{V}_{(2)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*), \widehat{\boldsymbol{\Delta}}_{(2)} \rangle \\ &\equiv T_1 + T_2.\end{aligned}$$

By the triangle inequality and the Cauchy–Schwarz inequality,

$$\begin{aligned}|T_1| = |\langle (\mathbf{C}^+)^T \mathbf{V}_{(1)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*), \mathbf{C} \widehat{\boldsymbol{\Delta}}_{(1)} \rangle| &\le \sum_{(i,j) \in E} |\langle (\mathbf{C}^+_{(i,j)})^T \mathbf{V}_{(1)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*), \mathbf{C}_{(i,j)} \widehat{\boldsymbol{\Delta}}_{(1)} \rangle| \\ &\le \sum_{(i,j) \in E} \|(\mathbf{C}^+_{(i,j)})^T \mathbf{V}_{(1)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2 \|\mathbf{C}_{(i,j)} \widehat{\boldsymbol{\Delta}}_{(1)}\|_2 \\ &\le \max_{(i,j) \in E} \|(\mathbf{C}^+_{(i,j)})^T \mathbf{V}_{(1)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2 \sum_{(i,j) \in E} \|\mathbf{C}_{(i,j)} \widehat{\boldsymbol{\Delta}}_{(1)}\|_2 \\ &= \max_{(i,j) \in E} \|(\mathbf{C}^+_{(i,j)})^T \mathbf{V}_{(1)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2 \mathcal{R}(\widehat{\boldsymbol{\Delta}}).\end{aligned}$$

Combined with the bound (20), this implies that, for $\lambda \ge 2n^{-1}\sqrt{\bar{c} N p \log(|E|p)/\lambda_G}$,

$$|T_1| \le \frac{\lambda}{2} \mathcal{R}(\widehat{\boldsymbol{\Delta}}) \tag{27}$$

with probability at least $1 - 2(|E|p)^{-1}$. Also, using (26) and (21),

$$|T_2| \le \|\mathbf{V}_{(2)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2 \|\widehat{\boldsymbol{\Delta}}_{(2)}\|_2 \le \frac{n}{\kappa_{\underline{c}} N} \|\mathbf{V}_{(2)}^T \nabla \widetilde{\ell}^M(\boldsymbol{\theta}^*)\|_2^2 \le \frac{\bar{c}}{4\kappa_{\underline{c}}} \left(\frac{rp}{n} + \frac{1}{n}\sqrt{rp \log n} \right) \tag{28}$$

with probability at least $1 - \exp\{-\min(c_1 \log n, c_2 \sqrt{rp \log n})\}$. Combining (24), (25), (27), and (28) and using the triangle inequality yields

$$\begin{aligned}\frac{\kappa_{\underline{c}} N}{n} \|\widehat{\boldsymbol{\Delta}}\|_2^2 &\le \lambda \mathcal{R}(\boldsymbol{\theta}^*) - \lambda \mathcal{R}(\widehat{\boldsymbol{\theta}}^M) + \frac{\lambda}{2} \mathcal{R}(\widehat{\boldsymbol{\Delta}}) + \frac{\bar{c}}{4\kappa_{\underline{c}}} \left(\frac{rp}{n} + \frac{1}{n}\sqrt{rp \log n} \right) \\ &\le \lambda \mathcal{R}(\boldsymbol{\theta}^*) - \lambda \mathcal{R}(\widehat{\boldsymbol{\theta}}^M) + \frac{\lambda}{2} (\mathcal{R}(\widehat{\boldsymbol{\theta}}^M) + \mathcal{R}(\boldsymbol{\theta}^*)) + \frac{\bar{c}}{4\kappa_{\underline{c}}} \left(\frac{rp}{n} + \frac{1}{n}\sqrt{rp \log n} \right) \\ &\le \frac{3\lambda}{2} \mathcal{R}(\boldsymbol{\theta}^*) + \frac{\bar{c}}{4\kappa_{\underline{c}}} \left(\frac{rp}{n} + \frac{1}{n}\sqrt{rp \log n} \right).\end{aligned}$$

Dividing both sides by $\kappa_{\underline{c}} N$ completes the proof.

B Supplementary Material

The supplementary material contains R code for implementing the proposed methods.

References

Ackerman, M., and Ben-David, S. (2016), "A Characterization of Linkage-Based Hierarchical Clustering," *Journal of Machine Learning Research*, 17, 1–17.

Anderlucci, L., and Viroli, C. (2020), "Mixtures of Dirichlet-Multinomial Distributions for Supervised and Unsupervised Classification of Short Text Data," *Advances in Data Analysis and Classification*, 14, 759–770.

Arthur, D., and Vassilvitskii, S. (2007), "k-means++: The Advantages of Careful Seeding," in *Proceedings of the 18th Annual ACM-SIAM Symposium on Discrete Algorithms*, pp. 1027–1035.

Barrat, A., Barthélemy, M., and Vespignani, A. (2008), *Dynamical Processes on Complex Networks*, Cambridge: Cambridge University Press.

Blei, D. M., Ng, A. Y., and Jordan, M. I. (2003), "Latent Dirichlet Allocation," *Journal of Machine Learning Research*, 3, 993–1022.

Blondel, V. D., Guillaume, J.-L., Lambiotte, R., and Lefebvre, E. (2008), "Fast Unfolding of Communities in Large Networks," *Journal of Statistical Mechanics*, 2008, P10008.

Bouveyron, C., Celeux, G., Murphy, T. B., and Raftery, A. E. (2019), *Model-Based Clustering and Classification for Data Science: With Applications in R*, Cambridge: Cambridge University Press.

Chen, J., and Chen, Z. (2008), "Extended Bayesian Information Criteria for Model Selection With Large Model Spaces," *Biometrika*, 95, 759–771.

Chen, J., and Li, H. (2013), "Variable Selection for Sparse Dirichlet-Multinomial Regression With an Application to Microbiome Data Analysis," *The Annals of Applied Statistics*, 7, 418–442.

Chi, E. C., and Lange, K. (2015), "Splitting Methods for Convex Clustering," *Journal of Computational and Graphical Statistics*, 24, 994–1013.

Chi, E. C., and Steinerberger, S. (2019), "Recovering Trees With Convex Clustering," *SIAM Journal on Mathematics of Data Science*, 1, 383–407.

Di Nuzzo, C., and Ingrassia, S. (2022), "A Mixture Model Approach to Spectral Clustering and Application to Textual Data," *Statistical Methods & Applications*, to appear.

Elkan, C. (2006), "Clustering Documents With an Exponential-Family Approximation of the Dirichlet Compound Multinomial Distribution," in *Proceedings of the 23rd International Conference on Machine Learning*, pp. 289–296.

Everitt, B. S., Landau, S., Leese, M., and Stahl, D. (2011), *Cluster Analysis* (5th ed.), Chichester: Wiley.

Godsil, C., and Royle, G. (2001), *Algebraic Graph Theory*, New York: Springer.

Guo, J., Levina, E., Michailidis, G., and Zhu, J. (2010), "Pairwise Variable Selection for High-Dimensional Model-Based Clustering," *Biometrics*, 66, 793–804.

Hallac, D., Leskovec, J., and Boyd, S. (2015), "Network Lasso: Clustering and Optimization in Large Graphs," *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 387–396.

Hocking, T. D., Joulin, A., Bach, F., and Vert, J.-P. (2011), "Clusterpath: An Algorithm for Clustering Using Convex Fusion Penalties," in *Proceedings of the 28th International Conference on Machine Learning*.

Holmes, I., Harris, K., and Quince, C. (2012), "Dirichlet Multinomial Mixtures: Generative Models for Microbial Metagenomics," *PLoS One*, 7, e30126.

Hopkins, B., and Skellam, J. G. (1954), "A New Method for Determining the Type of Distribution of Plant Individuals," *Annals of Botany*, 18, 213–227.

Li, G., and Pong, T. K. (2016), "Douglas–Rachford Splitting for Nonconvex Optimization With Application to Nonconvex Feasibility Problems," *Mathematical Programming*, Series A, 159, 371–401.

Lindsten, F., Ohlsson, H., and Ljung, L. (2011), "Clustering Using Sum-of-Norms Regularization: With Application to Particle Filter Output Computation," in *IEEE Statistical Signal Processing Workshop*, pp. 201–204.

Liu, J., Yuan, L., and Ye, J. (2013), "Guaranteed Sparse Recovery Under Linear Transformation," in *Proceedings of the 30th International Conference on Machine Learning*, pp. 91–99.

Manning, C. D., Raghavan, P., and Schütze, H. (2008), *Introduction to Information Retrieval*, Cambridge: Cambridge University Press.

Mikolov, T., Chen, K., Corrado, G., and Dean, J. (2013), "Efficient Estimation of Word Representations in Vector Space," in *ICLR Workshop Proceedings*.

Mosimann, J. E. (1962), "On the Compound Multinomial Distribution, the Multivariate β-Distribution, and Correlations Among Proportions," *Biometrika*, 49, 65–82.

Newman, M. E. J., and Leicht, E. A. (2007), "Mixture Models and Exploratory Analysis in Networks," *Proceedings of the National Academy of Sciences*, 104, 9564–9569.

Nigam, K., McCallum, A. K., Thrun, S., and Mitchell, T. (2000), "Text Classification From Labeled and Unlabeled Documents Using EM," *Machine Learning*, 39, 103–134.

Pelckmans, K., De Brabanter, J., Suykens, J. A. K., and De Moor, B. (2005), "Convex Clustering Shrinkage," in *PASCAL Workshop on Statistics and Optimization of Clustering*.

Radchenko, P., and Mukherjee, G. (2017), "Convex Clustering via l_1 Fusion Penalization," *Journal of the Royal Statistical Society*, Series B, 79, 1527–1546.

Rudelson, M., and Vershynin, R. (2013), "Hanson-Wright Inequality and Sub-Gaussian Concentration," *Electronic Communications in Probability*, 18, 1–9.

Sen, P., Namata, G., Bilgic, M., Getoor, L., Galligher, B., and Eliassi-Rad, T. (2008), "Collective Classification in Network Data," *AI Magazine*, 29, 93–106.

Steinley, D. (2006), "K-Means Clustering: A Half-Century Synthesis," *British Journal of Mathematical and Statistical Psychology*, 59, 1–34.

Tan, K. M., and Witten, D. (2015), "Statistical Properties of Convex Clustering," *Electronic Journal of Statistics*, 9, 2324–2347.

Tandon, A., Albeshri, A., Thayananthan, V., Alhalabi, W., and Fortunato, S. (2019), "Fast Consensus Clustering in Complex Networks," *Physical Review E*, 99, 042301.

Tang, Z.-Z., and Chen, G. (2019), "Zero-Inflated Generalized Dirichlet Multinomial Regression Model for Microbiome Compositional Data Analysis," *Biostatistics*, 20, 698–713.

Tibshirani, R., Saunders, M., Rosset, S., Zhu, J., and Knight, K. (2005), "Sparsity and Smoothness via the Fused Lasso," *Journal of the Royal Statistical Society*, Series B, 67, 91–108.

Watts, D. J., and Strogatz, S. H. (1998), "Collective Dynamics of 'Small-World' Networks," *Nature*, 393, 440–442.

Weir, B. S., and Hill, W. G. (2002), "Estimating F-Statistics," *Annual Review of Genetics*, 36, 721–750.

Wu, G. D., Chen, J., Hoffmann, C., Bittinger, K., Chen, Y.-Y., Keilbaugh, S. A., Bewtra, M., Knights, D., Walters, W. A., Knight, R., Sinha, R., Gilroy, E., Gupta, K., Baldassano, R., Nessel, L., Li, H., Bushman, F. D., and Lewis, J. D. (2011), "Linking Long-Term Dietary Patterns With Gut Microbial Enterotypes," *Science*, 334, 105–108.

Xu, X., Yuruk, N., Feng, Z., and Schweiger, T. A. J. (2007), "SCAN: A Structural Clustering Algorithm for Networks," in *Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 824–833.

Yin, P., Pham, M., Oberman, A., and Osher, S. (2018), "Stochastic Backward Euler: An Implicit Gradient Descent Algorithm for k-Means Clustering," *Journal of Scientific Computing*, 77, 1133–1146.

Zhang, J., and Lin, W. (2019), "Scalable Estimation and Regularization for the Logistic Normal Multinomial Model," *Biometrics*, 75, 1098–1108.

Zhou, H., and Lange, K. (2010), "MM Algorithms for Some Discrete Multivariate Distributions," *Journal of Computational and Graphical Statistics*, 19, 645–665.

Zhu, C., Xu, H., Leng, C., and Yan, S. (2014), "Convex Optimization Procedure for Clustering: Theoretical Revisit," in *Advances in Neural Information Processing Systems*, vol. 27, pp. 1619–1627.