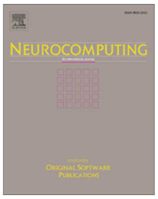
Contents lists available at ScienceDirect

Neurocomputing

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



Regression-based clustering network via combining prior information

Wei Xia [a], Quanxue Gao [a,*], Qianqian Wang [a], Xinbo Gao [b,c]

[a] State Key Laboratory of Integrated Services Networks, Xidian University, Shaanxi 710071, China
[b] School of Electronic Engineering, Xidian University, Shaanxi 710071, China
[c] Chongqing Key Laboratory of Image Cognition, Chongqing University of Posts and Telecommunications, Chongqing 400065, China

ARTICLE INFO

Article history:
Received 20 July 2020
Revised 13 January 2021
Accepted 11 March 2021
Available online 18 March 2021
Communicated by Zidong Wang

Keywords:
Clustering
Unsupervised learning
Adversarial learning

ABSTRACT

Despite the promising performance, existing regression-based clustering methods still have the following limitations. (1) They only extract the shallow discriminant features, resulting in unstable clustering performance on data with complex underlying subspaces. (2) It is difficult to optimize the objective due to the discretization of the elements in the cluster indicator matrix, resulting in suboptimal solution. (3) They fail to employ the structure prior information embedded in the clustering label matrix, resulting in suboptimal clustering performance. Targeting at above problems, we propose a novel **R**egression-based **C**lustering network via **C**ombining **P**rior **I**nformation (RC2PI), which consists of a convolutional auto-encoder, a priori information encoder, and a discriminator. Specifically, the auto-encoder is used to generate the ideal distribution to relax discrete cluster indicator matrix, which can help obtain optimal solution. The prior information encoder is employed to exploit the structure prior knowledge embedded in clustering label matrix, thereby boosting clustering via a self-supervised manner. The discriminator, as a connector of the above two sub-networks, is used for verifying the embedding process of prior information that will guide the auto-encoder to generate a more reliable actual distribution. Extensive experiments demonstrate the effectiveness of RC2PI over state-of-the-art methods.

1. Introduction

Linear regression, due to its super effectiveness of processing high dimensional data, has been successfully applied in many classic supervised learning areas, *e.g.*, object classification [1] and face recognition [2]. Nevertheless, there are few studies to utilize the property of regression to do clustering analysis. In practice, many high dimensional data may exhibit dense grouping in a low dimensional subspace, and the true cluster indicator matrix of high dimensional data can be always embedded in a low dimensional mapping of the data [3,11]. Thus, regression helps to guide the partitioning process by modeling the dissimilarity of each cluster in the low dimensional subspace.

To take advantage of this property, Han *et al.* [4] developed a local and global discriminative framework for balanced clustering (LGBC) via minimizing distribution entropy and the least-squares regression between cluster indicator matrix and low dimensional features. However, LGBC uses the continuous low-dimensional feature to approximate discrete cluster indicator matrix, resulting in suboptimal solution. To this end, Nie *et al.* [5] proposed spectral embedded clustering (SEC) to reduce the divergence between the cluster indicator matrix and the latent features of the data. SEC takes into account relaxing the discrete cluster indicator matrix, but keep the orthogonality intact. Similarly, Gao *et al.* [6] proposed a robust regression-based clustering method to tackle cancer genome data. However, a vital constraint is ignored by [5,6], *i.e.*, all the elements of the cluster indicator matrix should be nonnegative by definition.

Although aforementioned regression-based clustering methods provide impressive results, they still have several limitations. 1) They suffer from the fact that the model is hard to solve due to the discretization of the elements in the cluster indicator matrix. 2) They only extract the shallow discriminant features. 3) They require a postprocessing, *e.g.*, K-Means, to get the final clustering results, which increases the instability of the original performance. Recently, numerous deep neural networks (DNNs) based clustering methods [7–10] have been presented and achieved impressive clustering performance in extensive experiments. Based on the multi-layer stacked auto-encoder, these deep clustering models integrate representation learning and clustering into a unified framework, which helps to further learn better latent representations for clustering. Nonetheless, rare studies have been found to introduce the property of regression to deep clustering models.

[*] Corresponding author.
E-mail address: qxgao@xidian.edu.cn (Q. Gao).

https://doi.org/10.1016/j.neucom.2021.03.031
0925-2312/© 2021 Elsevier B.V. All rights reserved.

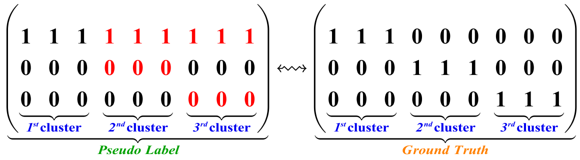


Fig. 1. Illustration of prior pseudo label information.

In addition, the aforementioned methods ignore the prior information of label matrix. Although the label matrix is unknown, it is formed by binary one and zero elements. Given an $N \times K$ pseudo label matrix, each row only has one element 1, while the others are 0, *i.e.*, the number of one and zero elements are N and $N \times (K-1)$, respectively, where N and K are the number of samples and clusters, respectively. Here, the position of element 1 indicates its cluster. Even though we do not know where element 1 is located, this structural prior knowledge of the unknown pseudo label matrix is still useful. For example, as shown in Fig. 1, on the right is the true one-hot label matrix, and a column of the matrix represents the label vector of one sample. For ease of illustration, suppose there are three clusters, each cluster has three samples. We can easily get the pseudo label matrix on the left, assuming all samples belong to the 1-st cluster. Now, for the samples of 1-st cluster, we get its true label. Only two elements are wrong in the label vector for other samples (See the red elements on the left in Fig. 1). More important, we can get the clustering label in each iteration of the algorithm, the information embedded in such pseudo label is useful. ***However, to our best knowledge, similar investigations for regression-based clustering have been found lacking so far, which is one of the motivations behind this work.***

Inspired by the above insight analysis, we propose a novel deep clustering model, namely regression-based clustering network via combining prior information (RC2PI). It incorporates all above concerns into a unified framework. We highlight the contribution of this work as the following.

- RC2PI uses an auto-encoder to generate the continuous ideal distribution to relax the discrete cluster indicator matrix, which can help to obtain a more satisfactory solution.
- RC2PI employs a prior information encoding network to take advantage of the structure prior knowledge embedded in pseudo label matrix in clustering tasks, and demonstrate the superior results over previous works.
- A discriminator is used to verify the embedding process of prior information that will guide the auto-encoder to generate the more reliable actual distribution.
- Compared with existing regression-based clustering methods, RC2PI can directly obtain the clustering label without extra postprocessing, which improves the stability of the model.

2. Methodology

2.1. Problem formulation

Regression-based clustering [11] is one of the most representative clustering methods. The objective is

$$\min_{\mathbf{W},\mathbf{b},\mathbf{F}} \|\mathbf{W}^T\mathbf{X} + \mathbf{1b} - \mathbf{F}\|_F^2 + \xi\|\mathbf{W}\|_F^2 + R(\mathbf{F}), \tag{1}$$

where $\mathbf{X} \in \mathbb{R}^{d\times N}$ is the data matrix, $\mathbf{W} \in \mathbb{R}^{d\times K}$ is the projection matrix, $\mathbf{b} \in \mathbb{R}^{K}$ is the bias vector, $\mathbf{F} \in \mathbb{R}^{N\times K}$ is the clustering label. ξ is the penalty coefficient. $R(\mathbf{F})$ represents the constraint on $\mathbf{F}$. By employing different constraint, several regression-based clustering methods are proposed, *e.g.*, orthogonal constraint [6], spectral embedded constraint [5], distribution entropy [4] and so on. Problem 1 leverages hard constraint to make the continuous low-dimensional features to approximate the discrete cluster indicator matrix $\mathbf{F}$.

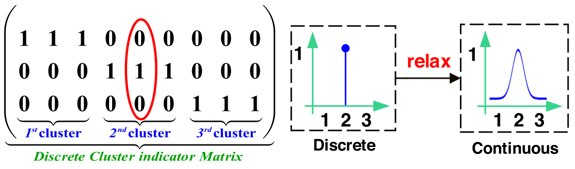


Fig. 2. Illustration of relaxing hard regression constraint.

However, discrete zero and one elements are too ideal, leading to a suboptimal solution. Although some methods usually relax the cluster indicator matrix and keep the orthogonality intact. Under this circumstance, the relaxed solution may severely deviate from the true solution and thus degrade the clustering performance. Because all the elements of the cluster indicator matrix should be nonnegative by definition. In addition, they use a post-processing, *e.g.*, K-Means, to get the final clustering results, resulting in suboptimal performance due to the uncertainty of K-Means. Moreover, they only extract the shallow discriminant features, resulting in unstable clustering performance on complex real-life data. Finally, they ignore the prior information embedded in the pseudo label matrix. This structure prior information is important for clustering.

To integrate all above concerns into one optimization framework, an appropriate continuous distribution is introduced to relax the discrete cluster indicator matrix, thereby relaxing the hard regression constraint (See Fig. 2, where discrete one and zero elements are relaxed to continuous). Meanwhile, we introduce a structure prior information encoder to embed the prior knowledge, thereby getting a robust actual distribution to make the RC2PI more effective and solid.

2.2. The framework of RC2PI

The framework is shown in Fig. 3. RC2PI consists of three components: a convolutional auto-encoder, a prior knowledge encoder, and a distribution consistency discriminator. The details of RC2PI will be described in the following.

2.2.1. Regression-based clustering

Lex $\mathbf{X} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ denote the samples with K clusters. For arbitrary data point $\mathbf{x}_i \in \mathbf{X}$, the actual distribution[1] $\mathbf{z}_i \in \mathbf{Z}^{N\times K}$ can be extracted by the mapping of a multi-layer convolutional encoder in Fig. 3. Thus, we have

$$\mathbf{Z} = \mathbf{E}(\mathbf{X}; \Theta_\mathbf{E}), \tag{2}$$

where $\mathbf{E}(\cdot)$ refers to the mapping function. $\Theta_\mathbf{E}$ is the parameters of the encoder. Hence, it is crucial to find an ideal continuous distribution $\mathcal{P}_{(\mathbf{Z})}$ as a cluster indicator matrix to approximate actual distribution $\mathbf{Z}$ for regression-based clustering.

Inspired by *t-SNE* algorithm [12], instead of measuring the similarity between data point $\mathbf{x}_i$ and $\mathbf{x}_j$, we employ *Student's t-distribution* as a mapping function to measure the similarity between $\mathbf{z}_i$ of data point $\mathbf{x}_i$ and the centroid of each cluster $\mu_j (j = 1, 2, \ldots, K)$. Hence, we can calculate the probability distribution $\mathcal{Q}_{(\mathbf{Z})} \in \mathbb{R}^{N\times K}$ by

$$q_{ij} = \frac{\left(1 + ||\mathbf{z}_i - \mu_j||^2\right)^{-1}}{\sum_{j'}\left(1 + ||\mathbf{z}_i - \mu_{j'}||^2\right)^{-1}}, \tag{3}$$

[1] In the whole text, actual distribution refers to low dimensional representation, because its dimension is $N \times K$.

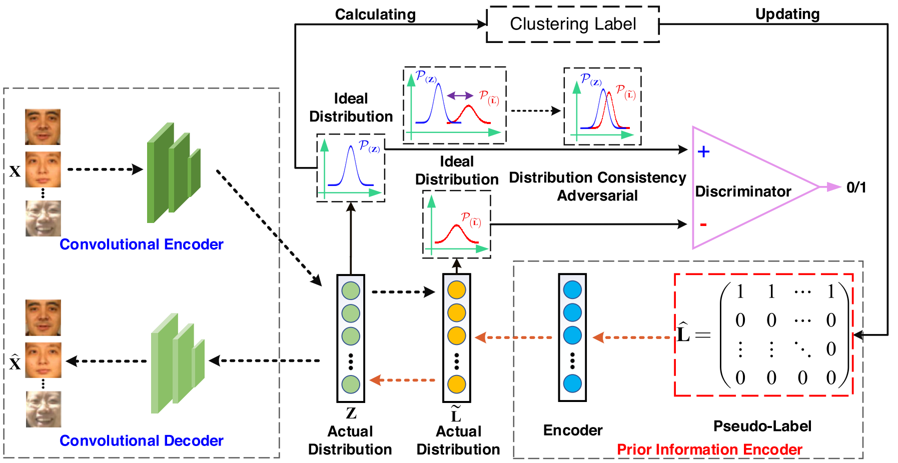

Calculating
Clustering Label
Updating
Ideal Distribution
$\mathcal{P}_{(\mathbf{Z})}$
$\mathcal{P}_{(\tilde{\mathbf{L}})}$
X
Convolutional Encoder
Ideal Distribution
Distribution Consistency Adversarial
+
-
Discriminator
0/1
$\hat{\mathbf{X}}$
Convolutional Decoder
Z
Actual Distribution
$\tilde{\mathbf{L}}$
Actual Distribution
Encoder
$\hat{\mathbf{L}} = \begin{pmatrix} 1 & 1 & \cdots & 1 \\ 0 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & 0 \\ 0 & 0 & 0 & 0 \end{pmatrix}$
Pseudo-Label
Prior Information Encoder

Fig. 3. The overall framework of RC2PI.

where $q_{ij} \in \mathcal{Q}_{(\mathbf{Z})}$ represents the probability of clustering sample i to cluster j. We introduce K-Means on actual distribution $\mathbf{Z}$ to obtain the initial cluster centroids μ_j. Motivated by [13], the highly trustworthy ideal continuous distribution $\mathscr{P}_{(\mathbf{Z})} \in \mathbb{R}^{N \times K}$ can be defined as

$$p_{ij} = \frac{q_{ij}^2 / \sum_i q_{ij}}{\sum_{j'} q_{ij'}^2 / \sum_i q_{ij'}}, \tag{4}$$

where $\sum_i q_{ij}$ is soft cluster frequency. In fact, ideal distribution $\mathscr{P}_{(\mathbf{Z})}$ is an enhancement of the probability distribution $\mathcal{Q}_{(\mathbf{Z})}$ (Transforming the high value of $\mathcal{Q}_{(\mathbf{Z})}$ becomes higher, small value becomes smaller) and concentrate more on the assigned data with high confidence.

We can **calculate the clustering labels directly from the last optimized** $\mathscr{P}_{(\mathbf{Z})}$, and the cluster estimated for sample i can be calculated by $\mathbf{s}_i = \max index(\mathbf{p}_i)$, where $\max index(\cdot)$ is set to find the index of max probability value in i-th row of $\mathscr{P}_{(\mathbf{Z})}, \mathbf{S} \in \mathbb{R}^{N \times K}$ is the clustering label matrix.

According to Eqs. (2), (4), the objective of regression-based clustering can be defined as

$$\min_{\Theta_{\mathbf{E}}} \mathscr{L}_R = \lambda_1 \|\mathbf{Z} - \mathscr{P}_{(\mathbf{Z})}\|_F^2, \tag{5}$$

where λ_1 is a tradeoff parameter.

To extract localized actual distribution $\mathbf{Z}$ while preserving spatial locality, a convolutional decoder $\mathbf{D}$ with a symmetric structure to the encoder is adopted. Each decoder layer tries to reverse the process of its corresponding encoder layer. The corresponding reconstructed sample $\hat{\mathbf{X}}$ can be represented by

$$\hat{\mathbf{X}} = \mathbf{D}(\mathbf{E}(\mathbf{X}; \Theta_{\mathbf{E}}); \Theta_{\mathbf{D}}) = \mathbf{D}(\mathbf{Z}; \Theta_{\mathbf{D}}), \tag{6}$$

where $\mathbf{D}(\cdot)$ refers to multiple decoder layers and $\Theta_{\mathbf{D}}$ is the learnable parameters of them. We utilize $\Theta_{\mathbf{Z}} = \{\Theta_{\mathbf{E}}, \Theta_{\mathbf{D}}\}$ to represent the parameter of auto-encoder. Thus, the auto-encoder reconstruction loss w.r.t. $\Theta_{\mathbf{Z}}$ can be defined as

$$\min_{\Theta_{\mathbf{Z}}} \mathscr{L}_{AE} = \frac{1}{N} \|\mathbf{X} - \hat{\mathbf{X}}\|_F^2. \tag{7}$$

In Eq. (7), we minimize the squared Frobenius norm of the difference between reconstructed samples and the raw samples to optimize the auto-encoder network in Fig. 3.

2.2.2. Structure prior information encoding

In real-world applications, the label of the sample is unknown. For a $K \times N$ label matrix, the number of one and zero elements are N and $N \times (K-1)$, respectively. This structure prior information is known and important. At the beginning, we can easily gain a initial pseudo label matrix $\hat{\mathbf{L}}$ as shown in Fig. 3, in which each column is a label vector of one sample, we initialize its first row to element 1. In this case, for the samples of the first cluster, their labels are correct, and for the samples of other clusters, there are only two elements wrong in the label. In the process of network iterative optimization, we can utilize the clustering labels to update the pseudo label matrix, thereby well exploiting the structure prior information embedded in pseudo label matrix. To take advantage of this prior information, a prior information encoding network is introduced. In particular, we utilize a multi-layer fully connected network to encode the discrete pseudo label matrix $\hat{\mathbf{L}}$, then we can gain another actual distribution $\tilde{\mathbf{L}}$, thus, we have

$$\tilde{\mathbf{L}} = \mathbf{E}_{\mathbf{L}^\sim}\left(\hat{\mathbf{L}}; \Theta_{\mathbf{L}^\sim}\right). \tag{8}$$

where $\mathbf{E}_{\tilde{\mathbf{L}}}(\cdot)$ refers to the prior knowledge encoder and $\Theta_{\tilde{\mathbf{L}}}$ is the corresponding parameters. To better embed the prior knowledge, we also constrain the mismatch between the actual distribution $\tilde{\mathbf{L}}$ and its corresponding ideal distribution $\mathscr{P}_{(\tilde{\mathbf{L}})}$, where $\mathscr{P}_{(\tilde{\mathbf{L}})}$ is calculated by Eqs. (3), (4). Thus the prior knowledge encoding loss can be defined by

$$\min_{\Theta_{\mathbf{L}^\sim}} \mathscr{L}_{PI} = \lambda_1 \left(\| \tilde{\mathbf{L}} - \mathscr{P}_{(\mathbf{L}^\sim)} \|_F^2\right), \tag{9}$$

where we set same tradeoff parameter λ_1 for Eqs. (5), (9).

Table 1
Descriptions of datasets.

Dataset	Dimensions	Cluster Number	Sample Number
MNIST-*Full*	$28 \times 28 \times 1$	10	70,000
MNIST-*test*	$28 \times 28 \times 1$	10	10,000
USPS	$16 \times 16 \times 1$	10	9,289
FRGC	$32 \times 32 \times 3$	20	2,462
YTF	$55 \times 55 \times 3$	41	10,000

Table 2
Details of the structure of the auto-encoder networks.

Dataset	encoder-1/ decoder-3	encoder-2/ decoder-2	encoder-3/ decoder-1
MNIST-*Full*	$4 \times 4 \times 50$	$5 \times 5 \times 50$	–
MNIST-*test*	$4 \times 4 \times 50$	$5 \times 5 \times 50$	–
USPS	$4 \times 4 \times 50$	$5 \times 5 \times 50$	–
FRGC	$5 \times 5 \times 50$	$5 \times 5 \times 50$	$3 \times 3 \times 50$
YTF	$5 \times 5 \times 50$	$5 \times 5 \times 50$	$3 \times 3 \times 50$

Table 3
Clustering results of various methods on five datasets. Best results are highlighted in **bold**. ⊗ means the results are unavailable from the corresponding paper or code. The data marked with ☆ in the upper right corner is obtained by running the code provided by the author. RC2PI-U means updating the pseudo label matrix by clustering labels.

Dataset	MNIST-*full*		MNIST-*test*		USPS		FRGC		YTF	
Method \ Metric	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
K-means [19]	0.534	0.500	0.547	0.501	0.460	0.450	0.243	0.287	0.560	0.752
N-Cuts [20]	0.327	0.411	0.304	0.753	0.314	0.675	0.235	0.285	0.536	0.742
SC-ST [21]	0.311	0.416	0.454	0.756	0.308	0.726	0.358	0.431	0.290	0.620
SC-LS [22]	0.714	0.706	0.740	0.756	0.659	0.681	0.407	0.550	0.544	0.759
SEC [11]	0.804	0.779	0.815	0.790	0.732☆	0.691☆	0.443☆	0.544☆	0.591☆	0.757☆
AC-PIC [23]	0.115	0.017	0.920	0.853	0.855	0.840	0.320	0.415	0.472	0.679
JULE [18]	0.964	0.913	0.961	0.915	0.950	0.913	0.461	0.574	0.684	0.848
DEC [13]	0.844	0.816	0.859	0.827	0.619	0.586	0.425☆	0.561☆	0.422☆	0.602☆
IDEC [7]	0.881	0.867	0.846	0.802	0.759	0.777	0.453☆	0.576☆	0.445☆	0.626☆
DEPICT [8]	0.965	0.917	0.963	0.915	0.964	0.927	0.470	0.610	0.621	0.802
DSC-L2 [14]	0.715☆	0.704☆	0.717☆	0.700☆	0.689☆	0.735☆	0.429☆	0.512☆	0.568☆	0.741☆
SpectralNet [24]	0.971	0.924	0.773	0.760	⊗	⊗	⊗	⊗	0.685	0.798
DPSC [25]	0.797	0.823	0.806	0.791	⊗	⊗	⊗	⊗	⊗	⊗
DSCDAN [10]	0.978	0.941	0.980	0.946	0.869	0.857	0.356☆	0.519☆	0.691	0.857
RC2PI	0.980	0.968	0.981	0.967	0.969	0.965	0.484	0.631	0.735	0.865
RC2PI-U	**0.990**	**0.985**	**0.988**	**0.980**	**0.969**	**0.985**	**0.508**	**0.633**	**0.755**	**0.891**

2.2.3. Distribution consistency adversarial

How to ensure the actual distribution $\tilde{\mathbf{L}}$, learned via updating the prior information encoding networks, is effective? To this end, we not only control the mismatch between actual distributions $\mathbf{Z}$ and $\mathbf{L}$, but also control the mismatch between the ideal distributions $\mathscr{P}_{(\mathbf{Z})}$ and $\mathscr{P}_{(\tilde{\mathbf{L}})}$. Thus, the distribution consistency loss can be defined as

$$\min_{\Theta_{\mathbf{Z}},\Theta_{\mathbf{L}^{\sim}}} \mathscr{L}_d = \lambda_2\left(\| \tilde{\mathbf{L}} - \mathbf{Z}\|_F^2 + \|\mathscr{P}_{(\mathbf{L}^{\sim})} - \mathscr{P}_{(\mathbf{Z})}\|_F^2\right), \tag{10}$$

where parameter λ_2 is set to balance other constraint terms.

When optimizing problem (10), due to the scale issue of $\mathscr{P}_{(\mathbf{Z})}$ and $\mathscr{P}_{(\tilde{\mathbf{L}})}$,[2] the single distribution consistency loss $\mathscr{L}_d$ may degrade the performance. To assure the diversity of two distributions, a distribution consistency discriminator $\mathscr{D}$ is adopted to complement $\mathscr{L}_d$. The discriminator $\mathscr{D}$ consists of a three-layer fully connected network. The goal of the $\mathscr{D}$ is to distinguish continuous distributions $\mathscr{P}_{(\mathbf{Z})}$ (real) and $\mathscr{P}_{(\mathbf{L})}$ (fake), and subsequently better utilize the prior information. Thus, the adversarial learning loss is introduced as

$$\min_{\Theta_{\mathbf{Z}},\Theta_{\tilde{\mathbf{L}}}}\max_{\Omega}\mathscr{L}_a = \tfrac{1}{2}\mathbb{E}\left[\log\left(\mathscr{D}(\mathscr{P}_{(\mathbf{Z})})\right)\right] + \tfrac{1}{2}\mathbb{E}\left[\log\left(1-\mathscr{D}\left(\mathscr{P}_{(\tilde{\mathbf{L}})}\right)\right)\right], \tag{11}$$

where Ω is the parameters of discriminator $\mathscr{D}$. For $\mathscr{D}$, we hope it can distinguish that $\mathscr{P}_{(\mathbf{Z})}$ is the real ideal distribution, and $\mathscr{P}_{(\tilde{\mathbf{L}})}$ is the fake ideal distribution. We minimize the adversarial loss to update the parameters of the convolutional auto-encoder and prior information encoding network until two distributions get similar, which also shows that the auto-encoder network has learned a satisfactory actual distribution $\mathbf{Z}$ with reasonable prior information. Thus, we rewrite the model (10) as

$$\min_{\Theta_{\mathbf{Z}},\Theta_{\tilde{\mathbf{L}}}}\max_{\Omega}\mathscr{L}_d = \lambda_2\left(\|\mathbf{L}-\mathbf{Z}\|_F^2 + \|\mathscr{P}_{(\tilde{\mathbf{L}})} - \mathscr{P}_{(\mathbf{Z})}\|_F^2\right) + \mathscr{L}_a. \tag{12}$$

By feeding back such supervision to the front, the auto-encoder and prior knowledge encoder is then enforced to maximize the discriminator loss, leading to better actual distribution $\mathbf{Z}$ and clustering performances. We integrate the above three components into a unified framework. Therefore, the overall objective can be written as

$$\min_{\Theta_{\mathbf{Z}},\Theta_{\mathbf{L}^{\sim}}}\max_{\Omega}\mathscr{L}_{total} = \mathscr{L}_{AE}+\mathscr{L}_R+\mathscr{L}_{PI}+\mathscr{L}_d+\delta\|\mathbf{H}\|_2^2. \tag{13}$$

where, $\|\mathbf{H}\|_2^2$ is setting to avoid overfitting, $\mathbf{H}$ represents the weights of all three sub-networks in our framework, and δ is a trade-off parameters.

2.3. Optimization

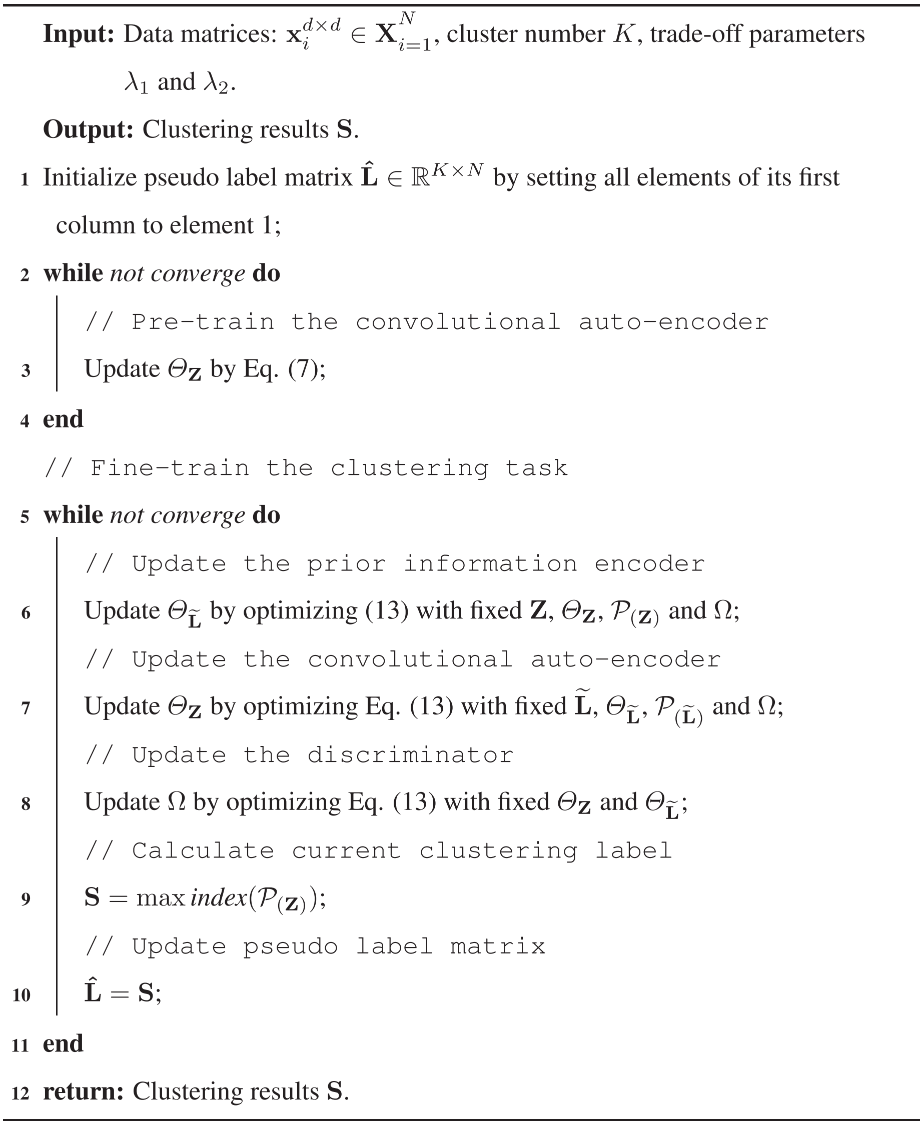
The optimization step can be roughly divided into two steps: 1) pre-training the convolutional auto-encoder, 2) fine-tuning the clustering network. To be specific, before solving the problem (13), like other unsupervised learning methods [14], we pre-train the convolutional auto-encoder networks by minimizing Eq. (7) to produce semantically meaningful and well-separated initial $\mathbf{Z}$. There are three blocks of variables in RC2PI, and the objective of the RC2PI is not jointly convex for all these variables. Therefore, in the phase of fine-tuning the RC2PI, we optimize problem (13) by employing Alternating Direction Method of Multipliers (ADMM) [15] strategy. To adopt the ADMM strategy, the optimization is cycled over the following three sub-steps: updating prior information encoding networks, updating the auto-encoder networks, and updating the discriminator networks. The optimization for each sub-step is as follows:

- **Updating prior information encoder**. To generate the reasonable actual distribution $\tilde{\mathbf{L}}$, we first update the parameters $\Theta_{\tilde{\mathbf{L}}}$ of prior information encoder by applying Adam optimizer to optimize $\mathscr{L}_{PI}$ and $\mathscr{L}_d$ in Eqs. (9), (12) with fixed $\mathbf{Z}, \Theta_{\mathbf{Z}}, \mathscr{P}_{(\mathbf{Z})}$ and Ω.
- **Updating the convolutional auto-encoder network**. To make full use of priori information, we update $\Theta_{\mathbf{Z}}$ of convolutional auto-encoder by applying Adam optimizer to optimize $\mathscr{L}_{AE}, \mathscr{L}_R$ and $\mathscr{L}_d$ in Eqs. (7), (5), (12) with fixed $\tilde{\mathbf{L}}, \Theta_{\tilde{\mathbf{L}}}, \mathscr{P}_{(\tilde{\mathbf{L}})}$ and Ω.
- **Updating the discriminator network**. To update the discriminator, we follow a similar way as updating the convolutional auto-encoder network. Hence, we should optimize Eq. (11) w.r.t Ω with fixed $\Theta_{\mathbf{Z}}$ and $\Theta_{\tilde{\mathbf{L}}}$.

The pseudo code of method is summarized in Algorithm 1.

[2] Although the difference value of distribution $\mathscr{P}_{(\tilde{\mathbf{L}})}$ and $\mathscr{P}_{(\mathbf{Z})}$ is small, the magnitude of element values in a distribution can be dramatically different.

Algorithm 1: RC2PI



```
Input: Data matrices: $\mathbf{x}_i^{d\times d} \in \mathbf{X}_{i=1}^{N}$, cluster number $K$, trade-off parameters
       $\lambda_1$ and $\lambda_2$.
Output: Clustering results $\mathbf{S}$.
Initialize pseudo label matrix $\hat{\mathbf{L}} \in \mathbb{R}^{K\times N}$ by setting all elements of its first
 column to element 1;
while not converge do
    // Pre-train the convolutional auto-encoder
    Update $\Theta_{\mathbf{Z}}$ by Eq. (7);
end
// Fine-train the clustering task
while not converge do
    // Update the prior information encoder
    Update $\Theta_{\widetilde{\mathbf{L}}}$ by optimizing (13) with fixed $\mathbf{Z}$, $\Theta_{\mathbf{Z}}$, $\mathcal{P}_{(\mathbf{Z})}$ and $\Omega$;
    // Update the convolutional auto-encoder
    Update $\Theta_{\mathbf{Z}}$ by optimizing Eq. (13) with fixed $\widetilde{\mathbf{L}}$, $\Theta_{\widetilde{\mathbf{L}}}$, $\mathcal{P}_{(\widetilde{\mathbf{L}})}$ and $\Omega$;
    // Update the discriminator
    Update $\Omega$ by optimizing Eq. (13) with fixed $\Theta_{\mathbf{Z}}$ and $\Theta_{\widetilde{\mathbf{L}}}$;
    // Calculate current clustering label
    $\mathbf{S} = \max index(\mathcal{P}_{(\mathbf{Z})})$;
    // Update pseudo label matrix
    $\hat{\mathbf{L}} = \mathbf{S}$;
end
return: Clustering results $\mathbf{S}$.
```

3. Experiments

3.1. Experimental setting

Datasets and Evaluation Metrics. We have chosen two handwritten digit datasets (*i.e.*, MNIST [16] and USPS) and two face image datasets (*i.e.*, FRGCv2.0 and Youtube-Face(YTF) [17]) for showing that the RC2PI works well. For the *MNIST-Full* dataset, we concatenate $60,000$ training and $10,000$ testing samples when applicable. The *MNIST-test* dataset consists of $10,000$ monochrome images from *MNIST* testing set. The *USPS* dataset contains $9,298$ gray images with the size of $16\times16\times1$ from envelopes by the *U.S.* postal services. The *FRGC* dataset consists of 20 randomly selected subjects in [18,8] from the raw dataset. The *YTF* dataset has 41 subjects face images, which were chosen from *Youtube-Face* like [17]. We summarize the statistics of these datasets in Table 1. For fair comparison, all datasets used in the experiments are consistent with [18,8]. For all tasks, two frequently-used measures: accuracy (ACC) and normalized mutual information (NMI) [10]) are adopted.

3.1.1. Comparison methods

We compare RC2PI with eight baseline clustering methods, including K-Means [19], normalized cuts (N-Cuts) [20], self-tuning spectral clustering (SC-ST) [21], large-scale spectral clustering (SC-LS) [22], SEC [11], agglomerative clustering via maximum incremental path integral (AC-PIC) [23]. In addition, we also evaluate the performance of the RC2PI with several state-of-the-art deep clustering models, including deep embedded clustering (DEC) [13], improved DEC (IDEC) [7], joint unsupervised learning (JULE) [18], deep embedded regularized clustering (DEPICT) [8], deep subspace clustering (DSC) [14], deep Spectral Net [24], latent distribution preserving deep subspace clustering (DPSC) [25] and deep spectral clustering using dual auto-encoder network (DSCDAN) [10].

3.1.2. Implementation details

In our experiment, detailed convolutional auto-encoder network structure, including kernel size and channel numbers, is shown in Table 2, we set stride $= 2$ for all convolutional layers. The dimension of the latent representation is set to K, where K is the number of clusters for different datasets. For all layers of auto-encoder, *ReLU* activation function is employed. The dimension of prior information encoder is set $K \rightarrow 128 \rightarrow K$ for all datasets, and the dimension of discriminator networks $\mathscr{D}$ is $K \rightarrow 64 \rightarrow 128 \rightarrow 1$. *Sigmoid* activation function is set for output layer of $\mathscr{D}$. We use TensorFlow 1.13.1 to implement our approach. For all optimizing steps, the *Adam* optimizer is adopted. During pre-training step, the learning rate is set to 3×10^{-3}. Then we set the learning rate to 10^{-2} to update prior information encoder and the learning rate to 10^{-3} to update auto-encoder. For all datasets, we set learning rate to 10^{-4} to update the discriminator $\mathscr{D}$. In actual experiments, we observe that the loss values of $\mathscr{L}_R$ and $\mathscr{L}_{PI}$ are much larger than other terms. In order to optimize reasonably, the values of parameters λ_1 and λ_2 are set to 1×10^{-4} and 1, respectively. δ is set to 1×10^{-5} for all datasets.

3.2. Clustering performance

We herein evaluate RC2PI on five widely used datasets, the clustering results are summarized in Table 3. Specifically, our model achieves the best performance on both handwritten digit clustering and face image clustering in both two metrics. Comparing with the traditional clustering model, *e.g.,* K-Means [19], N-Cuts [20], and spectral clustering [21], RC2PI greatly boosts the clustering results. The reason is that RC2PI utilizes the convolutional auto-encoder to extract low dimensional representations, which can easily deal with the complex handwritten images (shifting, rotation, and so on).

Comparing with the shallow regression-based clustering model SEC [11], RC2PI behaviors more excellent, this is because RC2PI introduces an appropriate soft regression constraint to minimize the difference between continuous actual distribution and continuous cluster indicator matrix, which can help to learn more powerful representations. Further observation, we find deep clustering methods (such as DEC [13], JULE [18], DPSC [25]) achieve the highest metric than the shallow models (such as K-Means [19]), by which we conclude that representation learning is significant to face image clustering.

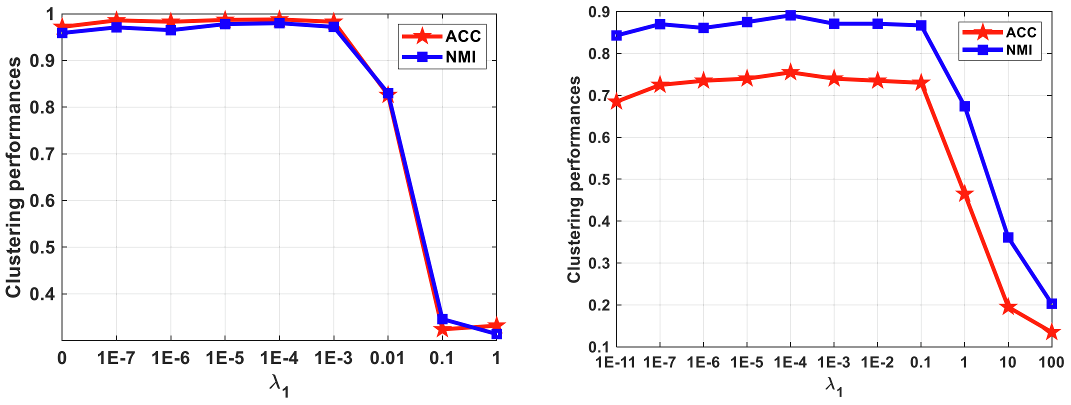
RC2PI also demonstrates superior results over previous deep clustering models, especially DEC [13], IDEC [7], DSC [14], and Spectral Net [24]. This is because RC2PI makes full use of the significant spital structure prior knowledge of the pseudo label matrix. For a comprehensive comparison, RC2PI still brings about 6.4%and 3.4% improvement in terms of ACC, NMI over the DSCDAN [10] on the YTF dataset.

It is obvious that the clustering performances are improved by updating the pseudo label matrix with clustering label, these results indicates that the pseudo clustering label is useful for subsequent clustering tasks.

3.3. Model discussion

Sensitivity Analysis. In this section, we simultaneously adjust the parameters λ_1 and λ_2 on both MNIST-*test* and YTF datasets to test the sensitivity of the RC2PI. To illustrate the results conveniently, we fix one and vary the other parameter in our experiment. We firstly analyze the sensitivity of the parameter λ_1 with fixed $\lambda_2 = 1$. As shown in Fig. 4, we present the clustering performance of RC2PI with different tuning parameters λ_1. We tune the parameter λ_1 in the range of $\left\{0, 10^{-7}, \ldots, 1\right\}$ on MNIST-*test* dataset and $\left\{0, 10^{-7}, \ldots, 100\right\}$ on YTF. We utilize 10^{-11} to denote value 0 in Fig. 4, from (a) we can observe that RC2PI is robust because changes of λ_1 has a little influence on the clustering performance in the range of $\left\{10^{-7} \sim 10^{-3}\right\}$. The performance of RC2PI decreases sharply when λ_1 is relatively big (*e.g.,* 10^{-2}), the main reason is the $\mathscr{L}_{\mathscr{R}}$ and $\mathscr{L}_{\mathscr{P}\mathscr{I}}$ dominate in this case, leading to the difference between $\mathscr{L}_d$ and other constraints are huge. The same situation also occurs on YTF dataset in (b). Obviously, when the tradeoff parameter λ_1 is set properly, both soft regression constraint and prior information help improve clustering performance.

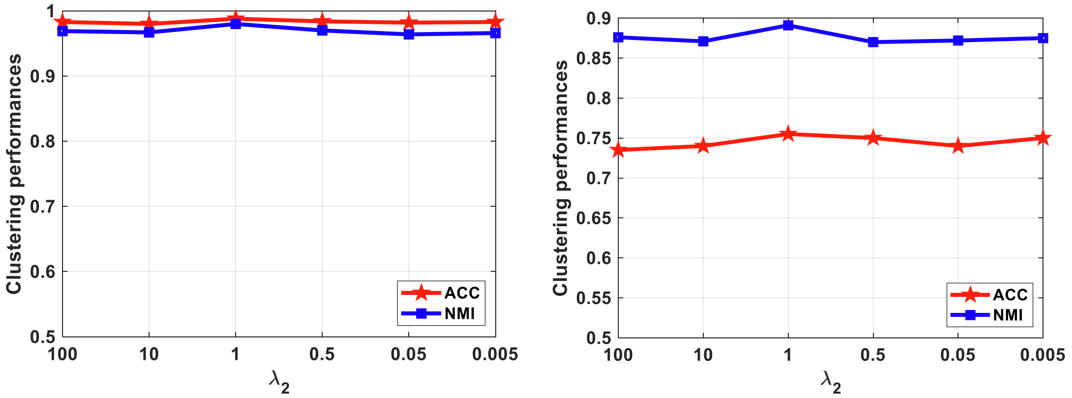
Then, we test the sensitivity of the tuned parameter λ_2 with fixed $\lambda_1 = 10^{-4}$. We vary λ_2 in the range of $\{0.005,\ 0.05,\ 0.5,\ 1,\ 10,\ 100\}$ on both MNIST-*test* and YTF data-



(a) MNIST-*test* Dataset

(b) YTF Dataset

Fig. 4. Sensitivity analysis of parameter λ_1 of RC2PI.



(a) MNIST-*test* dataset

(b) YTF dataset

Fig. 5. Sensitivity analysis of parameter λ_2 of RC2PI.

Table 4
Ablation study on five datasets, where ✗ denotes discarding the corresponding constraint term and ✓ is exactly the opposite.

$\mathscr{L}_{\mathscr{AE}}$	$\mathscr{L}_{\mathscr{R}}$	$\mathscr{L}_{\mathscr{PI}} + \mathscr{L}_d$	MNIST-*full*		MNIST-*test*		USPS		FRGC		YTF	
			ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI	ACC	NMI
✓	✗	✗	0.841	0.756	0.887	0.790	0.776	0.699	0.305	0.402	0.638	0.810
✓	✓	✗	0.970	0.957	0.969	0.949	0.938	0.953	0.469	0.620	0.665	0.833
✓	✓	✓	**0.990**	**0.985**	**0.988**	**0.980**	**0.969**	**0.985**	**0.508**	**0.633**	**0.755**	**0.891**

set. As shown in Fig. 5, it intuitively demonstrates that the RC2PI maintains acceptable performance within a wide range of λ_2, it also shows that RC2PI is relatively stable.

Ablation Study. Now we validate the effectiveness of the proposed prior information encoding networks and the soft regression constraint term via image clustering task on the five datasets. There are three cases as shown in Table 4.

- **Case 1**: Reconstruction loss $\mathscr{L}_{AE}$ only. It indicates the objective function only maintain auto-encoder reconstruction loss in Eq. (7). Under the same hyper-parameters, we perform K-Means on the actual distribution $\mathbf{Z}$ obtained from the pre-trained auto-encoder. Compared to the results of K-Means in Table 3, whose input is raw image vector, it can be noticed the convolutional auto-encoder is helpful to clustering because we can use it to learn more powerful representations with spatial locality preserved.
- **Case 2**: Reconstruction loss $\mathscr{L}_{AE}$ with regression-based clustering loss $\mathscr{L}_R$. It indicates the network is trained by Eqs. (5) and (7). Obviously, when we train the auto-encoder with regression clustering constraint, the clustering performances are relatively improved. These results demonstrate the soft regression constraint is helpful for clustering. This is because for each sample $\mathbf{x}_i$, we utilize a continuous distribution to constrain the actual distribution $\mathbf{z}_i$, the soft regression term for each sample can help with the actual distribution $\mathbf{Z}$ to capture the locally discriminative information, thereby improving clustering performance without the manifold assumption.
- **Case 3**: All constraints are retained. RC2PI produced the best results. This is because not only prior information helps actual distribution learning, but also the RC2PI can self-optimize the continuous distribution $\mathscr{P}_{(\mathbf{Z})}, \mathscr{P}_{(\tilde{\mathbf{L}})}$ according to current cluster centroid μ, actual distributions $\mathbf{Z}$ and $\tilde{\mathbf{L}}$.

- **Running Time Comparison.** In order to evaluate the efficiency of our proposed method in dealing with image data, we compare the running time costs of our method with some representative deep clustering method, *i.e.*, DEC, IDEC, JULE, DEPICT and DSCDAN. We run our method and the released codes of above five methods on a machine with two NVIDIA Tesla P100-PICE GPUs, the Intel (R) Xeon (R) Gold 6230 CPU and 128 GB RAM. Fig. 6 illustrates the execution times (in seconds) of our proposed method and the comparing methods on five datasets. As reported in Fig. 6, except for MNIST-Full dataset, we can see that our method is faster than other comparison methods

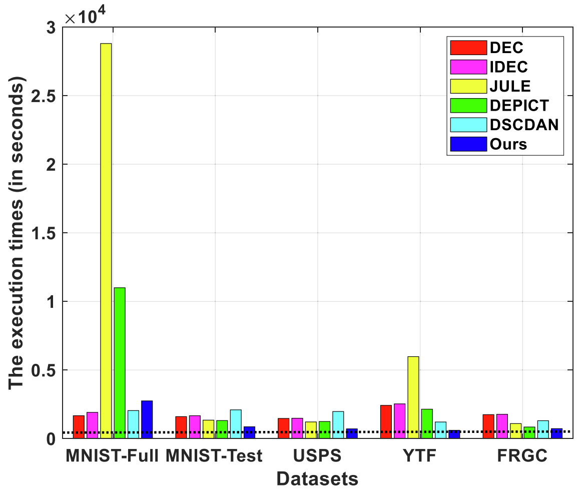


Fig. 6. Comparison of the execution times (in seconds) of different deep clustering methods on five datasets.

when dealing with image data. This performance again demonstrates the practicality of our method for real-world image clustering tasks.

4. Conclusion and future work

In this paper, we propose a novel regression-based clustering framework, named RC2PI, which consists of an auto-encoder, a structure prior information encoder and a discriminator. The proposed model can obtain more discriminative and satisfactory actual distribution to boost clustering performance. Comparing with existing regression-based clustering, RC2PI behaves stable, because it can directly obtain the clustering result without post-processing operation, like K-Means. Experimental results on five image datasets demonstrate the validity of RC2PI and have shown the advantage over the various clustering methods. In the future, we'll take multi-view learning into account.

CRediT authorship contribution statement

Wei Xia: Conceptualization, Methodology, Software, Investigation, Writing - original draft. **Quanxue Gao:** Conceptualization, Methodology, Supervision, Writing - review & editing. **Qianqian Wang:** Validation, Visualization, Software, Writing - review & editing. **Xinbo Gao:** Supervision, Writing - review & editing.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Acknowledgements

The authors would like to thank the anonymous reviewers and AE for their constructive comments and suggestions. This work is supported by the National Natural Science Foundation of China under Grants 61773302, Natural Science Basic Research Plan in Shaanxi Province under Grant 2020JZ-19 and 2020JQ-327, the Initiative Postdocs Supporting Program BX20190262, the Fundamental Research Funds for the Central Universities and the Innovation Fund of Xidian University.

References

[1] S. Zheng, X. Cai, C.H.Q. Ding, F. Nie, H. Huang, A closed form solution to multi-view low-rank regression, AAAI (2015) 1973–1979.
[2] L. Chen, Dual linear regression based classification for face cluster recognition, CVPR (2014) 2673–2680.
[3] J. Ye, Least squares linear discriminant analysis, in: ICML, 2007, pp. 1087–1093.
[4] J. Han, H. Liu, F. Nie, A local and global discriminative framework and optimization for balanced clustering, IEEE Trans. Neural Netw. Learning Syst. 30 (10) (2019) 3059–3071.
[5] F. Nie, D. Xu, I.W. Tsang, C. Zhang, Spectral embedded clustering, IJCAI (2009) 1181–1186.
[6] H. Gao, X. Wang, H. Huang, New robust clustering model for identifying cancer genome landscapes, in: ICDM, 2016, pp. 151–160.
[7] X. Guo, L. Gao, X. Liu, J. Yin, Improved deep embedded clustering with local structure preservation, IJCAI (2017) 1753–1759.
[8] K. Ghasedi Dizaji, A. Herandi, C. Deng, W. Cai, H. Huang, Deep clustering via joint convolutional autoencoder embedding and relative entropy minimization, in: ICCV, 2017, pp. 5736–5745.
[9] S. Mukherjee, H. Asnani, E. Lin, S. Kannan, Clustergan: Latent space clustering in generative adversarial networks, in: AAAI, 2019, pp. 4610–4617.
[10] X. Yang, C. Deng, F. Zheng, J. Yan, W. Liu, Deep spectral clustering using dual autoencoder network, CVPR (2019) 4066–4075.
[11] F. Nie, Z. Zeng, I.W. Tsang, D. Xu, C. Zhang, Spectral embedded clustering: A framework for in-sample and out-of-sample spectral clustering, IEEE Trans. Neural Networks 22 (11) (2011) 1796–1808.
[12] L. v. d. Maaten, G. Hinton, Visualizing data using t-sne, JLMR 9 (2008) 2579–2605.
[13] J. Xie, R. Girshick, A. Farhadi, Unsupervised deep embedding for clustering analysis, ICML (2016) 478–487.
[14] P. Ji, T. Zhang, H. Li, M. Salzmann, I. Reid, Deep subspace clustering networks, NIPS (2017) 24–33.
[15] Z. Lin, R. Liu, Z. Su, Linearized alternating direction method with adaptive penalty for low-rank representation, NIPS (2011) 612–620.
[16] Y. LeCun, L. Bottou, Y. Bengio, P. Haffner, et al., Gradient-based learning applied to document recognition, Proc. of the IEEE 86 (11) (1998) 2278–2324.
[17] L. Wolf, T. Hassner, I. Maoz, Face recognition in unconstrained videos with matched background similarity, CVPR (2011) 529–543.
[18] J. Yang, D. Parikh, D. Batra, Joint unsupervised learning of deep representations and image clusters, CVPR (2016) 5147–5156.
[19] J. MacQueen, et al., Some methods for classification and analysis of multivariate observations, in: Mathematical Statistics and Probability, 1967, pp. 281–297.
[20] J. Shi, J. Malik, Normalized cuts and image segmentation, Departmental Papers (CIS) (2000) 107.
[21] L. Zelnik-Manor, P. Perona, Self-tuning spectral clustering, NIPS (2005) 1601–1608.
[22] X. Chen, D. Cai, Large scale spectral clustering with landmark-based representation, AAAI (2011) 313–318.
[23] W. Zhang, D. Zhao, X. Wang, Agglomerative clustering via maximum incremental path integral, Pattern Recog. 46 (11) (2013) 3056–3065.
[24] U. Shaham, K. P. Stanton, H. Li, R. Basri, B. Nadler, Y. Kluger, Spectralnet: Spectral clustering using deep neural networks, in: ICLR, 2018.
[25] L. Zhou, X. Bai, D. Wang, X. Liu, J. Zhou, E.R. Hancock, Latent distribution preserving deep subspace clustering, IJCAI (2019) 4440–4446.

Wei Xia received the B.Eng. degree in Communication Engineering from Lanzhou University of Technology, Lanzhou, China, in 2018. He is currently pursuing the Ph.D. degree in communication and information system in Xidian University, Xi'an, China. His research interests include pattern recognition, machine learning and deep learning.

Quanxue Gao received the B. Eng. degree from Xi'an Highway University, Xi'an, China, in 1998, the M.S. degree from the Gansu University of Technology, Lanzhou, China, in 2001, and the Ph.D. degree from Northwestern Polytechnical University, Xi'an China, in 2005. He was an associate research with the Biometrics Center, The Hong Kong Polytechnic University, Hong Kong from 2006 to 2007. He is currently a professor with the School of Telecommunications Engineering, Xidian University, and a key member of State Key Laboratory of Integrated Services Networks. His current research interests include pattern recognition and machine learning.

Qianqian Wang received the B.Eng. degree in communication engineering from Lanzhou University of Technology, China, in 2014, the Ph.D. degree from Xidian University, X'ian China, in 2019. She is currently a lecturer with the School of Telecommunications Engineering, Xidian University, China. Her research interests include pattern recognition, dimensionality reduction, sparse representation, and face recognition.

Xinbo Gao received the B. Eng., M.S., and Ph.D. degrees in signal and information processing from Xidian University, Xi'an, China, in 1994, 1997, and 1999, respectively. From 1997 to 1998, he was a Research Fellow with the Department of Computer Science, Shizuoka University, Japan. From 2000 to 2001, he was a Post-Doctoral Research Fellow with the Department of Information Engineering, The Chinese University of Hong Kong, Hong Kong. Since 2001, he has been with the School of Electronic Engineering, Xidian University. He is currently a Cheung Kong Professor with the Ministry of Education, a Professor of pattern recognition and intelligent system, and the Director of the State Key Laboratory of Integrated Services Networks, Xidian University. He has authored six books and around 200 technical articles in refereed journals and proceedings. His current research interests include multimedia analysis, computer vision, pattern recognition, machine learning, and wireless communications. He served as the General Chair/Co-Chair, a Program Committee Chair/Co-Chair, or a PC Member for around 30 major international conferences. He is a fellow of the Institute of Engineering and Technology, and the Chinese Institute of Electronics. He is on the editorial boards of several journals including Signal Processing (Elsevier) and Neurocomputing (Elsevier).