

# 互补与一致的多视角子空间聚类网络

何士豪, 张玉龙, 唐启凡

(西安交通大学软件学院, 710049, 西安)

**摘要:** 针对在非线性子空间中同时探索多视角数据的互补性与一致性的问题,提出了一种互补与一致的多视角子空间聚类网络 C2MSCN。该网络由编码器、自表达层、谱聚类模块和解码器 4 部分组成。通过编码器将数据映射到潜在的非线性空间,探索多视角数据间复杂的非线性关系;利用自表达层学习多视角共享的自表达系数矩阵和单视角私有的自表达系数矩阵,同时挖掘多视角数据间的互补与一致信息;将系数矩阵供给谱聚类模块以获得聚类标签;通过解码器将自表达数据还原至原始空间。为充分挖掘多视角数据间的互补信息,针对私有自表达系数矩阵加入多样性规范化项;为利用聚类标签监督自表达系数的学习,在目标函数中加入自监督规范化项。实验结果表明:所提算法在 4 个评价指标下 6 个数据集的对比实验中取得了 1.25 的平均排名;在 Yale 数据集上,用准确率评价的聚类准确度较线性空间多视角聚类算法 CSMSC、深度多模态子空间聚类算法 DMSCN 和自编码器算法 Ae2-nets 分别提高了 8.3%、6.3% 和 11.7%;参数敏感度实验和消融性实验表明,所提算法能有效在非线性子空间中探索数据互补性与一致性,且在不同数据集上表现稳定。

**关键词:** 多视角聚类;自编码器;子空间聚类;自监督;自表达

**中图分类号:** TP181 **文献标志码:** A

**DOI:** 10.7652/xjtuxb202107019 **文章编号:** 0253-987X(2021)07-0166-13



OSID 码

## Complementary and Consistent Multi-View Subspace Clustering Network

HE Shihao, ZHANG Yulong, TANG Qifan

(School of Software Engineering, Xi'an Jiaotong University, Xi'an 710049, China)

**Abstract:** A complementary and consistent multi-view subspace clustering network (C2MSCN) is proposed to better explore the complementarity and consistency of multi-view data in nonlinear subspace. The network consists of four parts: encoder, self-expression layer, spectral clustering module, and decoder. This network firstly maps the data to the potential nonlinear space through the encoder to explore the complex nonlinear relationship among multi-view data. Then the self-expression layer is used to simultaneously learn the self-expression coefficient matrix shared by multiple views and private self-expression coefficient matrixes of multiple views to simultaneously mine the complementary and consistent information in multi-view data. Then the coefficient matrixes are fed to the spectral clustering module to obtain the clustering label. Finally, the self-expressed data is restored to the original space through the decoder. A multiplicity normalized item for private self-expression coefficient matrixes is added to fully mine the complementary information among multi-view data. Meanwhile, a self-supervised normalized term is added to supervise the learning process of self-expression coefficient matrixes. Experimental results show

收稿日期: 2021-01-20。 作者简介: 何士豪(1998—),男,硕士生;张玉龙(通信作者),男,讲师。 基金项目: 国家自然科学基金资助项目(61573273)。

网络出版时间: 2021-03-01

网络出版地址: <http://kns.cnki.net/kcms/detail/61.1069.T.20210301.1318.002.html>

that the proposed algorithm achieves an average ranking of 1.25 in six data sets under four evaluation indexes. On the Yale dataset, the clustering accuracy evaluated by ACC is 8.3%, 6.3%, and 11.7% higher than that of the linear space multi-view clustering algorithm CSMSC, the deep multi-view subspace clustering algorithm DMSCN and the autoencoder algorithm Ae2-nets, respectively. Parameter sensitivity experiments and ablation experiments indicate that the proposed algorithm can effectively explore data complementarity and consistency in nonlinear subspace, and performs stably in different data sets.

**Keywords:** multi-view clustering; autoencoder; subspace clustering; self-supervision; self-expression

近年来,随着计算机和网络技术的快速发展,“大数据”一词开始频频出现在大众视野,随着海量数据的涌现,如何有效地在数据中挖掘出有价值的信息成为热门的研究课题。其中,多视角数据因其包含更全面的信息而成为热点研究数据。多视角数据是指同源数据的多种表示形式,例如,一幅图片可以由颜色、纹理、边缘等特征表示,一篇文章可以由多种语言表示等。聚类分析一直以来是数据挖掘的主要任务之一,其中子空间聚类是聚类分析中的重要研究部分。传统的子空间聚类算法在处理多视角数据时,通常是将其多视角数据简单拼接为单视角数据,然后对其应用经典的单视角子空间聚类算法<sup>[1-3]</sup>。这种做法的优点是简单快捷,但同时带来了一系列缺点。首先,多视角数据具有互补性与一致性的特点<sup>[4]</sup>。互补性是指各视角包含其他视角所不具备的信息,各视角所含数据信息具有互补性。一致性是指来自同一数据源的各视角数据所包含的基本信息是一致的<sup>[4]</sup>。单视角聚类算法仅将多视角数据简单拼接,无法有效地挖掘多视角所包含的丰富信息,因此无法取得较好的聚类效果。其次,多视角数据中含有大量重复数据和噪声数据,简单地将其拼接成单视角数据时,往往会带来维度灾难,导致数据庞大而冗余。因此,近年来多视角聚类算法受到越来越广泛的关注。

多视角聚类算法通过提取多视角间的互补性<sup>[5-7]</sup>或一致性信息<sup>[8-12]</sup>来提升聚类性能。然而,随着研究的深入,单一地探索多视角数据的互补性或一致性信息已不能满足聚类性能要求。于是,一系列同时探索多视角数据的互补性和一致性信息的聚类算法被提出<sup>[13-15]</sup>,这些算法通过对多视角共享的邻接矩阵和单视角私有的邻接矩阵添加规范项来更全面地探索数据信息,进一步提升了聚类性能。

这些多视角聚类算法具有开创性意义,且聚类性能均达到非常优秀的水平,其中关于多视角子空

间聚类研究主要集中在子空间的聚类上。然而,现实中数据不一定严格地符合线性子空间模型,例如,在人脸图像聚类中,反射率通常是非朗伯式的,而且拍摄对象的姿势经常变化,在这些条件下,面部图像更像是在一个非线性的子空间中<sup>[15]</sup>。针对这个问题,文献[16-18]通过核技巧<sup>[19]</sup>进行了尝试,但目前仍需依靠经验选择核类型,并且没有严格的证明表示核所属的隐式特征空间可以完美地适配子空间聚类。近年来,越来越多的研究工作将深度学习与子空间聚类结合,利用自编码器将数据映射到非线性子空间,通过在编码器与解码器间的全连接层学习邻接矩阵进行子空间聚类<sup>[15,20-22]</sup>。实验表明,自编码器可以有效挖掘多视角数据间的非线性子空间关系,从而提升聚类性能。

为了在非线性子空间中同时挖掘多视角数据中的互补性与一致性信息、最大程度地利用多视角数据的有效信息提升聚类性能,一种行之有效的方法是将深度学习与传统的多视角聚类算法相结合,例如:文献[23]利用两个网络分别挖掘数据的互补性与一致性信息,然而由于两个网络的独立性,多视角共享的相似度矩阵和多视角独有的相似度矩阵是独立的,两种相似度矩阵缺乏关联性,未能充分挖掘相关信息,导致某些数据集上的实验效果欠佳;文献[24]利用全局结构和局部结构使得多视角共享的系数矩阵与多视角私有的潜在特征表示产生连接,同时利用规范项挖掘多视角数据的互补性信息,然而其缺乏对聚类标签的有效利用,导致系数矩阵的学习缺乏反馈过程,很难学习到好的特征表示。尽管聚类标签不一定正确地标记所有数据点的类信息,但其包含一些有价值的信息,因此可以用聚类标签来监督自表达系数的训练过程,以提升聚类性能<sup>[25]</sup>。

本文提出一种互补与一致的多视角子空间聚类网络(C2MSCN),用来解决自监督式地在非线性空间中同时挖掘多视角数据的互补性与一致性信息的

问题。C2MSCN 由编码器、自表达层、谱聚类模块和解码器组成。C2MSCN 中的数据流通过程主要有 4 个阶段:①编码器将原始数据映射到非线性空间;②自表达层在非线性空间同时学习各视角共享的自表达系数矩阵和各视角私有的自表达系数矩阵;③将自表达系数矩阵供给谱聚类模块聚类,得到聚类标签,谱聚类模块将聚类标签返回网络;④解码器将自表达数据还原到原始空间。C2MSCN 算法在 4 个评价指标下 6 个数据集的对比实验中取得了 1.25 的平均排名;参数敏感度实验和消融性实验表明,C2MSCN 算法能够有效地在非线性子空间中探索数据互补性与一致性,并且在不同数据集上表现稳定。

## 1 相关工作

### 1.1 子空间聚类

假设高维数据来自低维子空间,子空间聚类的任务是将一群高维数据分为多组,为每组数据寻找一个适配的低维子空间<sup>[4]</sup>。自表达是指用属于同一子空间的数据的线性组合来表示高维数据,数据的线性组合系数被称为自表达系数,由所有数据的自表达系数组成的矩阵被称为自表达系数矩阵。子空间聚类的流程是:①构造自表达系数矩阵;②构造相似度矩阵;③应用谱聚类得到聚类结果。子空间聚类的重点是如何利用自表达系数矩阵正确地恢复数据的子空间结构。近年来,子空间聚类算法主要研究如何学习一个优质的数据自表达系数矩阵。其中一些优秀的算法包括:SSC 通过对自表达系数矩阵添加  $L_1$  范数,保证了自表达系数矩阵的稀疏性<sup>[1]</sup>;LRR 通过对自表达系数矩阵添加核范数约束,挖掘了自表达系数矩阵的低秩性<sup>[2]</sup>;深度子空间聚类网络(DSCN)在编码器与解码器间加入自表达层,在编码器映射的低维非线性空间中学习自表达系数矩阵,可以对具有复杂或非线性潜在结构的数据有效地聚类,从而弥补了传统子空间聚类算法只能探索线性数据关系的弊端<sup>[15]</sup>;自监督卷积神经网络(S2SCN)在 DSCN 的基础上加入了自监督模块,将自表达系数矩阵的学习与谱聚类结合,首次利用聚类标签信息监督自表达系数矩阵的学习<sup>[25]</sup>;多级深度子空间表示学习(MLRDSC)取用自编码器网络多个中间层数据,并利用自表达层学习中间层数据共享的自表达系数矩阵和中间层数据私有的自表达系数矩阵,融合了多个中间层数据的互补性与一致性信息,通过挖掘网络的中间层数据提高了聚类效

果<sup>[26]</sup>;基于差分演化算法的软子空间聚类在目标函数中综合利用了模糊加权类内相似性和约束权重矩阵,并用复合差分演化算法搜寻子空间聚类,有效地提高了聚类效果<sup>[27]</sup>。

### 1.2 多视角聚类

多视角聚类的关键是如何充分利用多视角数据的有效信息来提升聚类效果。目前,一系列多视角聚类算法通过对多视角数据的关系进行建模或者学习数据的内在表示来提高聚类效果<sup>[23]</sup>。RMSC 通过学习公共低秩概率转移矩阵来融合多视角数据的有效信息,通过学习各视角的私有稀疏噪声矩阵来处理多视角数据中的噪声,有效地融合了多视角数据信息,过滤了噪声<sup>[8]</sup>;LMSC 首先将数据从原始空间映射到潜在的数据空间,然后在潜在空间学习自表达系数矩阵,从而挖掘了更深层的数据信息<sup>[9]</sup>;Co-Reg 假设多视角数据具有相同的类别关系,对不同视角间的特征向量矩阵添加协同约束,利用一个视角的数据信息去优化其他视角特征向量矩阵的学习,以此挖掘多视角数据信息<sup>[6]</sup>;一致与特殊多视角子空间聚类(CSMSC)通过学习多视角共享的自表达系数矩阵和多视角私有的自表达系数矩阵,在线性空间同时挖掘了多视角数据间的互补性与一致性信息<sup>[4]</sup>;一种双重加权的多视角聚类算法利用互信息学习各个视角的权重,并将各视角权重用于基于内容和基于上下文的多视角表示,充分利用了两种数据形式下的多视角数据间的互补性信息<sup>[28]</sup>。

### 1.3 自编码器

自编码器通过将数据映射到非线性低维空间来提取数据的内在非线性特征。因此,在子空间聚类中,可以利用自编码器来挖掘多视角间潜在的非线性关系<sup>[15]</sup>。深度多模态子空间聚类(DMSCN)将自编码器与多视角聚类结合,采用端对端的训练算法,利用多视角数据间的一致信息,在非线性空间中学习公共自表达系数矩阵,从而克服了传统多视角聚类算法无法探索多视角数据间非线性关系的弊端<sup>[21]</sup>;自编码器网络中的自编码器算法(Ae2-nets)由内部自编码器和外部自编码器组成,内部自编码器由自编码器网络组成,外部自编码器由退化网络和内部自编码器网络组成,首先单独训练内部自编码器,然后通过退化网络将潜在特征退化为内部自编码器的中间层特征表示  $Z'$ ,通过优化真正的中间层特征表示  $Z$  与  $Z'$  的差距得到良好的潜在特征,从而自适应地平衡多视角数据间的互补性与一致性信息<sup>[29]</sup>。

## 2 互补与一致的多视角子空间聚类网络

### 2.1 网络结构

图 1 是互补与一致的多视角子空间聚类网络,网络由多视角卷积编码器、自表达模块、谱聚类模块和多视角卷积解码器共 4 部分组成。假设  $N$  个数据

样本共  $V$  个视角,令  $\{\mathbf{X}^i\}_{i=1}^V$  表示  $V$  个视角的数据,  $\mathbf{X}^i \in \mathbf{R}^{n \times d_i}$  表示第  $i$  视角的数据。 $d_i$  表示第  $i$  视角数据的维度,互补与一致的多视角子空间聚类网络允许各视角数据的维度不一致。为提高自编码器的学习能力,降低神经网络的参数量和计算复杂度,C2MSCN 采用卷积神经网络作为自编码器的网络层。

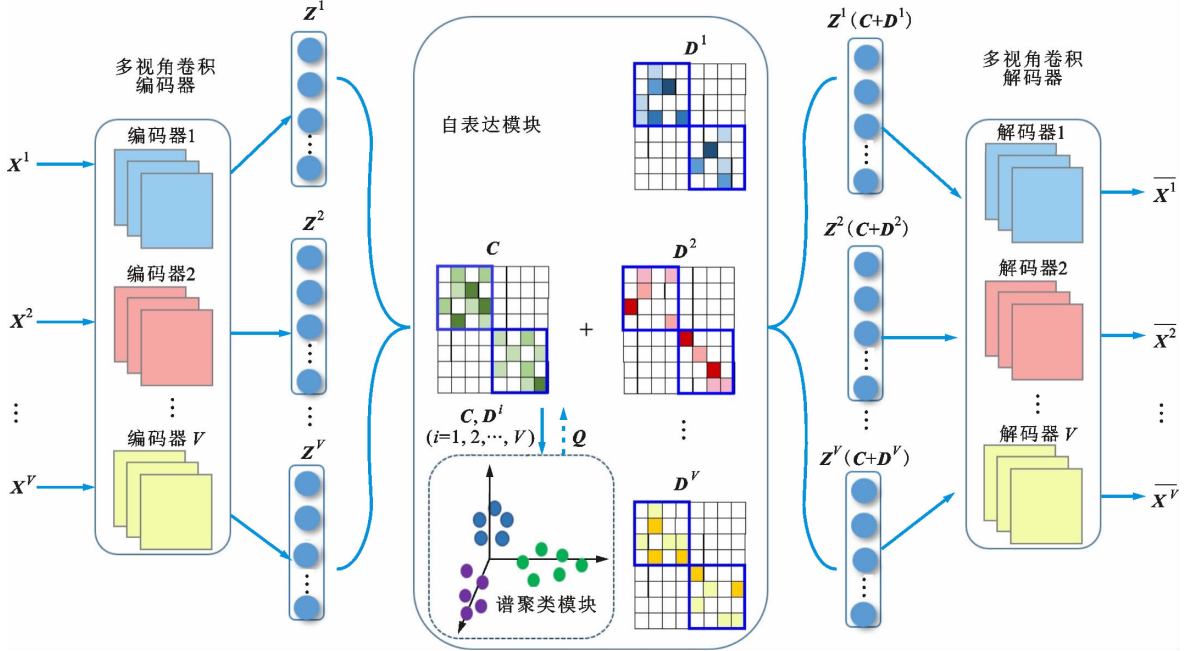


图 1 互补与一致的多视角子空间聚类网络

Fig. 1 Complementary and consistent multi-view subspace clustering network

(1)多视角卷积编码器。为适应不同视角数据维度的不一致性,C2MSCN 中的每个视角均有自己的卷积自编码器。卷积自编码器将数据  $\{\mathbf{X}^i\}_{i=1}^V$  映射为非线性空间的潜在特征表示  $\{\mathbf{Z}^i\}_{i=1}^V$ 。

(2)自表达模块。通过全连接层学习多视角数据共享的自表达系数矩阵  $\mathbf{C}$  和多视角数据私有的自表达系数矩阵  $\{\mathbf{D}^i\}_{i=1}^V$ ,得到自表达特征  $\{\mathbf{Z}^i(\mathbf{C} + \mathbf{D}^i)\}_{i=1}^V$ 。

(3)谱聚类模块。谱聚类模块利用共享自表达系数  $\mathbf{C}$  和私有自表达系数  $\{\mathbf{D}^i\}_{i=1}^V$  构造相似度矩阵  $\mathbf{S}$ ,基于相似度矩阵进行谱聚类,得到聚类结果,生成聚类标签信息  $\mathbf{Q}$ ,利用  $\mathbf{Q}$  构造自监督规范化项为自表达系数的学习提供反馈。

(4)多视角卷积解码器。多视角卷积解码器与多视角卷积编码器具有相反的网络结构,卷积解码器将自表达特征  $\{\mathbf{Z}^i(\mathbf{C} + \mathbf{D}^i)\}_{i=1}^V$  恢复到原始空间得到重构数据  $\{\bar{\mathbf{X}}^i\}_{i=1}^V$ 。

### 2.2 目标函数

根据 C2MSCN 的网络结构,目标函数由重构损

失、自表达损失、多样性规范化项、自监督规范化项和其他约束共 5 部分组成。

(1)重构损失。由于端对端的训练方式,输入编码器的原始数据与输出解码器的重构数据间存在重构损失。重构损失项通过约束自编码器的网络参数,确保原始数据经编码器编码获得的隐式特征可以通过解码器还原到原始数据,从而保证隐式特征的有效性。重构损失定义为

$$L_{\text{RE}} = \sum_{i=1}^V \|\mathbf{X}^i - \bar{\mathbf{X}}^i\|_{\text{F}}^2 \quad (1)$$

(2)自表达损失。潜在特征表示与自表达特征间存在自表达损失。自表达损失项通过约束自表达系数,确保原始数据与其自表达系数表示形式具有一致性。自表达损失定义为

$$L_{\text{self-exp}} = \sum_{i=1}^V \|\mathbf{Z}^i - \mathbf{Z}^i(\mathbf{C} + \mathbf{D}^i)\|_{\text{F}}^2 \quad (2)$$

(3)多样性规范化项。为充分挖掘多视角数据间的互补性信息,参考文献[24],在私有自表达系数矩阵间引入多样性规范化项

$$H(\mathbf{D}^i, \mathbf{D}^j) = \|\mathbf{D}^i \odot \mathbf{D}^j\|. \quad (3)$$

式中:  $\odot$  表示哈达玛积;  $\|\cdot\|$  表示矩阵的  $L_0$  范数;  $\mathbf{D}^i$  表示第  $i$  视角自表达系数矩阵。利用多样性规范化项使两个矩阵尽可能正交。理想情况下, 如果  $\mathbf{D}^i$  的  $(a, b)$  位置不等于 0, 规范化项将迫使  $\mathbf{D}^j$  的相同位置为 0, 使得私有自表达系数基于位置尽可能不相同。因此, 多样性规范化项可以挖掘不同视角数据在不同集群中的互补信息。由于  $L_0$  范数的非凸性, 将  $L_0$  范数松弛为  $L_1$  范数<sup>[18]</sup>,  $\mathbf{D}^i$  与  $\mathbf{D}^j$  间多样性规范化变为

$$H(\mathbf{D}^i, \mathbf{D}^j) = \|\mathbf{D}^i \odot \mathbf{D}^j\|_1 = \text{tr}((\mathbf{D}^i)^T \mathbf{D}^j) \quad (4)$$

所有视角的私有自表达系数矩阵两两之间的多样性规范化之和构成总的多样性规范化项

$$L_D = \sum_{m,n} \text{tr}((\mathbf{D}^m)^T \mathbf{D}^n) \quad (5)$$

(4) 自监督规范化项。参考文献[25]中的研究工作, 即使谱聚类产生的聚类标签不一定与真实聚类相同, 聚类标签仍包含一些价值信息, 可以用谱聚类的输出来监督自表达系数的训练过程。从自表达模块学习到自表达系数矩阵应包含足够的预测数据点的类标签信息。因此, 将谱聚类的目标函数作为自监督规范化项, 具有监督自表达模块训练的效果。在得到自表达系数矩阵后, 计算出相似度矩阵  $\mathbf{A} = (|\mathbf{C}| + |\mathbf{C}|^T)/2$ , 其中  $|\mathbf{C}|$  代表对矩阵  $\mathbf{C}$  求绝对值。通过最小化损失函数

$$\min_Q \sum_{i,j} a_{ij} \|\mathbf{q}_i - \mathbf{q}_j\|_2^2, \quad \text{s. t. } \mathbf{Q} \in M \quad (6)$$

对相似度矩阵  $\mathbf{A}$  应用谱聚类得到聚类指示矩阵  $\mathbf{Q}$ 。式(6)中,  $M = \{\mathbf{Q} \in \{0, 1\}^{k \times N}, \text{ s. t. } \mathbf{1}^T \mathbf{Q} = \mathbf{1}^T, \text{ rank}(\mathbf{Q}) = k\}$  是具有  $k$  类的有效聚类指示矩阵的集合,  $\mathbf{q}_m$  和  $\mathbf{q}_n$  表示  $\mathbf{Q}$  的第  $m$  列和第  $n$  列向量, 分别指示第  $m$  个视角数据与第  $n$  个视角数据所属的类别,  $\mathbf{1}$  为元素全是 1 的向量, 利用谱聚类模块为网络提供的聚类标签信息  $\mathbf{Q}$  构造自监督规范化项

$$L_{\text{self-sup}} = \sum_{i=1}^V \frac{1}{2} \sum_{m,n} S_{m,n}^i \|\mathbf{q}_m - \mathbf{q}_n\|_2^2 \quad (7)$$

$\mathbf{S}$  为相似度矩阵, 构造算法为

$$\mathbf{S} = \frac{|\mathbf{C}| + |\mathbf{C}|^T}{2} + \frac{1}{v} \sum_{i=1}^v \frac{|\mathbf{D}^i| + |\mathbf{D}^i|^T}{2} \quad (8)$$

由式(7)可以看出, 自监督项可以衡量相似度矩阵  $\mathbf{S}$  和聚类指示矩阵  $\mathbf{Q}$  间的差异性。在已知  $\mathbf{Q}$  的情况下, 最小化式(7)会有如下效果: 当  $\mathbf{q}_m$  不等于  $\mathbf{q}_n$  时, 迫使相似度矩阵  $\mathbf{S}$  的  $(m, n)$  位置项  $S_{m,n}$  趋近于 0; 当且仅当  $\mathbf{q}_m$  等于  $\mathbf{q}_n$  时,  $S_{m,n}$  才有可能为非 0 项。背后的意义是: 当第  $i$  项数据和第  $j$  项数据不属于同一

簇时, 迫使它们之间的相似度趋于 0; 只有当它们属于同一簇时, 它们之间的相似度才有可能为非 0 项。相似度矩阵由公有自表达系数  $\mathbf{C}$  和私有自表达系数  $\mathbf{D}^i$  组成, 于是式(7)间接地约束  $\mathbf{C}$  和  $\mathbf{D}^i$ , 从而为其自表达系数的训练提供反馈, 达到自监督公有自表达系数和私有自表达系数的效果。

(5) 其他规范化项。理想情况下, 各个子空间应该是相互独立的, 子空间中的数据应仅由属于同一子空间的其他数据表示<sup>[26]</sup>, 于是公有自表达系数应具有块状结构, 即稀疏的矩阵结构,  $L_1$  范数可以迫使它具有系数的矩阵结构。另外, 对私有自表达系数加入  $F$  范数约束可以保证连通性, 使得私有自表达系数更紧密, 从而消除连通性问题<sup>[4]</sup>。为保证自表达系数的这些特征, 参考文献[26]的思想, 对公共系数  $\mathbf{C}$  和私有系数  $\mathbf{D}^i$  添加约束

$$L_c = \|\mathbf{Q}^T |\mathbf{C}| \mathbf{Q}\|_1; \quad L_d = \sum_i \|\mathbf{D}^i\|_F^2 \quad (9)$$

式中  $\mathbf{Q}^T |\mathbf{C}|$  的第  $(i, j)$  位置元素表示在构建数据  $j$  时使用属于第  $i$  个簇的数据比例。与  $\|\mathbf{C}\|_1$  促使整个矩阵所有元素稀疏性不同,  $L_c$  仅促进样本簇成员数据间的稀疏性。换言之, 它鼓励每个数据仅由具有相同类标签的样本表示, 从而使样本对不同子空间的成员预测变得平滑。此外, 正则化项  $L_d$  使得不同矩阵的元素在数值上趋于相似, 从而增强了与相似度矩阵的连通性<sup>[26]</sup>。

将所有约束项相加得到目标函数

$$\left. \begin{aligned} L &= L_{\text{RE}} + \lambda_1 L_{\text{self-exp}} + \lambda_2 L_d^2 + \lambda_3 L_d + \\ &\quad \lambda_4 L_c + \lambda_5 L_{\text{self-sup}} \\ \text{s. t. } &\text{diag}(\mathbf{C}) = 0, \quad \text{diag}(\mathbf{D}^i) = 0 \\ &i = 1, 2, \dots, V, \quad \mathbf{Q}^T \mathbf{Q} = \mathbf{I} \end{aligned} \right\} \quad (10)$$

目标函数有  $\lambda_1 \sim \lambda_5$  共 5 个超参数:  $\lambda_1$  负责调节自表达重构约束的强度;  $\lambda_2$  和  $\lambda_4$  分别负责调节共享系数矩阵和私有系数矩阵的稀疏程度;  $\lambda_3$  和  $\lambda_5$  分别调节私有自表达系数矩阵的多样性程度和自监督规范化项程度。

## 2.3 训练策略

C2MSCN 算法分为 3 个阶段: 第 1 阶段预训练自编码器网络; 第 2 阶段正式训练整个网络; 第 3 阶段利用自表达系数矩阵构造相似度矩阵进行聚类。

(1) 预训练阶段。设置  $\lambda_1 = \dots = \lambda_5 = 0$ , 优化重构损失  $L_{\text{RE}}$ 。在预训练阶段, 将多视角数据  $\{\mathbf{X}^i\}_{i=1}^V$  输入多视角编码器, 经映射得到潜在特征表示  $\{\mathbf{Z}^i\}_{i=1}^V$ , 然后将其输入多视角解码器, 得到多视角重构数据  $\{\bar{\mathbf{X}}^i\}_{i=1}^V$ 。

(2)微调阶段。首先,使用预训练阶段得到的多视角编码器参数和多视角解码器参数初始化自编码器。然后,迭代优化编码器参数、解码器参数和自表达层参数(共享自表达系数与私有自表达系数)以获得最优参数。参考文献[15]的训练策略,使用Adam优化器优化目标函数,并设置优化器的初始学习率为 $1.0 \times 10^{-3}$ 。由于加入了自监督规范化项,在训练过程中需利用谱聚类模块更新 $\mathbf{Q}$ 。采用间隔式的方式更新 $\mathbf{Q}$ ,即每隔一定的迭代次数更新一次 $\mathbf{Q}$ ,原因有两点:①由于自表达系数是随机初始化的,在初始迭代阶段,其包含的正确信息不多,这会导致由其构造的相似度矩阵 $\mathbf{S}$ 具有很强的误导性,最终导致聚类指示矩阵 $\mathbf{Q}$ 具有误导性,如果此时利用 $\mathbf{Q}$ 约束自表达系数,会导致自表达系数偏离正确的优化方向,影响迭代速度;②更新 $\mathbf{Q}$ 的过程就是进行谱聚类过程,如果每迭代一次都进行一次谱聚类,将导致很大程度地增加算法的收敛时间,得不偿失。因此,间隔式更新 $\mathbf{Q}$ 既能保证自监督效果,又能节省时间,是权衡时间与效果后的结果。

(3)聚类阶段。在目标函数收敛后,利用 $\mathbf{C}$ 和 $\{\mathbf{D}^i\}_{i=1}^V$ 构造相似度矩阵 $\mathbf{S}$ ,构造算法如式(8)所示。

最后,基于相似度矩阵应用谱聚类得到聚类结果。算法流程如下。

输入:多视角数据 $\{\mathbf{X}^1, \mathbf{X}^2, \dots, \mathbf{X}^V\}$ ,超参数 $\lambda_1 \sim \lambda_5$ ,预训练迭代数 $M'$ ,学习率 $\alpha$ , $\mathbf{Q}$ 的更新周期 $T_0$ ,最大迭代数 $T_{\max}$ 。

```

1  设置 $\lambda_1 = \dots = \lambda_5 = 0$ ,随机初始化自编码器网络参数,设置 $t_1 = 0$ 。
2  WHILE  $t_1 < M'$  DO
3      优化 $L_{\text{RE}}$ ,更新自编码器网络参数;
4       $t_1 \leftarrow t_1 + 1$ ;
5  END WHILE
6  随机初始化 $\mathbf{C}, \mathbf{D}^i (i=1, 2, \dots, V), \mathbf{Q}$ ,使用预训练阶段得到的网络参数初始化自编码器网络,设置 $t=0$ ;
7  WHILE  $t < T_{\max}$  AND 目标函数未收敛 DO
8      IF  $t \% T_0 = 0$  THEN
9          通过谱聚类模块更新 $\mathbf{Q}$ ;
10     END IF
11     优化目标函数,更新整个网络参数;
12      $t \leftarrow t + 1$ ;
13 END WHILE
14 构造相似度矩阵,应用谱聚类得到聚类结果。
输出:聚类结果

```

## 3 实验

### 3.1 实验设置

3.1.1 数据集 本文使用6种广泛应用的多视角数据集进行实验。Yale是一个包含15人共165张人脸图像的数据集,每张图像的光线、人脸的表情等特征均有所不同。ORL是一个包含40种不同对象的数据集,每个对象含10张在不同光线、面部表情和脸部细节下采集的图像。对于Yale和ORL数据集,提取强度、局部二值模式(LBP)和Gabor共3种图像特征作为3个视角。BBCSport<sup>[8]</sup>数据集里包含了BBC Sport网站的544篇运动新闻文章,文章涵盖两个视角下的5种专题领域。3sources<sup>[30]</sup>数据集包含BBC、Reuters和Guardian共3个视角的新闻,共有169篇文章和6个主题。Newsgroups(NGs)<sup>[30]</sup>数据集是20Newsgroups数据集的子集,由3个视角的500个样本构成。Movie数据集包含了17个电影种类下的617部电影,由演员和关键字两种视角描述。

3.1.2 评价指标 本文使用4种评价指标评估聚类算法的聚类效果:归一化互信息 $N$ 、准确率 $A$ 、 $F$ 分数 $F$ 和修正随机指数 $R$ 。4种指标都是值越大代表聚类效果更好。

(1)归一化互信息。计算预测结果与实际结果的归一化相似度,定义为

$$N = \frac{\sum_{i,j=1}^k \log(Bn_{i,j}/n_i n_j)}{\sqrt{(\sum_{i=1}^k n_i \log(n_i/B))(\sum_{j=1}^k n_j \log(n_j/B))}} \quad (11)$$

式中: $B$ 是样本总数; $n_i$ 和 $n_j$ 分别为真实簇标签是第 $i$ 簇和第 $j$ 簇的样本数; $n_{i,j}$ 为真实簇标签是第 $i$ 簇而聚类结果标签为第 $j$ 簇的样本数。

(2)准确率。用来评估聚类的准确率,定义为

$$A = \max_f \frac{\sum_{i=1}^B \{l_i = f(c_i)\}}{B} \quad (12)$$

式中: $l_i$ 是真实簇标签; $c_i$ 是聚类结果簇标签; $f(c_i)$ 是映射函数,将聚类标签映射到真实标签; $\{l_i = f(c_i)\}$ 的定义为

$$\{l_i = f(c_i)\} = \begin{cases} 0, & l_i \neq f(c_i) \\ 1, & l_i = f(c_i) \end{cases} \quad (13)$$

(3) $F$ 分数。基于准确率与召回率的评价指标,定义为

$$F = \frac{2AK}{A+K} \quad (14)$$

式中  $K$  为召回率。

(4)修正随机指数。随机指数  $r$  的改良版本,定义为

$$R = \frac{r - E(r)}{\max(r) - E(r)}; r = \frac{a + b}{C_b^2} \quad (15)$$

式中: $a$  是在真实簇标签和聚类结果中是同一簇标签的样本数; $b$  是在真实簇标签和聚类结果中是不同一簇标签的样本数; $C_b^2$  是样本所有可能组合数; $E(r)$  是  $r$  的期望。

3.1.3 对比算法 对比算法选取 2 种单视角聚类算法和 8 种多视角聚类算法,多视角聚类算法包括 5 种经典算法和 3 种深度学习算法。

单视角聚类算法:①BSV,对多个视角进行谱聚类,选择最优的聚类效果;②LRRbest<sup>[2]</sup>,使用 LRR 算法对多个视角进行聚类,选择最优的聚类效果。

多视角聚类算法:①特征拼接(FeaCon),合并所有视角的特征,对合并的数据进行谱聚类;②Co-Reg<sup>[6]</sup>,通过协同约束的算法隐式地将多视角的图融合,利用一个视角的数据去优化其他视角特征向量矩阵的学习;③RMSC<sup>[8]</sup>,通过低秩稀疏分解得到共享的概率转移矩阵,将其输入基于马尔可夫链的谱聚类算法进行聚类;④LMSC<sup>[9]</sup>,寻找数据的潜在表示空间,在潜在表示空间进行子空间聚类;⑤CSMSC<sup>[4]</sup>,在原始数据空间同时学习共享自表达系数矩阵和私有自表达系数矩阵,最后利用其构造邻接矩阵进行谱聚类;⑥DMSCN<sup>[21]</sup>,用卷积神经网络提取深层数据特征,利用自表达层融合多视角数据信息得到共享自表达系数矩阵,使用共享自表达系数构造相似度矩阵进行谱聚类;⑦Ae2-nets<sup>[29]</sup>,利

用自编码器非线性提取深层数据特征,通过最优化退化网络的输出与自编码器中间层数据的损失来优化退化网络以及多视角数据共享的潜在特征表示,在目标函数收敛后将潜在特征表示输入  $K$ -means 算法得到聚类结果;⑧DMSC-UDL<sup>[24]</sup>,利用自编码器将数据映射到非线性空间,通过约束多视角数据潜在特征和自表达系数,同时挖掘多视角数据的互补与一致信息。

DMSC-UDL 与本文算法相同之处在于:二者均利用自编码器将多视角数据映射到非线性空间,以更好地适应具有非线性关系的子空间;二者都同时挖掘了多视角数据的互补性与一致性信息以提升聚类效果。不同之处在于:DMSC-UDL 在自表达过程中仅学习了公共的自表达系数,未学习私有自表达系数,且是通过多视角数据的潜在空间表示进行约束来挖掘多视角数据间的互补信息,本文算法在自表达过程中同时学习了公共的自表达系数和私有自表达系数,且通过对私有自表达系数进行约束来挖掘互补信息;DMSC-UDL 未加入自监督约束,而是加入了局部结构约束来挖掘数据的局部特征,本文算法利用聚类信息监督自表达系数的学习过程,以学习到更好的自表达系数,提升聚类效果。

3.2 结果分析

3.2.1 对比实验 表 1~6 展示了 C2MSCN 算法与对比算法在 6 个数据集上的 4 种评价指标效果。调节所有对比算法的超参数,记录其最优效果。对所有算法均运行 30 次,并在表中以“平均值(标准差)”的形式记录。表中效果最好的数据用加粗表示,次好的用加粗下划线表示,再次的用下划线表示。

表 1 Yale 数据集对比实验结果

Table 1 Experimental results of different algorithms on the Yale data set

| 算法       | N                          | A                          | R                          | F                          |
|----------|----------------------------|----------------------------|----------------------------|----------------------------|
| BSV      | 0.656(0.033)               | 0.614(0.048)               | 0.426(0.052)               | 0.463(0.049)               |
| FeaCon   | 0.593(0.015)               | 0.394(0.023)               | 0.353(0.025)               | 0.550(0.030)               |
| Co-Reg   | 0.648(0.002)               | 0.564(0)                   | 0.436(0.002)               | 0.466(0)                   |
| LRRbest  | 0.713(0.009)               | 0.703(0.012)               | 0.515(0.004)               | 0.556(0.015)               |
| RMSC     | 0.684(0.033)               | 0.642(0.036)               | 0.485(0.046)               | 0.517(0.043)               |
| LMSC     | 0.702(0.013)               | 0.670(0.012)               | 0.472(0.018)               | 0.506(0.018)               |
| CSMSC    | <u>0.784(0.001)</u>        | 0.752(0.001)               | <u>0.615(0.005)</u>        | 0.640(0.004)               |
| DMSCN    | 0.769(0.003)               | <u>0.772(0.004)</u>        | 0.582(0.012)               | 0.598(0.006)               |
| Ae2-nets | 0.748(0.026)               | 0.718(0.037)               | 0.566(0.050)               | <u>0.663(0.030)</u>        |
| DMSC-UDL | <b><u>0.796(0.002)</u></b> | <b><u>0.832(0.007)</u></b> | <b><u>0.713(0.001)</u></b> | <b><u>0.732(0.004)</u></b> |
| C2MSCN   | <b>0.852(0.003)</b>        | <b>0.835(0.006)</b>        | <b>0.723(0.008)</b>        | <b>0.740(0.008)</b>        |

表 2 ORL 数据集对比实验结果

Table 2 Experimental results of different algorithms on the ORL data set

| 算法       | <i>N</i>                   | <i>A</i>                   | <i>R</i>                   | <i>F</i>                   |
|----------|----------------------------|----------------------------|----------------------------|----------------------------|
| BSV      | 0.888(0.013)               | 0.683(0.031)               | 0.675(0.032)               | 0.753(0.025)               |
| FeaCon   | 0.811(0.015)               | 0.531(0.033)               | 0.519(0.034)               | 0.639(0.031)               |
| Co-Reg   | 0.843(0.008)               | 0.725(0.021)               | 0.602(0.004)               | 0.598(0.065)               |
| LRRbest  | 0.886(0.006)               | 0.754(0.006)               | 0.713(0.002)               | 0.707(0.007)               |
| RMSC     | 0.872(0.012)               | 0.723(0.025)               | 0.635(0.029)               | 0.663(0.003)               |
| LMSC     | 0.931(0.016)               | 0.819(0.017)               | 0.769(0.044)               | 0.758(0.009)               |
| CSMSC    | <b><u>0.936(0.008)</u></b> | <u>0.850(0.020)</u>        | <u>0.805(0.023)</u>        | <u>0.810(0.022)</u>        |
| DMSCN    | 0.928(0.001)               | 0.833(0.008)               | 0.790(0.005)               | 0.787(0.001)               |
| Ae2-nets | 0.865(0.004)               | 0.695(0.012)               | 0.595(0.010)               | 0.609(0.021)               |
| DMSC-UDL | <b>0.942(0.002)</b>        | <b><u>0.865(0.001)</u></b> | <b>0.832(0.005)</b>        | <b><u>0.821(0.003)</u></b> |
| C2MSCN   | <u>0.933(0.003)</u>        | <b>0.876(0.007)</b>        | <b><u>0.820(0.006)</u></b> | <b>0.824(0.006)</b>        |

表 3 BBCSport 数据集对比实验结果

Table 3 Experimental results of different algorithms on the BBCSport data set

| 算法       | <i>N</i>               | <i>A</i>               | <i>R</i>                   | <i>F</i>                   |
|----------|------------------------|------------------------|----------------------------|----------------------------|
| BSV      | 0.719(0.002)           | 0.766(0.001)           | 0.696(0.002)               | 0.853(0.007)               |
| FeaCon   | 0.607(0.021)           | 0.668(0.021)           | 0.561(0.025)               | 0.722(0.027)               |
| Co-Reg   | 0.737(0.003)           | 0.746(0.006)           | 0.744(0.005)               | 0.805(0.004)               |
| LRRbest  | 0.699(0)               | 0.797(0.001)           | 0.697(0.008)               | 0.761(0)                   |
| RMSC     | 0.818(0.017)           | 0.871(0.022)           | 0.845(0.047)               | 0.869(0.035)               |
| LMSC     | 0.826(0.006)           | 0.900(0.044)           | <b><u>0.893(0.012)</u></b> | 0.887(0.071)               |
| CSMSC    | <u>0.861(0)</u>        | <u>0.955(0)</u>        | <u>0.888(0)</u>            | <u>0.914(0)</u>            |
| DMSCN    | 0.813(0.002)           | 0.888(0.003)           | 0.859(0.001)               | 0.854(0.002)               |
| Ae2-nets | 0.843(0.033)           | 0.948(0.013)           | 0.869(0.011)               | 0.902(0.027)               |
| DMSC-UDL | <b>0.890(0.001)</b>    | <b>0.961(0.003)</b>    | 0.889(0.001)               | <b><u>0.917(0.004)</u></b> |
| C2MSCN   | <b><u>0.880(0)</u></b> | <b><u>0.959(0)</u></b> | <b>0.899(0)</b>            | <b>0.923(0)</b>            |

表 4 3sources 数据集对比实验结果

Table 4 Experimental results of different algorithms on the 3sources data set

| 算法       | <i>N</i>                   | <i>A</i>                   | <i>R</i>                   | <i>F</i>               |
|----------|----------------------------|----------------------------|----------------------------|------------------------|
| BSV      | 0.453(0.049)               | 0.462(0.047)               | 0.320(0.061)               | 0.531(0.074)           |
| FeaCon   | 0.523(0.038)               | 0.488(0.038)               | 0.347(0.054)               | 0.555(0.025)           |
| Co-Reg   | 0.533(0.760)               | 0.543(0.360)               | 0.450(0.010)               | 0.505(0.770)           |
| LRRbest  | 0.634(0.007)               | 0.678(0.013)               | 0.531(0)                   | 0.615(0.018)           |
| RMSC     | 0.612(0.007)               | 0.575(0.014)               | 0.514(0.010)               | 0.562(0.010)           |
| LMSC     | 0.669(0.016)               | 0.704(0.017)               | 0.611(0.023)               | 0.631(0.017)           |
| CSMSC    | 0.693(0.002)               | 0.720(0.007)               | 0.562(0.006)               | 0.660(0.006)           |
| DMSCN    | <b>0.779(0)</b>            | <b>0.899(0)</b>            | <b>0.817(0)</b>            | <u>0.861(0)</u>        |
| Ae2-nets | 0.602(0.001)               | 0.781(0.011)               | 0.660(0.015)               | 0.678(0.007)           |
| DMSC-UDL | <b><u>0.701(0.001)</u></b> | <b><u>0.812(0.003)</u></b> | <b><u>0.731(0.003)</u></b> | <b>0.883(0.001)</b>    |
| C2MSCN   | <u>0.696(0)</u>            | <u>0.792(0)</u>            | <u>0.701(0)</u>            | <b><u>0.876(0)</u></b> |

表 5 NGs 数据集对比实验结果

Table 5 Experimental results of different algorithms on the NGs data set

| 算法       | <i>N</i>                   | <i>A</i>                   | <i>R</i>                   | <i>F</i>                   |
|----------|----------------------------|----------------------------|----------------------------|----------------------------|
| BSV      | 0.241(0.025)               | 0.353(0.005)               | 0.061(0.011)               | 0.268(0.001)               |
| FeaCon   | 0.112(0.014)               | 0.328(0.003)               | 0.035(0.012)               | 0.290(0.018)               |
| Co-Reg   | 0.421(0)                   | 0.494(0.001)               | 0.430(0)                   | 0.481(0.001)               |
| LRRbest  | 0.340(0.020)               | 0.421(0.018)               | 0.354(0.002)               | 0.391(0.005)               |
| RMSC     | 0.519(0.007)               | 0.597(0.006)               | 0.450(0.009)               | 0.520(0.004)               |
| LMSC     | 0.905(0.007)               | 0.970(0.002)               | 0.931(0.003)               | 0.941(0.005)               |
| CSMSC    | <u>0.946(0)</u>            | <u>0.984(0)</u>            | <u>0.960(0)</u>            | <u>0.968(0)</u>            |
| DMSCN    | 0.655(0.008)               | 0.709(0.009)               | 0.544(0.019)               | 0.653(0.012)               |
| Ae2-nets | 0.852(0.017)               | 0.940(0.011)               | 0.850(0.028)               | 0.891(0.018)               |
| DMSC-UDL | <b><u>0.954(0.002)</u></b> | <b><u>0.992(0.003)</u></b> | <b><u>0.973(0.002)</u></b> | <b><u>0.972(0.004)</u></b> |
| C2MSCN   | <b><u>0.958(0.004)</u></b> | <b><u>0.988(0.001)</u></b> | <b><u>0.969(0.003)</u></b> | <b><u>0.975(0.002)</u></b> |

表 6 Movie 数据集对比实验结果

Table 6 Experimental results of different algorithms on the Movie data set

| 算法       | <i>N</i>                   | <i>A</i>                   | <i>R</i>                   | <i>F</i>                   |
|----------|----------------------------|----------------------------|----------------------------|----------------------------|
| BSV      | 0.244(0.007)               | 0.136(0.009)               | 0.077(0.010)               | <b><u>0.235(0.014)</u></b> |
| FeaCon   | 0.252(0.008)               | 0.144(0.008)               | 0.086(0.009)               | <b><u>0.248(0.011)</u></b> |
| Co-Reg   | 0.260(0.001)               | 0.257(0.002)               | 0.087(0.001)               | 0.158(0.001)               |
| LRRbest  | 0.269(0.006)               | 0.276(0.009)               | 0.092(0.001)               | 0.156(0.004)               |
| RMSC     | <u>0.296(0.002)</u>        | <u>0.298(0.004)</u>        | <u>0.133(0.001)</u>        | 0.181(0.002)               |
| LMSC     | 0.279(0.009)               | 0.269(0.013)               | 0.105(0.001)               | 0.160(0.008)               |
| CSMSC    | 0.294(0.007)               | 0.295(0.009)               | 0.131(0.006)               | 0.186(0.006)               |
| DMSCN    | 0.266(0.003)               | 0.295(0.006)               | 0.112(0.007)               | 0.176(0.006)               |
| Ae2-nets | 0.253(0.018)               | 0.283(0.017)               | 0.106(0.014)               | 0.198(0.016)               |
| DMSC-UDL | <b><u>0.331(0.003)</u></b> | <b><u>0.342(0.005)</u></b> | <b><u>0.151(0.001)</u></b> | <u>0.213(0.007)</u>        |
| C2MSCN   | <b><u>0.320(0.003)</u></b> | <b><u>0.320(0.009)</u></b> | <b><u>0.154(0.005)</u></b> | 0.205(0.005)               |

由表 1~6 可知,在 Yale 数据集上,C2MSCN 算法在 *N*、*A*、*R*、*F* 上比第二名算法分别提升了 6.846%、6.337%、10.826 3% 和 7.735 3%。对比传统多视角聚类算法 RMSC 和 LMSC,C2MSCN 算法在所有数据集上的 4 个评价指标均优于它们的。在 ORL 数据集上,C2MSCN 算法的 *N* 低于 CSMSC 算法的 0.3%;在其他数据集上,C2MSCN 算法的 4 个评价指标上均优于 CSMSC 算法的。对比近年来的深度多视角子空间聚类算法 DMSCN 和 Ae2-nets。在 3sources 数据集上,C2MSCN 的 *F* 优于 DMSCN 算法的 1.5%;在其他数据集上,C2MSCN 的 4 个评价指标均优于 DMSCN 和 Ae2-nets 的,这验证了 C2MSCN 算法的优越性。对比类似的深度多视角子空间聚类算法 DMSC-UDL。在

Yale 数据集上,本文算法的 4 个指标均位居第一;在其他数据集上,本文算法与 DMSC-UDL 不相上下。这是因为有些数据集如 BBCSport、3sources 等具有明显的局部结构,挖掘其局部结构会明显提升聚类效果,所以 DMSC-UDL 效果更好,而有些数据集如 Yale 并不具备明显的局部结构,所以本文算法效果更好。

由表 4 可知,在 3sources 数据集上,C2MSCN 的 *A*、*N*、*R* 低于最好算法 DMSCN 的约 10%。这是因为 3sources 数据集是由 BBC、Reuters 和 Guardian 这 3 家知名新闻网站围绕 6 个主题报导的共 169 篇文章组成。由于不同新闻网站的语言风格大相径庭,且语言具有很大的发挥空间,有时带有歧义,导致了即使是同一主题,不同新闻网站报道的文章在

文字层面具有很强的属于新闻网站的独特色彩,这些不一致信息非但不具有互补性,反而会影响聚类结果。所以,对于同时利用多视角的互补性与一致性信息的算法,如 CSMSC、Ae2-nets、C2MSC,过度挖掘多视角间的不一致信息反而会使得聚类效果比仅挖掘多视角间一致性信息的 DMSCN 算法的效果差。

相比其他多视角聚类算法,C2MSCN 算法的优越性主要体现在 3 个方面:①深度卷积网络提取非线性数据特征,经过全连接层同时学习共享自表达系数和私有自表达系数,使 C2MSCN 算法可以在非线性空间中同时探索数据的互补性与一致性信息,在更复杂的子空间关系中充分挖掘了多视角数据的有效信息;②多样性规范化使算法更充分地挖掘多视角数据间的互补信息;③自监督规范化项利用聚类标签信息对自表达模型反馈,有效地监督自表达系数的学习过程。

3.2.2 参数敏感度实验 为了分析不同参数对算法的影响,对所有参数进行了敏感度实验。图 2~6 展示了参数在 Yale 数据集上的敏感度实验结果。在研究某一参数时,固定其他参数。每组参数实验均运行 5 次。

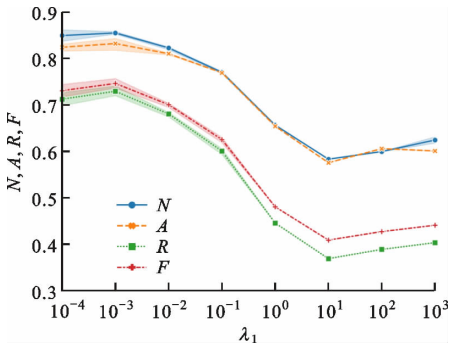


图 2  $\lambda_1$  敏感度实验结果  
Fig. 2  $\lambda_1$  sensitivity experiment results

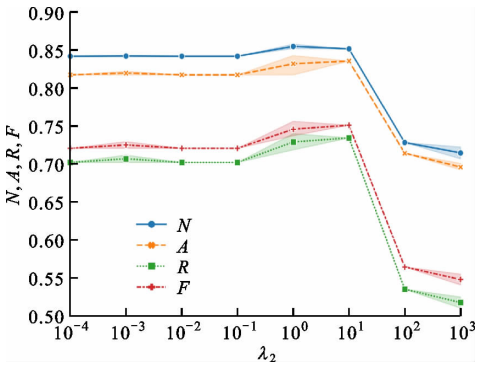


图 3  $\lambda_2$  敏感度实验结果  
Fig. 3  $\lambda_2$  sensitivity experiment results

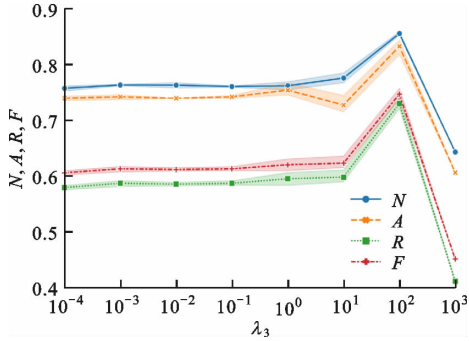


图 4  $\lambda_3$  敏感度实验结果  
Fig. 4  $\lambda_3$  sensitivity experiment results

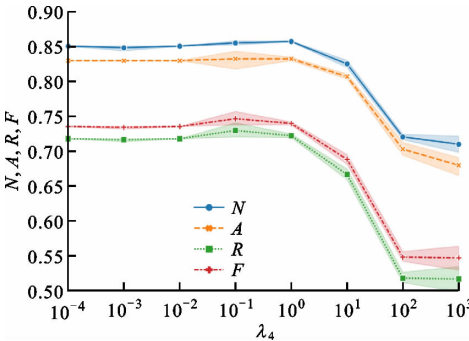


图 5  $\lambda_4$  敏感度实验结果  
Fig. 5  $\lambda_4$  sensitivity experiment results

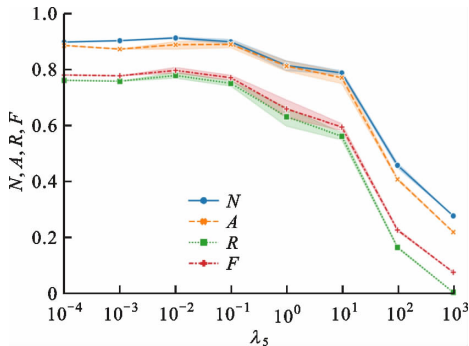


图 6  $\lambda_5$  敏感度实验结果  
Fig. 6  $\lambda_5$  sensitivity experiment results

$\lambda_1 \sim \lambda_5$  取自  $\{10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}, 10^0, 10^1, 10^2, 10^3\}$ 。由于本文算法参数较多,很难在研究单一参数时获取其他参数的最佳值,在权衡时间与效果后,进行了一定的妥协。在进行参数敏感度实验时,首先固定其他参数为 1,在  $[10^{-2}, 10^2]$  内以 10 倍间隔对参数  $\lambda_1$  进行选择;获得此时  $\lambda_1$  最佳值后固定  $\lambda_1$ ,在  $[10^{-2}, 10^2]$  内以 10 倍间隔对参数  $\lambda_2$  进行选择;获得此时  $\lambda_1$  和  $\lambda_2$  最佳值后固定  $\lambda_1$  和  $\lambda_2$ ,在  $[10^{-2}, 10^2]$  内以 10 倍间隔对参数  $\lambda_3$  进行选择,直到获得所有参数的妥协最佳值。当获得所有参数的最佳值后,为测试参数的敏感度,在更大的范围内如

$[10^{-4}, 10^3]$ 以 10 倍间隔对所有参数进行敏感度实验。实验结果显示,在固定其他参数的情况下,当  $\lambda_1$  小于  $10^{-2}$  时,聚类效果良好且在较小的范围内波动。同样,当  $\lambda_2$  小于  $10^1$ 、 $\lambda_2$  小于 100、 $\lambda_5$  小于  $10^{-1}$  时,C2MSCN 均能取得良好的聚类效果,且参数具有较广泛的选择范围。这说明 C2MSCN 的参数选择具有一定的鲁棒性。图 4 表明,选择合适的  $\lambda_3$  对聚类效果具有很大的提升,恰当地挖掘多视角数据间的互补信息是提升多视角聚类效果的关键。

3.2.3 收敛性实验 为验证本文算法的收敛性,对 Yale 数据集进行收敛性实验,结果如图 7 所示。可以看出,在 200 次迭代后,本文算法的聚类效果已达到最优且在小范围内上下波动。在其他数据集上同样如此。由此,本文算法的收敛性得到了验证。

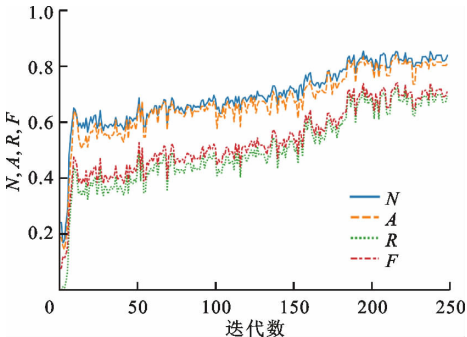


图 7 收敛性实验结果  
Fig. 7 Convergence experiment results

3.2.4 消融性实验 为验证多样性规范化和自监督规范化项的有效性,在 Yale 数据集上对 C2MSCN 算法进行了消融性实验研究,结果如表 7 所示。表中:M1 为不带自监督规范化项和多样性规范化的算法;M2 为只有自监督规范化项的算法;M3 为只有多样性规范化的算法;M4 为 C2MSCN 算法;效果最好的数据用加粗表示。

表 7 在 Yale 数据集上的消融性实验结果 1  
Table 7 Ablation experiment results on Yale: I

| 算法 | N                   | A                   | R                   | F                   |
|----|---------------------|---------------------|---------------------|---------------------|
| M1 | 0.730(0.003)        | 0.727(0.004)        | 0.540(0.007)        | 0.569(0.007)        |
| M2 | 0.755(0.002)        | 0.733(0.005)        | 0.576(0.008)        | 0.602(0.006)        |
| M3 | 0.815(0.003)        | 0.793(0.003)        | 0.645(0.005)        | 0.668(0.006)        |
| M4 | <b>0.852(0.003)</b> | <b>0.835(0.006)</b> | <b>0.723(0.008)</b> | <b>0.740(0.008)</b> |

由表 7 可以看出:M2 性能超过了 M1,说明自监督规范化项对自表达系数矩阵学习的监督有效提升了聚类效果;M3 的性能超越了 M1,说明多样性规范化有效地挖掘多视角间的互补信息,提升了聚

类效果;M4 的性能超过了 M1、M2、M3,说明自监督规范化项和多样性规范化能同时提聚类效果。由此,加入多样性规范化和自监督规范化项的有效性得到了验证。

为验证公有自表达系数和私有自表达系数对聚类效果结果的影响,在 Yale 数据集上对 C2MSCN 算法进行了消融性实验研究,结果如表 8 所示。表中:N1 为只利用式(10)第 1 项构造相似度矩阵  $S$  的算法;N2 为只利用式(10)第 2 项构造相似度矩阵  $S$  的算法;N3 为 C2MSCN 算法;效果最好的数据用加粗表示。

表 8 在 Yale 数据集上的消融性实验结果 2  
Table 8 Ablation experiment results on Yale: II

| 算法 | N                   | A                   | R                   | F                   |
|----|---------------------|---------------------|---------------------|---------------------|
| N1 | 0.811(0.002)        | 0.794(0.001)        | 0.639(0.005)        | 0.662(0.006)        |
| N2 | 0.735(0.004)        | 0.739(0.002)        | 0.551(0.004)        | 0.579(0.005)        |
| N3 | <b>0.852(0.003)</b> | <b>0.835(0.006)</b> | <b>0.723(0.008)</b> | <b>0.740(0.008)</b> |

由表 8 可以看出,N1 的性能超过了 N2,即仅利用多视角数据的一致性信息对聚类性能的影响超越了仅利用多视角数据的互补性信息,说明在本实验中多视角数据的一致性信息包含了更多有效的聚类信息;N3 的性能超过了 N1,说明同时利用多视角数据的互补性与一致性信息能有效地提聚类效果。由此,公有自表达系数和私有自表达系数对结果的影响得到了验证。

4 结 论

多视角子空间聚类算法的研究中,如何在非线性子空间中同时充分地挖掘多视角数据间的互补性与一致性信息是提升聚类效果的关键。针对这个问题,本文提出了一种互补与一致的多视角子空间聚类网络,用来自监督式地在非线性子空间中同时探索多视角数据的互补性与一致性信息,提升聚类效果。本文的主要结论如下。

- (1)提出了一种多视角子空间网络。该网络通过深度自编码器将多视角数据映射到非线性空间,以探索多视角数据的非线性子空间关系。通过规定公共自表达矩阵和私有自表达矩阵,同时探索数据的互补性与一致性信息。利用多样性规范化充分挖掘多视角数据间的互补信息。添加谱聚类模块对中间结果聚类获得聚类标签信息,并利用自监督规范化项监督自表达系数矩阵的学习。
- (2)6 个数据集上的实验结果表明,本文算法的

$N$ 、 $A$ 、 $R$ 、 $F$  这 4 个评价指标在大部分数据集上优于其他算法的,其中在 Yale 数据集上,比近期提出的 CSMSC、DMSCN 和 Ae2-Nets 算法分别提高了 8.3%、6.3%和 11.7%。

(3)由参数敏感度实验可以看出,多样性规范化项参数的稳定性不够,选择范围不够宽,如何更稳定地挖掘互补信息有待后续研究。

## 参考文献:

- [1] ELHAMIFAR E, VIDAL R. Sparse subspace clustering: algorithm, theory, and applications [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(11): 2765-2781.
- [2] LIU Guangcan, LIN Zhouchen, YAN Shuicheng, et al. Robust recovery of subspace structures by low-rank representation [J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(1): 171-184.
- [3] ZHENG Qinghai, ZHU Jihua, LI Zhongyu, et al. Feature concatenation multi-view subspace clustering [J]. Neurocomputing, 2020, 379: 89-102.
- [4] LUO Shirui, ZHANG Changqing, ZHANG Wei, et al. Consistent and specific multi-view subspace clustering [C]//Proceedings of the 32nd AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI, 2018: 3730-3737.
- [5] KUMAR A, DAUME H. A co-training approach for multi-view spectral clustering [C]//Proceedings of the 28th International Conference on Machine Learning. New York, USA: ACM, 2011: 393-400.
- [6] KUMAR A, RAI P, DAUME H. Co-regularized multi-view spectral clustering [C]//Proceedings of the Advances in Neural Information Processing Systems. New York, USA: Curran Associates Inc., 2011: 1413-1421.
- [7] CAO Xiaochun, ZHANG Changqing, FU Huazhu, et al. Diversity-induced multi-view subspace clustering [C]//Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2015: 586-594.
- [8] XIA Rongkai, PAN Yan, DU Lei, et al. Robust multi-view spectral clustering via low rank and sparse decomposition [C]//Proceedings of the 28th AAAI Conference on Artificial Intelligence. Palo Alto, CA, USA: AAAI, 2014: 2149-2155.
- [9] ZHANG Changqing, HU Qinghua, FU Huazhu, et al. Latent multi-view subspace clustering [C]//Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2017: 4333-4341.
- [10] BRBIĆ M, KOPRIVA I. Multi-view low-rank sparse subspace clustering [J]. Pattern Recognition, 2018, 73: 247-258.
- [11] TANG Chang, ZHU Xinzong, LIU Xinwang, et al. Learning a joint affinity graph for multiview subspace clustering [J]. IEEE Transactions on Multimedia, 2019, 21(7): 1724-1736.
- [12] XIE Deyan, ZHANG Xiangdong, GAO Quanxue, et al. Multiview clustering by joint latent representation and similarity learning [J]. IEEE Transactions on Cybernetics, 2020, 50(11): 4848-4854.
- [13] WANG Xiaobo, GUO Xiaojie, LEI Zhen, et al. Exclusivity-consistency regularized multi-view subspace clustering [C]//Proceedings of the 2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2017: 1-9.
- [14] YANG Zhiyong, XU Qianqian, ZHANG Weigang, et al. Split multiplicative multi-view subspace clustering [J]. IEEE Transactions on Image Processing, 2019, 28(10): 5147-5160.
- [15] JI Pan, ZHANG Tong, LI Hongdong, et al. Deep subspace clustering networks [C]//Proceedings of the 31st Advances in Neural Information Processing Systems. Vancouver, Canada: NIPS, 2017: 24-33.
- [16] CHEN Guangliang, ATEV S, LERMAN G. Kernel spectral curvature clustering (KSCC) [C]//Proceedings of the 2009 IEEE 12th International Conference on Computer Vision Workshops. Piscataway, NJ, USA: IEEE, 2009: 765-772.
- [17] PATEL V M, VIDAL R. Kernel sparse subspace clustering [C]//Proceedings of the 2014 IEEE International Conference on Image Processing (ICIP). Piscataway, NJ, USA: IEEE, 2014: 2849-2853.
- [18] YIN Ming, GUO Yi, GAO Junbin, et al. Kernel sparse subspace clustering on symmetric positive definite manifolds [C]//Proceedings of the 2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2016: 5157-5164.
- [19] SHAWE-TAYLOR J, CRISTIANINI N. Kernel methods for pattern analysis [M]. Cambridge, UK: Cambridge University Press, 2004: 290-430.
- [20] ZHOU Pan, HOU Yunqing, FENG Jiashi. Deep adversarial subspace clustering [C]//Proceedings of the 2018 IEEE/CVF Conference on Computer Vision and

- Pattern Recognition. Piscataway, NJ, USA: IEEE, 2018: 1596-1604.
- [21] ABAVISANI M, PATEL V M. Deep multimodal subspace clustering networks [J]. IEEE Journal of Selected Topics in Signal Processing, 2018, 12(6): 1601-1614.
- [22] ZHANG Tong, JI Pan, HARANDI M, et al. Neural collaborative subspace clustering [C]// Proceedings of the 36th International Conference on Machine Learning. Princeton, NJ, USA: IMLS, 2019: 2777-12786.
- [23] ZHU Pengfei, HUI Binyuan, ZHANG Changqing, et al. Multi-view deep subspace clustering networks [EB/OL]. [2020-10-01]. <https://arxiv.org/pdf/1908.01978.pdf>.
- [24] WANG Qianqian, CHENG Jiafeng, GAO Quanxue, et al. Deep multi-view subspace clustering with unified and discriminative learning [J/OL]. IEEE Transactions on Multimedia [2020-10-01]. <https://ieeexplore.ieee.org/stamp/stamp.jsp?tp=&arnumber=9204408>.
- [25] ZHANG Junjian, LI Chunguang, YOU Chong, et al. Self-supervised convolutional subspace clustering network [C]// Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2019: 5468-5477.
- [26] KHEIRANDISHFARD M, ZOHRIZADEH F, KAMANGAR F. Multi-level representation learning for deep subspace clustering [C]// Proceedings of the 2020 IEEE Winter Conference on Applications of Computer Vision (WACV). Piscataway, NJ, USA: IEEE, 2020: 2028-2037.
- [27] 毕志升, 王甲海, 印鉴. 基于差分演化算法的软子空间聚类 [J]. 计算机学报, 2012, 35(10): 2116-2128.  
BI Zhisheng, WANG Jiahai, YIN Jian. Subspace clustering based on differential evolution [J]. Chinese Journal of Computers, 2012, 35(10): 2116-2128.
- [28] 胡世哲, 姜铮铮, 王若彬, 等. 一种双重加权的多视角聚类方法 [J]. 计算机学报, 2020, 43(9): 1708-1720.  
HU Shizhe, LOU Zhengzheng, WANG Ruobin, et al. Dual-weighted multi-view clustering [J]. Chinese Journal of Computers, 2020, 43(9): 1708-1720.
- [29] ZHANG Changqing, LIU Yeqing, FU Huazhu. Ae2-nets: autoencoder in autoencoder networks [C]// Proceedings of the 2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). Piscataway, NJ, USA: IEEE, 2019: 2572-2580.
- [30] WANG Hao, YANG Yan, LIU Bing. GMC: graph-based multi-view clustering [J]. IEEE Transactions on Knowledge and Data Engineering, 2020, 32(6): 1116-1129.

(编辑 陶晴)