

# 融合选择性聚类集成的深度聚类算法

钱悦, 姚世新\*, 吴添君, 方雪垠, 曾令斌

(国防科技大学 教研保障中心, 湖南 长沙 410073)

**摘要:** 为使深度聚类算法在面对不同类型的复杂数据时, 能取得鲁棒性较好的聚类结果, 本文将选择性集成策略与深度聚类算法相结合, 提出了一种融合选择性聚类集成的深度聚类算法, 有效改善了深度聚类算法的鲁棒性和聚类性能。该算法利用不同初始化参数的自编码深度聚类算法生成多个不同的基聚类结果, 构建基聚类的集成相似性和多样性评估度量, 选出一定数目相似性更高、多样性丰富的基聚类作为聚类集成候选, 在聚类集成策略中考虑簇的可靠性来构建加权图共识函数。实验结果表明, 融合了选择性聚类集成的深度聚类算法的鲁棒性得到了提高, 并且在多个不同类型数据上, 相比较许多现有的聚类集成算法获得了更好的聚类结果。

**关键词:** 深度聚类; 选择性集成; 聚类集成

中图分类号: TP391      文献标志码: A      文章编号: XX

## Deep clustering algorithm with fusion selective clustering ensemble

QIAN Yue, YAO Shixin\*, WU Tianjun, FANG Xueyin, ZENG Lingbin

(Center for Teaching and Research Support, National University of Defense Technology, Changsha 410073, China)

**Abstract:** To achieve better robustness of the deep clustering algorithm when facing complex data of different types, this paper combines a selective ensemble strategy with the deep clustering algorithm, proposing a deep clustering algorithm with fusion selective clustering ensemble. This approach effectively enhances the robustness and clustering performance of the deep clustering algorithm. The algorithm utilizes an autoencoder-based deep clustering algorithm with different initialization parameters to generate multiple diverse base clustering results. It constructs measures for ensemble similarity and diversity of base clusterings. A certain number of base clusterings with higher similarity and richer diversity were selected as candidates for clustering ensemble. The clustering ensemble strategy considers the reliability of clusters to construct a weighted graph consensus function. Experimental results demonstrate that the deep clustering algorithm with fusion selective clustering ensemble shows improved robustness and achieves better clustering results on various types of data compared to many existing clustering ensemble algorithms.

**Keywords:** deep clustering; selective ensemble; clustering ensemble

---

收稿日期: 2023-06-13

基金项目: 国家部委基金资助项目 (145BZB210055287X)

第一作者: 钱悦 (1980—), 男, 江苏扬州人, 博士, 正高级工程师, 硕士生导师, E-mail: yueqian@nudt.edu.cn

\*通讯作者: 姚世新 (1994—), 男, 湖北荆州人, 硕士, 工程师, E-mail: yaoshixin22@nudt.edu.cn

信息科技的快速发展产生并沉淀了海量数据,受其影响,大数据技术随之成为一个热门研究领域。大数据技术在各行各业中应用非常广泛<sup>[1]</sup>,典型代表有商业、医疗、金融等领域。大数据在商业领域可以应用于销售预测、供应链管理等,帮助企业做出更好的商业决策<sup>[2]</sup>;在医疗领域可以用于病例分析、药物研发、模型训练等,帮助医疗机构提高医疗水平和效率<sup>[3]</sup>;在金融领域可以服务于风险预警、欺诈检测等,帮助金融从业者更好的管理风险和提高盈利能力<sup>[4]</sup>。然而伴随着数据来源差异化和生成速度指数化增长,数据体量庞大、结构复杂、标记数据严重不足、相关性复杂等特征日益明显。如何从海量数据中高效挖掘出简洁且有意义的信息便成了一个亟待解决的问题。

聚类分析作为一种可以在没有任何先验知识的情况下,从未标记的数据中发现其潜在知识规律和模式的机器学习算法<sup>[5]</sup>,常被用作理解和分析原始数据的有效工具,尤其是对于缺乏先验知识或获取先验知识成本昂贵的问题。常用的聚类算法有  $k$  均值<sup>[6]</sup>、密度聚类<sup>[7]</sup>、层次聚类<sup>[8]</sup>等,这些传统的聚类算法在许多数据上有着不俗的表现。但是,在面对结构复杂、维度高的数据时,传统聚类算法效果并不理想甚至失效。这是因为这些数据复杂度高且分布更为稀疏,传统的聚类算法只能利用数据的浅层特征,难以在更高维度发现数据的内在关联<sup>[9]</sup>。

深度聚类算法<sup>[10,11,12]</sup>的出现为解决这些挑战提供了可行路径,它通过将具有强大表示学习能力的深度神经网络与聚类方法相结合,从复杂数据中提取有利于聚类的低维特征,进而完成聚类任务。然而深度聚类算法效果高度依赖深度神经网络所提取的低维特征,受深度神经网络参数初始化影响明显,鲁棒性较差<sup>[13,14]</sup>。

聚类集成是基于多个基聚类结果,采用一致性函数生成最终聚类结果的方法<sup>[15,16,17]</sup>,它对噪声和异常值具有较强的鲁棒性。近些年聚类集成技术得到了长足的发展。文献[18]通过从稀疏图构建概率轨迹相似度来融合全局信息,最终提出了两种一致性函数:概率轨迹累加(Probability Trajectory Accumulation, PTA)和概率轨迹图分割(Probability Trajectory Graph Partitioning, PTGP)。文献[19]利用社区检测领域和图聚类理念提出了加权谱聚类集成算法(Weighted Spectral Cluster Ensemble, WSCE)。文献[20]通过揭示谱聚类集成与加权  $k$  均值之间的等效关系,提出了  $k$  均值

加权谱聚类集成(Spectral Ensemble Clustering via Weighted K-means, SECWK)方法,大幅降低了谱聚类集成的时空复杂度。文献[21]考虑同一基聚类中簇的局部多样性,提出了两种新的一致性函数:局部加权证据积累(Locally Weighted Evidence Accumulation, LWEA)和局部加权图分割(Locally Weighted Graph Partitioning, LWGP)。文献[22]融合低秩张量逼近和聚类集成方法,提出了一种基于低秩张量逼近(Low-Rank Tensor Approximation, LRTA)的聚类集成新算法。文献[23]利用高度可靠的信息和去噪误差连接,提出了共同关联矩阵自增强集成聚类算法(Ensemble Clustering via Co-association Matrix Self-enhancement, ECCMS),该方法理论上保证了收敛性和全局最优。文献[24]利用密集表示模型构建成对相似性矩阵,通过 Ncut 得到最终的集成聚类结果。文献[25]针对资源有限的超大规模数据集的光谱聚类的可扩展性和鲁棒性,提出超可扩展光谱聚类和超可扩展集成聚类方法。文献[26]通过将主动聚类集成算法嵌入到一个自适应学习框架中,提出了自适应主动聚类集成(Self-Paced Active Clustering Ensemble, SPACE)算法。

为使深度聚类算法在应对复杂数据时,能取得更稳定和质量更高的聚类结果。本文将深度聚类与聚类集成相结合,内嵌选择策略,提出了一种新的融合选择性聚类集成的深度聚类算法(Deep Clustering with Fusion Selective Clustering Ensemble, DCFSCE)。DCFSCE 算法包含面向聚类损失优化的深度基聚类生成和基于标准化互信息度量的选择性集成两部分,前者通过深度神经网络提取特征,融合经典聚类方法生成基聚类;后者基于前者生成的基聚类集合,采取多样化的选择策略生成候选聚类,然后构建加权图,并使用 Tcut<sup>[27]</sup>进行图分割获得最终聚类结果。大量实验证明 DCFSCE 方法高效可行。

## 1 选择性深度聚类集成

### 1.1 面向聚类损失优化的深度基聚类生成方法

在这个阶段主要包含两个步骤:一是预训练深度特征提取器;二是构建深度基聚类生成模型。

#### 1.1.1 预训练深度特征提取器

DCFSCE 使用自编码器<sup>[28]</sup>(Autoencoder, AE)作为提取样本特征的深度神经网络模型。

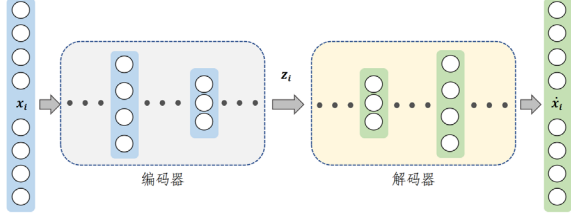


图 1 自编码器网络结构

Fig.1 Autoencoder Network Architecture

如图 1 所示，自编码器由编码器和解码器两部分组成，其重构损失函数  $L_r$  如公式 (1) 所示，函数表征意义为原始数据样本  $x$  与自编码器解码过程生成的重构样本  $\hat{x}$  的欧氏距离。

$$L_r = \|x - \hat{x}\|^2 \quad (1)$$

当自编码器完成训练后，其编码器即为预训练完毕的深度特征提取器。

#### 1.1.2 构建深度基聚类生成模型

深度基聚类生成模型网络结构如图 2 所示。深度基聚类生成模型结构由预训练后的深度特征提取器和聚类层组成。具体而言，输入样本  $x_i$  经深度特征提取器处理后得到特征点  $z_i$ ，聚类层通过学生  $t$  分布将每个特征点  $z_i$  映射到软簇分配  $q_i$ ，并将聚类中心  $\mu_j$  作为可训练权重。软簇分配

$Q = \{q_{ij}\}$  的计算方式为：

$$q_{ij} = \frac{\left(1 + \|z_i - \mu_j\|^2\right)^{-1}}{\sum_j \left(1 + \|z_i - \mu_j\|^2\right)^{-1}} \quad (2)$$

$q_{ij}$  是  $q_i$  的第  $j$  个条目，表示  $z_i$  属于第  $j$  个簇的概率， $j'$  是包含  $j$  的集合，即  $j' = \{1, \dots, j, \dots, k\}$ ， $k$  是聚类簇数。

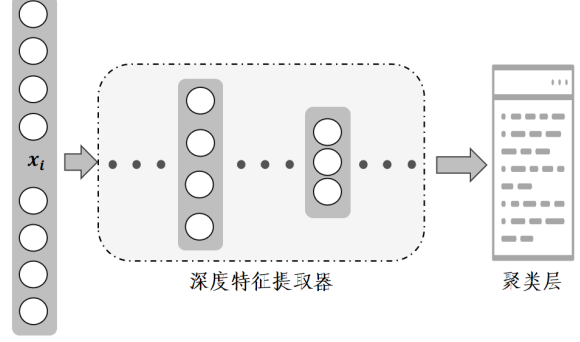


图 2 深度基聚类生成模型结构

Fig.2 Structure of deep base clustering generative model

深度基聚类生成模型的损失，即聚类损失为软簇分配和预定义的辅助目标分布之间的相对熵<sup>[29]</sup>，记作  $L_e$ ， $L_e$  的计算方式为：

$$L_e = KL(P \parallel Q) = \sum_i \sum_j p_{ij} \log \frac{p_{ij}}{q_{ij}} \quad (3)$$

这里  $P$  是辅助分布，通过将软簇分配  $Q$  的平方归一化所得：

$$p_{ij} = \frac{q_{ij}^2 / h_j}{\sum_j q_{ij}^2 / h_j}, h_j = \sum_i q_{ij} \quad (4)$$

随后计算  $L_e$  对每个聚类中心  $\mu_j$  和对每个嵌入点

$z_i$  的梯度，并将  $L_e$  对  $z_i$  的梯度传递给深度神经网络，然后通过反向传播方法调整深度神经网络参数。当连续两次迭代的簇更新小于阈值时，则停止迭代，并输出最终的簇分配。

为了清晰的展示深度基聚类生成模型，在算法 1 中总结了具体步骤。

#### 算法 1 深度基聚类生成

##### Alg.1 Deep base clustering generation

**输入：** 原始数据  $X$ ，基聚类个数  $r$ ，最大迭代次数  $MaxIter$ ，参数更新间隔  $inval$ ，阈值  $\varepsilon$   
**输出：** 深度基聚类集合  $\Pi$

**for** rounds  $\in \{1, 2, \dots, r\}$

    参数初始化

**for** iter  $\in \{1, 2, \dots, MaxIter\}$

**if** iter%inval = 0

            计算所有的嵌入点  $Z$

利用 k-means 获得初始聚类 and 聚类中心

通过  $Z$  更新辅助分布  $P$

获取现有簇分配标记  $pre$  和之前标记  $pre_{before}$

if  $sum(pre_{before} \neq pre) / n < \varepsilon$

停止训练

end if

end if

更新参数

end for

保存第  $rounds$  个基聚类结果

end for

从深度基聚类生成模型损失计算方式不难看出，深度聚类的结果受到深度特征提取器提取的特征质量和网络初始参数影响。为了缓解深度聚类方法应用于复杂数据时鲁棒性较差的问题，本文将把聚类集成和选择策略应用于深度聚类方法。

## 1.2 基于标准化互信息度量的选择性集成

本阶段将分为三步进行处理，首先是对前一阶段生成的深度基聚类进行评估度量；其次根据策略差异，选取相对应的基聚类作为聚类集成候选；最后通过一致性函数将集成候选的基聚类进行融合，得到最终的聚类结果。

### 1.2.1 集成度量

DCFSCE 基于标准化互信息<sup>[30]</sup> (Normalized Mutual Information, NMI) 方法 (详见公式 12)，从相似性和多样性两个维度对基聚类集合  $\Pi = \{\pi^1, \pi^2, \dots, \pi^r\}$  进行评估度量。NMI 主要用于度量两个聚类结果的相近程度，值越高表示结果越相近，是聚类结果评估中常用的一个外部指标。

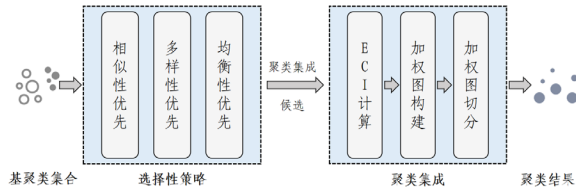


图3 基于标准化互信息度量的选择性集成结构

Fig.3 Structures of selective integration based on Normalized Mutual Information

#### (1) 趋势相似性度量

如公式 (5) 所示，DCFSCE 采用标准化互信息求和 (Summed NMI, SNMI) 的方式对相似性进

行度量。具体来说，SNMI 衡量了一个特定的聚类  $\pi^t (1 \leq t \leq r)$  与集合  $\Pi$  中总体趋势的相似性程度。

$$SNMI(\pi^t, \Pi) = \sum_{i=1}^r NMI(\pi^t, \pi^i) \quad (5)$$

#### (2) 集成多样性度量

关于集成的多样性评估，DCFSCE 采用成对标准化互信息 (Pair-wise NMI, PNMI) 进行度量：

$$PNMI = \sum_{i \neq j, \pi^i, \pi^j \in \Pi} NMI(\pi^i, \pi^j) \quad (6)$$

显然，当  $PNMI$  的值越小时，集成  $\Pi$  中的多样性越高。选择上述的 NMI 作为相似性和多样性评估基础度量，是因为它已被证实会影响聚类集成的效果<sup>[31]</sup>。

#### 1.2.2 选择策略

在定义了相似性和多样性评估标准后，DCFSCE 提供多种策略从集合  $\Pi$  中选取基聚类<sup>[32]</sup>，以作为聚类集成候选，具体如下：

##### (1) 相似性优先

对于给定的集合  $\Pi$ ，相似性优先策略使用  $SNMI(\pi^t, \Pi)$  对  $\Pi$  中所有基聚类进行计算，并按从大到小排序，然后选出前  $s (s \leq r)$  个基聚类作为集成候选。因为较高的  $SNMI$  值表明该聚类与整个集合的总体趋势具有较高相似性，应该优先考虑纳入集成中。

##### (2) 多样性优先

对于给定的集合  $\Pi$ ，多样性优先策略从集合  $\Pi$  中筛选出  $s (s \leq r)$  个基聚类，使得  $PNMI$  值最小，从而确保集成多样性最大化。该问题可以看成求解权重最大子图的问题。考虑该问题是 NP 难的，在多项式时间内无法得到结果，因此 DCFSCE 将采取贪婪策略进行近似求解。

##### (3) 均衡优先

均衡优先策略是指均衡考虑集成中基聚类的相似性以及集成的多样性，即将  $SNMI$  与  $PNMI$  两个度量联合构成一个新的度量  $BNMI$ ：

$$BNMI = \alpha \sum_{i=1,2,\dots,s} SNMI(\pi^i, \Pi) + (1 - \alpha) \sum_{i \neq j} (1 - NMI(\pi^i, \pi^j)) \quad (7)$$

其中  $\alpha$  是调整系数, 用于均衡相似性和多样性的权重占比。

### 1.2.3 聚类集成

在聚类集成步骤中完成集成驱动簇指标构建、一致性函数构建, DCFSCCE 将输出最终的聚类结果。

#### (1) 集成驱动簇指标构建

集成驱动簇指标 (Ensemble-driven Cluster Index, ECI) 主要用于度量簇的可靠性<sup>[21]</sup>, 其定义为:

$$ECI(C_i) = e^{-\frac{H^\Pi(C_i)}{\theta \cdot R}} \quad (8)$$

其中,  $H^\Pi(C_i)$  是度量集合  $\Pi$  中簇  $C_i$  不确定性的指标,  $\theta$  ( $\theta > 0$ ) 是调整簇不确定性影响的参数,  $R$  是集合  $\Pi$  中基聚类的个数。  $H^\Pi(C_i)$  具体计算方式如下所示:

$$H^\Pi(C_i) = \sum_{r=1}^R H^r(C_i) \quad (9)$$

其中  $H^r(C_i)$  是衡量基聚类  $\pi^r = \{C_1^r, C_2^r, \dots, C_{n^r}^r\}$  与簇  $C_i$  之间不确定性的指标<sup>[21]</sup>,  $\pi^r$  表示  $\Pi$  中的第  $r$  个基聚类,  $n^r$  是基聚类  $\pi^r$  的簇数,  $C_j^r$  是  $\pi^r$  的第  $j$  个簇。

$H^r(C_i)$  具体计算如下:

$$H^r(C_i) = -\sum_{j=1}^{n^r} \rho(C_i, C_j^r) \log_2 \rho(C_i, C_j^r) \quad (10)$$

其中  $\rho(C_i, C_j^r) = \frac{|C_i \cap C_j^r|}{|C_i|}$ 。  $|C_i|$  表示的簇  $C_i$  中样本的数量。

显然, 当簇  $C_i$  的不确定性越小,  $ECI$  的值就越大,  $ECI(C_i)$  的取值范围为  $(0, 1]$ 。

#### (2) 一致性函数构建

在  $ECI$  的基础上, 利用加权图构建一致性函

数, 加权图定义为  $G = (V, L)$ , 其中  $V$  为节点集,

$L$  为边集。两个节点  $v_i$  和  $v_j$  之间的边权重为:

$$l_{ij} = \begin{cases} ECI(v_j) & \text{当 } v_i \in X, v_j \in \pi, v_i \in v_j \\ ECI(v_i) & \text{当 } v_j \in X, v_i \in \pi, v_j \in v_i \\ 0 & \text{否则} \end{cases} \quad (11)$$

DCFSCCE 使用 Tcut 算法对加权图进行划分, 得到最终的聚类结果。为了清晰的展示选择性聚类集成阶段, 在算法 2 中总结了具体步骤。

#### 算法 2 选择性聚类集成

##### Alg.2 Selective clustering ensemble

**输入:**  $r$  个深度基聚类结果

$\Pi = \{\pi^1, \pi^2, \dots, \pi^r\}$ , 调整簇不确定性的参数  $\theta$ , 候选集成规模  $s$ , 选择策略。

**输出:** 集成聚类结果,  $s$  个满足选择性策略的候选基聚类  $\Pi^s = \{\pi^{i_1}, \pi^{i_2}, \dots, \pi^{i_s}\}$

确定选择策略

根据选择的方案计算出  $s$  个候选基聚类  $\Pi^s$

计算  $\Pi^s$  中的簇不确定性  $H^{\Pi^s}(C_i)$

计算  $\Pi^s$  中的 ECI 度量

如公式(11)构建加权图

采用 Tcut 算法对加权图进行图划分

获得最终的集成聚类结果

## 2 实验结果与分析

### 2.1 实验数据与聚类评估

为了验证 DCFSCCE 算法的有效性, 这一节中在 6 个数据集上进行了一系列实验, 将 DCFSCCE 与 11 种聚类集成算法进行了比较。其中 Cars 和 Iris 是常用的 UCI 数据集, MNIST 和 Mnist5 是常用图像数据集, Strain 和 Species 是经典的微生物 DNA 序列数据集。数据集的相关统计汇总信息如表 1 所示。

表 1 数据集统计信息

Tab.1 Statistics of the datasets

| 数据名称    | 样本数量  | 特征维度 | 类别数 |
|---------|-------|------|-----|
| Cars    | 392   | 8    | 3   |
| Iris    | 150   | 4    | 3   |
| MNIST   | 70000 | 784  | 10  |
| Mnist5  | 3495  | 784  | 10  |
| Strain  | 9401  | 200  | 20  |
| Species | 37585 | 232  | 101 |

聚类性能的评估指标采用两种广泛使用的评价指标，即标准化互信息和调整兰德指数<sup>[33]</sup> (Adjusted Rand Index, ARI)。值得一提的是，这两个指标的值越大，表明聚类结果越好。

NMI 的计算方式如公式 12 所示：

$$NMI = \frac{\sum_k \sum_t n_{k,t} \log \left( \frac{n \times n_{k,t}}{n_k n_t} \right)}{\sqrt{\left( \sum_k n_k \log \left( \frac{n_k}{n} \right) \right) \left( \sum_t n_t \log \left( \frac{n_t}{n} \right) \right)}} \quad (12)$$

其中  $n_k$  和  $n_t$  分别代表簇  $\mathcal{C}_k$  和  $\mathcal{C}_t$  中的样本数， $\mathcal{C}_k$

和  $\mathcal{C}_t$  分别是由聚类算法和真实标记生成的簇，

$n_{k,t}$  是表示同时在  $\mathcal{C}_k$  和  $\mathcal{C}_t$  中出现的样本数。

ARI 的计算方式如式 13 所示：

$$ARI(\mathcal{C}_k, \mathcal{C}_t) = \frac{2(N_{00}N_{11} - N_{01}N_{10})}{(N_{00} + N_{01})(N_{01} + N_{11}) + (N_{00} + N_{10})(N_{10} + N_{11})} \quad (13)$$

其中  $N_{11}$  表示样本对同时出现在  $\mathcal{C}_k$  和  $\mathcal{C}_t$  中的数量， $N_{00}$  表示样本对出现在不同  $\mathcal{C}_k$  和  $\mathcal{C}_t$  中的数量，

$N_{10}$  表示样本对出现在  $\mathcal{C}_k$  但在不同的  $\mathcal{C}_t$  中的数量， $N_{01}$  表示样本对出现在不同的  $\mathcal{C}_k$  但在相同的

$\mathcal{C}_t$  中的数量。

## 2.2 实验设置与聚类效果对比

本文提出的 DCFSC 算法与 11 种主流的聚类集成算法进行了对比，这些算法分别是

PTA(PTAAL, PTACL, PTASL)<sup>[18]</sup>，PTGP<sup>[18]</sup>，WSCE<sup>[19]</sup>，SECWK<sup>[20]</sup>，LWEA<sup>[21]</sup>，LWGP<sup>[21]</sup>，LRTA<sup>[22]</sup>，ECCMS<sup>[23]</sup>，SPACE<sup>[26]</sup>。对比算法实验设置如表 2 所示，其余相关参数均按照相应文献中的推荐参数进行设置，最后将 20 次运行结果的 NMI 和 ARI 平均值与方差作为评估依据。

表 2 实验设置

| Tab.2 Setup of experiments |              |           |      |      |
|----------------------------|--------------|-----------|------|------|
| 算法类别                       | 初始聚类规模总数 (r) | 参与集成基聚类规模 | 簇数设置 | 运行次数 |
| PTGP 等 11 种算法              | 50           | 20        | 真实类别 | 20   |
| DCFSC                      | 50           | [10, r/2] | 真实类别 | 20   |

DCFSC 参数设置如表 2 所示，其中参与集成的基聚类规模  $s$  的范围在  $[10, r/2]$ ， $r$  为初始聚类规模总数，调整簇不确定性影响的参数  $\theta$  统一设置为 0.4。本文对三种选择策略均进行了实验，其中均衡优先策略的调整系数  $\alpha = 0.5$ 。同样地，将 20 次运行结果的 NMI 和 ARI 平均值与方差作为评估结果。

聚类性能比较结果见表 3 至表 6，对于每个数据集，最高的三个得分用粗体突出显示。根据 5% 显著性水平的两两 t 检验，表中所有粗体结果在统计学上都优于其他算法。

需要说明的是，DCFSCESF、DCFSCEDF、DCFSCBEF 分别指采用相似性优先、多样性优先、均衡性优先策略的 DCFSC 的算法。

结合表 3 和表 4 可见，在 Iris 和 MNIST 两个数据集上，DCFSC 三种策略的聚类性能都优于其他对比算法。特别是在 MNIST 图像数据集上，DCFSC 取得了优秀的聚类性能，这是因为深度聚类过程会将原始数据映射为低维表示，结合聚类损失后能生成更加聚类友好的深度表示，从而提升聚类性能。WSCE、LRTA、ECCMS 和 SPACE 四种算法无法在常规时间下得出 MNIST 数据集的聚类结果，这里用“timeout”表示。在 Cars 数据集上，SPACE 算法虽然取得了最好的 NMI 分值，但其 ARI 得分的标准差为 0.045，可以得知这并非一个稳定的聚类结果。LWEA 和 LWGP 在 Cars 数据集上取得了最好的 ARI 得分，但其 NMI 得分比 DCFSC 低。综合来看，他们的聚类结果不如 DCFSC。

表 3 DCFSC 与对比算法在 Cars、Iris 和 MNIST 数据集上的平均 NMI 结果

Tab.3 Average NMI results of DCFSC and compared

| method on Cars, Iris, and MNIST dataset |                     |                     |                     |
|-----------------------------------------|---------------------|---------------------|---------------------|
| 算法                                      | Cars                | Iris                | MNIST               |
| PTAAL                                   | 0.139(0.055)        | 0.547(0.216)        | 0.497(0.013)        |
| PTACL                                   | 0.142(0.055)        | 0.554(0.207)        | 0.491(0.016)        |
| PTASL                                   | 0.133(0.061)        | 0.498(0.213)        | 0.038(0.034)        |
| PTGP                                    | 0.167(0.026)        | 0.741(0.016)        | 0.494(0.017)        |
| WSCE                                    | 0.246(0.001)        | <b>0.758(0.001)</b> | timeout             |
| SECWK                                   | 0.104(0.070)        | 0.572(0.223)        | 0.470(0.026)        |
| LWEA                                    | 0.207(0.003)        | 0.748(0.008)        | 0.506(0.007)        |
| LWGP                                    | 0.206(0.004)        | 0.747(0.008)        | 0.498(0.004)        |
| LRTA                                    | 0.005(0.003)        | 0.015(0.012)        | timeout             |
| ECCMS                                   | 0.206(0.003)        | 0.747(0.008)        | timeout             |
| SPACE                                   | <b>0.294(0.016)</b> | 0.497(0.039)        | timeout             |
| DCFSCESEF                               | <b>0.259(0.002)</b> | <b>0.764(0.012)</b> | <b>0.787(0.001)</b> |
| DCFSCEDF                                | 0.232(0.008)        | 0.755(0.005)        | <b>0.758(0.002)</b> |
| DCFSCEBF                                | <b>0.259(0.002)</b> | <b>0.774(0.008)</b> | <b>0.789(0.001)</b> |

表 4 DCFSCCE 与对比算法在 Cars、Iris 和 MNIST 数据集上的平均 ARI 结果

Tab.4 Average ARI results of DCFSCCE and compared method on Cars, Iris, and MNIST dataset

| 算法        | Cars                | Iris                | MNIST               |
|-----------|---------------------|---------------------|---------------------|
| PTAAL     | -0.018(0.115)       | 0.409(0.198)        | 0.382(0.024)        |
| PTACL     | -0.019(0.115)       | 0.427(0.190)        | 0.372(0.022)        |
| PTASL     | -0.015(0.114)       | 0.363(0.187)        | 0.003(0.004)        |
| PTGP      | -0.026(0.107)       | 0.682(0.071)        | 0.377(0.032)        |
| WSCE      | 0.053(0.001)        | <b>0.725(0.001)</b> | timeout             |
| SECWK     | -0.034(0.046)       | 0.467(0.220)        | 0.346(0.036)        |
| LWEA      | <b>0.072(0.003)</b> | 0.722(0.007)        | 0.397(0.014)        |
| LWGP      | <b>0.073(0.003)</b> | 0.721(0.007)        | 0.384(0.008)        |
| LRTA      | -0.001(0.004)       | 0.003(0.014)        | timeout             |
| ECCMS     | 0.074(0.003)        | 0.721(0.007)        | timeout             |
| SPACE     | 0.059(0.045)        | 0.388(0.128)        | timeout             |
| DCFSCESEF | 0.058(0.002)        | <b>0.732(0.012)</b> | <b>0.801(0.002)</b> |
| DCFSCEDF  | <b>0.146(0.015)</b> | 0.725(0.007)        | <b>0.742(0.002)</b> |
| DCFSCEBF  | 0.058(0.002)        | <b>0.740(0.009)</b> | <b>0.801(0.001)</b> |

表 5 DCFSCCE 与对比算法在 Mnist5、Strain 和 Species 数据集上的平均 NMI 结果

Tab.5 Average NMI results of DCFSCCE and compared method on Mnist5, Strain and Species dataset

| 算法    | Mnist5       | Strain       | Species      |
|-------|--------------|--------------|--------------|
| PTAAL | 0.495(0.012) | 0.815(0.045) | 0.899(0.007) |
| PTACL | 0.501(0.011) | 0.813(0.040) | 0.894(0.008) |
| PTASL | 0.354(0.080) | 0.788(0.050) | 0.914(0.011) |
| PTGP  | 0.500(0.011) | 0.833(0.038) | 0.860(0.019) |
| WSCE  | 0.543(0.008) | 0.936(0.001) | timeout      |
| SECWK | 0.470(0.028) | 0.762(0.056) | 0.836(0.016) |
| LWEA  | 0.513(0.006) | 0.945(0.007) | 0.981(0.005) |

|           |                     |                     |                     |
|-----------|---------------------|---------------------|---------------------|
| LWGP      | 0.495(0.005)        | 0.947(0.008)        | 0.986(0.003)        |
| LRTA      | 0.528(0.007)        | 0.950(0.007)        | timeout             |
| ECCMS     | 0.345(0.114)        | 0.946(0.006)        | timeout             |
| SPACE     | 0.444(0.006)        | 0.861(0.011)        | timeout             |
| DCFSCESEF | <b>0.652(0.006)</b> | <b>0.959(0.002)</b> | <b>0.993(0.002)</b> |
| DCFSCEDF  | <b>0.652(0.007)</b> | <b>0.959(0.001)</b> | <b>0.993(0.003)</b> |
| DCFSCEBF  | <b>0.668(0.002)</b> | <b>0.958(0.001)</b> | <b>0.993(0.001)</b> |

表 6 DCFSCCE 与对比算法在 Mnist5、Strain 和 Species 数据集上的平均 ARI 结果

Tab.6 Average ARI results of DCFSCCE and compared method on Mnist5, Strain and Species dataset

| 算法        | Mnist5              | Strain              | Species             |
|-----------|---------------------|---------------------|---------------------|
| PTAAL     | 0.372(0.020)        | 0.567(0.142)        | 0.463(0.037)        |
| PTACL     | 0.380(0.019)        | 0.578(0.126)        | 0.449(0.046)        |
| PTASL     | 0.174(0.084)        | 0.476(0.137)        | 0.582(0.068)        |
| PTGP      | 0.382(0.018)        | 0.673(0.106)        | 0.353(0.067)        |
| WSCE      | 0.377(0.016)        | 0.906(0.001)        | timeout             |
| SECWK     | 0.317(0.048)        | 0.410(0.138)        | 0.175(0.034)        |
| LWEA      | 0.393(0.009)        | 0.932(0.024)        | 0.846(0.021)        |
| LWGP      | 0.370(0.006)        | 0.934(0.024)        | 0.869(0.018)        |
| LRTA      | 0.404(0.013)        | 0.932(0.024)        | timeout             |
| ECCMS     | 0.170(0.073)        | 0.894(0.036)        | timeout             |
| SPACE     | 0.319(0.003)        | 0.724(0.037)        | timeout             |
| DCFSCESEF | <b>0.575(0.005)</b> | <b>0.946(0.002)</b> | <b>0.972(0.004)</b> |
| DCFSCEDF  | <b>0.575(0.013)</b> | <b>0.946(0.003)</b> | <b>0.971(0.003)</b> |
| DCFSCEBF  | <b>0.589(0.003)</b> | <b>0.945(0.001)</b> | <b>0.972(0.004)</b> |

表 5 和表 6 展现 DCFSCCE 和对比方法在 Mnist5、Strain 和 Species 三个数据集的聚类结果。由于在 MNIST 数据中 4 个对比算法无法在常规时间得出聚类结果，因此在 Mnist5 图像数据集上再次进行实验。可以看到 DCFSCCE 三种策略的聚类性能依旧优于其他对比算法。同时在 Strain 和 Species 两个经典微生物 DNA 序列数据上，DCFSCCE 同样取得了最好的聚类结果。WSCE、LRTA、ECCMS 和 SPACE 四种算法同样无法在常规时间下得出 Species 数据集的聚类结果。综合上述结果，DCFSCCE 在多种不同类型数据上相较于已有的聚类集成算法，确实能达到较好的聚类性能和稳定性，在真实应用的数据聚类中也具有不错的效果，这也表明 DCFSCCE 算法对不同类型数据都具有一定的适用性，实际应用场景广泛，在对原始数据进行收集处理后，就能应用 DCFSCCE 算法进行聚类。

实验记录了每个算法在不同数据上的执行时间，由于涉及的算法和实验数据较多，因此在本节中不进行展示。在对 Cars 和 Iris 的实验中，观察到除 LRTA 和 SPACE 算法外，其余算法都

在 1 秒内获得结果。这表明这些算法在较小的数据规模上能快速获得集成结果。然而随着样本量逐渐增加。从图像数据集 Mnist5 和 MNIST 的执行时间统计发现, SECWK 算法是计算速度最快的, 其次是 LWGP、LWEA 算法和 DCFSCSCE 算法。DCFSCSCE 集成的效率同 LWGP 近似, 但由于引入了选择策略, 因此算法执行时间比 LWGP 稍久。

运行效率较慢的是 WSCE, LRTA, ECCMS 和 SPACE 算法。WSCE 效率较慢可能是因为涉及核谱聚类, 需要准确计算特征值; LRTA 则是需要计算链接矩阵和共关联矩阵形成的三维张量, 随着样本和维度的增加, 这一过程耗时急剧增加; ECCMS 从理论上保证了收敛性和全局最优, 但同时涉及大量的矩阵分解, 这可能是其面对大规模数据无法在合理时间得出最终结果的原因; SPACE 是自适应的集成聚类算法, 需要尝试不同的聚类簇数以及选择查询不确定或不可靠的数据来注释, 因此耗时较多。

### 2.3 DCFSCSCE 消融实验

为了验证 DCFSCSCE 算法的集成有效性和选择策略的有效性, 这一节进行了消融实验。其中, Single 是指单独的自编码深度聚类算法, 用于对比验证集成的有效性; noSCE 是指没有使用选择策略的算法, 用于对比验证选择策略的有效性, 具体实验结果如表 7 至表 10 所示。从表 7 和表 8 的结果来看, 在 Cars、Iris 和 MNIST 三个数据集上, 经过集成后的 noSCE 算法的聚类结果鲁棒性得到了提升, 具体体现在对应的标准差都有所减小, 特别是在 Iris 数据上, 集成后标准差从 0.070 降低到 0.009。在融入选择性策略后, 聚类性能和鲁棒性都有进一步的提升。Cars 数据集成后的 ARI 值不如集成前的单独自编码深度聚类结果, 这可能是因为在选择策略中 DCFSCSCE 更偏向考虑 NMI 结果更高的进行集成。结合表 9 和表 10 在 Mnist5、Strain 和 Species 数据上的结果可以分析出类似地结果。因而这小节这些消融实验的结果证明了 DCFSCSCE 算法中集成的有效性和选择性策略的有效性。

表 7 消融实验在 Cars、Iris 和 MNIST 数据集上的平均 NMI 结果

Tab.7 Average NMI results of ablation experiment on Cars, Iris, and MNIST dataset

| 算法     | Cars         | Iris         | MNIST        |
|--------|--------------|--------------|--------------|
| Single | 0.224(0.036) | 0.732(0.070) | 0.750(0.025) |

|          |                     |                     |                     |
|----------|---------------------|---------------------|---------------------|
| noSCE    | 0.232(0.025)        | 0.754(0.009)        | 0.758(0.012)        |
| DCFSCESF | <b>0.259(0.002)</b> | <b>0.764(0.012)</b> | <b>0.787(0.001)</b> |
| DCFSCEDF | <b>0.232(0.008)</b> | <b>0.755(0.005)</b> | <b>0.758(0.002)</b> |
| DCFSCBEF | <b>0.259(0.002)</b> | <b>0.774(0.008)</b> | <b>0.789(0.001)</b> |

表 8 消融实验在 Cars、Iris 和 MNIST 数据集上的平均 ARI 结果

Tab.8 Average ARI results of ablation experiment on Cars, Iris, and MNIST dataset

| 算法       | Cars                | Iris                | MNIST               |
|----------|---------------------|---------------------|---------------------|
| Single   | 0.085(0.072)        | 0.719(0.091)        | 0.739(0.047)        |
| noSCE    | 0.056(0.012)        | 0.721(0.009)        | 0.745(0.013)        |
| DCFSCESF | 0.058(0.002)        | <b>0.732(0.012)</b> | <b>0.801(0.002)</b> |
| DCFSCEDF | <b>0.146(0.015)</b> | <b>0.725(0.007)</b> | <b>0.742(0.002)</b> |
| DCFSCBEF | 0.058(0.002)        | <b>0.740(0.009)</b> | <b>0.801(0.001)</b> |

表 9 消融实验在 Mnist5、Strain 和 Species 数据集上的平均 NMI 结果

Tab.9 Average NMI results of ablation experiment on Mnist5, Strain and Species dataset

| 算法       | Mnist5              | Strain              | Species             |
|----------|---------------------|---------------------|---------------------|
| Single   | 0.620(0.029)        | 0.943(0.007)        | 0.989(0.003)        |
| noSCE    | 0.643(0.021)        | 0.955(0.005)        | 0.991(0.003)        |
| DCFSCESF | <b>0.652(0.006)</b> | <b>0.959(0.002)</b> | <b>0.993(0.002)</b> |
| DCFSCEDF | <b>0.652(0.007)</b> | <b>0.959(0.001)</b> | <b>0.993(0.003)</b> |
| DCFSCBEF | <b>0.668(0.002)</b> | <b>0.958(0.001)</b> | <b>0.993(0.001)</b> |

表 10 消融实验在 Mnist5、Strain 和 Species 数据集上的平均 ARI 结果

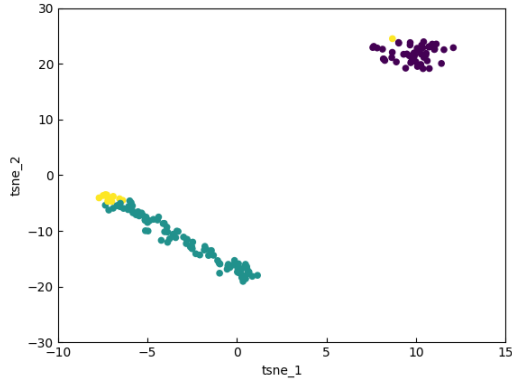
Tab.10 Average ARI results of ablation experiment on Mnist5, Strain and Species dataset

| 算法       | Mnist5              | Strain              | Species             |
|----------|---------------------|---------------------|---------------------|
| Single   | 0.515(0.049)        | 0.928(0.030)        | 0.948(0.034)        |
| noSCE    | 0.550(0.040)        | 0.938(0.010)        | 0.961(0.021)        |
| DCFSCESF | <b>0.575(0.005)</b> | <b>0.946(0.002)</b> | <b>0.972(0.004)</b> |
| DCFSCEDF | <b>0.575(0.013)</b> | <b>0.946(0.003)</b> | <b>0.971(0.003)</b> |
| DCFSCBEF | <b>0.589(0.003)</b> | <b>0.945(0.001)</b> | <b>0.972(0.004)</b> |

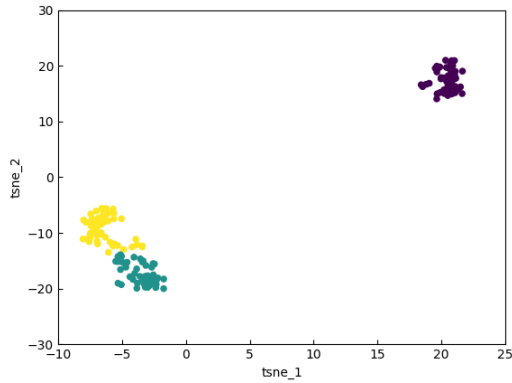
### 2.4 DCFSCSCE 聚类结果可视化

为了验证 DCFSCSCE 能获得良好的聚类结果, 这一小节采用 t-SNE 对 Iris 和 Mnist5 数据集上的自编码深度聚类步骤中产生的初始嵌入和最终嵌入进行了可视化, 初始嵌入是指没有与聚类损失联合优化的嵌入结果, 最终嵌入是指与聚类损失联合优化后的嵌入结果。可视化结果如图 3 和图 4 所示, 图中的 tsne\_1、tsne\_2 横纵坐标是对初始嵌入和最终嵌入进行 t-SNE 后降成二维特征的数值坐标。从该结果可知, 经过聚类损失联合优化后的最终嵌入对聚类更友好, 并且能产生性能更

好的聚类结果。Mnist5 数据的可视化结果对比非常明显，通过联合聚类损失训练的最终嵌入形成了更密集的簇团，混合的簇团及样本更少。对于 Iris 的可视化结果，由于 Iris 数据本身特点是只有一类是线性可分的，其他两类是线性不可分的。



(a) Iris 初始嵌入 t-SNE 可视化  
(a) t-SNE visualization on Iris initial embedding

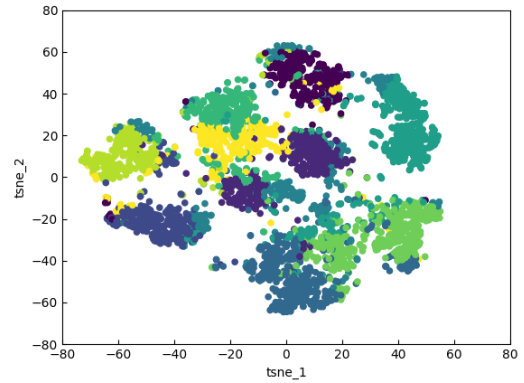


(b) Iris 最终嵌入 t-SNE 可视化  
(b) t-SNE visualization on Iris final embedding

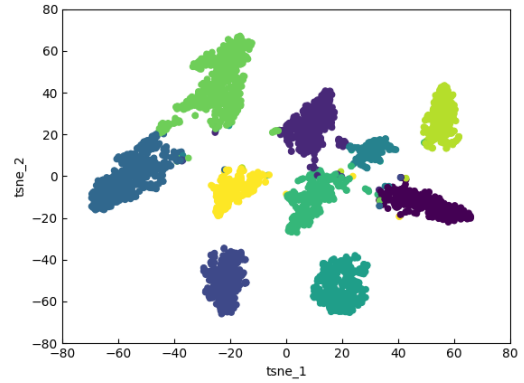
图 3 Iris 深度嵌入 t-SNE 可视化

Fig.3 t-SNE visualization on Iris deep embedding

因此，用 t-SNE 对 Iris 结果进行可视化处理的结果是存在两个无法区分开来的簇。不过，这并不意味着 Iris 的聚类结果差，Iris 的最终聚类性能指标证明了联合聚类损失训练的有效性。



(a) Mnist5 初始嵌入 t-SNE 可视化  
(a) t-SNE visualization on Mnist5 initial embedding



(b) Mnist5 最终嵌入 t-SNE 可视化  
(b) t-SNE visualization on Mnist5 final embedding

图 4 Mnist5 深度嵌入 t-SNE 可视化

Fig.4 t-SNE visualization on Mnist5 deep embedding

### 3 结论

本文提出了一种融合选择性聚类集成的深度聚类 DCFSCCE 算法，通过将选择策略、聚类集成技术、深度聚类算法三者相结合，有效应对现有深度聚类算法面临的挑战，使其在面对不同类型复杂数据时，能取得更稳定和质量更高的聚类结果。通过在不同类型数据上与多种集成聚类算法进行对比实验表明，DCFSCCE 算法产生的聚类结果鲁棒性更强，聚类性能更好，实用性较强，能为聚类技术的发展提供基础支撑。

### 参考文献 (References)

- [1] Rani S, Bhambri P, Kataria A, et al. Big Data, Cloud Computing and IoT: Tools and Applications[M]. America: CRC Press, 2023.

- [2] Mikalef P, Pappas I O, Krogstie J, et al. Big data and business analytics: a research agenda for realizing business value[J]. *Information and Management*, 2020, 57(1): 103237.
- [3] Rehman A, Naz S, Razzak I. Leveraging big data analytics in healthcare enhancement: trends, challenges and opportunities[J]. *Multimedia Systems*, 2022, 28(4): 1339-1371.
- [4] Venkateswara Rao M, Vellela S, Reddy V, et al. Credit Investigation and Comprehensive Risk Management System based Big Data Analytics in Commercial Banking[C]//*Proceedings of International Conference on Advanced Computing and Communication Systems (ICACCS)*, IEEE, 2023: 2387-2391.
- [5] Garcia-Dias R, Vieira S, Pinaya W H L, et al. Clustering analysis[J]. *Machine Learning*, 2020: 227-247.
- [6] HARTIGAN J A, WONG M A. Algorithm AS 136: a K-means clustering algorithm[J]. *Applied Statistics*, 1979, 28(1): 100-108.
- [7] Bhattacharjee P, Mitra P. A survey of density based clustering algorithms[J]. *Frontiers of Computer Science*, 2021, 15: 1-27.
- [8] Murtagh F, Contreras P. Algorithms for hierarchical clustering: an overview[J]. *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, 2012, 2(1): 86-97.
- [9] Ertöz L, Steinbach M, Kumar V. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data[C]//*Proceedings of SIAM international conference on data mining (SDM)*, Society for Industrial and Applied Mathematics, 2003: 47-58.
- [10] Caron M, Bojanowski P, Joulin A, et al. Deep clustering for unsupervised learning of visual features[C]//*Proceedings of the European conference on computer vision (ECCV)*, 2018: 132-149.
- [11] 董永峰, 邓亚哈, 董瑶, 等. 基于深度学习的聚类综述[J]. *计算机应用*, 2022, 42(4): 1021-1028.  
DONG Y F, DENG Y H, DONG Y, et al. Survey of clustering based on deep learning[J]. *Journal of Computer Applications*, 2022, 42(4): 1021-1028.(in Chinese)
- [12] Zong B, Song Q, Min M R, et al. Deep autoencoding gaussian mixture model for unsupervised anomaly detection[C]//*Proceedings of International Conference on Learning Representations (ICLR)*, 2018: 1-19.
- [13] Caron M, Misra I, Mairal J, et al. Unsupervised learning of visual features by contrasting cluster assignments[J]. *Advances in neural information processing systems*, 2020, 33: 9912-9924.
- [14] Wang Z Q, Le Roux J, Hershey J R. Alternative objective functions for deep clustering[C]//*Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, IEEE, 2018: 686-690.
- [15] Golalipour K, Akbari E, Hamidi S S, et al. From clustering to clustering ensemble selection: A review[J]. *Engineering Applications of Artificial Intelligence*, 2021, 104: 104388.
- [16] Alqurashi T, Wang W. Clustering ensemble method[J]. *International Journal of Machine Learning and Cybernetics*, 2019, 10: 1227-1246.
- [17] Wu X, Ma T, Cao J, et al. A comparative study of clustering ensemble algorithms[J]. *Computers and Electrical Engineering*, 2018, 68: 603-615.
- [18] HUANG D, LAI J H, WANG C D. Robust ensemble clustering using probability trajectories[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2016, 28(5): 1312-1326.
- [19] YOUSEFNEZHAD M, ZHANG D Q. Weighted spectral cluster ensemble[C]//*Proceedings of the IEEE International Conference on Data Mining*, 2015: 549-558.
- [20] LIU H F, WU J J, LIU T L, et al. Spectral ensemble clustering via weighted K-means: theoretical and practical evidence[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2017, 29(5): 1129-1143.
- [21] HUANG D, WANG C D, LAI J H. Locally weighted ensemble clustering[J]. *IEEE Transactions on Cybernetics*, 2018, 48(5): 1460-1473.
- [22] JIA Y H, LIU H, HOU J H, et al. Clustering ensemble meets low-rank tensor approximation[J]. *Proceedings of the AAAI Conference on Artificial Intelligence*, 2021, 35(9): 7970-7978.
- [23] JIA Y, TAO S, WANG R, et al. Ensemble clustering via co-association matrix self-enhancement[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2024, 35(8): 11168-11179.
- [24] Zhou J, Zheng H, Pan L. Ensemble clustering based on dense representation[J]. *Neurocomputing*, 2019, 357: 66-76.
- [25] Huang D, Wang C D, Wu J S, et al. Ultra-scalable spectral clustering and ensemble clustering[J]. *IEEE Transactions on Knowledge and Data Engineering*, 2019, 32(6): 1212-1226.
- [26] ZHOU P, SUN B C, LIU X W, et al. Active clustering ensemble with self-paced learning[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2024, 35(9): 12186-12200.
- [27] Li Z, Wu X M, Chang S F. Segmentation using superpixels: a bipartite graph partitioning approach[C]//*Proceedings of IEEE conference on computer vision and pattern recognition (CVPR)*, IEEE, 2012: 789-796.
- [28] Michelucci U. An introduction to autoencoders[J]. *Arxiv preprint arxiv*, 2022:2201.03898.
- [29] Zhou P, Du L, Wang H, et al. Learning a robust consensus matrix for clustering ensemble via Kullback-Leibler divergence minimization[C]//*Proceedings of Twenty-Fourth International Joint Conference on Artificial*

Intelligence (IJCAI), 2015: 4112-4118.

- [30] Estévez P A, Tesmer M, Perez C A, et al. Normalized mutual information feature selection[J]. IEEE Transactions on neural networks, 2009, 20(2): 189-201.
- [31] FERN X Z, LIN W. Cluster ensemble selection[J]. Statistical Analysis and Data Mining: the ASA Data Science Journal, 2008, 1(3): 128-141.
- [32] Abbasi S O, Nejatian S, Parvin H, et al. Clustering ensemble selection considering quality and diversity[J]. Artificial Intelligence Review, 2019, 52(2): 1311-1340.
- [33] Steinley D. Properties of the hubert-arable adjusted rand index[J]. Psychological methods, 2004, 9(3): 386.