Deep Clustering with Incomplete Noisy Pairwise Annotations: A Geometric Regularization Approach

Tri Nguyen [1] **Shahana Ibrahim** [1] **Xiao Fu** [1]

Abstract

The recent integration of deep learning and pairwise similarity annotation-based constrained clustering—i.e., *deep constrained clustering* (DCC)—has proven effective for incorporating weak supervision into massive data clustering: Less than 1% of pair similarity annotations can often substantially enhance the clustering accuracy. However, beyond empirical successes, there is a lack of understanding of DCC. In addition, many DCC paradigms are sensitive to annotation noise, but performance-guaranteed noisy DCC methods have been largely elusive. This work first takes a deep look into a recently emerged logistic loss function of DCC, and characterizes its theoretical properties. Our result shows that the logistic DCC loss ensures the identifiability of data membership under reasonable conditions, which may shed light on its effectiveness in practice. Building upon this understanding, a new loss function based on geometric factor analysis is proposed to fend against noisy annotations. It is shown that even under *unknown* annotation confusions, the data membership can still be *provably* identified under our proposed learning criterion. The proposed approach is tested over multiple datasets to validate our claims.

1. Introduction

Clustering is one of the most prominent unsupervised learning tasks (Jain & Dubes, 1988). Classic clustering paradigms, e.g., K-means and spectral clustering, are designed to work without any label information. In practice, it was observed that limited supervision may significantly boost the clustering performance. The so-called *constrained clustering* (CC) method (Wagstaff & Cardie, 2000) is one of such weak (or semi-)supervised approaches. The CC paradigm has many variants (see (Basu et al., 2008; Schultz & Joachims, 2003; Yeung & Yeo, 1996; Davidson et al., 2010; Zhang et al., 2019; 2021)). The arguably most widely used paradigm annotates the similarity (using binary codes) of data pairs and restrains the clustering outcomes to respect these pairwise constraints; see, e.g., (Basu et al., 2004a; Segal et al., 2003; Wagstaff et al., 2001; Givoni & Frey, 2009; Hsu & Kira, 2015; Manduchi et al., 2021). Note that annotating similarity is considerably easier than annotating the exact class labels of data, yet it was observed that such "simple" annotations could boost the performance of data categorization. These findings may assist designing economical annotation mechanisms in the era of data explosion.

Early CC approaches are often combined with existing clustering modules like K-means (Basu et al., 2004a; Wagstaff et al., 2001; Bilenko et al., 2004; Li et al., 2009; Kamvar et al., 2003; Wang et al., 2014; Cucuringu et al., 2016). The pairwise annotations are used to construct regularization terms that are similar to graph-Laplacian based regularization. Variants using Bayesian perspectives were also proposed (Basu et al., 2004b; Law et al., 2005). These methods were observed to largely outperform their unsupervised counterparts, e.g., K-means and spectral clustering, even only as few as 1% of the pairs are annotated.

In recent years, clustering modules and deep neural network-based feature extractors are proposed to be learned jointly—leading to the so-called end-to-end deep clustering approaches; see, e.g., (Yang et al., 2017; Caron et al., 2018). These methods use clustering structures in the latent space to regularize the neural networks, and the neural networks offers enhanced transformation power to find latent spaces where the data can be well clustered. The CC methods also benefited from similar ideas. The so-called *deep constrained clustering* (DCC) approaches were proposed in (Zhang et al., 2019; 2021; Manduchi et al., 2021), and substantial performance enhancement relative to classic CC methods was observed.

Although the DCC methods have enjoyed empirical successes, some notable challenges remain. First, there is a general lack of understanding to the effectiveness of DCC. It is

[1]School of Electrical Engineering and Computer Science, Oregon State University, OR, USA. Correspondence to: Xiao Fu <xiao.fu@oregonstate.edu>.

Proceedings of the 40^{th} International Conference on Machine Learning, Honolulu, Hawaii, USA. PMLR 202, 2023.

unclear under what conditions the loss functions constructed for DCC could succeed or fail in finding the ground-truth cluster membership of the data entities. However, understanding the *identifiability* of the membership is critical for designing principled and robust DCC systems. The interplay of key aspects, e.g., neural network complexity and the generalization ability of the learned feature extractor, in the context of DCC is also of great interest—yet no pertinent study exists. Second, most of the existing (D)CC methods (implicitly) assumed that annotations are accurate; see, e.g., (Basu et al., 2004a; Wagstaff et al., 2001; Li et al., 2009; Zhang et al., 2019; Ren et al., 2019; Hsu et al., 2018). Hence, many of these methods may not be robust to annotation noise. This is particularly detrimental to DCC methods as large over-parameterized neural models easily overfit (Du et al., 2019). Some works took noisy annotations into consideration (e.g., (Luo et al., 2018; Manduchi et al., 2021)), but no guarantees of recovering the cluster membership.

Contributions. In this work, we take a deeper look at the DCC problem from a membership-identifiability analysis viewpoint. Our contribution is twofold:

First, we re-examine a recently emerged effective loss function of DCC, namely, the logistic loss-based DCC criterion, which was proposed by (Hsu et al., 2018; Zhang et al., 2021). We show that, if the pairwise annotations are generated following a model that is reminiscent of the mixed-membership stochastic blockmodel (MMSB) (Airoldi et al., 2008; Huang & Fu, 2019), then, the logistic loss can provably recover (i) the data entities' cluster membership and (ii) the nonlinear function that maps the data features to the membership indicator for both seen and unseen data—and thus generalization of the learned neural network is guaranteed.

Second, using our understanding, we propose a noisy annotation-robust version of logistic loss for DCC. We explicitly model the annotators' confusion as a probability transition matrix, which is inspired by classic noisy label analysis such as the Dawid-Skene model (Dawid & Skene, 1979; Ghosh et al., 2011; Zhang et al., 2014). We propose a geometric factor analysis (Fu et al., 2018; 2019) based learning criterion to *provably* ensure the identifiability of the ground-truth cluster membership, in the presence of annotation confusion.

We test our method over a series of DCC tasks and observe that the proposed approach significantly improves the performance over existing paradigms, especially when annotation noise exists. Our finding shows the significance of identifiability in DCC, echoing observations made in similar semi-supervised/unsupervised problems, e.g., (Arora et al., 2013; Kumar et al., 2013; Anandkumar et al., 2014; Zhang et al., 2014). We also evaluate the algorithms using real data collected through the Amazon Mechanical Turk (AMT) platform. The code is published at `github.com/ductri/VolMaxDCC`.

Notation. We use x, $\boldsymbol{x}$ and $\boldsymbol{X}$ to denote scalar, vector and matrix, respectively; both $[\boldsymbol{x}]_k, x_k$ refer to the kth element of vector $\boldsymbol{x}$; X_{ij} (and $[\boldsymbol{X}]_{i,j}$) is the element in the ith row and jth column of $\boldsymbol{X}$; $\boldsymbol{I}_K$ denotes the identity matrix of size K; $\mathbf{0}$ and $\mathbf{1}$ are all-zero and all-one matrices with proper sizes; $\langle \boldsymbol{x}, \boldsymbol{y} \rangle$ and $\langle \boldsymbol{X}, \boldsymbol{Y} \rangle$ denote dot products between two vectors and two matrices, respectively; $\|\boldsymbol{x}\|$ denotes the ℓ_2-norm; $\|\boldsymbol{X}\|_{\mathrm{F}}$ and $\|\boldsymbol{X}\|_2$ denote the Frobenius norm and spectral norm of $\boldsymbol{X}$, respectively; $\sigma_{\max}(\boldsymbol{X}), \sigma_{\min}(\boldsymbol{X})$, and $\sigma_i(\boldsymbol{X})$ represent the largest, the smallest, and the ith singular value of matrix $\boldsymbol{X}$, respectively; $[N]$ is the set of natural numbers from 1 to N, i.e., $[N] = \{1, \ldots, N\}$; $[N] \times [N]$ denotes the set of all possible pairs of (i, j) where $i \in [N], j \in [N]$; $\mathrm{cone}(\boldsymbol{X})$ to denote conic hull of the column vectors of $\boldsymbol{X}$, i.e., $\mathrm{cone}(\boldsymbol{X}) = \{\boldsymbol{y} \mid \boldsymbol{y} = \boldsymbol{X}\boldsymbol{\theta}, \boldsymbol{\theta} \geq \mathbf{0}\}$.

2. Background

Problem Setting. We consider the CC setting as follows: There are N data samples $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$. Each sample belongs to one or multiple clusters of in total K clusters. The association of $\boldsymbol{x}_n$ with the clusters is represented by a vector $\boldsymbol{m}_n \in \mathbb{R}^K$. The element $[\boldsymbol{m}_n]_k$ represents the probability that data n belongs to cluster k. Note that in hard clustering, $[\boldsymbol{m}_n]_k \in \{0, 1\}$ for all $k \in [K]$, which means the clusters have no overlaps. In more general cases, we have

$$\mathbf{1}^\top \boldsymbol{m}_n = 1, \ \boldsymbol{m}_n \geq \mathbf{0}. \tag{1}$$

A collection of M pairwise annotations are available, which are denoted by $(i_1, j_1, y_1), \ldots, (i_M, j_M, y_M)$. Here,

$$y_m = \begin{cases} 1, & \boldsymbol{x}_{i_m}, \boldsymbol{x}_{j_m} \text{ are "similar"}, \\ 0, & \text{otherwise}, \end{cases} \tag{2}$$

where the similarity of the membership of $\boldsymbol{x}_{i_m}$ and $\boldsymbol{x}_{j_m}$ is often deemed by an annotator. Note that there are in total $N(N-1)/2$ such data pairs, and we often have

$$M \ll N(N-1)/2;$$

that is, only a small portion of the data pairs are annotated. The objective of pairwise annotation-based CC is to find the cluster membership vector $\boldsymbol{m}_n$ of each $\boldsymbol{x}_n$ using the data and the M annotations.

Early CC Methods. The task of clustering with the pairwise constraints can be dated back to the early 2000s, where (Wagstaff & Cardie, 2000; Wagstaff et al., 2001) used the pairwise annotations to impose extra constraints of the classic K-means iterations. The work (Basu et al., 2004a) considered using pairwise annotation-induced "soft" constraints

(or, regularization terms) to modify K-means; see similar ideas in (Bilenko et al., 2004). Instead of modifying K-means, another line of approaches proposed to work with pairwise constraints under the spectral clustering framework. The idea is to incorporate the pairwise annotation-based constraints into the construction of the graph affinity matrix; see, e.g., (Kulis et al., 2005; Lu & Carreira-Perpinan, 2008; Li et al., 2009; Cucuringu et al., 2016).

DCC Developments. In recent years, deep neural network-based feature extractors were proposed to combine with CC—leading to the *deep constrained clustering* (DCC) paradigms, e.g., (Manduchi et al., 2021; Luo et al., 2018; Zhang et al., 2019; 2021). In DCC, a deep neural network (DNN) $\boldsymbol{f}_{\boldsymbol{\theta}}(\cdot)$ is used to link the data vector $\boldsymbol{x}_n$ with its membership, i.e.,

$$\boldsymbol{m}_n = \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_n).$$

The use of DNN helps nonlinearly transform the data to spaces that are "friendly" to clustering (Yang et al., 2017). Learning $\boldsymbol{f}_{\boldsymbol{\theta}}(\cdot)$ also allows the neural network to generalize to unseen data. By (1),

$$\boldsymbol{m}_{i_m}^\top \boldsymbol{m}_{j_m} \in [0,1]$$

can be considered as the probability that $\boldsymbol{x}_{i_m}$ and $\boldsymbol{x}_{j_m}$ belong to the same cluster, i.e., $y_m \sim \mathsf{Bernoulli}(\boldsymbol{m}_{i_m}^\top \boldsymbol{m}_{j_m})$. From this perspective, recent works have proposed an extension of logistic regression to incorporate pairwise annotations (Hsu et al., 2018; Zhang et al., 2019; 2021):

$$\mathsf{Loss}_{\mathrm{cc}}(\boldsymbol{\theta}) = \frac{1}{M}\sum_{m=1}^{M}\Big(y_m \log \frac{1}{\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{i_m})^\top \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{j_m})} + (1-y_m)\log\frac{1}{1-\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{i_m})^\top \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{j_m})}\Big). \quad (3)$$

In the literature, the $\mathsf{Loss}_{\mathrm{cc}}$ term is sometimes used with other loss functions; e.g., (Zhang et al., 2019; 2021) used an overall loss function consisting of two terms:

$$\text{minimize } \mathsf{Loss}_{\mathrm{recon}} + \lambda \mathsf{Loss}_{\mathrm{cc}}, \quad (4)$$

where $\lambda \geq 0$ and the reconstruction loss $\mathsf{Loss}_{\mathrm{recon}}$ was realized using an autoencoder. Such combination is advocated for practical reasons, e.g., utilizing all available data. Nonetheless, as we will show, $\mathsf{Loss}_{\mathrm{cc}}$ itself suffices to offer strong guarantees under reasonable conditions.

Challenges. Although there have been abundant empirical evidence demonstrating the effectiveness of CC and DCC, theoretical understanding has been largely behind. This is particularly obvious for the DCC case, where aspects such as the identifiability of $\boldsymbol{m}_n$ and the generalization ability of the learned $\boldsymbol{f}_{\boldsymbol{\theta}}(\cdot)$ are of great interest—yet no theoretical support exists, to our best knowledge.

Another challenge lies in noise robustness. Although it has been widely observed that noisy annotations could greatly impact the performance of CC and DCC (Liu et al., 2017; Covoes et al., 2013; Pelleg & Baras, 2007; Manduchi et al., 2021; Luo et al., 2018; Chang et al., 2017b; Zhang et al., 2021; Zhu et al., 2015), effective solutions—especially performance-guaranteed ones—for handling this problem have been largely lacking. Many works did test their algorithms with noisy labels (see, e.g., (Cucuringu et al., 2016; Wang et al., 2014)), but no special care was taken to alleviate the impact of such noisy labels. Some heuristics—such as pre-processing the data (Yi et al., 2012), modeling annotation uncertainties (Manduchi et al., 2021), and introducing concepts reflecting human behaviors (Luo et al., 2018; Chang et al., 2017b) and annotators' accuracy (Luo et al., 2018; Chang et al., 2017b)—were also proposed in the literature. However, performance guarantees have been elusive.

3. DCC Loss Revisited: Identifiability and Generalization

In this section, we take a deeper look into the DCC loss in (3) and understand its theoretical properties. Such understanding will allow us to design a performance-guaranteed new DCC loss in the presence of noisy pairwise annotations.

3.1. A Generative Model of Annotations

To better present the results, in this section, we use the superscript "$\natural$" to denote all the ground-truth terms; e.g., $\boldsymbol{f}^\natural(\cdot)$ denotes the ground-truth nonlinear mapping from data to membership and $\boldsymbol{m}_n^\natural = \boldsymbol{f}^\natural(\boldsymbol{x}_n)$ denotes the ground-truth membership vector of sample n. We propose to employ the following generative model of y_m: Given $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N \sim \mathcal{P}_{\mathcal{X}}, \boldsymbol{x}_n \in \mathcal{X}$,

$$i, j \text{ are sampled over } [N] \times [N]\,; \quad (5a)$$

$$\boldsymbol{m}_i^\natural = \boldsymbol{f}^\natural(\boldsymbol{x}_i) \text{ and } \boldsymbol{m}_j^\natural = \boldsymbol{f}^\natural(\boldsymbol{x}_j); \quad (5b)$$

$$y_{i,j} \sim \mathsf{Bernoulli}(\langle \boldsymbol{m}_i^\natural, \boldsymbol{m}_j^\natural \rangle). \quad (5c)$$

Note that the logistic loss in (3) is the maximum likelihood estimator (MLE) of the parameters in the generative model. The model is reminiscent of the classic generative models of logistic regression and network analysis, particularly, MMSB (Airoldi et al., 2008). In MMSB, the nonlinear mapping from the data features to the membership vectors were not considered. Incorporating the nonlinear mapping $\boldsymbol{f}^\natural$ follows the ideas from supervised learning, where the relationship between the data features and the data class is often modeled as the following conditional probability represented by $\boldsymbol{f}^\natural$ (Shalev-Shwartz & Ben-David, 2014):

$$[\boldsymbol{m}_n^\natural]_k = \mathsf{Pr}(y = k | \boldsymbol{x}_n) = [\boldsymbol{f}^\natural(\boldsymbol{x}_n)]_k.$$

Once $\boldsymbol{f}^\natural$ is learned, it can be used as a multi-class classifier. This perspective was also mentioned in (Hsu et al., 2018; Zhang et al., 2019; 2021)—for algorithm design purpose. However, membership identifiability was not addressed.

In this section, we will show that the logistic loss (3) ensures identifying the membership vectors $\boldsymbol{m}_n$ under the generative model in (5). It also ensures that $\boldsymbol{f}_{\boldsymbol{\theta}} \approx \boldsymbol{f}^\natural$, under reasonable conditions. These findings may shed some light onto the effectiveness and good generalization performance of DCC using (3) in the literature.

3.2. Performance Analysis

Finite-Sample Identifiability and Generalization. We first show that Loss_{cc} is a sound criterion for identifying $\boldsymbol{m}_n^\natural$ and $\boldsymbol{f}^\natural(\cdot)$ by itself. Specifically, let us denote

$$\boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} \mathsf{Loss}_{\mathrm{cc}}(\boldsymbol{\theta}) \tag{6}$$

and $\boldsymbol{f}^\star = \boldsymbol{f}_{\boldsymbol{\theta}^\star}$. Here, $\boldsymbol{f}_{\boldsymbol{\theta}}$ is represented by a deep neural network. To be more specific, we consider $\boldsymbol{f}_{\boldsymbol{\theta}}$ belonging to a function class $\mathcal{F}$ defined by

$$\mathcal{F} \triangleq \left\{ \mathtt{softmax}(\mathtt{net}(\boldsymbol{x};\boldsymbol{\theta})) \mid \forall \boldsymbol{x} \in \mathcal{X} \right\},$$

where $\mathtt{net}(\cdot;\boldsymbol{\theta})$ is a neural network that maps $\boldsymbol{x}$ to $\mathbb{R}^K$, and $[\mathtt{softmax}(\boldsymbol{x})]_k \triangleq \exp(x_k)/\sum_{\ell=1}^K \exp(x_\ell)$ is imposed onto the output layer of the neural network, which is used to reflect the constraints in (1).

We will show that $\boldsymbol{f}^\star \approx \boldsymbol{f}^\natural$ and $\boldsymbol{m}_n^\star = \boldsymbol{f}^\star(\boldsymbol{x}_n) \approx \boldsymbol{m}_n^\natural$ under reasonable conditions. To proceed, let us invoke the following assumptions:

Assumption 3.1 (Anchor Sample Condition (ASC))**.** Let $\boldsymbol{m}_n^\natural = \boldsymbol{f}^\natural(\boldsymbol{x}_n)$, $\boldsymbol{M}^\natural = [\boldsymbol{m}_1^\natural, \ldots, \boldsymbol{m}_N^\natural]$. $\boldsymbol{M}^\natural$ satisfies ASC if there exists a set $\mathcal{K}$ of K indices such that $\boldsymbol{M}^\natural[:,\mathcal{K}] = \boldsymbol{I}$. Accordingly, define $\boldsymbol{V}^\natural \triangleq \boldsymbol{M}^\natural[:,\mathcal{K}^c], \mathcal{K}^c \triangleq [N] \setminus \mathcal{K}$.

Assumption 3.2. (Function Class) There exist $0 < \nu < 1$ and $\widetilde{\boldsymbol{f}} \in \mathcal{F}$ such that

$$\left\| \widetilde{\boldsymbol{f}}(\boldsymbol{x}) - \boldsymbol{f}^\natural(\boldsymbol{x}) \right\| \le \nu, \forall \boldsymbol{x} \in \mathcal{X}. \tag{7}$$

In addition, $\alpha < \boldsymbol{f}(\boldsymbol{x})^\top \boldsymbol{f}(\boldsymbol{y}) < 1-\alpha,\ \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}, \forall \boldsymbol{f} \in \mathcal{F}$, for some $0 < \alpha < 1$. The complexity measure of the neural network $\mathtt{net}(\boldsymbol{x};\boldsymbol{\theta})$ is $R_{\mathtt{NET}}$.

The ASC means that there exist samples that solely belong to a single cluster, which are called the anchor samples. This assumption is reminiscent of the anchor point assumption in the community detection literature (Panov et al., 2017; Mao et al., 2017). Condition (7) takes the approximation error of the employed neural network class $\mathcal{F}$ into consideration. The assumption on α is a regularity condition that prevents pathological unbounded cases of the logistic loss from happening. The constant $R_{\mathtt{NET}}$ is proportional to the upper bound of the so-called spectral complexity in (Bartlett et al., 2017). A formal definition of $R_{\mathtt{NET}}$ is given in Lemma A.10. The parameters ν and $R_{\mathtt{NET}}$ present a tradeoff: Roughly speaking, if one has a deeper and wider neural network, then ν is smaller—but $R_{\mathtt{NET}}$ is bigger.

Define $\boldsymbol{P}^\natural \in \mathbb{R}^{N\times N}$ such that $P_{ij}^\natural = \langle \boldsymbol{f}^\natural(\boldsymbol{x}_i), \boldsymbol{f}^\natural(\boldsymbol{x}_j)\rangle$, $\boldsymbol{P}^\star \in \mathbb{R}^{N\times N}$ such that $P_{ij}^\star = \langle \boldsymbol{f}^\star(\boldsymbol{x}_i), \boldsymbol{f}^\star(\boldsymbol{x}_j)\rangle$, and $\boldsymbol{S}_{\boldsymbol{X}} = [\boldsymbol{x}_{i_1}, \boldsymbol{x}_{j_1}, \ldots, \boldsymbol{x}_{i_M}, \boldsymbol{x}_{i_M}]$. We first show that the matrix $\boldsymbol{P}^\natural$ is approximately recoverable via minimizing (3):

Lemma 3.3. *Let $S = \{(i_1, j_1, y_1), \ldots, (i_M, j_M, y_M)\}$, where $(i_1, j_1), \ldots, (i_M, j_M)$ are drawn independently and uniformly at random from $[N] \times [N]$. Suppose that $y_m \mid (i_m, j_m) \sim \mathsf{Bernoulli}(P^\natural_{i_m,j_m})$ following the generative model in (5) and that $\mathcal{F}$ satisfies Assumption 3.2. Then, with probability at least $1-\delta$, we have*

$$\frac{1}{N^2} \left\| \boldsymbol{P}^\star - \boldsymbol{P}^\natural \right\|_{\mathrm{F}}^2 \le \epsilon(M,\delta)^2,$$

where $\epsilon(M,\delta)^2$ is defined as follows:

$$\epsilon(M,\delta)^2 \triangleq \frac{64\log(1/\alpha)}{M} + \frac{96\sqrt{2}\log M}{\alpha M \log 2} \left\| \boldsymbol{S}_{\boldsymbol{X}} \right\|_{\mathrm{F}} \sqrt{R_{NET}} + 64\log(1/\alpha)\sqrt{\frac{2\log(4/\delta)}{M}} + \frac{16\nu}{\alpha}. \tag{8}$$

The proof of Lemma 3.3 is relegated to Appendix A. It is not surprising that $\boldsymbol{P}^\natural$ can be recovered from minimizing the logistic loss, as the problem of recovering $\boldsymbol{P}^\natural$ can be regarded as a generalized 1-bit matrix completion (MC) problem. Unlike the conventional 1-bit MC frameworks that leveraged the low-rank structure of the complete data (see, e.g., (Davenport et al., 2014)), here we exploit the neural generative model in (5), which is also a low-dimensional model, as long as $R_{\mathtt{NET}}$ is sufficiently small (compared to M and N).

Before proceeding to showing recovery of $\boldsymbol{M}^\natural$, let us observe the following fact based on Lemma 3.3: It can be seen in (8) that $\epsilon(M, M^{-0.5})^2$ is decreasing with a rate of $\mathcal{O}(\sqrt{(\log M)/M})$, hence there exists $M_0 \in \mathbb{N}$ independent to N such that $\forall M > M_0$,

$$\epsilon'(M)^2 \triangleq M^{0.25}\epsilon(M, M^{-0.5})^2 + M^{-0.25} \le \frac{1}{8K^2}. \tag{9}$$

Using the above fact, we show that

Theorem 3.4. *Under the same assumptions as in Lemma 3.3, further assume that Assumption 3.1 is satisfied. Let $\boldsymbol{M}^\star = [\boldsymbol{f}^\star(\boldsymbol{x}_1), \ldots, \boldsymbol{f}^\star(\boldsymbol{x}_N)]$, and consider $M > M_0$, then there exists a permutation matrix $\boldsymbol{\Pi}^\star$ such that*

$$\frac{1}{NK} \left\| \boldsymbol{\Pi}^\star \boldsymbol{M}^\star - \boldsymbol{M}^\natural \right\|_{\mathrm{F}}^2 \le \frac{4N}{K}\epsilon(M,\delta)^2 + \frac{2K}{N}(1 + 8\sigma_{\max}^2(\boldsymbol{V}^\natural))\epsilon'(M),$$

holds with probability at least $1-\delta-K^2/M^{0.25}$*, where* $\epsilon'(M)$ *is defined in* (9).

The proof of Theorem 3.4 is in Appendix B. We should mention that the sample complexity in Theorem 3.4 is based on a worst-case analysis, and thus the bound tends to be pessimistic—it starts to make sense when

$$N \leq \mathcal{O}(K\sqrt{M}),$$

as reflected in the first term on the right hand side. In practice, $\mathsf{Loss}_{\text{cc}}$ can work fairly well using a much smaller M. Nonetheless, Theorem 3.4 for the first time shows the soundness of using $\mathsf{Loss}_{\text{cc}}$ in (3), in terms of being able to guarantee the identifiability of $\boldsymbol{M}^\natural$.

With Theorem 3.4, it is readily to show the following:

Theorem 3.5 (Generalization)**.** *Under the same conditions as in Theorem 3.4, with probability at least* $1-\delta-K^2/M^{1/4}$*,*

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{P}_{\mathcal{X}}}\left[\frac{1}{K}\left\|\boldsymbol{\Pi}^\star\boldsymbol{f}^\star(\boldsymbol{x})-\boldsymbol{f}^\natural(\boldsymbol{x})\right\|^2\right] \leq \frac{4N}{K}\epsilon(M,\delta)^2 + \frac{2K}{N}(1+8\sigma_{\max}^2(\boldsymbol{V}^\natural))\epsilon'(M) + \frac{4}{NK}+\frac{12\sqrt{2R_{NET}}\left\|\boldsymbol{X}\right\|_{\text{F}}\log N}{NK\log 2}+8\sqrt{\frac{2\log(4/\delta)}{NK^2}}.$$

See the proof in Appendix C. Theorem 3.5 confirms that the learned $\boldsymbol{f}^\star$ via minimizing $\mathsf{Loss}_{\text{cc}}$ can generalize to classify unseen data, which was observed in the literature (Hsu et al., 2018; Zhang et al., 2019; 2021) but never formally shown.

Enhanced Identifiability with Large Sample. In Theorem 3.4, the identifiability of $\boldsymbol{M}^\natural$ was established under finite samples. The proof relies on the ASC. One may argue that the anchor samples may not exist in some cases, as some data naturally exhibit mixed membership, e.g., social network users (Airoldi et al., 2008) and multi-topic documents (Blei et al., 2003). Here, we relax the finite sample assumption to show that at the limit of large M, the logistic loss enjoys enhanced membership identifiability that does not hinge on the ASC.

To see this, let us introduce the following condition:

Assumption 3.6 (Sufficiently Scattered Condition (SSC))**.** Let $\boldsymbol{m}_n^\natural=\boldsymbol{f}^\natural(\boldsymbol{x}_n)$, $\boldsymbol{M}^\natural=[\boldsymbol{m}_1^\natural,\ldots,\boldsymbol{m}_N^\natural]$. $\boldsymbol{M}^\natural$ satisfies SSC if there exists a subset $\mathcal{S}\in[N]$ such that

$$\mathcal{C}\subseteq\text{cone}(\boldsymbol{M}^\natural[:,\mathcal{S}]),$$

where $\mathcal{C}=\left\{\boldsymbol{x}\in\mathbb{R}^K\mid\sqrt{K-1}\left\|\boldsymbol{x}\right\|\leq\boldsymbol{1}^\top\boldsymbol{x}\right\}$, and $\text{cone}(\boldsymbol{M}^\natural[:,\mathcal{S}])\subseteq\text{cone}(\boldsymbol{Q})$ does not hold for any orthogonal $\boldsymbol{Q}$ except for the permutation matrices.

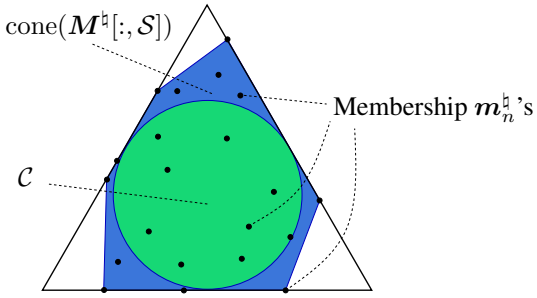


Figure 1: An example of $\boldsymbol{M}^\natural$ satisfying SSC when $K=3$.

The SSC is illustrated in Fig. 1 with a $K=3$ case. The triangle represents the nonnegative orthant and the green circle the second-order cone $\mathcal{C}$. SSC means that the conic hull of a subset of the $\boldsymbol{m}_n^\natural$'s (the blue region) covers the green region as a subset. This means that the columns of $\boldsymbol{M}^\natural$ are sufficiently different. The SSC is widely used in the nonnegative matrix factorization literature; see (Fu et al., 2018; 2015b). Note that $\boldsymbol{M}^\natural$ satisfying the SSC is much more relaxed compared to the ASC, which means that $\text{cone}(\boldsymbol{M}^\natural)=\mathbb{R}_+^K$, i.e., the blue region in Fig. 1 covers the entire triangle. In the context of membership learning, an SSC-satisfying $\boldsymbol{M}^\natural$ means that the membership of the data should contain enough diversity, but an ASC-satisfying $\boldsymbol{M}^\natural$ means that some single-membership data samples need to exist—which is more stringent.

Using the SSC, we show that the following holds:

Theorem 3.7. *(Enhanced Identifiability) Under the same assumptions as in Lemma 3.3, suppose that* $\boldsymbol{M}^\natural$ *satisfies SSC (Assumption 3.6) and that* $\boldsymbol{f}^\natural\in\mathcal{F}$*. Then, at the limit of* $\max\left(\log(1/\alpha),\log(M)\sqrt{R_{NET}}/\alpha\right)/\sqrt{M}\to 0$*, the following statements hold:*

(i) There exists a permutation matrix $\boldsymbol{\Pi}^\star$ *such that* $\boldsymbol{\Pi}^\star\boldsymbol{M}^\star=\boldsymbol{M}^\natural$.

(ii) The learned neural network $\boldsymbol{f}^\star$ *satisfies*

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{P}_{\mathcal{X}}}\left[\frac{1}{K}\left\|\boldsymbol{\Pi}^\star\boldsymbol{f}^\star(\boldsymbol{x})-\boldsymbol{f}^\natural(\boldsymbol{x})\right\|^2\right]\leq\frac{8}{NK}+\frac{12\sqrt{2R_{NET}}\left\|\boldsymbol{X}\right\|_{\text{F}}\log N}{NK\log 2}+8\sqrt{\frac{2\log(4/\delta)}{NK^2}}$$

with probability at least $1-\delta$.

The proof of Theorem 3.7 is in Appendix D. Theorem 3.7 has plausible implications in practice: When the sample size M is large and $\mathcal{F}$ is expressive, even no anchor samples exist, identifying $\boldsymbol{M}^\natural$ and finding a generalizable $\boldsymbol{f}^\star$ are still possible.

4. DCC with Noisy Labels: Geometric Regularization

Incorporating Annotation Confusion. We should mention that the generative model in (5) can already model annotation noise to a certain extent: The Bernoulli sampling process can encode some 0-1 flipping probability of observing y_m. Nonetheless, this level of noise consideration is not enough, as annotations could be grossly inaccurate. To take more severe annotation errors into consideration, we modify the generative model as follows. We assume that the annotator confuses class i with class j with probability $\mathsf{Pr}(j|i)$. Let $A_{i,j} = \mathsf{Pr}(i|j)$, we have a "confusion matrix" $\boldsymbol{A} \in \mathbb{R}^{K\times K}$ where $[\boldsymbol{A}]_{i,j} = A_{i,j}$. Hence, the annotator's "confused membership vector" is modeled as

$$\boldsymbol{m}_n^{\text{confused}} = \boldsymbol{A}\boldsymbol{m}_n^\natural = \boldsymbol{A}\boldsymbol{f}^\natural(\boldsymbol{x}_n). \tag{10}$$

Then, the annotator's output is sampled from the following:

$$y_m \sim \mathsf{Bernoulli}(\langle \boldsymbol{A}\boldsymbol{f}^\natural(\boldsymbol{x}_{i_m}), \boldsymbol{A}\boldsymbol{f}^\natural(\boldsymbol{x}_{j_m})\rangle). \tag{11}$$

Note that using a confusion matrix to model noisy labels' generating process is widely seen in noisy label learning—but mostly under the supervised learning setting; see, e.g., (Dawid & Skene, 1979; Liu et al., 2012; Zhang et al., 2014; Chu et al., 2021). We argue that this confusion model is also suitable for pairwise annotation. The rationale is that the error happened in comparison is mainly caused by the annotator's confusion on the membership of $\boldsymbol{x}_{i_m}$ *or* that of $\boldsymbol{x}_{j_m}$—which is exactly reflected in (11).

Volume Maximization DCC. To proceed, we propose the following modified logistic loss:

$$\mathsf{Loss}'_{\text{cc}}(\boldsymbol{\theta}, \boldsymbol{B}) = \frac{1}{M}\sum_{m=1}^{M}\Big(y_m \log \frac{1}{\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{i_m})^\top \boldsymbol{B}\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{j_m})} + (1-y_m)\log\frac{1}{1-\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{i_m})^\top \boldsymbol{B}\boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_{j_m})}\Big),$$

where $\boldsymbol{B} \in \mathbb{R}^{K\times K}$ satisfies $0 \le \boldsymbol{B} \le 1$, as it is induced by $\boldsymbol{B} = \boldsymbol{A}^\top\boldsymbol{A}$. Note that we will use $\boldsymbol{\theta}$ and $\boldsymbol{B}$ as our optimization variables (instead of $\boldsymbol{A}$) as it simplifies the loss function. In addition, as our ultimate goal is to learn $\boldsymbol{M}^\natural$ and $\boldsymbol{f}^\natural$, the intermediate variable $\boldsymbol{A}$ does not need to be explicitly estimated.

We show that minimizing $\mathsf{Loss}'_{\text{cc}}(\boldsymbol{\theta}, \boldsymbol{B})$ provably recovers the data membership and finds a generalizable $\boldsymbol{f}^\star$, with an additional volume requirement satisfied by the solution. Specifically, we have the following theorem:

Theorem 4.1 (Identifiability of Noisy Case)**.** *Assume that the assumptions in Lemma 3.3 hold, except that the generative model is replaced by* (11)*. Suppose that $\boldsymbol{M}^\natural$ satisfies SSC (Assumption 3.6) and that $\boldsymbol{f}^\natural \in \mathcal{F}$. Also assume that* $\text{rank}(\boldsymbol{A}^\top\boldsymbol{A}\boldsymbol{M}^\natural) = K$*. Denote*

$$(\boldsymbol{\theta}^\star, \boldsymbol{B}^\star) = \arg\min\ \mathsf{Loss}'_{\text{cc}}(\boldsymbol{\theta}, \boldsymbol{B}) \tag{12}$$

and $\boldsymbol{f}^\star = \boldsymbol{f}_{\boldsymbol{\theta}^\star}$ and $\boldsymbol{m}_n^\star = \boldsymbol{f}^\star(\boldsymbol{x}_n)$. In addition, assume that $\mathsf{Loss}'_{\text{cc}}$ is minimized with a solution $\boldsymbol{M}^\star, \boldsymbol{B}^\star$ such that $\log\det(\boldsymbol{M}^\star(\boldsymbol{M}^\star)^\top)$ is maximized among all possible optimal solutions. Then, at the limit of $\max\big(\log(1/\alpha), \log(M)\sqrt{R_{NET}}/\alpha\big)/\sqrt{M} \to 0$, the following statements hold:

(i) There exists a permutation matrix $\boldsymbol{\Pi}^\star$ such that $\boldsymbol{\Pi}^\star\boldsymbol{M}^\star = \boldsymbol{M}^\natural$.

(ii) The learned neural network $\boldsymbol{f}^\star$ satisfies

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{P}_{\mathcal{X}}}\left[\frac{1}{K}\left\|\boldsymbol{\Pi}^\star\boldsymbol{f}^\star(\boldsymbol{x}) - \boldsymbol{f}^\natural(\boldsymbol{x})\right\|^2\right] \le \frac{8}{NK} + \frac{12\sqrt{2R_{NET}}\,\|\boldsymbol{X}\|_{\text{F}}\log N}{NK\log 2} + 8\sqrt{\frac{2\log(4/\delta)}{NK^2}}$$

with probability at least $1-\delta$.

The proof of Theorem 4.1 is in Appendix E. The take-home point is that, when one has large samples and an expressive neural network, the membership identifiability and generalization performance of using $\mathsf{Loss}'_{\text{cc}}$ can be as good as that of using $\mathsf{Loss}_{\text{cc}}$—as if there is no annotation confusion.

Of course, there are more requirements to satisfy under Theorem 4.1. Particularly, the maximal $\log\det(\boldsymbol{M}^\star(\boldsymbol{M}^\star)^\top)$ requirement is nontrivial. In practice, the optimization criterion in Theorem 4.1 can be approximated via a regularized version of $\mathsf{Loss}'_{\text{cc}}$ as follows:

$$\mathop{\text{minimize}}_{\boldsymbol{\theta}, \boldsymbol{0}\le\boldsymbol{B}\le\boldsymbol{1}}\ \mathsf{Loss}'_{\text{cc}} + \mathsf{Loss}_{\text{vol}}, \tag{13}$$

where $\mathsf{Loss}_{\text{vol}} = -\lambda\log\det(\boldsymbol{M}\boldsymbol{M}^\top)$ and $\boldsymbol{m}_n = \boldsymbol{f}_{\boldsymbol{\theta}}(\boldsymbol{x}_n)$. Note that $\log\det(\boldsymbol{M}\boldsymbol{M}^\top)$ is proportional to the volume of the Gram matrix $\boldsymbol{M}\boldsymbol{M}^\top$ (Boyd et al., 2004). Hence, the term can be regarded as a geometry-driven regularization. We name our method using (13) as the *Volume Maximization-Regularized Deep Constrained Clustering* (`VolMaxDCC`). An overall architecture is shown in Fig. 2.

5. Related Work

Recovering the underlying unseen matrix from incomplete and binary measurement is related to 1-bit low-rank matrix completion (Davenport et al., 2014), which was often studied in the context of recommender systems. The generative model in (5) is reminiscent of the MMSB (Airoldi et al., 2008; Huang & Fu, 2019) that has been a workhorse in overlapped community detection. The MMSB model with

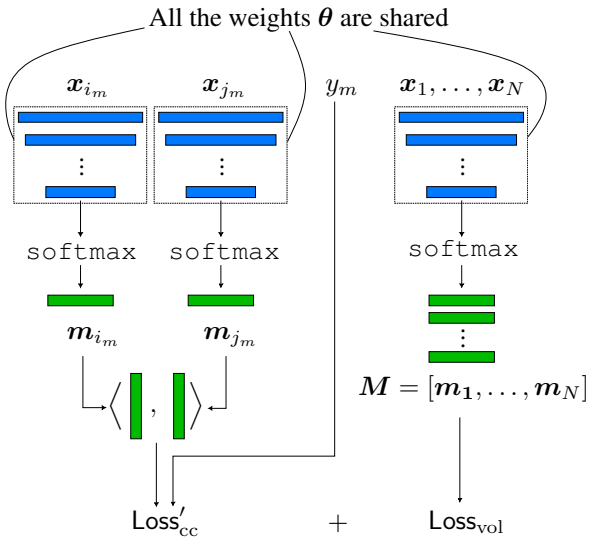


Figure 2: The architecture of the VolMaxDCC approach.

missing links was also used for clustering-related tasks, e.g., crowdclustering (Gomes et al., 2011). The ASC and SSC are commonly seen conditions in identifiability analysis of nonnegative matrix factorization (Fu et al., 2018; 2015a; 2019; 2016; Huang et al., 2014; Donoho & Stodden, 2003; Gillis, 2014; Gillis & Vavasis, 2014; Gillis & Luce, 2014; Gillis, 2020; Nguyen et al., 2022). Using volume maximization/minimization to enhance NMF identifiability appeared as early as 1994 (Craig, 1994) in the context of a blind source separation (BSS) problem in spectral image analysis. The volume-based geometric regularization was connected to ASC- and SSC-like conditions (e.g., the so-called "pure pixel condition" and "local dominance condition") in (Chan et al., 2009) and (Fu et al., 2015b; 2016; Lin et al., 2015), respectively, to attain uniqueness of matrix factorization models, again, in the context of BSS. All these models do not involve nonlinear function learning or deep neural networks. In addition, the classic geometric factorization models were developed with continuous low-rank matrix data—instead of binary data generated from models involving complex *unknown* nonlinear function. Confusion matrices are often used in supervised noisy label learning and crowdsourcing (Dawid & Skene, 1979; Zhang et al., 2014; Rodrigues & Pereira, 2018), to model probabilistic label transition in the annotating process. The ASC and SSC were also used to establish identifiability in supervised (crowdsourced) noisy label learning (Xia et al., 2019; Li et al., 2021a; Ibrahim et al., 2019; Ibrahim & Fu, 2021; Ibrahim et al., 2023). Incorporating the idea of confusion matrix-based modeling in similarity annotation was not seen before. The proposed generative model connects label confusion matrices, volume maximization (more generally, ASC/SSC-based identifiable factor analysis), and DCC for the first time, to our best knowledge.

6. Experiments

Datasets. We use STL-10 (Coates et al., 2011), ImageNet-10 (Chang et al., 2017a), and CIFAR-10 (Krizhevsky et al., 2009). For three datasets, we use $N = 10000$ samples as the seen data, and set aside 2000, 2000 and 45000 unseen data, respectively, for testing the generalization performance. In all experiments, $M = 10,000$ pairwise constraints are uniformly and randomly drawn from $[N] \times [N]$. There are no more than 0.02% of the total number of pairs annotated for all three datasets.

Baselines. We compare our method with several baselines, including the classic CC methods, i.e., PCKMeans (Basu et al., 2004a), COP-KMeans (Wagstaff et al., 2001), and the DCC methods, namely, DC-GMM (Manduchi et al., 2021) and C-IDEC (Zhang et al., 2019; 2021). We also include the plain-vanilla K-means as a reference. We use a validation set for the baselines whenever proper for parameter tuning and algorithm stopping. The sizes of the validation sets are $N_{\text{valid}} = 1000$ for STL-10 and ImageNet-10 and $N_{\text{valid}} = 5000$ for CIFAR-10. For C-IDEC, the regularization parameters is chosen from $\{0, 1e{-}1, 1e{-}2, 1e{-}3, 1e{-}4, 1e{-}5\}$. For DC-GMM, we use their heuristic to set the constraint violation penalty with the true (oracle) label flipping rate. For the proposed VolMaxDCC, we also choose λ among $\lambda \in \{0, 1e{-}1, 1e{-}2, 1e{-}3, 1e{-}4, 1e{-}5\}$. We also include the result of using the simple logistic loss Loss_{cc} in (3), which is referred to as VanillaDCC.

Neural Network Settings. For all the DCC methods, we employ the unsupervised pre-training method by (Li et al., 2021b) to convert the images to feature vectors $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\} \subseteq \mathbb{R}^{512}$ (Li et al., 2021b). The feature vectors are then fed to a two-hidden-layer fully connected neural network $\boldsymbol{f}_{\boldsymbol{\theta}}(\cdot)$, where each hidden layer has 512 ReLU activation functions. The output layer of $\boldsymbol{f}_{\boldsymbol{\theta}}(\cdot)$ has $K = 10$ dimensions with the softmax constraints. The classic methods also work with the pre-trained feature vectors.

Algorithm Implementation. To tackle the proposed criterion in (13), we first parameterize $\boldsymbol{B}$ such that each element $B_{ij} = 1/(1 + \exp(-B'_{ij}))$, where $B'_{ij} \in \mathbb{R}$ is a trainable parameter. By doing so, (13) becomes an unconstrained optimization problem. We then employ the commonly used stochastic gradient-based solvers to tackle the re-parameterized problem. In our implementation, we use stochastic gradient descent with a batch size of 128. We set the learning rate for $\boldsymbol{B}'$ and $\boldsymbol{\theta}$ to be 0.1 and 0.5, respectively. The initialization of $\boldsymbol{\theta}$ is chosen randomly following uniform distributions with parameters depending on output dimension of each layer. To initialize $\boldsymbol{B}'$, we make the

Table 1: Clustering performance of (seen data, unseen data) on STL10; $N_{\text{unseen}} = 2000$.

Noise level \ Methods		Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.71, —	0.66, —	0.70, —	0.88, 0.87	0.89, 0.88	**0.93**, **0.91**	0.91, 0.89
	NMI	0.75, —	0.67, —	0.72, —	0.82, 0.80	0.81, 0.80	**0.84**, **0.81**	0.82, 0.80
	ARI	0.54, —	0.52, —	0.55, —	0.80, 0.78	0.79, 0.77	**0.85**, **0.83**	0.84, 0.82
8.3%	ACC	—	0.70, —	0.70, —	0.75, 0.76	0.77, 0.79	0.78, 0.80	**0.80**, **0.81**
	NMI	—	0.64, —	**0.71**, —	0.67, **0.69**	0.67, **0.69**	0.59, 0.62	0.64, 0.65
	ARI	—	0.51, —	0.56, —	0.57, 0.59	0.59, 0.61	0.69, 0.71	**0.73**, **0.74**
10.3%	ACC	—	0.62, —	0.69, —	0.70, 0.72	0.70, 0.71	0.72, 0.73	**0.79**, **0.81**
	NMI	—	0.59, —	**0.73**, —	0.62, 0.64	0.60, 0.62	0.50, 0.51	0.68, **0.70**
	ARI	—	0.44, —	0.55, —	0.51, 0.52	0.50, 0.52	0.62, 0.64	**0.77**, **0.78**
15.0%	ACC	—	0.62, —	0.64, —	0.60, 0.61	0.57, 0.57	0.56, 0.58	**0.79**, **0.81**
	NMI	—	0.54, —	**0.72**, —	0.54, 0.55	0.50, 0.50	0.33, 0.35	0.68, **0.69**
	ARI	—	0.41, —	0.52, —	0.38, 0.39	0.38, 0.39	0.50, 0.51	**0.76**, **0.77**

Table 2: Clustering performance of (seen data, unseen data) on CIFAR10; $N_{\text{unseen}} = 45000$.

Noise level \ Methods		Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.78, —	0.67, —	0.67, —	0.91, 0.89	0.90, 0.89	**0.92**, **0.90**	0.91, **0.90**
	NMI	0.71, —	0.66, —	0.71, —	0.83, **0.81**	0.83, **0.81**	**0.84**, 0.80	0.83, 0.80
	ARI	0.62, —	0.54, —	0.55, —	0.82, 0.79	0.81, 0.79	**0.85**, 0.81	0.84, **0.82**
4.9%	ACC	—	0.75, —	0.70, —	**0.86**, **0.86**	**0.86**, **0.86**	**0.86**, **0.86**	**0.86**, **0.86**
	NMI	—	0.69, —	0.69, —	**0.77**, **0.77**	**0.77**, **0.77**	0.73, 0.73	0.73, 0.74
	ARI	—	0.60, —	0.57, —	0.73, 0.74	0.73, 0.74	**0.77**, **0.77**	**0.77**, **0.77**
8.7%	ACC	—	0.64, —	0.72, —	0.76, 0.76	0.76, 0.76	0.76, 0.77	**0.83**, **0.83**
	NMI	—	0.63, —	0.69, —	**0.71**, **0.71**	0.70, 0.70	0.58, 0.59	0.70, 0.70
	ARI	—	0.49, —	0.59, —	0.58, 0.59	0.59, 0.60	0.70, 0.70	**0.75**, **0.75**
10.9%	ACC	—	0.68, —	0.70, —	0.74, 0.74	0.73, 0.73	0.68, 0.69	**0.82**, **0.82**
	NMI	—	0.61, —	**0.68**, —	0.67, **0.68**	0.65, 0.66	0.48, 0.49	**0.68**, **0.68**
	ARI	—	0.50, —	0.57, —	0.55, 0.55	0.56, 0.57	0.62, 0.63	**0.74**, **0.74**

diagonal elements to be 1 and the other elements -1. The baselines are handled by their respective author-provided code for optimization.

6.1. Noisy Machine Annotations.

Noisy Annotation Settings. We first conduct experiments under noise-controlled settings. To be specific, we divide the experiments into two cases, where the annotations are accurate and noisy, respectively. In the former case, the annotations are set to 1 if the pair of data are from the same class, and set to 0 otherwise. In the latter case, to generate noisy pairwise constraints, several supervised machine annotators (i.e., classifiers) are trained using different number of training samples. By doing this, the trained classifiers have different prediction errors. The pairs are then annotated based on the class predictions made by these imperfect classifiers. If the pair of samples share the same predicted membership by the machine annotator, the annotation is set to be 1 (and 0 otherwise). The annotations are noisy as the machine annotators are far from perfect. The annotation noise level can be controlled by tuning the prediction error of the classifiers, via using various amounts of training samples. The statistics of the annotation errors are provided with our experiment results (see the "noise level" column in the tables).

Results. We report average performance on the seen set $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N\}$ and the unseen data over 5 random trials in Tables 1, 2, and 3—which correspond to STL10, CIFAR10, and ImageNet10, respectively. For K-means, COP-Kmeans, and PCKMeans, we only report results on the seen set since

Table 3: Clustering performance of (seen data, unseen data) on ImageNet10; $N_{\text{unseen}} = 2000$.

Noise level \ Methods		Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.85, —	0.79, —	0.70, —	**0.97**, **0.96**	**0.97**, **0.96**	**0.97**, **0.96**	**0.97**, **0.96**
	NMI	0.80, —	0.77, —	0.75, —	0.93, 0.91	0.93, **0.92**	**0.94**, 0.91	**0.94**, 0.91
	ARI	0.68, —	0.66, —	0.55, —	**0.94**, **0.92**	**0.94**, **0.92**	0.93, 0.91	0.93, 0.91
3.4%	ACC	—	0.79, —	0.66, —	0.93, 0.92	0.93, 0.93	0.92, 0.91	**0.94**, **0.94**
	NMI	—	0.75, —	0.73, —	0.86, 0.85	0.87, 0.86	0.83, 0.82	**0.88**, **0.87**
	ARI	—	0.65, —	0.52, —	0.84, 0.83	0.86, 0.85	0.84, 0.84	**0.89**, **0.88**
6.9%	ACC	—	0.74, —	0.72, —	0.84, 0.84	0.88, 0.88	0.84, 0.84	**0.92**, **0.91**
	NMI	—	0.70, —	0.76, —	0.79, 0.79	0.82, 0.82	0.70, 0.70	**0.84**, **0.83**
	ARI	—	0.58, —	0.59, —	0.71, 0.71	0.77, 0.77	0.77, 0.77	**0.88**, **0.87**
11.2%	ACC	—	0.72, —	0.62, —	0.71, 0.72	0.80, 0.81	0.65, 0.66	**0.91**, **0.90**
	NMI	—	0.64, —	0.73, —	0.68, 0.70	0.74, 0.76	0.49, 0.52	**0.83**, **0.82**
	ARI	—	0.54, —	0.51, —	0.56, 0.58	0.66, 0.68	0.62, 0.63	**0.87**, **0.86**

Table 4: Clustering performance of (seen data, unseen data) on ImageNet10 with pairwise annotations acquired from the AMT platform; $N_{\text{unseen}} = 2000$, noise level: 23.09%.

Metrics \ Methods	Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VolMaxDCC
ACC	0.84, —	0.68, —	0.84, —	0.87, 0.85	0.91, 0.90	**0.95**, **0.94**
NMI	0.79, —	0.49, —	0.79, —	0.88, 0.86	0.86, 0.85	**0.89**, **0.87**
ARI	0.67, —	0.42, —	0.67, —	0.82, 0.79	0.83, 0.82	**0.89**, **0.88**

these methods do not have the notion of generalization. The performance is measured by three commonly seen metrics, namely, clustering accuracy (ACC) (Cai et al., 2010), normalized mutual information (NMI) (Cai et al., 2010), and adjusted rank index (ARI) (Yeung & Ruzzo, 2001). For all metrics, a higher score indicates a better performance.

In case where accurate pairwise constraints are used (i.e., the rows in all the tables corresponding to "Noise Level = 0%"), most methods work reasonably well. As expected, all the DCC methods exhibit tangible edges over the CC methods that do not use deep neural networks. This is consistent with the observations made from previous DCC works (Manduchi et al., 2021; Zhang et al., 2019; 2021). The good performance of VanillaDCC on both training and testing set are also as expected, per our identifiability analysis.

The rows in the tables associated with nonzero noise levels show that the performance of DC-GMM, C-IDEC, and VanillaDCC drops quickly. For example, in Table 2, the ACC of DC-GMM drops from (0.91,0.89) to (0.74,0.74) when the noise level changes from 0% to 10.9%. Similar performance degradation is observed for C-IDEC and VanillaDCC, which are both DCC methods that do not explicit consider annotation noise. Nonetheless, the proposed VolMaxDCC's performance decline is much more graceful on all three datasets. In particular, Table 3 shows that the ACC of VolMaxDCC is still at (0.91,0.90) when the noise level reaches 11.2%, while the baselines have a best ACC of (0.80,0.81). The results show the usefulness of our confusion model, as well as the effectiveness of our identifiability-driven loss function design. More experiment results can be seen in Appendix F.

6.2. Noisy AMT Annotations.

Data Acquisition. In addition to using machine classifier-annotated data, we also conduct experiments using pairwise annotations that are obtained from the AMT platform. We uploaded 8994 data pairs to AMT, where the samples are from the ImageNet10 dataset. The annotators were asked to provide their judgement on the similarity of the pairs. Recall that there are $N = 10000$ samples in the ImageNet10 dataset, which means that 0.018% of all the pairs were annotated. We manually checked the error rate, which was found to be 23.09%. The annotated pairs are also released with the code.

Results. Table 4 shows the results on this AMT dataset. As before, we use the available pairs to learn the membership of training data and observe the testing accuracy over $N_{\text{unseen}} = 2,000$ samples. One can see that the proposed `VolMaxDCC` exhibits the highest clustering accuracy over the seen and unseen data. The clustering accuracy of the second best baseline is 4% lower than that of `VolMaxDCC` over both seen and unseen data. The margins of the proposed method over the baselines in terms of NMI and ARI are also obvious. The performance on the noisy AMT data speaks for the usefulness and effectiveness of the proposed method in real-world scenarios.

7. Conclusion

We revisited the pairwise annotation-based DCC problem from a membership identifiability viewpoint. We showed that a recently emerged logistic DCC loss is a sound criterion in terms of model identification—if the annotations are generated following a model that is reminiscent of the MMSB and deep learning based classifier learning. Based our understanding to the vanilla logistic loss, we moved forward to consider the noisy annotation case and proposed a confusion-matrix based generative model. We proposed a modified logistic loss with a geometric regularization for provable membership identification—whose identifiability guarantee is the first of the kind under noisy annotations-based DCC, to our best knowledge. We tested our new design over various datasets under multiple noisy levels. We observed tangible improvements over all cases, showing our confusion-based modeling and identifiability-driven design are promising.

Limitations. The proposed approach has a couple of notable limitations. First, the model used for annotator noise relies on a confusion matrix model, assuming uniform confusion across all data samples, which may not always hold true in practical scenarios. Developing a framework that takes into account more realistic confusion models could lead to further improvements in performance. Second, the establishment of membership identifiability under the SSC assumption lacks finite sample analysis (cf. Theorem 3.7 and Theorem 4.1). The assumption that M reaches infinity can never be met in practice. It is of great interest to show how the performance of the volume-based criterion scales with different sample sizes.

Acknowledgement. This work is supported in part by the National Science Foundation (NSF) under project NSF IIS-2007836.

References

Airoldi, E. M., Blei, D., Fienberg, S., and Xing, E. Mixed membership stochastic blockmodels. *Advances in Neural Information Processing Systems*, 21, 2008.

Anandkumar, A., Ge, R., Hsu, D., and Kakade, S. M. A tensor approach to learning mixed membership community models. *The Journal of Machine Learning Research*, 15 (1):2239–2312, 2014.

Arora, S., Ge, R., Halpern, Y., Mimno, D., Moitra, A., Sontag, D., Wu, Y., and Zhu, M. A practical algorithm for topic modeling with provable guarantees. In *International Conference on Machine Learning*, pp. 280–288. PMLR, 2013.

Bartlett, P. L., Foster, D. J., and Telgarsky, M. J. Spectrally-normalized margin bounds for neural networks. *Advances in Neural Information Processing Systems*, 30, 2017.

Basu, S., Banerjee, A., and Mooney, R. J. Active semi-supervision for pairwise constrained clustering. In *Proceedings of the 2004 SIAM International Conference on Data Mining*, pp. 333–344, 2004a.

Basu, S., Bilenko, M., and Mooney, R. J. A probabilistic framework for semi-supervised clustering. In *Proceedings of the Tenth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 59–68, 2004b.

Basu, S., Davidson, I., and Wagstaff, K. *Constrained clustering: Advances in algorithms, theory, and applications*. CRC Press, 2008.

Bilenko, M., Basu, S., and Mooney, R. J. Integrating constraints and metric learning in semi-supervised clustering. In *Proceedings of the Twenty-First International Conference on Machine Learning*, pp. 11, 2004.

Blei, D. M., Ng, A. Y., and Jordan, M. I. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3 (Jan):993–1022, 2003.

Boyd, S., Boyd, S. P., and Vandenberghe, L. *Convex optimization*. Cambridge university press, 2004.

Cai, D., He, X., and Han, J. Locally consistent concept factorization for document clustering. *IEEE Transactions on Knowledge and Data Engineering*, 23(6):902–913, 2010.

Cai, T. T. and Zhou, W.-X. Matrix completion via max-norm constrained optimization. *Electronic Journal of Statistics*, 10(1):1493 – 1525, 2016.

Caron, M., Bojanowski, P., Joulin, A., and Douze, M. Deep clustering for unsupervised learning of visual features. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pp. 132–149, 2018.

Chan, T.-H., Chi, C.-Y., Huang, Y.-M., and Ma, W.-K. A convex analysis-based minimum-volume enclosing simplex algorithm for hyperspectral unmixing. *IEEE Transactions on Signal Processing*, 57(11):4418–4432, 2009.

Chang, J., Wang, L., Meng, G., Xiang, S., and Pan, C. Deep adaptive image clustering. In *Proceedings of the IEEE International Conference on Computer Vision*, pp. 5879–5887, 2017a.

Chang, Y., Chen, J., Cho, M. H., Castaldi, P. J., Silverman, E. K., and Dy, J. G. Multiple clustering views from multiple uncertain experts. In *International Conference on Machine Learning*, pp. 674–683, 2017b.

Chu, Z., Ma, J., and Wang, H. Learning from crowds by modeling common confusions. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pp. 5832–5840, 2021.

Coates, A., Ng, A., and Lee, H. An analysis of single-layer networks in unsupervised feature learning. In *Proceedings of the Fourteenth International Conference on Artificial Intelligence and Statistics*, volume 15, pp. 215–223. PMLR, 11–13 Apr 2011.

Covoes, T. F., Hruschka, E. R., and Ghosh, J. A study of K-means-based algorithms for constrained clustering. *Intelligent Data Analysis*, 17(3):485–505, 2013.

Craig, M. D. Minimum-volume transforms for remotely sensed data. *IEEE Transactions on Geoscience and Remote Sensing*, 32(3):542–552, 1994.

Cucuringu, M., Koutis, I., Chawla, S., Miller, G., and Peng, R. Simple and scalable constrained clustering: A generalized spectral method. In *Artificial Intelligence and Statistics*, pp. 445–454, 2016.

Davenport, M. A., Plan, Y., van den Berg, E., and Wootters, M. 1-bit matrix completion. *Information and Inference: A Journal of the IMA*, 3(3):189–223, 2014.

Davidson, I., Ravi, S., and Shamis, L. A SAT-based framework for efficient constrained clustering. In *Proceedings of the 2010 SIAM International Conference on Data Mining*, pp. 94–105. SIAM, 2010.

Dawid, A. P. and Skene, A. M. Maximum likelihood estimation of observer error-rates using the EM algorithm. *Journal of the Royal Statistical Society: Series C (Applied Statistics)*, 28(1):20–28, 1979.

Donoho, D. and Stodden, V. When does non-negative matrix factorization give a correct decomposition into parts? In *Advances in Neural Information Processing Systems*, volume 16, pp. 1141–1148, 2003.

Du, S. S., Zhai, X., Poczos, B., and Singh, A. Gradient descent provably optimizes over-parameterized neural networks. In *International Conference on Learning Representations*, 2019.

Fu, X., Ma, W.-K., Chan, T.-H., and Bioucas-Dias, J. M. Self-dictionary sparse regression for hyperspectral unmixing: Greedy pursuit and pure pixel search are related. *IEEE Journal of Selected Topics in Signal Processing*, 9 (6):1128–1141, 2015a.

Fu, X., Ma, W.-K., Huang, K., and Sidiropoulos, N. D. Blind separation of quasi-stationary sources: Exploiting convex geometry in covariance domain. *IEEE Transactions on Signal Processing*, 63:2306–2320, 2015b.

Fu, X., Huang, K., Yang, B., Ma, W.-K., and Sidiropoulos, N. D. Robust volume minimization-based matrix factorization for remote sensing and document clustering. *IEEE Transactions on Signal Processing*, 64(23):6254–6268, 2016.

Fu, X., Huang, K., and Sidiropoulos, N. D. On identifiability of nonnegative matrix factorization. *IEEE Signal Processing Letters*, 25(3):328–332, 2018.

Fu, X., Huang, K., Sidiropoulos, N. D., and Ma, W.-K. Nonnegative matrix factorization for signal and data analytics: Identifiability, Algorithms, and Applications. *IEEE Signal Process. Mag.*, 36(2):59–80, March 2019.

Ghosh, A., Kale, S., and McAfee, P. Who moderates the moderators? crowdsourcing abuse detection in user-generated content. In *Proceedings of the 12th ACM Conference on Electronic Commerce*, pp. 167–176, 2011.

Gillis, N. The why and how of nonnegative matrix factorization. In *Regularization, Optimization, Kernels, and Support Vector Machines*, pp. 275–310. Chapman and Hall/CRC, 2014.

Gillis, N. *Nonnegative matrix factorization*. SIAM, 2020.

Gillis, N. and Luce, R. Robust near-separable nonnegative matrix factorization using linear optimization. *The Journal of Machine Learning Research*, 15(1):1249–1280, 2014.

Gillis, N. and Vavasis, S. A. Fast and robust recursive algorithmsfor separable nonnegative matrix factorization. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(4):698–714, 2014.

Givoni, I. and Frey, B. Semi-supervised affinity propagation with instance-level constraints. In *Artificial Intelligence and Statistics*, pp. 161–168. PMLR, 2009.

Gomes, R., Welinder, P., Krause, A., and Perona, P. Crowdclustering. *Advances in Neural Information Processing Systems*, 24, 2011.

Hsu, Y.-C. and Kira, Z. Neural network-based clustering using pairwise constraints. *arXiv preprint arXiv:1511.06321*, 2015.

Hsu, Y.-C., Lv, Z., Schlosser, J., Odom, P., and Kira, Z. A probabilistic constrained clustering for transfer learning and image category discovery. *CVPR Deep-Vision workshop*, 2018.

Huang, K. and Fu, X. Detecting overlapping and correlated communities without pure nodes: Identifiability and algorithm. In *Proceedings of the 36th International Conference on Machine Learning*, volume 97, pp. 2859–2868. PMLR, 09–15 Jun 2019.

Huang, K., Sidiropoulos, N. D., and Swami, A. Nonnegative matrix factorization revisited: Uniqueness and algorithm for symmetric decomposition. *IEEE Transactions on Signal Processing*, 62:211–224, 2014.

Huang, K., Fu, X., and Sidiropoulos, N. D. Anchor-free correlated topic modeling: Identifiability and algorithm. *Advances in Neural Information Processing Systems*, 29, 2016.

Ibrahim, S. and Fu, X. Crowdsourcing via annotator cooccurrence imputation and provable symmetric nonnegative matrix factorization. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139, pp. 4544–4554. PMLR, 18–24 Jul 2021.

Ibrahim, S., Fu, X., Kargas, N., and Huang, K. Crowdsourcing via pairwise co-occurrences: Identifiability and algorithms. *Advances in Neural Information Processing Systems*, 32, 2019.

Ibrahim, S., Nguyen, T., and Fu, X. Deep learning from crowdsourced labels: Coupled cross-entropy minimization, identifiability, and regularization. In *The Eleventh International Conference on Learning Representations*, 2023.

Jain, A. K. and Dubes, R. C. *Algorithms for Clustering Data*. Prentice-Hall, Inc., USA, 1988. ISBN 013022278X.

Kamvar, K., Sepandar, S., Klein, K., Dan, D., Manning, M., and Christopher, C. Spectral learning. In *International Joint Conference of Artificial Intelligence*. Stanford InfoLab, 2003.

Krizhevsky, A., Hinton, G., et al. Learning multiple layers of features from tiny images. 2009.

Kulis, B., Basu, S., Dhillon, I., and Mooney, R. Semi-supervised graph clustering: A kernel approach. In *Proceedings of the 22nd International Conference on Machine Learning*, pp. 457–464, 2005.

Kumar, A., Sindhwani, V., and Kambadur, P. Fast conical hull algorithms for near-separable non-negative matrix factorization. In *International Conference on Machine Learning*, pp. 231–239, 2013.

Law, M. H. C., Topchy, A., and Jain, A. K. Model-based clustering with probabilistic constraints. In *Proceedings of the 2005 SIAM International Conference on Data Mining (SDM)*, pp. 641–645, 2005.

Li, X., Liu, T., Han, B., Niu, G., and Sugiyama, M. Provably end-to-end label-noise learning without anchor points. In *International Conference on Machine Learning*, pp. 6403–6413. PMLR, 2021a.

Li, Y., Hu, P., Liu, Z., Peng, D., Zhou, J. T., and Peng, X. Contrastive clustering. In *2021 AAAI Conference on Artificial Intelligence (AAAI)*, 2021b.

Li, Z., Liu, J., and Tang, X. Constrained clustering via spectral regularization. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 421–428, 2009.

Lin, C.-H., Ma, W.-K., Li, W.-C., Chi, C.-Y., and Ambikapathi, A. Identifiability of the simplex volume minimization criterion for blind hyperspectral unmixing: The no-pure-pixel case. *IEEE Transactions on Geoscience and Remote Sensing*, 53(10):5530–5546, 2015.

Liu, H., Tao, Z., and Fu, Y. Partition level constrained clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(10):2469–2483, 2017.

Liu, Q., Peng, J., and Ihler, A. T. Variational inference for crowdsourcing. *Advances in Neural Information Processing Systems*, 25, 2012.

Lu, Z. and Carreira-Perpinan, M. A. Constrained spectral clustering through affinity propagation. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pp. 1–8, 2008.

Luo, Y., Tian, T., Shi, J., Zhu, J., and Zhang, B. Semi-crowdsourced clustering with deep generative models. *Advances in Neural Information Processing Systems*, 31, 2018.

Manduchi, L., Chin-Cheong, K., Michel, H., Wellmann, S., and Vogt, J. Deep conditional gaussian mixture model for constrained clustering. *Advances in Neural Information Processing Systems*, 34:11303–11314, 2021.

Mao, X., Sarkar, P., and Chakrabarti, D. On mixed memberships and symmetric nonnegative matrix factorizations. In *International Conference on Machine Learning*, pp. 2324–2333, 2017.

Nguyen, T., Fu, X., and Wu, R. Memory-efficient convex optimization for self-dictionary separable nonnegative matrix factorization: A frank–wolfe approach. *IEEE Transactions on Signal Processing*, 70:3221–3236, 2022.

Panov, M., Slavnov, K., and Ushakov, R. Consistent estimation of mixed memberships with successive projections. *International Workshop on Complex Networks and their Applications*, pp. 53–64, 2017.

Pelleg, D. and Baras, D. K-means with large and noisy constraint sets. In *European Conference on Machine Learning*, pp. 674–682. Springer, 2007.

Ren, Y., Hu, K., Dai, X., Pan, L., Hoi, S. C., and Xu, Z. Semi-supervised deep embedded clustering. *Neurocomputing*, 325:121–130, 2019.

Rodrigues, F. and Pereira, F. Deep learning from crowds. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 32, 2018.

Schultz, M. and Joachims, T. Learning a distance metric from relative comparisons. *Advances in Neural Information Processing Systems*, 16, 2003.

Segal, E., Wang, H., and Koller, D. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, 19(suppl_1):i264–i272, 2003.

Shalev-Shwartz, S. and Ben-David, S. *Understanding machine learning: From theory to algorithms*. Cambridge university press, 2014.

Wagstaff, K. and Cardie, C. Clustering with instance-level constraints. *AAAI/IAAI*, 1097:577–584, 2000.

Wagstaff, K., Cardie, C., Rogers, S., Schrödl, S., et al. Constrained K-means clustering with background knowledge. In *International Conference on Machine Learning*, volume 1, pp. 577–584, 2001.

Wang, X., Qian, B., and Davidson, I. On constrained spectral clustering and its applications. *Data Mining and Knowledge Discovery*, 28(1):1–30, 2014.

Weyl, H. Das asymptotische verteilungsgesetz der eigenwerte linearer partieller differentialgleichungen (mit einer anwendung auf die theorie der hohlraumstrahlung). *Mathematische Annalen*, 71(4):441–479, 1912.

Xia, X., Liu, T., Wang, N., Han, B., Gong, C., Niu, G., and Sugiyama, M. Are anchor points really indispensable in label-noise learning? *Advances in Neural Information Processing Systems*, 32, 2019.

Yang, B., Fu, X., Sidiropoulos, N. D., and Hong, M. Towards K-means-friendly spaces: Simultaneous deep learning and clustering. In *Proceedings of the 34th International Conference on Machine Learning*, volume 70, pp. 3861–3870. PMLR, 06–11 Aug 2017.

Yeung, K. Y. and Ruzzo, W. L. Details of the adjusted rand index and clustering algorithms, supplement to the paper an empirical study on principal component analysis for clustering gene expression data. *Bioinformatics*, 17(9): 763–774, 2001.

Yeung, M. M. and Yeo, B.-L. Time-constrained clustering for segmentation of video into story units. In *Proceedings of 13th International Conference on Pattern Recognition*, volume 3, pp. 375–380. IEEE, 1996.

Yi, J., Jin, R., Jain, S., Yang, T., and Jain, A. Semi-crowdsourced clustering: Generalizing crowd labeling by robust distance metric learning. *Advances in Neural Information Processing Systems*, 25, 2012.

Zhang, H., Basu, S., and Davidson, I. A framework for deep constrained clustering-algorithms and advances. In *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*, pp. 57–72, 2019.

Zhang, H., Zhan, T., Basu, S., and Davidson, I. A framework for deep constrained clustering. *Data Mining and Knowledge Discovery*, 35(2):593–620, 2021.

Zhang, Y., Chen, X., Zhou, D., and Jordan, M. I. Spectral methods meet EM: A provably optimal algorithm for crowdsourcing. *Advances in Neural Information Processing Systems*, 27, 2014.

Zhu, X., Loy, C. C., and Gong, S. Constrained clustering with imperfect oracles. *IEEE Transactions on Neural Networks and Learning Systems*, 27(6):1345–1357, 2015.

A. Proof of Lemma 3.3

Denote (I, J) and Y as two random variables (RVs) such that $\mathsf{Pr}\,(I, J, Y) = \mathsf{Pr}\,((I, J))\,\mathsf{Pr}\,(Y|(I, J))$ and

$$\begin{cases} I, J \sim \mathcal{U}([N] \times [N]), \\ Y|(I, J) \sim \mathsf{Bernoulli}(P^\natural_{IJ}), \end{cases}$$

where $(i_1, j_1, y_1), \ldots, (i_M, j_M, y_M)$ are M i.i.d realizations of (I, J, Y). Denote $\mathcal{D}_{\boldsymbol{P}^\natural}$ as the joint distribution of (I, J, Y), i.e, $(I, J, Y) \sim \mathcal{D}_{\boldsymbol{P}^\natural}$. We first note the following relationship, which was also used in 1-bit matrix completion literature (Davenport et al., 2014; Cai & Zhou, 2016):

$$\begin{aligned}
D_{\mathsf{kl}}(\boldsymbol{P}^\natural \parallel \boldsymbol{P}^\star) &= \frac{1}{N^2} \sum_{(i,j)\in[N]^2} d_{\mathsf{kl}}(P^\natural_{ij} \parallel P^\star_{ij}) \quad \text{(by definition in (28))} \\
&= \mathop{\mathbb{E}}_{(I,J)\sim\mathcal{U}} \left[d_{\mathsf{kl}}(P^\natural_{IJ} \parallel P^\star_{IJ}) \right] \\
&= \mathop{\mathbb{E}}_{(I,J)\sim\mathcal{U}} \left[P^\natural_{IJ} \log \frac{P^\natural_{IJ}}{P^\star_{IJ}} + (1 - P^\natural_{IJ}) \log \frac{1 - P^\natural_{IJ}}{1 - P^\star_{IJ}} \right] \quad \text{(by definition in (29))} \\
&= \mathop{\mathbb{E}}_{(I,J)\sim\mathcal{U}} \left[\mathop{\mathbb{E}}_{Y|(I,J)\sim\mathrm{Bern}(P^\natural_{IJ})} \left[Y \log \frac{P^\natural_{IJ}}{P^\star_{IJ}} + (1 - Y) \log \frac{1 - P^\natural_{IJ}}{1 - P^\star_{IJ}} \right] \right] \\
&= \mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}} \left[Y \log P^\natural_{IJ} + (1 - Y) \log P^\natural_{IJ} \right] - \mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}} \left[Y \log P^\star_{IJ} + (1 - Y) \log P^\star_{IJ} \right] \\
&= L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\star) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\natural),
\end{aligned} \tag{14}$$

where we define $L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P})$ for any matrix $0 \le \boldsymbol{P} \le 1$ as

$$\begin{aligned}
L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}) &\triangleq \mathbb{E}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}} \left[\ell(\boldsymbol{P}, (I, J, Y)) \right], \text{ where} \\
\ell(\boldsymbol{P}, (I, J, Y)) &\triangleq -Y \log P_{IJ} - (1 - Y) \log(1 - P_{IJ}).
\end{aligned} \tag{15}$$

Recall $S \triangleq \{(i_1, j_1, y_1), \ldots, (i_M, j_M, y_M)\}$. Define $L_S(\boldsymbol{P}) \triangleq (1/M) \sum_{i=1}^M \ell(\boldsymbol{P}, (i_m, j_m, y_m))$. By this definition, $\boldsymbol{P}^\star$ is an optimal solution of the following problem

$$\mathop{\mathsf{minimize}}_{\boldsymbol{P}\in\mathcal{P}_{\mathcal{F},\boldsymbol{X}}} \quad L_S(\boldsymbol{P}),$$

where $\mathcal{P}_{\mathcal{F},\boldsymbol{X}}$ is a set of matrices defined by the function class $\mathcal{F}$ and the data set $\boldsymbol{X} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N]$. Specifically, it is defined as

$$\mathcal{P}_{\mathcal{F},\boldsymbol{X}} \triangleq \left\{ \boldsymbol{P} \in \mathbb{R}^{N\times N} \mid P_{ij} = \mathtt{dot}(\boldsymbol{f}'(\boldsymbol{x}_i, \boldsymbol{x}_j)) \right\}, \tag{16a}$$

$$\mathtt{dot}([\boldsymbol{u}; \boldsymbol{v}]) \triangleq \boldsymbol{u}^\top \boldsymbol{v}, \tag{16b}$$

$$\boldsymbol{f}'(\boldsymbol{x}, \boldsymbol{y}) \triangleq [\boldsymbol{f}(\boldsymbol{x}); \boldsymbol{f}(\boldsymbol{y})], \quad \boldsymbol{f} \in \mathcal{F}. \tag{16c}$$

Define $\widetilde{\boldsymbol{P}} \in \mathbb{R}^{N\times N}$ such that $\widetilde{P}_{ij} = \widetilde{\boldsymbol{f}}(\boldsymbol{x}_i)^\top \widetilde{\boldsymbol{f}}(\boldsymbol{x}_j)$ where $\widetilde{\boldsymbol{f}}$ is defined in Assumption 7, i.e., the "best approximation" for $\boldsymbol{f}^\natural$ by the class $\mathcal{F}$. Using (14) and the above notations, we have

$$\begin{aligned}
D_{\mathsf{kl}}(\boldsymbol{P}^\natural \parallel \boldsymbol{P}^\star) &= L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\star) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\natural) \\
&= \left(L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\star) - L_S(\boldsymbol{P}^\star) \right) + \left(L_S(\boldsymbol{P}^\star) - L_S(\widetilde{\boldsymbol{P}}) \right) + \left(L_S(\widetilde{\boldsymbol{P}}) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\widetilde{\boldsymbol{P}}) \right) + \left(L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\widetilde{\boldsymbol{P}}) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\natural) \right) \\
&\le \left(L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\star) - L_S(\boldsymbol{P}^\star) \right) + \left(L_S(\widetilde{\boldsymbol{P}}) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\widetilde{\boldsymbol{P}}) \right) + \left(L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\widetilde{\boldsymbol{P}}) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\natural) \right) \\
&\le 2 \sup_{\boldsymbol{P}\in\mathcal{P}_{\mathcal{F},\boldsymbol{X}}} \left| L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}) - L_S(\boldsymbol{P}) \right| + \left| L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\widetilde{\boldsymbol{P}}) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\natural) \right|,
\end{aligned} \tag{17}$$

where the first inequality holds because $L_S(\boldsymbol{P}^\star) \le L_S(\widetilde{\boldsymbol{P}})$ which holds by the definition of $\boldsymbol{P}^\star$, and the second inequality holds as both $\boldsymbol{P}^\star, \widetilde{\boldsymbol{P}} \in \mathcal{P}_{\mathcal{F},\boldsymbol{X}}$.

Bound the first term in (17). Notice that $L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P})$ and $L_S(\boldsymbol{P})$ are the expected loss and the empirical loss, respectively. Hence, the difference between these two quantities can be bounded using generalization analysis via concentration inequalities.

To this end, we first determine an upper bound on the loss function $\ell(\boldsymbol{P},(i,j,y))$. Under the assumption stated in Assumption 3.2,

$$\alpha < P_{ij} < 1-\alpha, \quad \forall (i,j) \in [N]\times[N],$$

which leads to

$$\ell(\boldsymbol{P},(i,j,y)) = y\log\frac{1}{P_{ij}} + (1-y)\log\frac{1}{1-P_{ij}} \le \log\frac{1}{P_{ij}} + \log\frac{1}{1-P_{ij}} \le 2\log\frac{1}{\alpha}, \quad \forall (i,j)\in[N]^2, y\in\{0,1\}, \tag{18}$$

As $\ell(\boldsymbol{P},(i,j,y))$ is bounded, the sample set S contains M samples of (i_m,j_m,y_m)'s, where each sample (i_m,j_m,y_m) is generated independently with the same distribution $\mathcal{D}_{\boldsymbol{P}^\natural}$, one can readily apply the following generalization bound (Shalev-Shwartz & Ben-David, 2014, Theorem 26.5):

$$\begin{aligned}\sup_{\boldsymbol{P}\in\mathcal{P}_{\mathcal{F},\boldsymbol{X}}} \left|L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}) - L_S(\boldsymbol{P})\right| &\le 2\mathcal{R}(\ell\circ\mathcal{P}_{\mathcal{F},\boldsymbol{X}}\circ S) + 8\log(1/\alpha)\sqrt{\frac{2\log(4/\delta)}{M}} \\ &\le \frac{8\log(1/\alpha)}{M} + \frac{12\sqrt{2}\log M}{\alpha M\log 2}\left\|\boldsymbol{S}_{\boldsymbol{X}}\right\|_{\mathrm{F}}\sqrt{R_{\mathrm{NET}}} + 8\log(1/\alpha)\sqrt{\frac{2\log(4/\delta)}{M}} \end{aligned} \tag{19}$$

holds with probability at least $1-\delta$, where we have applied Lemma A.3 to reach the final bound.

Bound the second term in (17). We have

$$\begin{aligned}\left|L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\widetilde{\boldsymbol{P}}) - L_{\mathcal{D}_{\boldsymbol{P}^\natural}}(\boldsymbol{P}^\natural)\right| &= \left|\mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[\ell(\widetilde{\boldsymbol{P}},(I,J,Y)) - \ell(\boldsymbol{P}^\natural,(I,J,Y))\right]\right| \\ &= \left|\mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[Y\left(\log P^\natural_{IJ} - \log\widetilde{P}_{IJ}\right) + (1-Y)\left(\log(1-P^\natural_{IJ}) - \log(1-\widetilde{P}_{IJ})\right)\right]\right| \\ &\le \mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[\left|Y\left(\log P^\natural_{IJ} - \log\widetilde{P}_{IJ}\right) + (1-Y)\left(\log(1-P^\natural_{IJ}) - \log(1-\widetilde{P}_{IJ})\right)\right|\right] \\ &\le \mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[Y\left|\log\frac{P^\natural_{IJ}}{\widetilde{P}_{IJ}}\right| + (1-Y)\left|\log\frac{1-P^\natural_{IJ}}{1-\widetilde{P}_{IJ}}\right|\right] \\ &\le \mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[\left|\log\frac{P^\natural_{IJ}}{\widetilde{P}_{IJ}}\right| + \left|\log\frac{1-P^\natural_{IJ}}{1-\widetilde{P}_{IJ}}\right|\right] \quad (\text{since } Y\in\{0,1\}) \\ &\le^{(a)} \mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[\left|\frac{P^\natural_{IJ}-\widetilde{P}_{IJ}}{\widetilde{P}_{IJ}}\right| + \left|\frac{\widetilde{P}_{IJ}-P^\natural_{IJ}}{1-\widetilde{P}_{IJ}}\right|\right] \\ &\le \frac{1}{\alpha}\mathop{\mathbb{E}}_{(I,J,Y)\sim\mathcal{D}_{\boldsymbol{P}^\natural}}\left[\left|P^\natural_{IJ}-\widetilde{P}_{IJ}\right|\right], \end{aligned} \tag{20}$$

where $^{(a)}$ holds by applying inequality $\log(x+1)\le x,\ \forall x > -1$.

In order to bound (20), consider arbitrary $\boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}$,

$$\begin{aligned}
&\left|\boldsymbol{f}^\natural(\boldsymbol{x})^\top \boldsymbol{f}^\natural(\boldsymbol{y}) - \widetilde{\boldsymbol{f}}(\boldsymbol{x})^\top \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right| \\
&\quad= \left|\left(\boldsymbol{f}^\natural(\boldsymbol{x}) - \widetilde{\boldsymbol{f}}(\boldsymbol{x}) + \widetilde{\boldsymbol{f}}(\boldsymbol{x})\right)^\top \left(\boldsymbol{f}^\natural(\boldsymbol{y}) - \widetilde{\boldsymbol{f}}(\boldsymbol{y}) + \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right) - \widetilde{\boldsymbol{f}}(\boldsymbol{x})^\top \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right| \\
&\quad= \left|\left(\boldsymbol{f}^\natural(\boldsymbol{x}) - \widetilde{\boldsymbol{f}}(\boldsymbol{x})\right)^\top \left(\boldsymbol{f}^\natural(\boldsymbol{y}) - \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right) + \left(\boldsymbol{f}^\natural(\boldsymbol{x}) - \widetilde{\boldsymbol{f}}(\boldsymbol{x})\right)^\top \widetilde{\boldsymbol{f}}(\boldsymbol{y}) + \widetilde{\boldsymbol{f}}(\boldsymbol{x})^\top \left(\boldsymbol{f}^\natural(\boldsymbol{y}) - \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right)\right| \\
&\quad\le \left\|\boldsymbol{f}^\natural(\boldsymbol{x}) - \widetilde{\boldsymbol{f}}(\boldsymbol{x})\right\| \left\|\boldsymbol{f}^\natural(\boldsymbol{y}) - \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right\| + \left\|\boldsymbol{f}^\natural(\boldsymbol{x}) - \widetilde{\boldsymbol{f}}(\boldsymbol{x})\right\| \left\|\widetilde{\boldsymbol{f}}(\boldsymbol{y})\right\| + \left\|\widetilde{\boldsymbol{f}}(\boldsymbol{x})\right\| \left\|\boldsymbol{f}^\natural(\boldsymbol{y}) - \widetilde{\boldsymbol{f}}(\boldsymbol{y})\right\| \\
&\quad\le \nu^2 + 2\nu \quad \text{(by Eq. (7))} \\
&\quad< 4\nu, \quad \text{(since } \nu \le \sqrt{2} \text{ due to the fact that all } \boldsymbol{f}(\boldsymbol{x}) \text{ are in the probability simplex)}
\end{aligned} \tag{21}$$

where we have used $\|\widetilde{\boldsymbol{f}}(\boldsymbol{x})\|_2 \le \|\widetilde{\boldsymbol{f}}(\boldsymbol{x})\|_1 = 1$ in the second inequality, as the neural network uses a softmax output layer. Substituting (21) into (20), we have

$$L_S(\widetilde{\boldsymbol{P}}) - L_S(\boldsymbol{P}^\natural) \le \frac{4\nu}{\alpha}. \tag{22}$$

Putting together. Combining (17), (19), (22) gives

$$D_{\mathsf{kl}}(\boldsymbol{P}^\natural \parallel \boldsymbol{P}^\star) \le \frac{16\log(1/\alpha)}{M} + \frac{24\sqrt{2}\log M}{\alpha M \log 2} \|\boldsymbol{S}_{\boldsymbol{X}}\|_{\mathrm{F}} \sqrt{R_{\text{NET}}} + 16\log(1/\alpha)\sqrt{\frac{2\log(4/\delta)}{M}} + \frac{4\nu}{\alpha}$$

By Lemma A.4, we can upper bound our quantity of interest by the following:

$$\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2 \le 4N^2 D_{\mathsf{kl}}(\boldsymbol{P}^\natural \parallel \boldsymbol{P}^\star). \tag{23}$$

That is,

$$\frac{1}{N^2}\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2 \le \frac{64\log(1/\alpha)}{M} + \frac{96\sqrt{2}\log M}{\alpha M \log 2} \|\boldsymbol{S}_{\boldsymbol{X}}\|_{\mathrm{F}} \sqrt{R_{\text{NET}}} + 64\log(1/\alpha)\sqrt{\frac{2\log(4/\delta)}{M}} + \frac{16\nu}{\alpha}$$

hold with probability at least $1-\delta$ over $S = \{(i_1, j_1, y_1) \ldots, (i_M, j_M, y_M)\}$.

A.1. Supporting lemmas for the proof of Theorem 3.3

A.1.1. Rademacher complexity bound for $\mathcal{P}_{\mathcal{F},\boldsymbol{X}}$

Definition A.1 (Rademacher complexity)**.** Let $A \subset R^m$ be a set of vectors. We define Rademacher complexity of A as

$$R(A) \triangleq \frac{1}{m}\mathbb{E}_{\boldsymbol{\sigma}}\left[\sup_{\boldsymbol{a}\in A}\sum_{i=1}^{m}\sigma_i a_i\right],$$

where $\sigma_1, \ldots, \sigma_m \in \{-1, 1\}$ are i.i.d. distributed according to $\mathsf{Pr}(\sigma_i = 1) = \mathsf{Pr}(\sigma_i = -1) = 0.5$.

Definition A.2 (Covering number)**.** Let $A \subset R^m$ be a set of vectors. We say that A is r-covered by a set A', with respect to metric d if for all $\boldsymbol{a} \in A$, there exists $\boldsymbol{a}' \in A'$ with $d(a, a') \le r$. We define by $\mathcal{N}(r, A, d)$ the cardinality of the smallest A' that r-covers A.

Lemma A.3 (Rademacher complexity bound)**.** *Consider function class $\mathcal{F}$ satisfying Assumptions 3.2 , and that $\alpha < \boldsymbol{f}(\boldsymbol{x})^\top \boldsymbol{f}(\boldsymbol{y}) < 1-\alpha,\ \forall \boldsymbol{x}, \boldsymbol{y} \in \mathcal{X}, \forall \boldsymbol{f} \in \mathcal{F}$. Then, we have*

$$\mathcal{R}(\ell \circ \mathcal{P}_{\mathcal{F},\boldsymbol{X}} \circ S) \le \frac{4\log(1/\alpha)}{M} + \frac{6\sqrt{2}\log M}{\alpha M \log 2} \|\boldsymbol{S}_{\boldsymbol{X}}\|_{\mathrm{F}} \sqrt{R_{NET}},$$

where $\ell \circ \mathcal{P}_{\mathcal{F},\boldsymbol{X}} \circ S \triangleq \left\{[\ell(\boldsymbol{P}, (i_1, j_1, y_1)), \ldots, \ell(\boldsymbol{P}, (i_M, j_M, y_M))] \in \mathbb{R}^M \mid \boldsymbol{P} \in \mathcal{P}_{\mathcal{F},\boldsymbol{X}}\right\}$, $\mathcal{R}(A)$ denotes Rademacher complexity of set $A \subseteq \mathbb{R}^M$, $\mathcal{P}_{\mathcal{F},\boldsymbol{X}}$ was defined in (16), $S = \{(i_1, j_1, y_1), \ldots, (i_M, j_M, y_M)\}$, and the loss function ℓ was defined in (15).

Proof. We start with deriving the covering number of an ϵ-net of the set $\ell \circ \mathcal{P}_{\boldsymbol{X}} \circ S$:

$$
\begin{aligned}
\mathcal{N}(\epsilon, \ell \circ \mathcal{P}_{\boldsymbol{X}} \circ S, \|\cdot\|) &= \mathcal{N}(\epsilon, \ell \circ \texttt{dot} \circ \boldsymbol{f}' \circ S, \|\cdot\|) \\
&\leq \mathcal{N}(\epsilon/L_\ell, \texttt{dot} \circ \boldsymbol{f}' \circ S, \|\cdot\|) && (24) \\
&\leq \mathcal{N}\big(\epsilon/(2L_\ell), \boldsymbol{f}' \circ S, \|\cdot\|_{\mathrm{F}}\big) && (25) \\
&\leq \mathcal{N}\big(\epsilon/(L_\ell\sqrt{2}), \boldsymbol{f} \circ S^{(1)}_{\boldsymbol{X}}, \|\cdot\|_{\mathrm{F}}\big)\mathcal{N}\big(\epsilon/(L_\ell\sqrt{2}), \boldsymbol{f} \circ S^{(2)}_{\boldsymbol{X}}, \|\cdot\|_{\mathrm{F}}\big) && (26) \\
&= \mathcal{N}\big(\epsilon/(L_\ell\sqrt{2}), \texttt{softmax} \circ \texttt{net} \circ S^{(1)}_{\boldsymbol{X}}, \|\cdot\|_{\mathrm{F}}\big)\mathcal{N}\big(\epsilon/(L_\ell\sqrt{2}), \texttt{softmax} \circ \texttt{net} \circ S^{(2)}_{\boldsymbol{X}}, \|\cdot\|_{\mathrm{F}}\big) \\
&\leq \mathcal{N}\big(\epsilon/(L_\ell\sqrt{2}), \texttt{net} \circ S^{(1)}_{\boldsymbol{X}}, \|\cdot\|_{\mathrm{F}}\big)\mathcal{N}\big(\epsilon/(L_\ell\sqrt{2}), \texttt{net} \circ S^{(2)}_{\boldsymbol{X}}, \|\cdot\|_{\mathrm{F}}\big) \\
&\leq \exp\left(\frac{\left\|\boldsymbol{S}^{(1)}_{\boldsymbol{X}}\right\|^2_{\mathrm{F}}}{\epsilon^2/(2L_\ell^2)} R_{\texttt{NET}} + \frac{\left\|\boldsymbol{S}^{(2)}_{\boldsymbol{X}}\right\|^2_{\mathrm{F}}}{\epsilon^2/(2L_\ell^2)} R_{\texttt{NET}}\right) \quad \text{(By Lemma A.10)} \\
&= \exp\left(\left(\left\|\boldsymbol{S}^{(1)}_{\boldsymbol{X}}\right\|^2_{\mathrm{F}} + \left\|\boldsymbol{S}^{(2)}_{\boldsymbol{X}}\right\|^2_{\mathrm{F}}\right)\frac{2R_{\texttt{NET}}L_\ell^2}{\epsilon^2}\right) && (27)
\end{aligned}
$$

Eq. (24) holds by applying Lemma A.7 and the fact that function $\phi_y(P_W) \triangleq -y\log(P_W) - (1-y)\log(1-P_W)$ is L_ℓ-Lipschitz continuous. Specifically, $L_\ell = (1/\alpha)$ since for $\alpha \leq p_1, p_2 \leq 1-\alpha$,

$$
\begin{aligned}
|\phi_y(p_1) - \phi_y(p_2)| &= \left|-y\log\frac{p_1}{p_2} - (1-y)\log\frac{1-p_1}{1-p_2}\right| \\
&\leq \left|\log\frac{p_1}{p_2}\right| + \left|\log\frac{1-p_1}{1-p_2}\right| \\
&\leq \left|\frac{p_1-p_2}{p_2}\right| + \left|\frac{p_2-p_1}{1-p_2}\right| \quad (\text{ by } \log(1+x) \leq x) \\
&\leq \frac{1}{\alpha}\left|p_1 - p_2\right| = L_\ell\left|p_1 - p_2\right|.
\end{aligned}
$$

Similarly, Eq. (25) holds by Lemma A.7 and the fact that for $\forall \boldsymbol{u}, \boldsymbol{v} \in \left\{\boldsymbol{x} \mid \boldsymbol{x} \geq 0, \mathbf{1}^\top\boldsymbol{x} = 1\right\}$, function $\texttt{dot}([\boldsymbol{u}; \boldsymbol{v}])$ defined in (16b) is 2-Lipschitz continuous. To see this, consider the following:

$$
\|\nabla \texttt{dot}([\boldsymbol{u}; \boldsymbol{v}])\| = \left\|\begin{bmatrix}\boldsymbol{u} \\ \boldsymbol{v}\end{bmatrix}\right\| \leq 2.
$$

Eq. (26) holds as a consequence of applying Lemma A.9 to the set $\boldsymbol{f}' \circ S$ which is

$$
\begin{aligned}
\boldsymbol{f}' \circ S &= \left\{[\boldsymbol{Z}_1, \boldsymbol{Z}_2] \in \mathbb{R}^{M \times 2K} \mid \boldsymbol{Z}_1 \in \boldsymbol{f} \circ S^{(1)}_{\boldsymbol{X}}, \boldsymbol{Z}_2 \in \boldsymbol{f} \circ S^{(2)}_{\boldsymbol{X}}\right\}, \\
\boldsymbol{S}^{(1)}_{\boldsymbol{X}} &= [\boldsymbol{x}_{\omega^1_1}, \boldsymbol{x}_{\omega^1_2}, \ldots, \boldsymbol{x}_{\omega^1_M}] \in \mathbb{R}^{D \times M}, \\
\boldsymbol{S}^{(2)}_{\boldsymbol{X}} &= [\boldsymbol{x}_{\omega^2_1}, \boldsymbol{x}_{\omega^2_2}, \ldots, \boldsymbol{x}_{\omega^2_M}] \in \mathbb{R}^{D \times M}.
\end{aligned}
$$

Now we can proceed to bound $\mathcal{R}(\ell \circ \mathcal{P}_{\boldsymbol{X}} \circ S)$ with the help of Dudley's entropy integral.

Apply Lemma A.6 onto the set $\ell \circ \mathcal{P}_{\boldsymbol{X}} \circ S$ with $c = 2\log(1/\alpha)\sqrt{M}$ (by (18)), for any integer $T > 0$,

$$\begin{aligned}
\mathcal{R}(\ell \circ \mathcal{P}_{\boldsymbol{X}} \circ S) &\leq \frac{c2^{-T}}{\sqrt{M}} + \frac{6c}{M}\sum_{t=1}^{T} 2^{-t}\sqrt{\log(\mathcal{N}(c2^{-t}, A))} \quad \text{(By Lemma A.6)} \\
&\leq \frac{c2^{-T}}{\sqrt{M}} + \frac{6c}{M}\sum_{t=1}^{T} 2^{-t}\sqrt{\left(\left\|\boldsymbol{S}_{\boldsymbol{X}}^{(1)}\right\|_{\mathrm{F}}^2 + \left\|\boldsymbol{S}_{\boldsymbol{X}}^{(2)}\right\|_{\mathrm{F}}^2\right)\frac{2R_{\mathtt{NET}}L_\ell^2}{c^2 2^{-2t}}} \quad \text{(By (27))} \\
&\leq \frac{c2^{-T}}{\sqrt{M}} + \frac{6T}{M}\sqrt{\left(\left\|\boldsymbol{S}_{\boldsymbol{X}}^{(1)}\right\|_{\mathrm{F}}^2 + \left\|\boldsymbol{S}_{\boldsymbol{X}}^{(2)}\right\|_{\mathrm{F}}^2\right)2R_{\mathtt{NET}}L_\ell^2} \\
&= \frac{c2^{-T}}{\sqrt{M}} + \frac{6\sqrt{2}T}{M}\left\|\boldsymbol{S}_{\boldsymbol{X}}\right\|_{\mathrm{F}} L_\ell\sqrt{R_{\mathtt{NET}}} \\
&= 2\log(1/\alpha)2^{-T} + \frac{6\sqrt{2}T}{M}\left\|\boldsymbol{S}_{\boldsymbol{X}}\right\|_{\mathrm{F}} L_\ell\sqrt{R_{\mathtt{NET}}} \quad \text{(substitute } c\text{)}.
\end{aligned}$$

Lastly, choosing $T = \lfloor \log_2 M \rfloor$ and substituting $L_\ell = 1/\alpha$ concludes the proof,

$$\begin{aligned}
\mathcal{R}(\ell \circ \mathcal{P}_{\mathcal{F},\boldsymbol{X}} \circ S) &\leq 2\log(1/\alpha)2^{-\lfloor \log_2 M \rfloor} + \frac{6\sqrt{2}\lfloor \log_2 M \rfloor}{M}\left\|\boldsymbol{S}_{\boldsymbol{X}}\right\|_{\mathrm{F}} L_\ell\sqrt{R_{\mathtt{NET}}} \\
&\leq \frac{4\log(1/\alpha)}{M} + \frac{6\sqrt{2}\log M}{\alpha M \log 2}\left\|\boldsymbol{S}_{\boldsymbol{X}}\right\|_{\mathrm{F}}\sqrt{R_{\mathtt{NET}}}.
\end{aligned}$$

This concludes the proof. □

A.1.2. SOME OTHERS LEMMAS

Lemma A.4. *For $0 \leq \boldsymbol{P}, \boldsymbol{Q} \leq 1$ with the same size $N \times N$,*

$$\|\boldsymbol{P} - \boldsymbol{Q}\|_{\mathrm{F}}^2 \leq 4N^2 D_H^2(\boldsymbol{P}, \boldsymbol{Q}) \leq 4N^2 D_{\mathrm{kl}}(\boldsymbol{Q}||\boldsymbol{P}),$$

where

$$\begin{aligned}
D_H^2(\boldsymbol{P}, \boldsymbol{Q}) &\triangleq \frac{1}{N^2}\sum_{\omega \in [N]\times[N]} d_H^2(P_\omega, Q_\omega), \\
D_{\mathrm{kl}}(\boldsymbol{P}||\boldsymbol{Q}) &\triangleq \frac{1}{N^2}\sum_{\omega \in [N]\times[N]} d_{\mathrm{kl}}(P_\omega||Q_\omega),
\end{aligned} \tag{28}$$

and $d_H(p, q)$, $d_{\mathrm{kl}}(p, q)$ are typical Hellinger distance and KL divergence for $0 \leq p, q \leq 1$, resp, i.e.,

$$\begin{aligned}
d_H^2(p, q) &\triangleq \left(\sqrt{p} - \sqrt{q}\right)^2 + \left(\sqrt{1-p} - \sqrt{1-q}\right)^2, \\
d_{\mathrm{kl}}(p, q) &\triangleq p\log\left(\frac{p}{q}\right) + (1-p)\log\left(\frac{1-p}{1-q}\right).
\end{aligned} \tag{29}$$

Proof. For the first inequality, we use the following result.

Lemma A.5 (Lemma 2, scalar version, (Davenport et al., 2014))**.** *Let $f : \mathbb{R} \to \mathbb{R}$ be any differential function, and s,t are two real numbers satisfying $|s|, |t| \leq \alpha$. Then*

$$d_H^2(f(s); f(t)) \geq \inf_{|x| \leq \alpha} \frac{f'(x)^2}{8f(x)(1 - f(x))}(s - t)^2.$$

As a result, for $f(x) = x$, $0 \leq p, q \leq 1$,

$$d_H^2(p, q) \geq \inf_{|x| \leq 1} \frac{1}{8x(1-x)}(p - q)^2 = \frac{1}{4}(p - q)^2.$$

Summing across all elements of $\boldsymbol{P}, \boldsymbol{Q}$ leads to,

$$\|\boldsymbol{P}-\boldsymbol{Q}\|_{\mathrm{F}}^2 \leq 4N^2 D_H^2(\boldsymbol{P},\boldsymbol{Q}).$$

The second inequality is simply derived from Jensen's inequality and the fact that $1-x \leq -\log x, \forall x>0$. □

Lemma A.6 (Dudley entropy integral)**.** *((Shalev-Shwartz & Ben-David, 2014, Lemma 27.4)) Let $A \subset \mathbb{R}^m$, $c = \min_{\overline{\boldsymbol{a}}\in\mathbb{R}^m} \max_{\boldsymbol{a}\in A} \|\overline{\boldsymbol{a}}-\boldsymbol{a}\|$. Then, for any integer $T>0$,*

$$R(A) \leq \frac{c2^{-T}}{\sqrt{m}} + \frac{6c}{m}\sum_{k=1}^{T} 2^{-k}\sqrt{\log(N(c2^{-k},A))}.$$

Lemma A.7. *For each $i \in [m]$, let $\boldsymbol{\phi}_i : \mathbb{R}^{d_1} \to \mathbb{R}^{d_2}$ be a ρ-Lipschitz function; namely, for all $\boldsymbol{x},\boldsymbol{y} \in \mathbb{R}^{d_1}$ we have $\|\boldsymbol{\phi}_i(\boldsymbol{x})-\boldsymbol{\phi}_i(\boldsymbol{y})\| \leq \rho\|\boldsymbol{x}-\boldsymbol{y}\|$. For $\boldsymbol{A} \in \mathbb{R}^{d_1\times m}$ let $\boldsymbol{\Phi}(\boldsymbol{A})$ denote the matrix $[\boldsymbol{\phi}_1(\boldsymbol{a}_1),\ldots,\boldsymbol{\phi}_m(\boldsymbol{a}_m)]$. For $\mathcal{A} \subset \mathbb{R}^{d_1\times m}$, let $\boldsymbol{\Phi}\circ\mathcal{A} = \{\boldsymbol{\Phi}(\boldsymbol{A}) \mid \boldsymbol{A}\in\mathcal{A}\} \subset \mathbb{R}^{d_2\times m}$. Then,*

$$\mathcal{N}(\rho r, \boldsymbol{\Phi}\circ\mathcal{A}, \|\cdot\|_{\mathrm{F}}) \leq \mathcal{N}(r,\mathcal{A},\|\cdot\|_{\mathrm{F}}). \tag{30}$$

Remark A.8. Firstly, the choice of Frobenius norm in (30) is tied to ℓ_2-norm defining the ρ-Lipschitz function. Secondly, Lemma A.7 is a natural generalization of (Shalev-Shwartz & Ben-David, 2014, Lemma 27.3).

Proof. Define $\mathcal{A}'$ as one of the smallest r-cover of $\mathcal{A}$, i.e., for all $\boldsymbol{A}\in\mathcal{A}$ there exists $\boldsymbol{A}'\in\mathcal{A}'$ such that $\|\boldsymbol{A}-\boldsymbol{A}'\|_{\mathrm{F}} \leq r$. By definition, $|\mathcal{A}'| = \mathcal{N}(r,\mathcal{A},\|\cdot\|_{\mathrm{F}})$. Since

$$\|\boldsymbol{\Phi}(\boldsymbol{A})-\boldsymbol{\Phi}(\boldsymbol{A}')\|_{\mathrm{F}}^2 = \sum_{i=1}^{m}\|\boldsymbol{\phi}_i(\boldsymbol{a}_i)-\boldsymbol{\phi}_i(\boldsymbol{a}_i')\|^2 \leq \sum_{i=1}^{m}\rho^2\|\boldsymbol{a}_i-\boldsymbol{a}_i'\|^2 = \rho^2\|\boldsymbol{A}-\boldsymbol{A}'\|_{\mathrm{F}}^2 \leq \rho^2 r^2,$$

$\boldsymbol{\Phi}\circ\mathcal{A}'$ is a ρr-cover of $\boldsymbol{\Phi}\circ\mathcal{A}$. That implies $\mathcal{N}(\rho r,\boldsymbol{\Phi}\circ\boldsymbol{A},\|\cdot\|_{\mathrm{F}}) \leq |\boldsymbol{\Phi}\circ\mathcal{A}'| = |\mathcal{A}'| = \mathcal{N}(r,\mathcal{A},\|\cdot\|_{\mathrm{F}})$. □

Lemma A.9. *Given two sets $\mathcal{A}_1,\mathcal{A}_2 \subseteq \mathbb{R}^{m\times d}$. Define $\mathcal{A}_3 \triangleq \{[\boldsymbol{A}_1,\boldsymbol{A}_2] \in \mathbb{R}^{m\times 2d} \mid \boldsymbol{A}_1\in\mathcal{A}_1, \boldsymbol{A}_2\in\mathcal{A}_2\}$, then*

$$\mathcal{N}(\epsilon/\sqrt{2},\mathcal{A}_3,\|\cdot\|_{\mathrm{F}}) \leq \mathcal{N}(\epsilon,\mathcal{A}_1,\|\cdot\|_{\mathrm{F}})\mathcal{N}(\epsilon,\mathcal{A}_2,\|\cdot\|_{\mathrm{F}}).$$

Proof. Let $\overline{\mathcal{A}}_1,\overline{\mathcal{A}}_2$ be minimum ϵ-cover sets of $\mathcal{A}_1,\mathcal{A}_2$, resp. By definition, for all $\boldsymbol{A}_1\in\mathcal{A}_1, \boldsymbol{A}_2\in\mathcal{A}_2$,

$$\begin{aligned}
&\exists\overline{\boldsymbol{A}}_1\in\overline{\mathcal{A}}_1, \quad \left\|\boldsymbol{A}_1-\overline{\boldsymbol{A}}_1\right\|_{\mathrm{F}}^2 \leq \epsilon^2, \text{ and}\\
&\exists\overline{\boldsymbol{A}}_2\in\overline{\mathcal{A}}_2, \quad \left\|\boldsymbol{A}_2-\overline{\boldsymbol{A}}_2\right\|_{\mathrm{F}}^2 \leq \epsilon^2\\
&\implies \quad \left\|[\boldsymbol{A}_1,\boldsymbol{A}_2]-[\overline{\boldsymbol{A}}_1,\overline{\boldsymbol{A}}_2]\right\|_{\mathrm{F}}^2 \leq 2\epsilon^2.
\end{aligned}$$

Therefore, set $\overline{\mathcal{A}}_3 \triangleq \{[\overline{\boldsymbol{A}}_1,\overline{\boldsymbol{A}}_2] \in \mathbb{R}^{m\times 2d} \mid \overline{\boldsymbol{A}}_1\in\overline{\mathcal{A}}_1, \overline{\boldsymbol{A}}_2\in\overline{\mathcal{A}}_2\}$ is a $(\epsilon/\sqrt{2})$-cover set of $\mathcal{A}_3$, which leads to our conclusion,

$$\mathcal{N}(\epsilon/\sqrt{2},\mathcal{A}_3,\|\cdot\|_{\mathrm{F}}) \leq \left|\overline{\mathcal{A}}_3\right| = \left|\overline{\mathcal{A}}_1\right|\left|\overline{\mathcal{A}}_2\right| = \mathcal{N}(\epsilon,\mathcal{A}_1,\|\cdot\|_{\mathrm{F}})\mathcal{N}(\epsilon,\mathcal{A}_2,\|\cdot\|_{\mathrm{F}}).$$

□

Lemma A.10. *Covering number of neural networks (Bartlett et al., 2017, Theorem 3.3) Let fixed nonlinearities $(\sigma_1,\ldots,\sigma_L)$ and reference matrices $(M_1,\ldots,M_L)$ be given, where σ_i is ρ_i-Lipschitz and $\sigma_i(0)=0$. Let spectral norm bounds $(s_1,\ldots,s_L)$, and matrix (2,1) norm bounds $b_1,\ldots,b_L$ be given. Let data matrix $\boldsymbol{X}\in\mathbb{R}^{n\times d}$ be given, where the n rows correspond to data points. Let $\mathcal{H}_{\boldsymbol{X}}$ denote the family of matrices obtained by evaluating $\boldsymbol{X}$ with all choices of network $F_{\mathcal{A}}$:*

$$\mathcal{H}_{\boldsymbol{X}} \triangleq \left\{F_{\mathcal{A}}(\boldsymbol{X}^\top) \mid \mathcal{A}=(\boldsymbol{A}_1,\ldots,\boldsymbol{A}_L), \|\boldsymbol{A}_i\|_\sigma \leq s_i, \left\|\boldsymbol{A}_i^\top-\boldsymbol{I}\right\|_{2,1} \leq b_i\right\},$$

where each matrix has dimension at most D along each axis. Then for any $\epsilon>0$,

$$\log\mathcal{N}(\epsilon,\mathcal{H}_{\boldsymbol{X}},\|\cdot\|_{\mathrm{F}}) \leq \frac{\|\boldsymbol{X}\|_{\mathrm{F}}^2 R_{NET}}{\epsilon^2},$$

where $R_{NET} = \log(2D^2)\left(\prod_{j=1}^{L}s_j^2\rho_j^2\right)\left(\sum_L^i (b_i/s_i)^{2/3}\right)^3$ is a constant depending only on the neural network's properties.

B. Proof of Theorem 3.4

Let $\boldsymbol{M}^\natural = [\boldsymbol{f}^\natural(\boldsymbol{x}_1), \dots, \boldsymbol{f}^\natural(\boldsymbol{x}_N)]$. As $\boldsymbol{M}^\natural$ satisfies separability condition, we assume that $\boldsymbol{M}^\natural = [\boldsymbol{I}, \boldsymbol{V}^\natural]$, which is w.o.l.g. Let $\boldsymbol{M}^\star = [\boldsymbol{f}^\star(\boldsymbol{x}_1), \dots, \boldsymbol{f}^\star(\boldsymbol{x}_N)]$ denote the predicted cluster memberships, and also partition $\boldsymbol{M}^\star$ as $\boldsymbol{M}^\star = [\boldsymbol{U}^\star, \boldsymbol{V}^\star], \boldsymbol{U}^\star \in \mathbb{R}^{K\times K}$. Then,

$$\begin{aligned}
\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2 &= \left\|(\boldsymbol{M}^\star)^\top \boldsymbol{M}^\star - (\boldsymbol{M}^\natural)^\top \boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2 \\
&= \left\| \begin{bmatrix} (\boldsymbol{U}^\star)^\top \\ (\boldsymbol{V}^\star)^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{U}^\star & \boldsymbol{V}^\star \end{bmatrix} - \begin{bmatrix} \boldsymbol{I}_K \\ (\boldsymbol{V}^\natural)^\top \end{bmatrix} \begin{bmatrix} \boldsymbol{I}_K & \boldsymbol{V}^\natural \end{bmatrix} \right\|_{\mathrm{F}}^2 \\
&= \left\|(\boldsymbol{U}^\star)^\top \boldsymbol{U}^\star - \boldsymbol{I}_K\right\|_{\mathrm{F}}^2 + 2\left\|(\boldsymbol{U}^\star)^\top \boldsymbol{V}^\star - \boldsymbol{V}^\natural\right\|_{\mathrm{F}}^2 + \left\|(\boldsymbol{V}^\star)^\top \boldsymbol{V}^\star - (\boldsymbol{V}^\natural)^\top \boldsymbol{V}^\natural\right\|_{\mathrm{F}}^2. \qquad (31)
\end{aligned}$$

Analysis of the first term in (31). Let $Z = \frac{1}{N^2}\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2, 0 \le Z \le 1$ be a random variable with a PDF $f(z)$. By Lemma 3.3,

$$\mathsf{Pr}\left(Z \le \epsilon(M,\delta)^2\right) \ge 1 - \delta.$$

We have

$$\begin{aligned}
\mathbb{E}[Z] = \int_0^1 z f(z) dz = \int_0^{\epsilon(M,\delta)^2} z f(z) dz + \int_{\epsilon(M,\delta)^2}^1 z f(z) dz &\le \int_0^{\epsilon(M,\delta)^2} \epsilon(M,\delta)^2 f(z) dz + \int_{\epsilon(M,\delta)^2}^1 f(z) dz \\
&\le \epsilon(M,\delta)^2 + \delta.
\end{aligned}$$

Recall $S = \{(i_1, j_1, y_1), \dots, (i_M, j_M, y_M)\}$. According to Lemma B.3,

$$\mathop{\mathbb{E}}_{\boldsymbol{X},S}\left[(P_{ij}^\star - P_{ij}^\natural)^2\right] = \text{const}, \quad \forall (i,j) \in [N]^2.$$

Therefore, for arbitrary $i, j \in [N]^2$,

$$\mathop{\mathbb{E}}_{\boldsymbol{X},S}\left[(P_{ij}^\star - P_{ij}^\natural)^2\right] = \mathop{\mathbb{E}}_{\boldsymbol{X},S}\left[\frac{1}{N^2}\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2\right] \le \epsilon(M,\delta)^2 + \delta.$$

Meanwhile, by Markov inequality, for $\tau > 0$,

$$\mathsf{Pr}\left(\left(P_{ij}^\star - P_{ij}^\natural\right)^2 \le \tau \mathop{\mathbb{E}}_{\boldsymbol{X},S}\left[\left(P_{ij}^\star - P_{ij}^\natural\right)^2\right]\right) \ge 1 - \frac{1}{\tau}.$$

Setting $\tau = M^{1/4}, \delta = e^{-\sqrt{M}}$ gives

$$\left(P_{ij}^\star - P_{ij}^\natural\right)^2 \le M^{1/4} \mathop{\mathbb{E}}_{\boldsymbol{X},S}\left[\left(P_{ij}^\star - P_{ij}^\natural\right)^2\right] \le M^{1/4}\epsilon(M, e^{-\sqrt{M}})^2 + M^{1/4} e^{-\sqrt{M}}$$

holds with probability at least $1 - 1/M^{1/4}$.

By union bound, the following holds with probability at least $1 - K^2/M^{1/4}$,

$$\left\|(\boldsymbol{U}^\star)^\top \boldsymbol{U}^\star - \boldsymbol{I}_K\right\|_{\mathrm{F}}^2 = \sum_{(i,j)\in[K]^2} (P_{ij}^\star - P_{ij}^\natural)^2 \le K^2 \underbrace{\left(M^{1/4}\epsilon(M, e^{-\sqrt{M}})^2 + M^{1/4} e^{-\sqrt{M}}\right)}_{\epsilon'(M)^2} = K^2 \epsilon'(M)^2, \qquad (32)$$

where $\epsilon'(M)$ is defined in (9). Inequality (32) implies

$$\begin{cases}
1 - K\epsilon'(M) \le \|\boldsymbol{u}_k^\star\|^2 \le 1, & \forall k \in [K] \qquad (33) \\
0 \le \langle \boldsymbol{u}_k^\star, \boldsymbol{u}_\ell^\star \rangle \le K\epsilon'(M), & \forall k \ne \ell, k, \ell \in [K]. \qquad (34)
\end{cases}$$

For any permutation matrix $\boldsymbol{\Pi} \in \mathbb{R}^{K\times K}$,

$$\begin{aligned}\|\boldsymbol{\Pi}\boldsymbol{U}^\star - \boldsymbol{I}\|_{\mathrm{F}}^2 = \sum_{k=1}^K \|\boldsymbol{u}_k^\star - \boldsymbol{\pi}_k\|^2 &= K + \sum_{k=1}^K \|\boldsymbol{u}_k^\star\|^2 - 2\sum_{k=1}^K \langle \boldsymbol{u}_k^\star, \boldsymbol{\pi}_k\rangle \\ &\le 2K - 2\sum_{k=1}^K \langle \boldsymbol{u}_k^\star, \boldsymbol{\pi}_k\rangle \quad \text{(Since } \boldsymbol{u}_k^\star \ge 0, \mathbf{1}^\top \boldsymbol{u}_k^\star = 1 \text{ and thus } \|\boldsymbol{u}_k^\star\| \le 1).\end{aligned}$$

Minimizing over $\boldsymbol{\Pi}$ on both sides,

$$\min_{\boldsymbol{\Pi}} \|\boldsymbol{\Pi}\boldsymbol{U}^\star - \boldsymbol{I}\|_{\mathrm{F}}^2 \le 2K - 2\max_{\boldsymbol{\Pi}} \sum_{k=1}^K \langle \boldsymbol{u}_k^\star, \boldsymbol{\pi}_k\rangle. \tag{35}$$

Denote $k_i^\star := \arg\max_k U_{ki}^\star$. Invoking Lemma B.2 where $K\epsilon'(M) < 0.381$ (by (9)) and $\boldsymbol{u}_k^\star$'s satisfy (33) and (34), it holds that $k_1^\star, \dots, k_K^\star$ are all different integers, and we assume w.o.l.g. that $k_1^\star = 1, \dots, k_K^\star = K$, i.e., $U_{ii}^\star = \arg\max_k U_{ki}^\star$. With that in mind, we get a simple lower bound,

$$\max_{\boldsymbol{\Pi}} \sum_{k=1}^K (\boldsymbol{u}_k^\star)^\top \boldsymbol{\pi}_k \ge \sum_{k=1}^K U_{kk}^\star. \tag{36}$$

Since $\|\boldsymbol{u}_1^\star\|^2 \ge 1 - K\epsilon'(M)$ by (33),

$$(U_{11}^\star)^2 \ge 1 - K\epsilon'(M) - \sum_{k=2}^K (U_{k1}^\star)^2 \ge 1 - K\epsilon'(m) - \left(\sum_{k=2}^K U_{k1}^\star\right)^2 = 1 - K\epsilon'(m) - (1 - U_{11}^\star)^2,$$

where the second inequality holds since $U_{k1}^\star \ge 0,\ \forall k \in [K]$, and the last equality holds since $\boldsymbol{u}_1$ is a probability simplex vector. As $U_{11}^\star = \max_k U_{k1}$, $U_{11}^\star \ge 1/K$. Solving the above quadratic inequality with respect to $1/K \le U_{11}^\star \le 1$ and $K\epsilon'(M) \le 0.381$ (by (9)) leads to

$$U_{11}^\star \ge \frac{1 + \sqrt{1 - 2K\epsilon'(M)}}{2} \ge 1 - K\epsilon'(M).$$

Applying the same argument for $U_{22}^\star, \dots, U_{KK}^\star$,

$$U_{kk}^\star \ge 1 - K\epsilon'(M), \quad k \in [K].$$

Therefore,

$$\min_{\boldsymbol{\Pi}} \|\boldsymbol{\Pi}\boldsymbol{U}^\star - \boldsymbol{I}\|_{\mathrm{F}}^2 \le 2K - 2\left(K - K^2\epsilon'(M)\right) = 2K^2\epsilon'(M). \tag{37}$$

This implies that $\boldsymbol{U}^\star$ is close to an identity matrix up to some permutation.

Analysis of the second term in (31). For a fixed $0 < \delta < 1$, suppose that the following event happens

$$\frac{1}{N^2}\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2 \le \epsilon(M,\delta)^2. \tag{38}$$

Recall that $\boldsymbol{M}^\star = [\boldsymbol{U}^\star, \boldsymbol{V}^\star], \boldsymbol{M}^\natural = [\boldsymbol{I}_K, \boldsymbol{V}^\natural]$, then

$$\left\|(\boldsymbol{U}^\star)^\top \boldsymbol{V}^\star - \boldsymbol{V}^\natural\right\|_{\mathrm{F}} \le \frac{N\epsilon(M,\delta)}{\sqrt{2}}.$$

Denote $\boldsymbol{\Pi}^\star$ as the optimal permutation matrix in LHS of (37). We have

$$\begin{aligned}\left\|(\boldsymbol{U}^\star)^\top \boldsymbol{V}^\star - \boldsymbol{V}^\natural - ((\boldsymbol{U}^\star)^\top(\boldsymbol{\Pi}^\star)^\top - \boldsymbol{I})\boldsymbol{V}^\natural\right\|_{\mathrm{F}} &\le \left\|(\boldsymbol{U}^\star)^\top \boldsymbol{V}^\star - \boldsymbol{V}^\natural\right\|_{\mathrm{F}} + \left\|((\boldsymbol{U}^\star)^\top(\boldsymbol{\Pi}^\star)^\top - \boldsymbol{I})\boldsymbol{V}^\natural\right\|_{\mathrm{F}} \\ &\le \frac{N\epsilon(M,\delta)}{\sqrt{2}} + \sigma_{\max}(\boldsymbol{V}^\natural)\left\|\boldsymbol{\Pi}^\star\boldsymbol{U}^\star - \boldsymbol{I}\right\|_{\mathrm{F}} \\ &\le \frac{N\epsilon(M,\delta)}{\sqrt{2}} + \sigma_{\max}(\boldsymbol{V}^\natural)K\sqrt{2\epsilon'(M)} \quad \text{(thanks to (37))}.\end{aligned} \tag{39}$$

On the other hand,

$$\begin{aligned}\left\|(\boldsymbol{U}^\star)^\top\boldsymbol{V}^\star-\boldsymbol{V}^\natural-((\boldsymbol{U}^\star)^\top(\boldsymbol{\Pi}^\star)^\top-\boldsymbol{I})\boldsymbol{V}^\natural\right\|_{\mathrm{F}}&=\left\|(\boldsymbol{U}^\star)^\top(\boldsymbol{V}^\star-(\boldsymbol{\Pi}^\star)^\top\boldsymbol{V}^\natural)\right\|_{\mathrm{F}}\\&\geq\sigma_{\min}(\boldsymbol{U}^\star)\left\|\boldsymbol{V}^\star-(\boldsymbol{\Pi}^\star)^\top\boldsymbol{V}^\natural\right\|_{\mathrm{F}}\\&=\sigma_{\min}(\boldsymbol{U}^\star)\left\|\boldsymbol{\Pi}^\star\boldsymbol{V}^\star-\boldsymbol{V}^\natural\right\|_{\mathrm{F}}.\end{aligned}\tag{40}$$

Using Lemma B.1, we can upper bound $\sigma_{\min}(\boldsymbol{U}^\star)$ as

$$|\sigma_{\min}(\boldsymbol{I}+\boldsymbol{\Pi}^\star\boldsymbol{U}^\star-\boldsymbol{I})-\sigma_{\min}(\boldsymbol{I})|\leq\|\boldsymbol{\Pi}^\star\boldsymbol{U}^\star-\boldsymbol{I}\|_{\mathrm{F}}\leq K\sqrt{2\epsilon'(M)}\quad\text{(thanks to (37))}$$

or,

$$\sigma_{\min}(\boldsymbol{U}^\star)=\sigma_{\min}(\boldsymbol{\Pi}^\star\boldsymbol{U}^\star)\geq 1-K\sqrt{2\epsilon'(M)}.\tag{41}$$

Combine (39), (40), (41),

$$\begin{aligned}\left\|\boldsymbol{\Pi}^\star\boldsymbol{V}^\star-\boldsymbol{V}^\natural\right\|_{\mathrm{F}}^2&\leq\left(\frac{N\epsilon(M,\delta)+2\sigma_{\max}(\boldsymbol{V}^\natural)K\sqrt{\epsilon'(M)}}{\sqrt{2}-2K\sqrt{\epsilon'(M)}}\right)^2\\&\leq 2\left(N\epsilon(M,\delta)+2\sigma_{\max}(\boldsymbol{V}^\natural)K\sqrt{\epsilon'(M)}\right)^2\quad\text{(Since }8K^2\epsilon'(M)\leq 1\text{ by (9))}\\&\leq 4N^2\epsilon(M,\delta)^2+16\sigma_{\max}^2(\boldsymbol{V}^\natural)K^2\epsilon'(M)\quad\text{(Cauchy-Schwarz inequality).}\end{aligned}\tag{42}$$

Lastly, combine (37) and (42), we finish our proof,

$$\begin{aligned}\left\|\boldsymbol{\Pi}^\star\boldsymbol{M}^\star-\boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2&=\|\boldsymbol{\Pi}^\star\boldsymbol{U}-\boldsymbol{I}\|_{\mathrm{F}}^2+\left\|\boldsymbol{\Pi}^\star\boldsymbol{V}-\boldsymbol{V}^\natural\right\|_{\mathrm{F}}^2\\&\leq 2K^2\epsilon'(M)+4N^2\epsilon(M,\delta)^2+16\sigma_{\max}^2(\boldsymbol{V}^\natural)K^2\epsilon'(M)\\&=4N^2\epsilon(M,\delta)^2+2K^2(1+8\sigma_{\max}^2(\boldsymbol{V}^\natural))\epsilon'(M).\end{aligned}$$

Note that the whole computation is derived based on the realizations of 2 independent events in (32), (38) with probability of happening at least $1-K^2/M^{1/4}$ and $1-\delta$, resp,

$$\frac{1}{N}\left\|\boldsymbol{\Pi}^\star\boldsymbol{M}^\star-\boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2\leq 4N\epsilon(M,\delta)^2+\frac{2}{N}K^2(1+8\sigma_{\max}^2(\boldsymbol{V}^\natural))\epsilon'(M).$$

holds with probability at least $1-(\delta+K^2/M^{1/4})$.

B.1. Supporting Lemmas for proof of Theorem 3.4

Lemma B.1 ((Weyl, 1912)). *Let* $\boldsymbol{X},\boldsymbol{\Delta}\in\mathbb{R}^{m\times n}$,

$$|\sigma_i(\boldsymbol{X}+\boldsymbol{\Delta})-\sigma_i(\boldsymbol{\Delta})|\leq\|\boldsymbol{\Delta}\|_2\quad(\leq\|\boldsymbol{\Delta}\|_{\mathrm{F}}),\quad 1\leq i\leq\min(m,n).$$

Lemma B.2. *Define* $A_\epsilon=\left\{\boldsymbol{x}\in\mathbb{R}^K\mid\mathbf{1}^\top\boldsymbol{x}=1,\boldsymbol{x}\geq 0,\|\boldsymbol{x}\|_2^2\geq 1-\epsilon\right\}$. *If* $\epsilon\leq 0.381$, *then*

$$\max_{\boldsymbol{x},\boldsymbol{y}\in A_\epsilon}\boldsymbol{x}^\top\boldsymbol{y}\leq\epsilon\quad\Rightarrow\quad\arg\max_k x_k\neq\arg\max_k y_k.$$

Proof. We use contradiction to prove the claim. Suppose that $\arg\max_k x_k=\arg\min_k y_k=\widetilde{k}$. Without loss of generality, let us assume $\widetilde{k}=1$. Using these assumptions, we will show that $\min_{\boldsymbol{x},\boldsymbol{y}\in A_\epsilon}\boldsymbol{x}^\top\boldsymbol{y}>\epsilon$. Indeed,

$$1-\epsilon\leq\|\boldsymbol{x}\|^2\leq x_1\left(\sum_{k=1}^K x_k\right)=x_1.$$

Similarly, we obtain $y_1\geq 1-\epsilon$, which leads to

$$\boldsymbol{x}^\top\boldsymbol{y}\geq(1-\epsilon)^2>\epsilon,$$

where the second inequality holds because $\epsilon\leq 0.381$. □

Lemma B.3. *Assume that $\omega = (\omega^1, \omega^2)$ is a uniform RV over $[N] \times [N]$ where $\omega^1 \in [N], \omega^2 \in [N]$, and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$ are i.i.d. Define the following mappings:*

$$\begin{aligned}
&\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N, \omega) &&\triangleq (\boldsymbol{x}_{\omega^1}, \boldsymbol{x}_{\omega^2}),\\
&\texttt{gen}(\omega_1, \ldots, \omega_M, y_1, \ldots, y_M, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N) &&\triangleq \{\boldsymbol{\psi}_1, \ldots, \boldsymbol{\psi}_M\}, \quad \boldsymbol{\psi}_i \triangleq (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N, \omega_i), y_i),
\end{aligned}$$

where

$$\begin{aligned}
&\omega_i = (k, \ell) \in [N]^2, && 1 \le i \le M,\\
&y_i \in \{0, 1\}, && 1 \le i \le M,\\
&\boldsymbol{x}_i \in \mathcal{X}, && 1 \le i \le N.
\end{aligned}$$

For a real-valued function $h(\boldsymbol{u}, \boldsymbol{v}, \texttt{gen}(\omega_1, \ldots, \omega_M, y_1, \ldots, y_M, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N))$ with $\boldsymbol{u}, \boldsymbol{v} \in \mathcal{X}$, we have: $\forall y_1, \ldots, y_M$, and $\forall i, j, i', j' \in [N]$,

$$\begin{aligned}
&\mathop{\mathbb{E}}_{\substack{\omega_i, 1 \le i \le M,\\ \boldsymbol{x}_i, 1 \le i \le N}} [h(\boldsymbol{x}_i, \boldsymbol{x}_j, \texttt{gen}(\omega_1, \ldots, \omega_M, y_1, \ldots, y_M, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N))] =\\
&\qquad\qquad \mathop{\mathbb{E}}_{\substack{\omega_i, 1 \le i \le M,\\ \boldsymbol{x}_i, 1 \le i \le N}} [h(\boldsymbol{x}_{i'}, \boldsymbol{x}_{j'}, \texttt{gen}(\omega_1, \ldots, \omega_M, y_1, \ldots, y_M, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N))]. \quad (43)
\end{aligned}$$

Proof. We have the following property on the mapping $\texttt{gen}$: $\forall i \ne j, i, j \in [N]$,

$$\begin{aligned}
&\texttt{gen}(\omega_1, \ldots, \omega_M, y_1, \ldots, y_M, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N) =\\
&\qquad\qquad \texttt{gen}(s_{ij}(\omega_1), \ldots, s_{ij}(\omega_M), y_1, \ldots, y_M, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N), \quad (44)
\end{aligned}$$

where $s_{ij} : [N]^2 \to [N]^2$,

$$s_{ij}(\omega) = s_{ij}((k, \ell)) \triangleq \begin{cases} (k, \ell) & \text{if } k \ne i \text{ and } \ell \ne j\\ (j, \ell) & \text{if } k = i \text{ and } \ell \ne j\\ (k, i) & \text{if } k \ne i \text{ and } \ell = j\\ (j, i) & \text{if } k = i \text{ and } \ell = j \end{cases}$$

To see this, let

$$\begin{aligned}
\mathcal{G}_1 &= \texttt{gen}(\omega_1, \ldots, \omega_M, y_1, \ldots, y_M, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N),\\
\mathcal{G}_2 &= \texttt{gen}(s_{ij}(\omega_1), \ldots, s_{ij}(\omega_M), y_1, \ldots, y_M, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N).
\end{aligned}$$

Pick $\boldsymbol{\psi} = (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N, \omega), y) \in \mathcal{G}_1$. Assume $\omega = (k, \ell)$. Consider the following cases,

- If $k \ne i$ and $\ell \ne j$, then $s_{ij}(\omega) = \omega$. So
$$\boldsymbol{\psi} = (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, s_{ij}(\omega)), y) \in \mathcal{G}_2.$$
- If $k = i$ and $\ell \ne j$, then $s_{ij}(\omega) = s_{ij}((i, \ell)) = (j, \ell)$. So
$$\begin{aligned}
\boldsymbol{\psi} &= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N, (i, \ell)), y)\\
&= (\boldsymbol{x}_i, \boldsymbol{x}_\ell, y)\\
&= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, (j, \ell)), y)\\
&= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, s_{ij}(\omega)), y) \in \mathcal{G}_2 \quad \text{(by definition of the mapping } \texttt{gen}).
\end{aligned}$$
- If $k \ne i$ and $\ell = j$, then $s_{ij}(\omega) = s_{ij}((k, j)) = (k, i)$. So
$$\begin{aligned}
\boldsymbol{\psi} &= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N, (k, j)), y)\\
&= (\boldsymbol{x}_k, \boldsymbol{x}_j, y)\\
&= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, (k, i)), y)\\
&= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, s_{ij}(\omega)), y) \in \mathcal{G}_2 \quad \text{(by definition of the mapping } \texttt{gen}).
\end{aligned}$$

- If $k = i$ and $\ell = j$, then $s_{ij}(\omega) = s_{ij}((i,j)) = (j,i)$. So

$$\begin{aligned}\boldsymbol{\psi} &= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_N, (i,j)), y)\\ &= (\boldsymbol{x}_i, \boldsymbol{x}_j, y)\\ &= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, (j,i)), y)\\ &= (\texttt{pick}(\boldsymbol{x}_1, \ldots, \boldsymbol{x}_j, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_N, s_{ij}(\omega)), y) \in \mathcal{G}_2 \quad \text{(by definition of the mapping } \texttt{gen}).\end{aligned}$$

Since $\psi \in \mathcal{G}_1$ is arbitrary, $\mathcal{G}_1 \subseteq \mathcal{G}_2$. Moreover, $|\mathcal{G}_1| = |\mathcal{G}_2|$, so $\mathcal{G}_1 = \mathcal{G}_2$, and hence (44) holds.

Notice that the mapping $s_{ij}(\omega)$ is surjective and onto $[N]^2$, we proceed with the following

$$\begin{aligned}&\mathop{\mathbb{E}}_{\substack{\boldsymbol{x}_i, \boldsymbol{x}_{i'},\\ \boldsymbol{x}_{k,k=[N]\setminus\{i,i'\}},\\ \omega_1,\ldots,\omega_M}} [h(\boldsymbol{x}_i, \boldsymbol{x}_j, \texttt{gen}(\omega_1, \ldots, \omega_M, y_1, ..., y_M, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_{i'}, \ldots))]\\ &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{x}_i, \boldsymbol{x}_{i'},\\ \boldsymbol{x}_{k,k=[N]\setminus\{i,i'\}},\\ \omega_1,\ldots,\omega_M}} [h(\boldsymbol{x}_i, \boldsymbol{x}_j, \texttt{gen}(s_{ii'}(\omega_1), \ldots, s_{ii'}(\omega_M), y_1, \ldots, y_M, \ldots, \boldsymbol{x}_{i'}, \ldots, \boldsymbol{x}_i, \ldots)]\\ &= \mathop{\mathbb{E}}_{\substack{\boldsymbol{u}, \boldsymbol{v},\\ \boldsymbol{x}_{k,k=[N]\setminus\{i,i'\}},\\ \omega_1,\ldots,\omega_M}} [h(\boldsymbol{u}, \boldsymbol{x}_j, \texttt{gen}(s_{ii'}(\omega_1), \ldots, s_{ii'}(\omega_M), y_1, \ldots, y_M, \ldots, \boldsymbol{v}, \ldots, \boldsymbol{u}, \ldots))]\\ &= \mathbb{E}_{\substack{\boldsymbol{x}_{i'}, \boldsymbol{x}_i,\\ \boldsymbol{x}_{k,k=[N]\setminus\{i,i'\}},\\ \omega_1,\ldots,\omega_M}} [h(\boldsymbol{x}_{i'}, \boldsymbol{x}_j, \texttt{gen}(s_{ii'}(\omega_1), \ldots, s_{ii'}(\omega_M), y_1, \ldots, y_M, \ldots, \boldsymbol{x}_i, \ldots, \boldsymbol{x}_{i'}, \ldots))]\,.\end{aligned}$$

The first equality holds by property (44), the second and the third equality hold as we just rename some random variables under the expectation. Proceed with the same argument to replace j with j', we obtain

$$\begin{aligned}&\mathbb{E}_{\substack{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N,\\ \omega_1,\ldots,\omega_M}} [h(\boldsymbol{x}_{i'}, \boldsymbol{x}_j, \texttt{gen}(s_{ii'}(\omega_1), \ldots, s_{ii'}(\omega_M), y_1, \ldots, y_M, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N))]\\ &\qquad = \mathbb{E}_{\substack{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_N,\\ \omega_1,\ldots,\omega_M}} [h(\boldsymbol{x}_{i'}, \boldsymbol{x}_{j'}, \texttt{gen}(s_{jj'}(s_{ii'}(\omega_1)), \ldots, s_{jj'}(s_{ii'}(\omega_M)), y_1, \ldots, y_M, \boldsymbol{x}_1, \ldots, \boldsymbol{x}_N))]\,. \quad (45)\end{aligned}$$

Since $s_{ii'}(\omega)$ is surjective and onto itself, $s_{ii'}(\omega)$ has the same probability distribution as ω. So $s_{jj'}(s_{ii'}(\omega))$ also define a RV with the same PDF as ω, and hence (45) implies (43).

□

C. Proof of Theorem 3.5

We start with the following key lemma:

Lemma C.1. *With probability at least $1-\delta$ over $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N$,*

$$\mathop{\mathbb{E}}_{\boldsymbol{x}\sim\mathcal{P}_{\mathcal{X}}} \left[\left\|\boldsymbol{\Pi}^\star \boldsymbol{f}^\star(\boldsymbol{x}) - \boldsymbol{f}^\natural(\boldsymbol{x})\right\|^2\right] \le \frac{1}{N}\left\|\boldsymbol{\Pi}^\star \boldsymbol{M}^\star - \boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2 + \frac{8}{N} + \frac{12\sqrt{2R_{NET}}\,\|\boldsymbol{X}\|_{\mathrm{F}} \log N}{N \log 2} + 8\sqrt{\frac{2\log(4\delta)}{N}},$$

where $\boldsymbol{\Pi}^\star$ is the optimal permutation matrix in LHS of (37) in the proof of Theorem 3.4.

Proof. Let us define the following:

$$\begin{aligned}L_{\boldsymbol{X}}(\boldsymbol{f}) &\triangleq \frac{1}{N}\sum_{n=1}^{N} \mathrm{SE}(\boldsymbol{f}, (\boldsymbol{x}_n, \boldsymbol{m}_n^\natural)),\\ \mathrm{SE}(\boldsymbol{f}, (\boldsymbol{x}, \boldsymbol{m})) &\triangleq \left\|\boldsymbol{\Pi}^\star \boldsymbol{f}(\boldsymbol{x}) - \boldsymbol{m}\right\|^2.\end{aligned}$$

Note that $L_{\boldsymbol{X}}(\boldsymbol{f})$ can be viewed as the empirical loss in statistical learning where a dataset of N i.i.d samples $(\boldsymbol{x}_1, \boldsymbol{m}_1^\natural), \ldots, (\boldsymbol{x}_N, \boldsymbol{m}_N^\natural)$ are given. Denote $\mathcal{P}_{\mathcal{X},\boldsymbol{f}^\natural}$ as the PDF of the joint distribution of $(\boldsymbol{x}_n, \boldsymbol{m}_n^\natural)$, and define

$$L_{\mathcal{P}_{\mathcal{X},\boldsymbol{f}^\natural}}(\boldsymbol{f}) \triangleq \mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{m}^\natural)\sim\mathcal{P}_{\mathcal{X},\boldsymbol{f}^\natural}} \left[\mathrm{SE}(\boldsymbol{f}, (\boldsymbol{x}, \boldsymbol{m}^\natural))\right],$$

which is the expected loss. Then it is readily to apply a standard generalization bound from (Shalev-Shwartz & Ben-David, 2014, Theorem 26.5) using the fact that $\mathrm{SE}(\boldsymbol{f},(\boldsymbol{x},\boldsymbol{m}))\leq 2$. That is, we have that

$$L_{\mathcal{P}_{\mathcal{X},\boldsymbol{f}^\natural}}(\boldsymbol{f}^\star)\leq L_{\boldsymbol{X}}(\boldsymbol{f}^\star)+2\mathcal{R}(\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X})+8\sqrt{\frac{2\log(4/\delta)}{N}},$$

holds with probability of at least $1-\delta$. The first term is

$$L_{\boldsymbol{X}}(\boldsymbol{f}^\star)=\frac{1}{N}\left\|\boldsymbol{\Pi}^\star\boldsymbol{M}^\star-\boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2.$$

The second term is bounded by following the proof of Lemma A.3,

$$\begin{aligned}\mathcal{N}\left(\epsilon,\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X},\|\cdot\|\right)&\leq\mathcal{N}\left(\frac{\epsilon}{2\sqrt{2}},\mathcal{F}\circ\boldsymbol{X},\|\cdot\|_{\mathrm{F}}\right)\qquad(46)\\&\leq\exp\left(\frac{8R_{\text{NET}}\left\|\boldsymbol{X}\right\|_{\mathrm{F}}^2}{\epsilon^2}\right)\quad\text{(By Lemma A.10).}\end{aligned}$$

Inequality (46) holds by applying Lemma A.7 and the fact that function $\phi_{\boldsymbol{m}}(\boldsymbol{x})\triangleq\|\boldsymbol{x}-\boldsymbol{m}\|^2$ is $2\sqrt{2}$-Lipschitz continuous on the domain $\boldsymbol{\Delta}=\left\{\boldsymbol{x}\in\mathbb{R}^K\mid\boldsymbol{x}\geq 0,\mathbf{1}^\top\boldsymbol{x}=1\right\}$. This is because of the following fact:

$$\|\nabla\phi_{\boldsymbol{m}}(\boldsymbol{x})\|=2\|\boldsymbol{x}-\boldsymbol{m}\|\leq 2\sqrt{2}.$$

We proceed by deriving a bound on $\mathcal{R}(\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X})$ with the help of Dudley's entropy integral technique. In particular, applying Lemma A.6 on the set $\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X}$, the following holds: For any integer $T>0$,

$$\begin{aligned}\mathcal{R}(\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X})&\leq\frac{c2^{-T}}{\sqrt{N}}+\frac{6c}{N}\sum_{t=1}^{T}2^{-t}\sqrt{\log(\mathcal{N}(c2^{-t},\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X}))}\\&\leq\frac{c2^{-T}}{\sqrt{N}}+\frac{6c}{N}\sum_{t=1}^{T}2^{-t}\sqrt{\frac{8R_{\text{NET}}\left\|\boldsymbol{X}\right\|_{\mathrm{F}}^2}{c^2 2^{-2t}}}\\&=\frac{c2^{-T}}{\sqrt{N}}+\frac{12T\sqrt{2R_{\text{NET}}}\left\|\boldsymbol{X}\right\|_{\mathrm{F}}}{N}.\end{aligned}$$

Substituting $c=2$ as the upper bound of the loss SE and setting $T=\lfloor 0.5\log_2(N)\rfloor$ gives

$$\mathcal{R}(\mathrm{SE}\circ\mathcal{F}\circ\boldsymbol{X})\leq\frac{4}{N}+\frac{6\sqrt{2R_{\text{NET}}}\left\|\boldsymbol{X}\right\|_{\mathrm{F}}\log N}{N\log 2}.$$

Therefore,

$$\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{m}^\natural)\sim\mathcal{P}_{\mathcal{X},\boldsymbol{f}^\natural}}\left[\left\|\boldsymbol{\Pi}^\star\boldsymbol{f}^\star(\boldsymbol{x})-\boldsymbol{m}^\natural\right\|^2\right]\leq\frac{1}{N}\left\|\boldsymbol{\Pi}^\star\boldsymbol{M}^\star-\boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2+\frac{8}{N}+\frac{12\sqrt{2R_{\text{NET}}}\left\|\boldsymbol{X}\right\|_{\mathrm{F}}\log N}{N\log 2}+8\sqrt{\frac{2\log(4/\delta)}{N}}$$

holds with probability of at least $1-\delta$. □

By Theorem 3.4,

$$L_{\boldsymbol{X}}(\boldsymbol{f}^\star)=\frac{1}{N}\min_{\boldsymbol{\Pi}}\sum_{n=1}^{N}\left\|\boldsymbol{\Pi}\boldsymbol{M}^\star-\boldsymbol{M}^\natural\right\|_{\mathrm{F}}^2\leq 4N\epsilon(M,\delta)^2+\frac{2}{N}K^2(1+8\sigma_{\max}^2(\boldsymbol{V}^\natural))\epsilon'(M)$$

holds with probability of at least $1-\delta-K^2/M^{0.25}$. Invoking Lemma C.1 gives the conclusion,

$$\begin{aligned}\mathop{\mathbb{E}}_{(\boldsymbol{x},\boldsymbol{m}^\natural)\sim\mathcal{P}_{\mathcal{X},\boldsymbol{f}^\natural}}\left[\left\|\boldsymbol{\Pi}^\star\boldsymbol{f}^\star(\boldsymbol{x})-\boldsymbol{m}^\natural\right\|^2\right]&\leq 4N\epsilon(M,\delta)^2+\frac{2}{N}K^2(1+8\sigma_{\max}^2(\boldsymbol{V}^\natural))\epsilon'(M)\\&\quad+\frac{8}{N}+\frac{12\sqrt{2R_{\text{NET}}}\left\|\boldsymbol{X}\right\|_{\mathrm{F}}\log N}{N\log 2}+8\sqrt{\frac{2\log(4/\delta)}{N}},\end{aligned}$$

hold with probability of at least $1-\delta-K^2/M^{0.25}$.

D. Proof of Theorem 3.7

As $\max\left(\log(1/\alpha), \log(M)\sqrt{R_{\texttt{net}}}/\alpha\right)/\sqrt{M} \to 0$, $\left\|\boldsymbol{P}^\star - \boldsymbol{P}^\natural\right\|_{\mathrm{F}}^2 \to 0$ accords to Lemma 3.3, which means

$$(\boldsymbol{M}^\star)^\top \boldsymbol{M}^\star = \boldsymbol{P}^\natural.$$

This gives a guarantee that there exists permutation matrix $\boldsymbol{\Pi}$ such that $\boldsymbol{\Pi}\boldsymbol{M}^\star = \boldsymbol{M}^\natural$ by invoking (Huang et al., 2014, Theorem 4). Lastly, applying Lemma C.1 concludes the proof.

E. Proof of Theorem 4.1

Invoking Lemma 3.3 and considering $\max\left(\log(1/\alpha), \log(M)\sqrt{R_{\texttt{net}}}/\alpha\right)/\sqrt{M} \to 0$ gives

$$\boldsymbol{P}^\star = (\boldsymbol{M}^\star)^\top (\boldsymbol{A}^\star)^\top \boldsymbol{A}^\star \boldsymbol{M}^\star = \boldsymbol{P}^\natural$$

at the limit. As assumed in Theorem 4.1, $\mathrm{rank}((\boldsymbol{M}^\natural)^\top(\boldsymbol{A})^\top\boldsymbol{A}) = K$, thus $\mathrm{rank}(\boldsymbol{M}^\star) = K$. As proved in (Huang et al., 2016), $\boldsymbol{M}^\natural$ satisfying SSC means $\mathrm{rank}(\boldsymbol{M}^\natural) = K$, which also means $\mathrm{rank}((\boldsymbol{M}^\natural)^\top(\boldsymbol{A})^\top\boldsymbol{A}) = K$ if $\mathrm{rank}(\boldsymbol{A}) = K$. It implies that there exists an invertible matrix $\boldsymbol{Q}$ such that $\boldsymbol{M}^\star = \boldsymbol{Q}^{-1}\boldsymbol{M}^\natural$. To proceed, we use the following proposition which is a part of the proof of Theorem 1 in (Fu et al., 2015b).

Proposition E.1. *For $K \leq N$, let $\boldsymbol{M} \in \mathbb{R}^{K\times N}$ be a matrix of rank K and $\boldsymbol{M} \geq 0, \mathbf{1}^\top\boldsymbol{M} = \mathbf{1}$. For any invertible matrix $\boldsymbol{Q}$ such that $\widehat{\boldsymbol{M}} \triangleq \boldsymbol{Q}^{-1}\boldsymbol{M}$ satisfies $\widehat{\boldsymbol{M}} \geq 0, \mathbf{1}^\top\widehat{\boldsymbol{M}} = \mathbf{1}^\top$, then $|\det \boldsymbol{Q}| \geq 1$. In addition, the equality holds if and only if $\boldsymbol{Q}$ is a permutation matrix.*

By Proposition E.1, $|\det(\boldsymbol{Q})| \geq 1$, and hence

$$\det(\boldsymbol{M}^\star(\boldsymbol{M}^\star)^\top) = \det(\boldsymbol{Q}^{-1}\boldsymbol{M}^\natural(\boldsymbol{M}^\natural)^\top\boldsymbol{Q}^{-\top}) = \det(\boldsymbol{Q})^{-2}\det\left(\boldsymbol{M}^\natural(\boldsymbol{M}^\natural)^\top\right) \leq \det\left(\boldsymbol{M}^\natural(\boldsymbol{M}^\natural)^\top\right). \tag{47}$$

As we take the solution of (12) with the $\boldsymbol{M}^\star$ that has the largest volume, the equality in (47) is attained. Therefore, $\boldsymbol{Q}$ can only be a permutation matrix. The rest follows by applying Lemma C.1.

F. Additional Experiments

F.1. Effect of Neural Network Complexity

In this section, we present some additional experiments. To be specific, we repeat experiments in Tables 1 and 2 using a more complex neural network for all the DCC methods. The purpose is to observe the effect of R_{NET}, i.e., the complexity of the neural network. Here, all the DCC methods use a three-hidden-layer fully connected neural networks, where each hidden layer has 512 ReLU activation functions. In the main text, the DCC methods used a two-hidden layer network, also with 512 activation functions in each layer.

Tables 5 and 6 show the new results. Overall, similar results as in the main text can be seen: `VolMaxDCC` outperforms all other baselines, especially when noise level is getting larger. In addition, `VolMaxDCC` enjoys a better clustering performance when using a more complex (and thus more expressive) neural network on STL10. When, the noise level is 15%, the new ACC of `VolMaxDCC` is (0.85,0.86), which is much higher than the previous case that used a two-hidden layer neural network (0.79,0.81). On CIFAR-10, some slight decrease of clustering accuracy occasionally appears. This may also suggest that using a deeper neural network may not always be the best choice. As implied in our theorems, a more complex neural network increases $R_{\texttt{NET}}$, which may increase the risk of overfitting—especially when N is not very large. In practice, the neural network structure may be selected following standard procedures, e.g., using validation sets, if available.

Table 5: Clustering performance of (seen data, unseen data) on STL10; $N_{\text{unseen}} = 2000$.

Noise level \ Methods		DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.89, 0.87	0.88, 0.87	0.88, 0.86	**0.92**, **0.91**
	NMI	0.83, 0.80	0.79, 0.78	0.79, 0.76	**0.84**, **0.82**
	ARI	0.80, 0.78	0.77, 0.76	0.82, 0.80	**0.84**, **0.83**
8.3%	ACC	0.78, 0.79	0.77, 0.79	0.76, 0.77	**0.85**, **0.86**
	NMI	0.68, 0.70	0.67, 0.69	0.56, 0.59	**0.73**, **0.75**
	ARI	0.59, 0.61	0.59, 0.61	0.66, 0.68	**0.77**, **0.78**
10.3%	ACC	0.72, 0.74	0.69, 0.70	0.69, 0.71	**0.84**, **0.85**
	NMI	0.63, 0.65	0.58, 0.60	0.47, 0.49	**0.72**, **0.73**
	ARI	0.51, 0.53	0.49, 0.51	0.60, 0.61	**0.77**, **0.78**
15.0%	ACC	0.59, 0.59	0.56, 0.57	0.54, 0.55	**0.85**, **0.86**
	NMI	0.52, 0.53	0.49, 0.50	0.33, 0.34	**0.72**, **0.74**
	ARI	0.36, 0.37	0.37, 0.38	0.48, 0.49	**0.77**, **0.78**

Table 6: Clustering performance of (seen data, unseen data) on CIFAR10; $N_{\text{unseen}} = 45000$.

Noise level \ Methods		DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.90, 0.89	0.88, 0.87	0.89, 0.87	**0.91**, **0.90**
	NMI	**0.83**, **0.80**	0.81, 0.79	0.80, 0.77	**0.83**, **0.80**
	ARI	0.82, 0.79	0.79, 0.76	0.82, 0.79	**0.84**, **0.82**
4.9%	ACC	**0.86**, **0.86**	0.85, **0.86**	0.85, **0.86**	**0.86**, **0.86**
	NMI	**0.77**, **0.77**	**0.77**, **0.77**	0.73, 0.73	0.73, 0.74
	ARI	0.73, 0.73	0.73, 0.74	**0.77**, **0.77**	**0.77**, **0.77**
8.7%	ACC	0.78, 0.78	0.76, 0.77	0.76, 0.77	**0.81**, **0.81**
	NMI	**0.72**, **0.72**	0.70, 0.70	0.58, 0.58	0.65, 0.66
	ARI	0.59, 0.60	0.59, 0.59	0.70, 0.70	**0.73**, **0.73**
10.9%	ACC	0.70, 0.71	0.74, 0.75	0.67, 0.68	**0.81**, **0.81**
	NMI	0.64, 0.65	**0.66**, **0.66**	0.47, 0.48	0.65, **0.66**
	ARI	0.50, 0.51	0.56, 0.57	0.61, 0.61	**0.74**, **0.74**

F.2. Effect of Pretrained Features

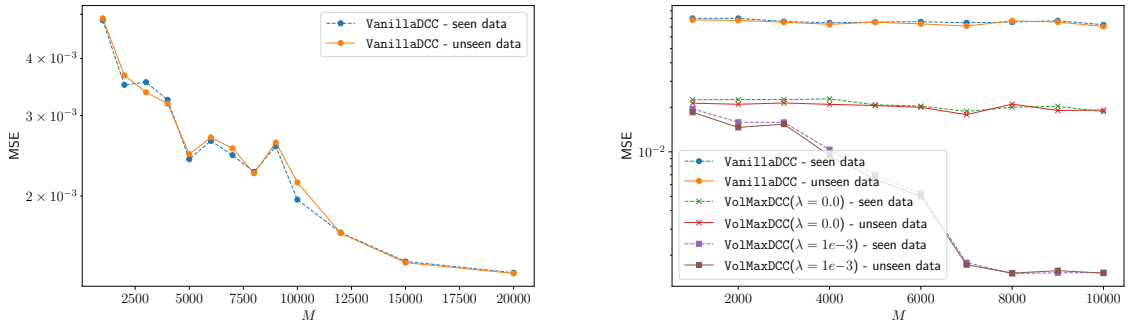
To see the effect of pre-training, we re-run the methods with feature vectors output by less well-trained feature extractors by (Li et al., 2021b). To be specific, here, the extractor is trained using 600 epochs, instead of 1000 epochs as in the main text. Nonetheless, similar results are observed in Table 7.

Table 7: Clustering performance of (seen data, unseen data) on ImageNet10; $N_{\text{unseen}} = 2000$; embedding vectors are extracted by training (Li et al., 2021b) for 600 epochs.

Noise level \ Methods		K-means	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.83, —	0.79±0.06, —	0.74±0.07, —	**0.97±0.00**, **0.96±0.00**	0.96±0.00, **0.96±0.00**	0.96±0.00, **0.96±0.00**	**0.97±0.00**, **0.96±0.00**
	NMI	0.78, —	0.76±0.03, —	0.76±0.03, —	0.92±0.00, 0.90±0.01	0.91±0.00, **0.91±0.00**	**0.93±0.00**, **0.91±0.00**	**0.93±0.00**, **0.91±0.00**
	ARI	0.66, —	0.66±0.06, —	0.59±0.10, —	**0.93±0.00**, 0.90±0.01	0.92±0.00, **0.91±0.00**	0.92±0.00, **0.91±0.00**	0.92±0.01, **0.91±0.00**
4.4%	ACC	—	0.81±0.03, —	0.66±0.05, —	0.91±0.00, 0.91±0.00	**0.94±0.01**, **0.94±0.01**	0.89±0.01, 0.90±0.01	0.92±0.02, 0.92±0.02
	NMI	—	0.75±0.01, —	0.73±0.02, —	0.85±0.01, 0.86±0.01	**0.89±0.01**, **0.89±0.01**	0.80±0.00, 0.80±0.01	0.85±0.02, 0.85±0.02
	ARI	—	0.66±0.03, —	0.54±0.07, —	0.82±0.01, 0.82±0.01	**0.88±0.02**, **0.87±0.02**	0.83±0.01, 0.84±0.01	0.87±0.01, **0.87±0.01**
5.8%	ACC	—	0.80±0.03, —	0.73±0.07, —	0.85±0.03, 0.85±0.04	**0.92±0.01**, **0.92±0.01**	0.81±0.00, 0.81±0.00	**0.92±0.01**, **0.92±0.01**
	NMI	—	0.74±0.01, —	0.76±0.04, —	0.81±0.01, 0.82±0.01	**0.86±0.01**, **0.87±0.00**	0.72±0.01, 0.72±0.01	0.84±0.01, 0.85±0.02
	ARI	—	0.65±0.03, —	0.60±0.06, —	0.75±0.03, 0.75±0.03	0.84±0.01, 0.84±0.01	0.79±0.01, 0.79±0.01	**0.86±0.01**, **0.87±0.01**

F.3. Validating Identifiability Claims using Synthetic Data

We generate synthetic data with $N = 2000, K = 3$. The first 1000 data points act as the seen samples, and the rest act as unseen data. In order to generate valid membership $\boldsymbol{M}^\natural$, we sample N random unit vectors, followed by adding i.i.d element-wise Gaussian noise with mean of 0 and variance of 0.1. The resulting matrix is then truncated to ensure that its elements are nonnegative. Lastly, the columns of $\boldsymbol{M}^\natural$ are normalized using their respective ℓ_1-norms to become valid probability mass functions. For constructing feature vectors $\boldsymbol{x}^\natural$ corresponding to membership $\boldsymbol{m}^\natural$, we choose a simple nonlinear function $\boldsymbol{f}$



(a) Average MSE of `VanillaDCC` of 5 random trials against different M's.

(b) Median MSE of 10 random trials against different M's.

who is defined through its inverse, i.e., $\boldsymbol{f}^{-1}$ defined as: $\boldsymbol{f}^{-1}(\boldsymbol{m}^\natural) \triangleq [2m_1^\natural, 3m_2^\natural + 1, m_1^\natural m_2^\natural + m_3^\natural - 2]^\top$. Such construction guarantees the existence of a function mapping from feature vector $\boldsymbol{x} \in \mathcal{X}$ to its corresponding membership vector $\boldsymbol{m}$.

To acquire the pairwise annotations, we sample M pairs of indices uniformly. The pair similarity labels are then drawn from the Bernoulli distribution as described in Section 3.1. The performance of estimated membership $\widehat{\boldsymbol{M}}$ is measured using the *mean squared error* (MSE), which is defined as

$$\text{MSE}(\widehat{\boldsymbol{M}}, \boldsymbol{M}^\natural) = \frac{1}{K} \min_{\boldsymbol{\Pi}} \sum_{k=1}^{K} \left\| \frac{\boldsymbol{\Pi}\boldsymbol{M}^\natural(k,:)}{\|\boldsymbol{\Pi}\boldsymbol{M}^\natural(k,:)\|} - \frac{\widehat{\boldsymbol{M}}(k,:)}{\|\widehat{\boldsymbol{M}}(k,:)\|} \right\|^2, \quad \boldsymbol{\Pi} \text{ is permutation matrix.}$$

Fig. 3a shows the median of the MSEs on the training and test sets of the `VanillaDCC` over 5 random trials. One can see the MSE is fairly low and decreases as M increases—which is consistent with our theory.

To simulate noisy pairwise constraints, we keep the above setting and include a confusion matrix $\boldsymbol{A}$ as follows

$$\boldsymbol{A} = \begin{bmatrix} 1.0 & 0.2 & 0.3 \\ 0.0 & 0.8 & 0.3 \\ 0.0 & 0.0 & 0.4 \end{bmatrix}.$$

Fig. 3b shows median MSE over 10 trials on seen and unseen data of `VanillaDCC` and `VolMaxDCC` using different number of pairwise constraints. One can see that using $\lambda > 0$, i.e., by promoting maximal volume of solutions, the MSE performance is much better, in the presence of noisy annotations. This shows the usefulness of the volume regularization in enhancing membership identifiability.

F.4. Results with Standard Deviation

In Tables 1, 2, 3, we only reported the average to avoid too-dense looking table. We report those results here with standard deviation in Table 8,9,10, respectively.

Table 8: Clustering performance of (seen data, unseen data) on STL10; $N_{\text{unseen}} = 2000$.

Noise level	Methods	Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	`VanillaDCC`	`VolMaxDCC`
0.0%	ACC	0.71±0.03, —	0.66±0.04 , —	0.70±0.08, —	0.88 ± 0.04, 0.87± 0.04	0.89±0.01, 0.88±0.02	**0.93±0.00**, **0.91±0.00**	0.91±0.03, 0.89±0.03
	NMI	0.75±0.01 , —	0.67±0.02 , —	0.72±0.02, —	0.82±0.02, 0.80±0.02	0.81±0.01, 0.80±0.02	**0.84±0.01**, **0.81±0.01**	0.82±0.03, 0.80±0.03
	ARI	0.54±0.03 , —	0.52±0.03 , —	0.55±0.06, —	0.80±0.04, 0.78±0.03	0.79±0.02, 0.77±0.02	**0.85±0.00**, **0.83±0.00**	0.84±0.01, 0.82±0.01
8.3%	ACC	—	0.70±0.04, —	0.70±0.04, —	0.75±0.03, 0.76±0.03	0.77±0.01, 0.79±0.01	0.78±0.00, 0.80±0.01	**0.80±0.01**, **0.81±0.00**
	NMI	—	0.64±0.03, —	**0.71**±0.02, —	0.67±0.01, **0.69±0.01**	0.67±0.01, **0.69±0.01**	0.59±0.00, 0.62±0.01	0.64±0.02, 0.65±0.02
	ARI	—	0.51±0.03, —	0.56±0.04, —	0.57±0.02, 0.59±0.02	0.59±0.02, 0.61±0.01	0.69±0.00, 0.71±0.01	**0.73±0.02**, **0.74±0.02**
10.3%	ACC	—	0.62±0.05, —	0.69±0.04, —	0.70±0.02, 0.72±0.03	0.70±0.03, 0.71±0.03	0.72±0.01, 0.73±0.01	**0.79±0.00**, **0.81±0.00**
	NMI	—	0.59±0.02, —	**0.73**±0.01, —	0.62±0.02, 0.64±0.02	0.60±0.02, 0.62±0.02	0.50±0.01, 0.51±0.01	0.68±0.00, **0.70±0.00**
	ARI	—	0.44±0.03, —	0.55±0.02, —	0.51±0.01, 0.52±0.02	0.50±0.03, 0.52±0.03	0.62±0.00, 0.64±0.01	**0.77±0.01**, **0.78±0.00**
15.0%	ACC	—	0.62±0.05, —	0.64±0.07, —	0.60±0.02, 0.61±0.02	0.57±0.01, 0.57±0.01	0.56±0.03, 0.58±0.03	**0.79±0.00**, **0.81±0.00**
	NMI	—	0.54±0.01, —	**0.72**±0.01, —	0.54±0.01, 0.55±0.02	0.50±0.02, 0.50±0.02	0.33±0.01, 0.35±0.01	0.68±0.01, **0.69±0.01**
	ARI	—	0.41±0.02, —	0.52±0.05, —	0.38±0.02, 0.39±0.01	0.38±0.01, 0.39±0.02	0.50±0.01, 0.51±0.01	**0.76±0.02**, **0.77±0.02**

Table 9: Clustering performance of (seen data, unseen data) on CIFAR10; $N_{\text{unseen}} = 45000$.

Noise level \ Methods		Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.78±0.00 , —	0.67±0.06, —	0.67±0.06, —	0.91±0.01, 0.89±0.01	0.90±0.01, 0.89±0.01	**0.92±0.00**, **0.90±0.00**	0.91±0.01, **0.90±0.00**
	NMI	0.71±0.00 , —	0.66±0.02, —	0.71±0.02, —	0.83±0.01, **0.81±0.01**	0.83±0.01, **0.81±0.01**	**0.84±0.00**, 0.80±0.00	0.83±0.01, 0.80±0.01
	ARI	0.62±0.00 , —	0.54±0.05, —	0.55±0.05, —	0.82±0.01, 0.79±0.01	0.81±0.01, 0.79±0.01	**0.85±0.00**, 0.81±0.00	0.84±0.00, **0.82±0.00**
4.9%	ACC	—	0.75±0.04, —	0.70±0.06, —	**0.86±0.00**, **0.86±0.00**	**0.86±0.00**, **0.86±0.00**	**0.86±0.00**, **0.86±0.00**	**0.86±0.00**, **0.86±0.00**
	NMI	—	0.69±0.01, —	0.69±0.02, —	**0.77±0.00**, **0.77±0.00**	**0.77±0.00**, **0.77±0.00**	0.73±0.00, 0.73±0.00	0.73±0.00, 0.74±0.00
	ARI	—	0.60±0.03, —	0.57±0.04, —	0.73±0.00, 0.74±0.00	0.73±0.00, 0.74±0.00	**0.77±0.00**, **0.77±0.00**	**0.77±0.00**, **0.77±0.00**
8.7%	ACC	—	0.64±0.06, —	0.72±0.06, —	0.76±0.04, 0.76±0.04	0.76±0.02, 0.76±0.02	0.76±0.02, 0.77±0.02	**0.83±0.01**, **0.83±0.01**
	NMI	—	0.63±0.02, —	0.69±0.02, —	**0.71±0.02**, **0.71±0.02**	0.70±0.01, 0.70±0.01	0.58±0.01, 0.59±0.01	0.70±0.01, 0.70±0.01
	ARI	—	0.49±0.04, —	0.59±0.03, —	0.58±0.03, 0.59±0.03	0.59±0.01, 0.60±0.01	0.70±0.01, 0.70±0.01	**0.75±0.01**, **0.75±0.01**
10.9%	ACC	—	0.68±0.06, —	0.70±0.04, —	0.74±0.03, 0.74±0.02	0.73±0.02, 0.73±0.02	0.68±0.00, 0.69±0.01	**0.82±0.01**, **0.82±0.01**
	NMI	—	0.61±0.03, —	**0.68**±0.01, —	0.67±0.02, **0.68±0.02**	0.65±0.01, 0.66±0.01	0.48±0.01, 0.49±0.01	**0.68±0.02**, **0.68±0.02**
	ARI	—	0.50±0.05, —	0.57±0.03, —	0.55±0.03, 0.55±0.03	0.56±0.03, 0.57±0.03	0.62±0.01, 0.63±0.01	**0.74±0.01**, **0.74±0.01**

Table 10: Clustering performance of (seen data, unseen data) on ImageNet10; $N_{\text{unseen}} = 2000$.

Noise level \ Methods		Kmeans	COP-Kmeans	PCKmeans	DC-GMM	C-IDEC	VanillaDCC	VolMaxDCC
0.0%	ACC	0.85±0.00 , —	0.79±0.06, —	0.70±0.06, —	**0.97±0.00**, **0.96±0.00**	**0.97±0.00**, **0.96±0.00**	**0.97±0.00**, **0.96±0.00**	**0.97±0.00**, **0.96±0.00**
	NMI	0.80±0.00, —	0.77±0.03, —	0.75±0.04, —	0.93±0.00, 0.91±0.00	0.93±0.00, **0.92±0.00**	**0.94±0.00**, 0.91±0.00	**0.94±0.00**, 0.91±0.00
	ARI	0.68±0.00, —	0.66±0.05, —	0.55±0.07, —	**0.94±0.00**, **0.92±0.00**	**0.94±0.00**, **0.92±0.00**	0.93±0.00, 0.91±0.00	0.93±0.00, 0.91±0.00
3.4%	ACC	—	0.79±0.09, —	0.66±0.12, —	0.93±0.01, 0.92±0.01	0.93±0.00, 0.93±0.00	0.92±0.01, 0.91±0.01	**0.94±0.01**, **0.94±0.01**
	NMI	—	0.75±0.04, —	0.73±0.05, —	0.86±0.01, 0.85±0.01	0.87±0.01, 0.86±0.01	0.83±0.02, 0.82±0.02	**0.88±0.01**, **0.87±0.02**
	ARI	—	0.65±0.07, —	0.52±0.12, —	0.84±0.01, 0.83±0.01	0.86±0.01, 0.85±0.01	0.84±0.01, 0.84±0.01	**0.89±0.01**, **0.88±0.01**
6.9%	ACC	—	0.74±0.07, —	0.72±0.08, —	0.84±0.01, 0.84±0.01	0.88±0.01, 0.88±0.01	0.84±0.00, 0.84±0.00	**0.92±0.00**, **0.91±0.00**
	NMI	—	0.70±0.03, —	0.76±0.04, —	0.79±0.01, 0.79±0.00	0.82±0.01, 0.82±0.01	0.70±0.01, 0.70±0.00	**0.84±0.00**, **0.83±0.00**
	ARI	—	0.58±0.05, —	0.59±0.09, —	0.71±0.01, 0.71±0.01	0.77±0.02, 0.77±0.02	0.77±0.01, 0.77±0.01	**0.88±0.00**, **0.87±0.00**
11.2%	ACC	—	0.72±0.05, —	0.62±0.07, —	0.71±0.03, 0.72±0.03	0.80±0.02, 0.81±0.02	0.65±0.01, 0.66±0.01	**0.91±0.01**, **0.90±0.01**
	NMI	—	0.64±0.02, —	0.73±0.03, —	0.68±0.01, 0.70±0.01	0.74±0.02, 0.76±0.02	0.49±0.03, 0.52±0.02	**0.83±0.01**, **0.82±0.01**
	ARI	—	0.54±0.03, —	0.51±0.09, —	0.56±0.02, 0.58±0.02	0.66±0.03, 0.68±0.03	0.62±0.03, 0.63±0.02	**0.87±0.01**, **0.86±0.00**