第 58 卷　第 4 期
2022 年 7 月

南京大学学报(自然科学)
JOURNAL OF NANJING UNIVERSITY
(NATURAL SCIENCE)

Vol. 58, No. 4
July, 2022

DOI:10.13232/j.cnki.jnju.2022.04.002

基于拓展约束投影的加权半监督聚类集成算法

张　鼎,杨有龙*,孙丽芹
(西安电子科技大学数学与统计学院,西安,710126)

摘　要:半监督聚类集成旨在利用成对约束提升聚类集成的精度,但在高维空间的聚类效果却显著降低,另外,当只有少量的成对约束可以利用时,聚类性能很难提升.针对这些问题,提出一种新颖的半监督聚类集成算法 WSCEC(Weighted Semi-supervised Clustering Ensemble Algorithm Based on Extended Constraint Projection).首先,利用多种聚类算法对数据的特征空间进行聚类,再使用随机子空间进行降维,以减少冗余特征的影响;其次,根据每对约束的 k 个最近或最远的样本以及约束间的传递关系来扩展原有的约束集,通过约束投影技术将原始数据空间投影到低维空间以满足尽可能多的约束;最后,设计了一个聚类解的加权策略,为每一个聚类解分配一个适当的权重以降低低质量聚类解的影响.在多个数据集上的实验结果证明了提出算法的有效性.

关键词:半监督聚类,聚类集成,随机子空间,约束投影
中图分类号:TP311.13　　**文献标志码:**A

Weighted semi-supervised clustering ensemble algorithm based on extended constraint projection

Zhang Ding, Yang Youlong*, Sun Liqin
(School of Mathematics and Statistics, Xidian University, Xi′an, 710126, China)

Abstract: Semi-supervised clustering ensemble aims at improving the accuracy of clustering ensemble by using pairwise constraints, but it achieves poor performance on high-dimensional datasets. In addition, clustering performance has little improvement when only a few pairwise constraints are available. To solve these problems, this paper proposes a novel semi-supervised clustering ensemble algorithm WSCEC (Weighted Semi-supervised Clustering ensemble algorithm based on Extended Constraint projection algorithm). Firstly, a variety of clustering algorithms are exploited to cluster the feature space of data, and then the random subspace is utilized for the dimension reduction to reduce the impact of redundant features. Secondly, the original constraint set is expanded according to the k nearest or farthest samples of constraints and the transitive relationship between constraints, and the original data space is projected into a low-dimensional space by constraint projection technique to satisfy as many constraints as possible. Finally, a weighting strategy of clustering solutions is designed, which assigns an appropriate weight to each clustering solution to reduce the impact of low-quality clustering solutions. Experimental results on several datasets prove the effectiveness of the proposed algorithm.

Key words: semi-supervised clustering, clustering ensemble, random subspace, constraint projection

聚类是数据挖掘中一种重要的技术,广泛应用于各个领域,如图像分类、文本分析、生物信息学、互联网安全等,其目标是基于一定标准对数据进行适当的分组.目前已经提出了大量的聚类算

基金项目:陕西省自然科学基金(2021JM-133)
收稿日期:2022-04-19
* 通讯联系人,E-mail:ylyang@mail.xidian.edu.cn

法用于解决不同的实际问题[1],如K-means、Spectral Clustering、密度峰值聚类(Density Peak Clustering,DPC)等.单个聚类算法在某些任务上取得了相当不错的效果,但缺点也尤为明显:一方面,不同的聚类算法往往基于不同的假设,面对不同的数据每个聚类算法都有其自身的优势和劣势;另一方面,面临高维或大规模数据时,参数的选择是一项耗时的工作.

作为集成学习的分支之一,聚类集成旨在组合一组聚类解来获得比任意单一聚类解更高质量的聚类结果,通过对不同聚类解的标签信息进行整合来克服单个聚类解仅能提供部分信息的缺点.总体地,聚类集成分两个阶段:聚类解的生成和一致划分.为了从聚类解中获得更多的信息,通常需要生成一组多样性高的聚类解,常见的方法有利用相同单一聚类算法的不同参数[2]、随机子空间技术[3]、随机投影[4]等.而大部分聚类集成算法都在一致划分阶段进行研究,即从不同的视角考虑如何利用一组聚类解信息.现有的聚类集成方法主要分三类.

基于对象共现的方法通过构造共联矩阵(Co-association Matrix)来表示聚类解的标签信息.Fred and Jain[5]首先提出证据积累的概念来构造共联矩阵,但是该方法仅利用了聚类解的粗略信息.为了克服这一方法的不足,Iam-On et al[6]考虑间接相关的簇之间的影响力来定义簇间相似性,进而细化共联矩阵.Huang et al[7]利用信息熵来描述每一个簇对于所有聚类解的可靠性,根据不同质量的簇对共联矩阵进行加权.

基于图的方法将一组聚类解的信息表示为图或超图,根据图划分算法来求解聚类集成问题.Strehl and Ghosh[8]提出三种超图划分算法,即CSPA(Cluster-based Similarity Partitioning Algorithm),HGPA(Hypergraph Partitioning Algorithm)和MCLA(Meta-CLustering Algorithm).其中,HGPA方法用顶点表示对象,用相同权重的超边表示簇,构造一个超图;MCLA方法则把所有的簇看成图的顶点,簇之间的相似性作为边的权重.Mimaroglu and Erdil[9]通过数据对象之间的关系来构建相似图,并通过寻找枢轴和增长簇来划分该图.

基于中值划分的方法将聚类集成问题转化为优化问题,通过寻找一个划分,使该划分与所有聚类解相似性之和最大.Li et al[10]基于非负矩阵分解对聚类集成问题求近似解,但这个方法只能找到局部极小值而不是全局极小值.Huang et al[11]将集成聚类问题转化为二元线性规划问题,并通过因子图模型求解,该方法在大数据量的集成聚类问题中降低了计算成本.

尽管上面三种方法尝试从各种角度求解聚类集成问题,但都没有充分利用数据中有限的先验信息(成对约束),这在实际应用情景中很常见.

与聚类集成相比,半监督聚类集成在生成阶段或一致划分阶段加入先验信息以提升聚类性能.Tian et al[12]将分层特征选择策略和约束传播技术进行结合,使先验信息能够被充分利用且聚类精度得到一定的提升.Yu et al[13]利用随机子空间技术对数据集进行降维,并利用增量学习框架迭代地选择一组最优的聚类解进行集成,细化了共联矩阵对实例间相似性的定义.然而,当先验信息较少时,半监督聚类集成方法对聚类效果难以有显著的提升且鲁棒性较差,另外,在高维数据空间容易受到维数灾难的影响导致聚类性能降低.针对这两个问题,本文提出一种基于拓展约束的加权半监督聚类集成算法WSCEC(Weighted Semi-supervised Clustering Ensemble Algorithm Based on Extended Constraint Projection).图1展示了该算法总体过程,主要分三部分:

首先,利用多种聚类算法对数据的特征空间

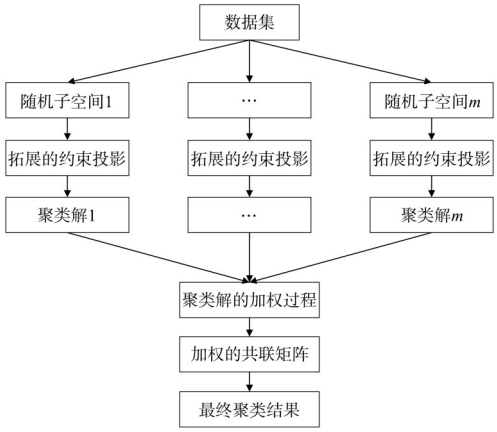

图1　WSCEC算法的框架

Fig. 1　The framework of the WSCEC algorithm

进行聚类,通过随机子空间对数据进行降维以减少冗余特征的影响.

其次,根据每对约束的k个最近或最远的样本以及约束间的关系,扩展原有的约束集以产生尽可能多的成对约束,再使用约束投影技术对数据进一步降维,并生成一组高质量的聚类解.

最后,设计了一个聚类解的加权策略,在一致划分阶段为每一个聚类解适当地分配一个权重,以减少低质量聚类解对最终聚类效果的影响,并利用图划分方法得到最终的聚类结果.

1 相关工作

1.1 "约束优先"的半监督聚类集成 由于半监督聚类算法在聚类精度上的优势,一些研究者在生成阶段利用半监督聚类算法来获得一组高质量的聚类解,该类算法称为"约束优先"的半监督聚类集成[14]. Iqbal et al[15]使用多种半监督聚类方法生成一组聚类解,并引入双参数加权机制,依靠重标记与基于投票的一致性函数得到最终聚类划分. Wei et al[16]考虑像素的空间近邻信息,改进了一种基于度量的半监督聚类算法,并根据基于约束和基于距离的半监督聚类算法共同生成一组聚类解,该算法在图像数据集上取得了较好的聚类效果. Yang and Jiang[17]提出一种双加权聚类集成算法,根据聚类解和簇的质量为每一个聚类解赋予两个权重. Yu et al[18]使用多种特征选择技术对高维数据进行降维,并选择一组最优的聚类解子集,可以显著减少无关特征对聚类过程的影响. 随后,Yu et al[19]设计了一种新的半监督聚类集成框架,通过自适应过程来搜索最佳子空间以产生更好的聚类划分,具有较强的鲁棒性. 尽管这类方法利用高质量的聚类解进行集成,但在高维空间相对低效且灵活性较差. 例如,当成对约束信息改变时,需要重新生成一组新的聚类解.

1.2 "集成优先"的半监督聚类集成 基于"约束优先"的半监督聚类集成的缺点,部分研究者开始利用无监督聚类算法生成一组聚类解,而在一致划分阶段考虑先验信息的影响. Wang et al[20]通过K-means等聚类算法生成一组聚类解,根据成对约束信息来调整所构造的共联矩阵,并利用图划分算法得到聚类结果. Yang et al[21]通过考虑满足约束的程度,结合聚类解的多样性和质量,提出一种新颖的聚类成员选择策略,实验性证明了先验信息对聚类集成的影响. Xiao et al[22]将先验信息与层次聚类算法结合,提出的模型在聚类集成中可以动态地适应先验信息的变化,并将该算法应用在高速列车故障诊断中. 最近,Lai et al[14]基于贝叶斯平均模型提出随机K-means集成的加权一致性算法(Weighted Consensus of Random K-means Ensemble,WECR),考虑每个簇中满足成对约束的程度与轮廓系数来加权共联矩阵,在保证精度提升的同时降低了时间复杂度.

尽管上述算法实现了不错的聚类效果并在某些领域取得成功的应用,但现有的半监督聚类集成算法没有同时考虑先验信息的稀缺和样本维数的过高对聚类集成带来的挑战. 本文提出的WSCEC算法能有效地处理这一问题. 该算法先通过双重降维的方式使成对约束在低维空间中仍能保持较高的一致性;其次,设计的聚类解加权策略自适应地为每一个聚类解赋予一个恰当的权重,可以在聚类集成中减少对样本的错误分配.

2 基于拓展约束投影的加权半监督聚类集成算法

2.1 基于融合聚类的随机子空间 随机子空间[23]是集成学习思想的一种应用,通过随机选取数据的部分特征来降低数据的维数,在降低冗余特征影响的同时还能减少算法运行的时间成本. 此外,划分的不同子空间可以提供不同的数据视角,从而降低训练得到的模型相关性,这恰好与聚类集成的思想一致.

传统的随机子空间技术通过随机选择不同的数据特征子集来生成不同的子空间,其缺点在于随机选择的特征子集不一定能准确地描述原始数据空间,即部分重要的特征可能无法被选中. 因此,本文先使用不同的聚类算法对原始的特征集进行聚类,从不同簇中随机选择特征组成一个子空间,最终得到一组子空间. 这样考虑是因为基于不同聚类算法得到的结果提供了不同视角的特征相关程度,对不同的簇中随机选择的多个特征可以尽可能避免重要的特征无法被选中的问题.

给定一个s维的数据集$\chi=\{x_1,\cdots,x_n\}$,D表示$n\times s$的数据矩阵.第i个子空间的生成步骤如下:

Step 1. 随机选择一个聚类算法对转置后的数据矩阵D^T进行聚类,得到一个聚类划分$C_i=\{c_{i1},\cdots,c_{im^i}\}$,$c_{ij}$为$C_i$的第$j$个簇.

Step 2. 给定一个特征采样率ρ,对于聚类划分C_i,从第j个簇c_{ij}中随机选择的特征数量为$\lfloor\rho\times s\div m^i\rfloor$,$j=1,\cdots,m^i$,$\lfloor\cdot\rfloor$为取整符号.

Step 3. 从C_i不同簇中选择的特征子集组成第i个子空间特征集$\{\lambda_1,\cdots,\lambda_{\lfloor\rho\times s\rfloor}\}$,其中$\lambda_i\in[1,s]$,且$\lambda_i\neq\lambda_j(i\neq j)$.

Step 4. 根据上述选择的特征,得到第i个子空间R_i.

基于融合聚类的随机子空间伪代码如下.

算法1　基于融合聚类的随机子空间

输入:子空间数量m,数据矩阵D,特征采样率ρ,多种聚类算法(K-means,DBSCAN,DPC等).

输出:m个随机子空间.

1. 随机选择一个聚类算法对D^T进行聚类,得到一个包含k个簇的聚类划分C.

2. 计算聚类划分C第i个簇中需要选择的特征数量$\lfloor\rho\times s\div k\rfloor$,分别从每个簇中随机选择特征.

3. 从每个簇中选择的特征组成最终的特征子集,得到一个子空间.

4. 重复步骤1至步骤3,得到m个子空间$R_1,\cdots,R_m$.

2.2　拓展的约束投影　约束投影[24]是半监督聚类中的一种重要技术,它根据已有的成对约束信息学习一个特征投影矩阵,将数据空间嵌入低维空间.一方面,成对约束信息在低维空间更容易满足;另一方面,在低维空间进行聚类可以提升运行效率且有更多的操作性.尽管这种方法处理高维数据效果不错,但前提是有足够的成对约束,而在实际的工程任务中,可能无法获得那么多约束.因此,本文考虑先扩展已有的约束集再利用约束投影技术将数据投影到一个新的空间.

令M和C分别为必连约束集(must-link set)和勿连约束集(cannot-link set).给定一对约束$(x_i,x_j)\in M$或$(x_i,x_j)\in C$,表示(x_i,x_j)在数据集中属于同一类或不属于同一类.

一方面,若$(x_i,x_j)\in M$,x_i和x_j与各自的k近邻成对加入相应的约束集M中.例如,当$k=2$时,x_1的两个近邻为x_3和x_5,x_2的两个近邻为x_4和x_6,则把$(x_1,x_3)(x_1,x_5)(x_2,x_4)(x_2,x_6)$加入约束集$M$中.类似地,约束集$C$中的约束则选择最远的$k$个样本进行扩展.

另一方面,约束集M和C拥有以下性质:

传递性:

$$(x_i,x_j)\in M,(x_j,x_z)\in M\Rightarrow(x_i,x_z)\in M$$

对称性:

$$(x_i,x_j)\in M\Leftrightarrow(x_j,x_i)\in M,(x_i,x_j)\in C\Leftrightarrow(x_j,x_i)\in C$$

因此,约束集M和C的约束分别根据最近和最远的k个样本进行扩展后,根据上述传递关系对约束集M和C再次进行扩展.需要注意的是,约束集C并不具备传递性,但仍然可以在以下情况进行扩展:若$(x_i,x_j)\in M$且$(x_i,x_k)\in C$,则有$(x_j,x_k)\in C$.扩展之后的约束集仍然记为M和C.

约束投影寻找一个投影矩阵$W=[w_1,\cdots,w_d]$,使约束集M的实例对在投影后的低维空间中尽可能接近,而约束集C中的实例对尽可能远离.定义该目标函数为$J(W)$,其中W满足$W^TW=I$:

$$J(W)=\frac{1}{2n_C}\sum_{(x_i,x_j)\in C}\left\|W^T(x_i-x_j)\right\|^2-\frac{1}{2n_M}\sum_{(x_i,x_j)\in M}\left\|W^T(x_i-x_j)\right\|^2 \tag{1}$$

其中,n_C和n_M分别表示约束集C和M的大小.令:

$$J_C=\frac{1}{2n_C}\sum_{(x_i,x_j)\in C}\left\|W^T(x_i-x_j)\right\|^2$$
$$J_M=\frac{1}{2n_M}\sum_{(x_i,x_j)\in M}\left\|W^T(x_i-x_j)\right\|^2 \tag{2}$$

从式(1)可以看出,为了最大化$J(W)$,需要最大化J_C且最小化J_M,则式(1)可以转化为特征值求解问题:

$$\begin{aligned}J_C&=\frac{1}{2n_C}\sum_{(x_i,x_j)\in C}\left\|W^T(x_i-x_j)\right\|^2=\\&\frac{1}{2n_C}\sum_{(x_i,x_j)\in C}\sum_l W_l^T(x_i-x_j)(x_i-x_j)^TW_l=\\&\frac{1}{2n_C}\sum_l W_l^TCC^TW_l=\\&\frac{1}{2n_C}\sum_l W_l^T\Lambda_CW_l\end{aligned} \tag{3}$$

同理,

$$J_M=\frac{1}{2n_M}\sum_l W_l^T\Lambda_M W_l \tag{4}$$

因此,目标函数$J(W)$转化为:

$$J(W)=\frac{1}{2n_C}\sum_l W_l^T\Lambda_C W_l-\frac{1}{2n_M}\sum_l W_l^T\Lambda_M W_l=\upsilon\sum_l W_l^T\left(\Lambda_C-\Lambda_M\right)W_l\propto\sum_l W_l^T(\Lambda)W_l \tag{5}$$

利用拉格朗日乘子法对上式进行优化,即:

$$L_{W_1,\cdots,W_K}=J\left(W_1,\cdots,W_K\right)-\sum_{l=1}^{k}\delta_l\left(W_l^TW_l-1\right) \tag{6}$$

对式(6)的W_l求偏导并令其为零,则有:

$$\frac{\partial L}{\partial W_l}=2\Lambda W_l-2\delta_l W_l=0$$
$$\forall l=1,\cdots,k\Rightarrow\Lambda W_l=\delta_l W_l,\forall l=1,\cdots,k \tag{7}$$

从式(7)可以看出,W_l是Λ的特征向量且δ_l为对应的特征值.此时,选择所有非负的特征值$\gamma_1,\cdots,\gamma_d$,记$\Lambda=diag\left(\gamma_1,\cdots,\gamma_d\right)$,对应的特征向量构成$W=\left[W_1,\cdots,W_d\right]$,即可最大化$J(W)$.$d$取式(7)中非负特征值的个数.最终,每个子空间$R_i$得到相应的投影矩阵$W_i$,第$i$个子空间投影后的数据矩阵为$Z_i=W_i^TD$.图2展示了二维数据空间中拓展约束投影的总体步骤.

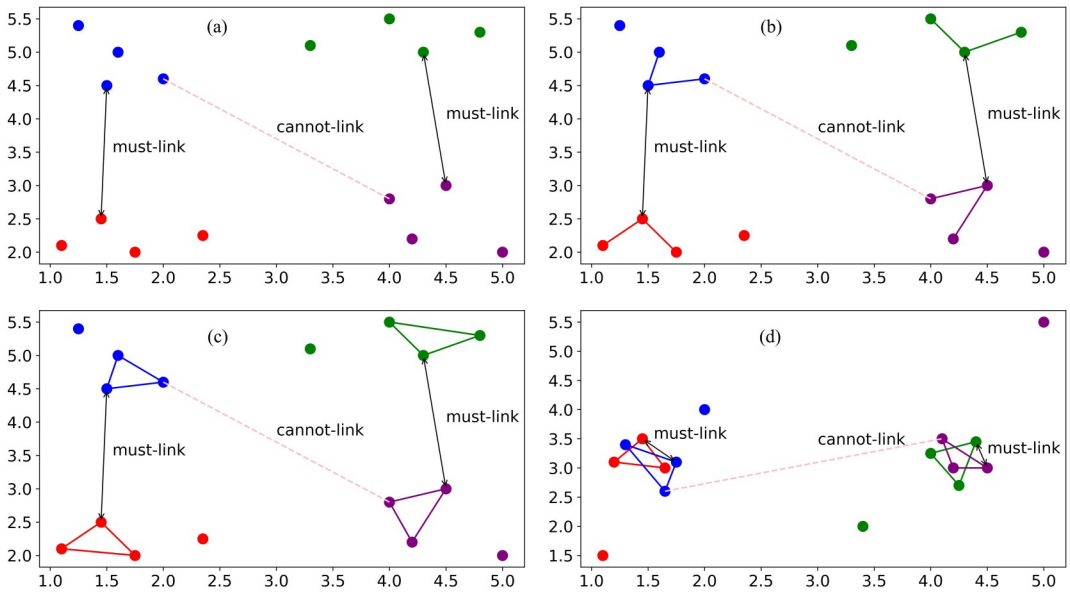

图2 二维数据空间的拓展约束投影示例过程:(a)带有约束的实例;(b)选择k个最近或最远的样本;(c)约束的扩展;(d)约束投影结果

Fig. 2 An example of expanded constraint projection process in two-dimensional data space: (a) some instances with constraint, (b) select k-nearest neighbor farthest samples, (c) extension of constraints, (d) results of constraint projection

2.3 聚类解的权重选择策略 对所有投影后的数据矩阵$Z_1,\cdots,Z_m$,使用穷尽有效的约束传播(Exhaustive and Efficient Constraint Propagation, E^{2}CP)[25]算法进行聚类,得到m个聚类解$\mu_1,\cdots,\mu_m$. E^{2}CP是一种高效的半监督聚类算法,能对约束信息在图上进行有效的传递.根据Fred and Jian[5]提出的共联矩阵作为一致性函数,定义如下:

$$A=\frac{1}{m}\sum_{k=1}^{m}A^k \tag{8}$$

其中,A_k为第k个聚类解的邻接矩阵.第i,j个元素定义如下:

$$a_{i,j}^k=\begin{cases}1, & \text{若}clu^k\left(x_i\right)=clu^k\left(x_j\right)\\ 0, & \text{其余}\end{cases} \tag{9}$$

其中,$clu^k\left(x_i\right)$表示实例x_i在聚类解μ_k中所属的簇.最后,利用图划分算法CSPA对上述矩阵进行划分,得到最终聚类结果.

然而,上述矩阵A隐含一个假设,即所有的聚类解都是同等重要的,而实际上不同聚类解的重要性是不同的.若一个聚类解比另一个聚类解

的质量更高,则该聚类解在最终的聚类划分中应该有更重要的地位.因此,考虑为每一个聚类解分配一个权重来定义它的重要性,使更高质量的聚类解对矩阵A的影响更明显,以至于对最终的聚类结果有更大的决定作用.

给定一个向量$B=(b_1,\cdots,b_m)$,修改后的共联矩阵定义如下:

$$A=\sum_{k=1}^{m}b_kA^k \tag{10}$$

且

$$b_1+\cdots+b_m=1 \tag{11}$$

接下来描述所设计的权重选择策略.假设向量B对应式(10)的聚类划分为$\{\pi_c\}_{c=1}^{k}$,则其质量评价为:

$$Q(B)=\sum_{j=1}^{k}\sum_{x\in\pi_j}dis(x,\pi_j)+\frac{1}{N^2}\|H-G\|_F^2 \tag{12}$$

式(12)右边第一项表示一个聚类划分的离散程度,π_j为第j个簇的簇中心,$dis(x,\pi_j)$表示样本x与所属簇的簇中心的欧式距离;第二项表示一个聚类划分中满足先验信息的程度,N表示所有成对约束的数量.其中,H的第i,j个元素表示如下:

$$h_{ij}=\begin{cases}1, & x_i^l=x_j^l,\text{且}(x_i,x_j)\in M\\ -1, & x_i^l\neq x_j^l,\text{且}(x_i,x_j)\in C\\ 0, & \text{其余}\end{cases} \tag{13}$$

其中,x_i^l和x_j^l分别表示x_i和x_j的真实标签.G的第i,j个元素表示如下:

$$g_{ij}=\begin{cases}1, & \text{若}(x_i,x_j)\in M\\ -1, & \text{若}(x_i,x_j)\in C\\ 0, & \text{其余}\end{cases} \tag{14}$$

质量评价函数$Q(B)$表示对任意向量B,相应的聚类划分的质量评价.$Q(B)$越小,说明该权重向量对应的聚类性能越好.

算法2描述了该策略的一般过程.首先,初始化一个大小为m的向量集$\phi=\{w_1,\cdots,w_m\}$,其中包含$m-1$个随机生成的向量$w_1,\cdots,w_{m-1}$和$w_m=(1/m,\cdots,1/m)$,且随机生成向量的元素满足式(11).由式(12),得到每一个向量的聚类质量评价$Q(w_i)$.根据评价结果,按升序对向量进行排序,记当前最优的向量为B_{best}.

其次,从向量集ϕ中随机选择两个向量进行竞争.令第i个向量被选择的概率$p(w_i)$和积累概率$q(w_i)$为:

$$p(w_i)=\frac{Q(w_i)}{\sum_{k=1}^{m}Q(w_k)} \tag{15}$$

$$q(w_i)=\sum_{j=1}^{i}p(w_j) \tag{16}$$

$$q(w_0)=0 \tag{17}$$

在均匀分布$[0,1]$上生成随机数a,若$a\in[q(w_{i-1}),q(w_i)]$,$i=1,\cdots,m$,则选择ϕ中第i个向量进行竞争.

对于任意两个被选择的向量:

$$\begin{aligned}W^s&=(w_s^1,\cdots,w_s^m)\\ W^t&=(w_t^1,\cdots,w_t^m)\end{aligned} \tag{18}$$

在均匀分布$[0,1]$上生成一个随机数b,则:

$$\begin{aligned}W^s&=(w_s^1,\cdots,w_s^{\lfloor m*b\rfloor},w_s^{\lfloor m*b\rfloor+1},\cdots,w_s^m)\\ W^t&=(w_t^1,\cdots,w_t^{\lfloor m*b\rfloor},w_t^{\lfloor m*b\rfloor+1},\cdots,w_t^m)\end{aligned} \tag{19}$$

交换向量W^s和W^t的前$\lfloor m*b\rfloor$个元素,生成两个新的向量,如下所示:

$$\begin{aligned}W^{s_1}&=(w_t^1,\cdots,w_t^{\lfloor m*b\rfloor},w_s^{\lfloor m*b\rfloor+1},\cdots,w_s^m)\\ W^{t_1}&=(w_s^1,\cdots,w_s^{\lfloor m*b\rfloor},w_t^{\lfloor m*b\rfloor+1},\cdots,w_t^m)\end{aligned} \tag{20}$$

此外,在均匀分布$[0,1]$上随机生成m个数$c_i(i=1,\cdots,m)$,若$c_i>0.5$,则随机生成一个$[0,1]$的数替换W^s中的第i个元素;否则,替换W^t中的第i个元素.经过m次替换,生成两个新的向量并进行归一化得到:

$$\begin{aligned}W^{s_2}&=(w_s^{1'},w_s^{2'},\cdots,w_s^{m'})\\ W^{t_2}&=(w_t^{1'},w_t^{2'},\cdots,w_t^{m'})\end{aligned} \tag{21}$$

根据式(12),计算新增向量W^{s_1},W^{t_1},W^{s_2},W^{t_2}的聚类质量评价并加入向量集ϕ中.

最后,对竞争结束后的向量集,根据聚类性能进行升序排序,排名在末尾的向量将被删除直至向量集ϕ大小为m,并更新最佳权重向量B_{best}.

重复上述过程直至迭代条件不满足.

算法2　聚类解的权重选择策略

输入:m个聚类解$\mu_1,\cdots,\mu_m$,最大迭代次数$t_{\max}$,竞争次数λ.

输出:最佳向量B_{best}.

1. 初始化一个包含 m 个向量的集合 ϕ,令 $t=0$.

2. 根据式(10)、式(11)、式(12)和CSPA算法,评估 ϕ 中每一个向量的相应聚类性能,根据它们的聚类性能升序排序,记录当前聚类性能最好的向量为 B_{best}.

3. 根据式(15)、式(16)和式(17),计算每一个向量被选择进行竞争的概率,随机从向量集 ϕ 中选择两个向量进行竞争;由式(18)至式(21),获得新的向量并加入向量集 ϕ. 重复进行 λ 次.

4. 对竞争后的向量根据聚类性能升序排序,记向量集 ϕ 中最优向量为 B^*_{best}. 若 $Q(B^*_{\text{best}})<Q(B_{\text{best}})$,则 $B^*_{\text{best}}=B_{\text{best}}$;令 $t=t+1$.

5. 删除排名靠后的向量,保持向量集 ϕ 的大小为 m.

6. 重复步骤3、步骤4和步骤5,直到 $t=t_{\max}$.

为了清晰地说明WSCEC算法,算法3总结了提出方法的过程.

算法3 基于拓展约束的加权半监督聚类集成算法

输入:数据矩阵 D.

输出:最终聚类划分 p_{result}.

1. 利用算法1得到 m 个子空间 $R_1,\cdots,R_m$.

2. 根据式(1)至式(7)计算每一个子空间的投影矩阵 $W_1,\cdots,W_m$.

3. 计算 m 个低维数据矩阵 $Z_1,\cdots,Z_m$,其中 $Z_i=W_i^TD$;由E^{2}CP算法得到 m 个聚类解 $\mu_1,\cdots,\mu_m$.

4. 根据算法2得到向量 B_{best}.

5. 根据式(10)和CSPA算法得到最终聚类划分 p_{result}.

3 实验

3.1 数据集 为了充分展示提出方法的有效性,在12个真实数据集上评估了WSCEC算法和其他聚类算法的聚类性能,包括六个UCI数据集Segmentation, Semeion, Optdigit, ISOLET, Sat, Cardiotocography;四个图像数据集COIL20, Yale, MNIST4000, ORL;两个基因表达数据集11_Tumors, Leukemia2. 表1给出了各数据集的具体信息.

使用归一化互信息(Normalized Mutual Information, *NMI*)、调整的兰德系数(Adjust Rand Index, *ARI*)和精度(Accuracy, *ACC*)为聚类质量的评价标准. *NMI*度量了两个聚类结果之间共享信息的程度,而 *ARI* 和 *ACC* 则是对两个聚类结果中成对实例匹配程度的度量. 因此, *NMI*, *ARI* 和 *ACC* 越近1说明聚类效果越好,反之亦然.

表1 实验中使用的数据集

Table 1 Datasets used in experiments

数据集	样本量	特征数	类别
Segmentation	2319	19	7
Semeion	1593	256	10
Optdigit	5620	64	10
ISOLET	1559	617	26
Cardiotocography	2126	23	10
Sat	6435	36	6
COIL20	1440	1024	20
Yale	165	1024	15
MNIST4000	4000	784	10
ORL	400	1024	40
11_Tumors	174	12533	11
Leukemia2	72	11225	3

给定一个聚类的真实划分 G 和预测的聚类划分 G', $G=\{C_1,\cdots,C_k\}$, $G'=\{C'_1,\cdots,C'_{k'}\}$. G 和 G' 之间的 *NMI* 定义如下:

$$NMI(G,G')=\frac{2H_1(G;G')}{H_2(G)+H_2(G')} \tag{22}$$

其中,

$$\begin{aligned}H_1(G;G')&=\sum_{i=1}^{k}\sum_{j=1}^{k'}\frac{|C_i\cap C'_j|}{n}\lg\frac{n|C_i\cap C'_j|}{|C_i||C'_j|}\\ H_2(G)&=-\sum_{i=1}^{k}\frac{|C_i|}{n}\lg\frac{|C_i|}{n}\\ H_2(G')&=-\sum_{j=1}^{k'}\frac{|C'_j|}{n}\lg\frac{|C'_j|}{n}\end{aligned} \tag{23}$$

n 表示样本的数量,$|\cdot|$ 表示一个簇的大小.

真实划分 G 和预测的聚类划分 G' 之间的 *ARI* 计算如下:

$$ARI(G,G')=\frac{2(n_{00}n_{11}-n_{01}n_{10})}{(n_{00}+n_{01})(n_{01}+n_{11})(n_{00}+n_{10})(n_{10}+n_{11})} \tag{24}$$

其中,n_{00} 为聚类划分 G 和 G' 都分配在一个簇中成对实例的数量,n_{11} 为聚类划分 G 和 G' 都不分配在一个簇中成对实例的数量,n_{01} 为聚类划分 G 分配在一个簇中而聚类划分 G' 不分配在一个簇中成对实例的数量,n_{10} 表示与 n_{01} 相反的情况.

真实划分 G 和预测的聚类划分 G' 之间的 *ACC* 计算如下:

$$ACC=\frac{\sum_{i=1}^{n}\delta\left(c_i,map\left(t_i\right)\right)}{n} \tag{25}$$

$$\delta(a,b)=\begin{cases}1, & \text{if } a=b\\ 0, & \text{otherwise}\end{cases} \tag{26}$$

其中,c_i是真实标签,t_i是预测标签,$map(\cdot)$函数为最佳类标签的重分配.

3.2　实验设置　将提出的WSCEC算法与多个半监督聚类算法和聚类集成算法进行比较,包括四个半监督聚类集成算法:约束k均值的聚类集成(Constrained K-means Clustering Ensemble, COPE-kmeans)[26]、约束邻域投影的半监督聚类集成(Constraint Neighborhood Projections for Semi-supervised K-means Ensemble, CNPE-kmeans)[27]、E^{2}CPE和WECR;三个聚类集成算法:MCLA、谱聚类集成(Spectral Ensemble Clustering,SEC)[28]、超大规模谱聚类集成(Ultra-scalable Ensemble Clustering,USENC)[29].

对每个数据集,WSCEC方法通过E^{2}CP算法生成20个聚类解,每个聚类解中簇的数量随机从$\left[2,\sqrt{n}\right]$选择.算法2中,令$t_{\max}=50$,$\lambda=10$.此外,为了尽可能避免偶然因素的影响,每种算法都重复运行10次,取平均值作为最终的结果.

所有的成对约束都基于数据集的真实类别随机生成.Segmentation, Semeion, Optdigit, ISOLET, Sat, Cardiotocography, COIL20和MNIST 4000数据集生成的成对约束数量为30个,而Yale,ORL,11_Tumors,Leukemia2生成的成对约束数量为10个.

3.3　实验结果及其分析

3.3.1　WSCEC与其他算法的比较　表2、表3和表4分别展示了所有聚类算法在12个数据集上的具体表现,表中黑体字表示性能最优.WSCEC算法的*NMI*,*ARI*和*ACC*在Segmentation,ISOLET,Sat,COIL20等九个数据集上最高,并在Semeion,Optdigit等三个数据集上第二高.根据表2,WSCEC算法在数据集ISOLET, Cardiotocography, Sat, COIL20, ORL和11_Tumors上的*NMI*分别比第二名的算法高4.4%,6.5%,4.7%,4%,4.6%,9.3%,优势明显.而在Semeion等三个数据集中和第一名的结果也较为接近.由表2、表3和表4还可以看到:

(1)聚类集成算法USENC的聚类性能和半监督聚类集成算法COPE-kmeans,CNPE-kmeans相比,有一定优势,和E^{2}CPE,WECR算法相比则旗鼓相当,但除了Semeion数据集,它们都不如WSCEC算法.导致这种结果可能的原因:当先验信息较少时,半监督聚类集成算法难以发挥其优势,在少数情况下甚至会导致聚类精度的下降,而WSCEC算法通过扩展约束集、约束投影等方式有效地处理了这个缺陷,证明了WSCEC算法的优势.

(2)WSCEC方法在高维数据中有明显优势,

表2　WSCEC与其他算法在12个数据集上的平均*NMI*对比

Table 2　Average *NMI* of WSCEC and other algorithms on twelve datasets

数据集	COPE-kmeans	CNPE-kmeans	E^{2}CPE	WECR	MCLA	SEC	USENC	WSCEC
Segmentation	0.513	0.547	0.634	0.623	0.539	0.562	0.638	**0.662**
Semeion	0.525	0.554	0.599	0.590	0.578	0.551	**0.649**	0.633
Optdigit	0.815	0.763	**0.916**	0.793	0.839	0.683	0.834	0.914
ISOLET	0.663	0.707	0.679	0.711	0.675	0.687	0.709	**0.755**
Cardiotocography	0.471	0.479	0.464	0.527	0.511	0.450	0.544	**0.609**
Sat	0.529	0.533	0.520	0.544	0.511	0.541	0.578	**0.625**
COIL20	0.760	0.776	0.793	0.801	0.744	0.723	0.799	**0.841**
Yale	0.417	**0.477**	0.438	0.413	0.351	0.414	0.417	0.464
MNIST4000	0.504	0.543	0.595	0.584	0.551	0.363	0.588	**0.632**
ORL	0.608	0.614	0.581	0.645	0.445	0.666	0.690	**0.736**
11_Tumors	0.544	0.569	0.554	0.543	0.515	0.577	0.524	**0.670**
Leukemia2	0.527	0.535	0.363	0.593	0.490	0.388	0.456	**0.627**

表 3 WSCEC 与其他算法在 12 个数据集上的平均 *ARI* 对比

Table 3 Average *ARI* of WSCEC and other algorithms on twelve datasets

数据集	COPE-kmeans	CNPE-kmeans	E^2CPE	WECR	MCLA	SEC	USENC	WSCEC
Segmentation	0.377	0.456	0.527	0.526	0.425	0.343	0.516	**0.567**
Semeion	0.391	0.454	0.487	0.499	0.464	0.457	**0.535**	0.522
Optdigit	0.776	0.723	**0.921**	0.761	0.806	0.671	0.780	0.914
ISOLET	0.415	0.482	0.440	0.488	0.386	0.391	0.454	**0.507**
Cardiotocography	0.254	0.262	0.236	0.303	0.278	0.235	0.304	**0.385**
Sat	0.439	0.449	0.434	0.465	0.421	0.353	0.486	**0.531**
COIL20	0.613	0.638	0.645	0.647	0.596	0.571	0.598	**0.681**
Yale	0.342	**0.400**	0.368	0.359	0.296	0.343	0.347	0.392
MNIST4000	0.386	0.452	0.490	0.475	0.429	0.286	0.469	**0.535**
ORL	0.217	0.186	0.235	0.311	0.063	0.277	0.327	**0.390**
11_Tumors	0.332	0.372	0.332	0.331	0.304	0.365	0.343	**0.536**
Leukemia2	0.535	0.538	0.341	0.596	0.461	0.32	0.397	**0.633**

表 4 WSCEC 与其他算法在 12 个数据集上的平均 *ACC* 对比

Table 4 Average *ACC* of WSCEC and other algorithms on twelve datasets

数据集	COPE-kmeans	CNPE-kmeans	E^2CPE	WECR	MCLA	SEC	USENC	WSCEC
Segmentation	0.543	0.563	0.65	0.662	0.574	0.573	0.645	**0.668**
Semeion	0.624	0.669	0.595	0.671	0.661	0.617	**0.724**	0.68
Optdigit	0.811	0.86	**0.944**	0.871	0.851	0.79	0.91	0.939
ISOLET	0.532	0.567	0.541	0.552	0.432	0.447	0.545	**0.565**
Cardiotocography	0.434	0.431	0.407	0.403	0.438	0.384	0.47	**0.533**
Sat	0.603	0.642	0.616	0.653	0.64	0.404	0.7	**0.73**
COIL20	0.693	0.705	0.732	0.717	0.634	0.553	0.617	**0.751**
Yale	0.372	**0.419**	0.38	0.348	0.312	0.359	0.355	0.397
MNIST4000	0.55	0.621	0.666	0.583	0.601	0.334	0.6	**0.687**
ORL	0.412	0.38	0.391	0.528	0.24	0.439	0.452	**0.553**
11_Tumors	0.513	0.522	0.509	0.539	0.508	0.489	0.512	**0.677**
Leukemia2	0.729	0.791	0.704	0.836	0.665	0.648	0.512	**0.677**

例如在数据集 11_Tumors 上的聚类精度比其余算法要高得多. 这可能是因为随机子空间和约束投影技术可以去除冗余的特征以降低维数灾难的影响,且在新的空间中能更有效地利用先验信息,从而使聚类算法生成的聚类解不仅具有一定的多样性,还能提升最终的聚类性能.

(3)尽管 WSCEC 方法进行了一定程度的降维,但在 Yale 数据集上的聚类效果仍然较差. 一方面,可能是 Yale 数据集的分布不依赖距离,其数据分布不符合选择聚类方法的假设;另一方面,可能是选择的降维方法不够准确,后续需要探索更多的降维方法.

3.3.2 集成大小的鲁棒性检验 进一步检验 WSCEC 方法在不同集成大小 E 下的 *NMI* 和 *ARI*. 随着集成规模的增加,聚类性能的波动范围越小说明鲁棒性和稳定性越强. 通过取 $E=\{10,20,30,40,50\}$ 来研究聚类性能,所有算法在每一个集成大小下的聚类结果都重复运行 10 次,取平均值.

图 3 和图 4 分别展示在不同的集成大小下,WSCEC 算法的平均 *NMI* 和 *ARI*. 尽管 USENC 算法在 Semeion 上优于 WSCEC,但在其余 11 个数据集上 WSCEC 算法都优于 USENC. 此外,从图 3 可以看出,随着集成大小的改变,WSCEC 算

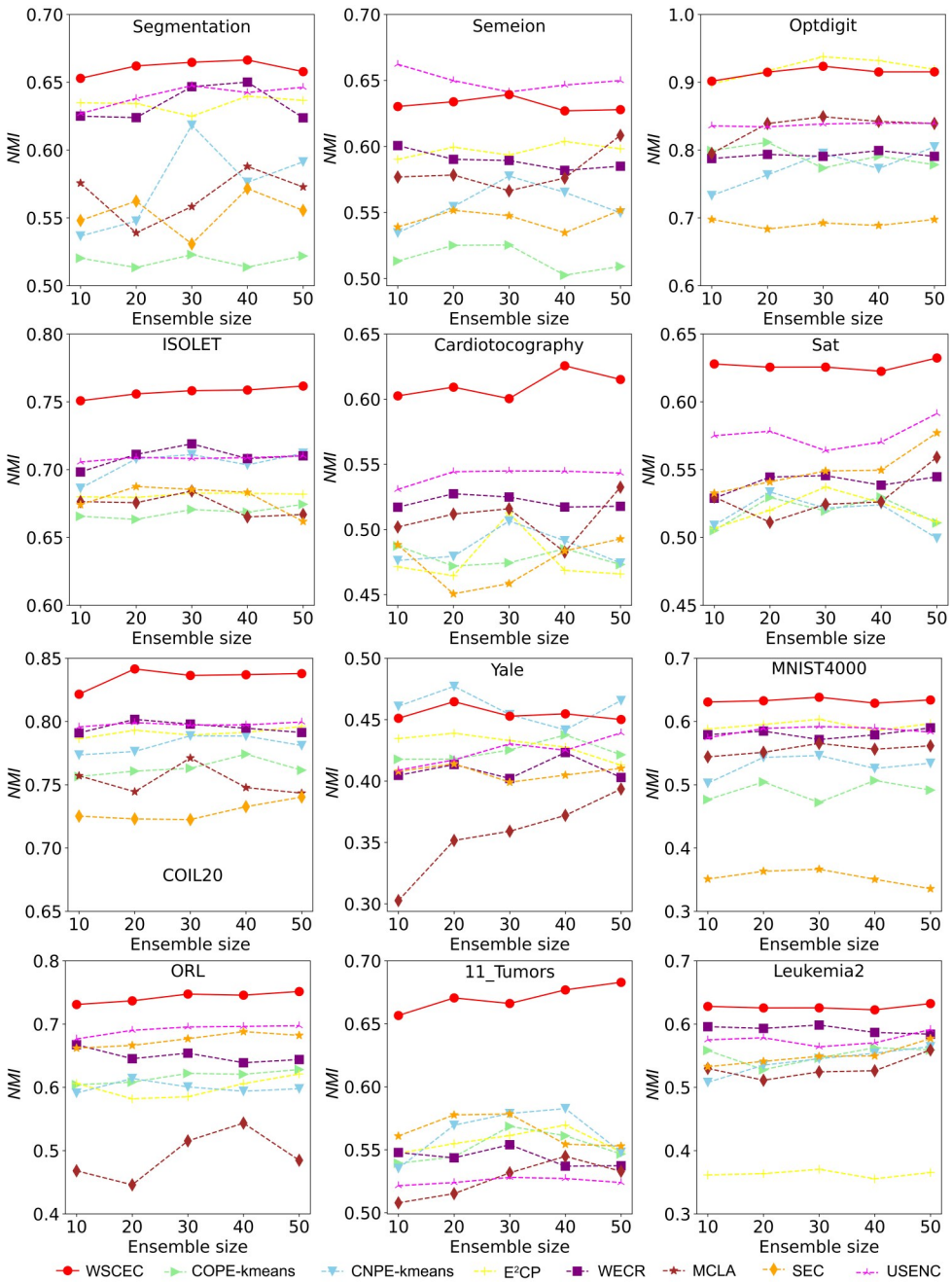

图3　不同算法在不同的集成大小下的平均*NMI*

Fig.3　Average *NMI* of different algorithms with different ensemble size

法在多数数据集上比其余算法更稳定且有最佳或接近最佳的*NMI*和*ARI*. 例如,在ISOLET,Sat,MNIST4000,11_Tumors等数据集上,WSCEC算法的*NMI*都比其余算法的波动更小. 从图4也能得出类似结论. 与大部分算法相比,WSCEC算法在正常数据集和高维数据集都能取得较优的聚类效果,有更广的适用性和更强的鲁棒性.

3.3.3　聚类解权重选择过程的影响　设计一组消融实验来探究聚类解权重的选择过程对集成聚类结果的影响. 将去除该过程的方法记为WSCEC_not,并在同样的条件下,在上述12个数据集上进行比较.

表5显示了WSCEC和WSCEC_not在所有数据集上的平均*NMI*,表中黑体字表示较优的结

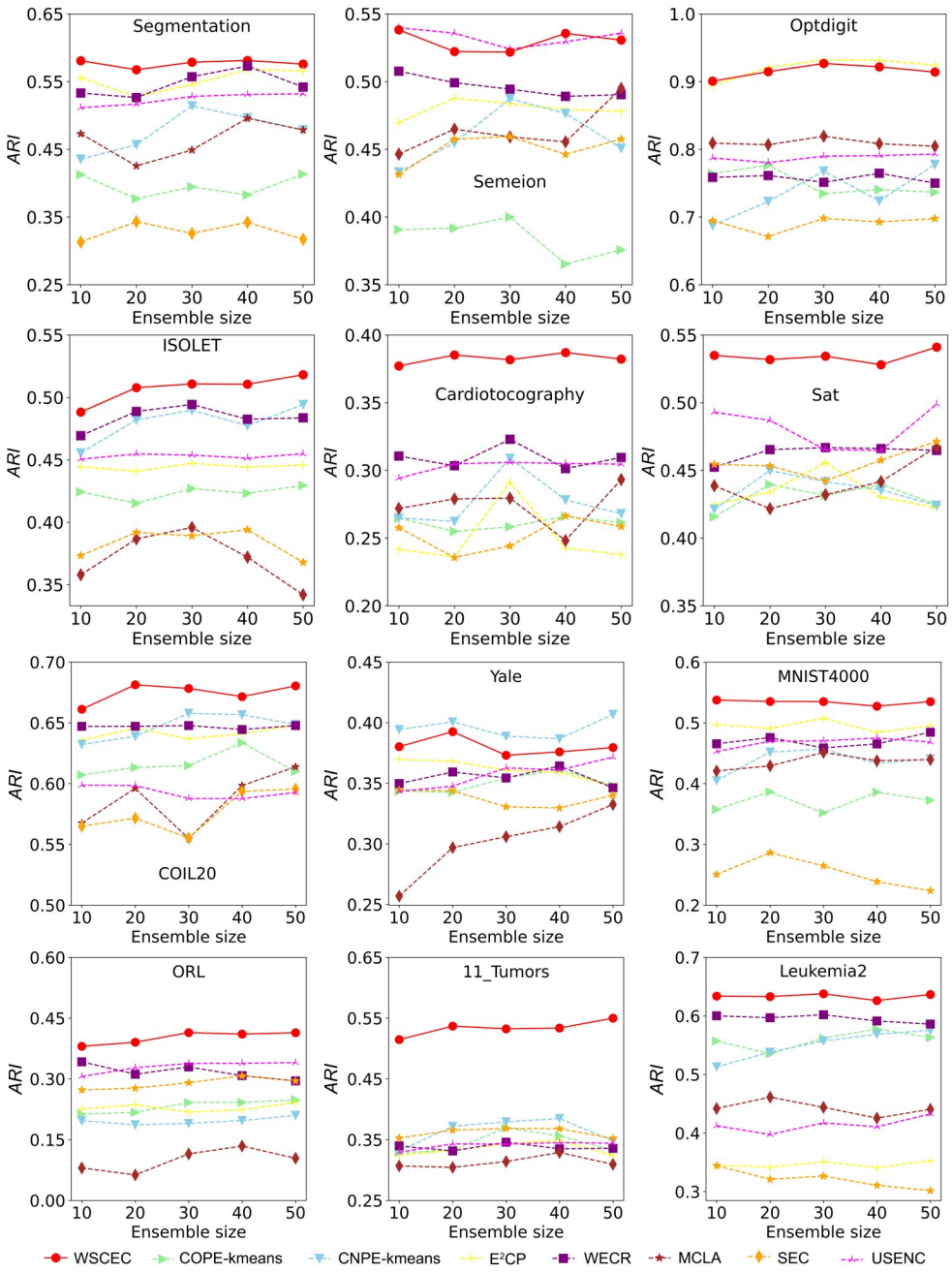

图 4　不同算法在不同的集成大小下的平均 *ARI*

Fig.4　Average *ARI* of different algorithms with different ensemble size

果. 由表可见,除了 Semeion 数据集,WSCEC 在其余 11 个数据集上都取得了最好聚类效果. 例如,在 Segmentation, Cardiotocography, Sat, Yale 和 11_Tumors 数据集上的 *NMI* 分别比 WSCEC_not 高 3.5%, 10.7%, 6.5%, 5.8% 和 11%,证明提出的聚类解权重选择策略有效,低质量的聚类解对最终聚类效果在一定程度上有负面作用. 权重选择过程通过模拟遗传算法,不断迭代更新较优的权重向量,使不同质量的聚类解可以被分配一个合适的权重,影响共联矩阵对实例间相似性判断,从而提高最终的聚类精度.

3.3.4　参数的敏感性分析　为了评估特征采样率 ρ 对 WSCEC 算法的影响,将 12 个数据集分成两组,维数少于 500 的数据集一组,其余数据集为

表5　WSCEC与WSCEC_not在不同数据集上的*NMI*

Table 5　*NMI* of WSCEC and WSCEC_not on different datasets

Datasets	WSCEC	WSCEC_not
Segmentation	**0.650**	0.615
Semeion	0.621	**0.644**
Optdigit	**0.928**	0.900
ISOLET	**0.752**	0.728
Cardiotocography	**0.615**	0.508
Sat	**0.625**	0.560
COIL20	**0.832**	0.806
Yale	**0.459**	0.401
MNIST4000	**0.638**	0.603
ORL	**0.732**	0.631
11_Tumors	**0.683**	0.573
Leukemia2	**0.623**	0.585

一组,取ρ在不同范围进行实验.对于Segmentation,Semeion等四个数据集,令$\rho=\{0.5,0.6,0.7,0.8,0.9\}$;对于COIL20,Yale等四个数据集,令$\rho=\{0.1,0.2,0.3,0.4,0.5\}$.

图5显示了算法在Segmentation等四个数据集上的实验结果.$\rho=0.7$时WSCEC取得最佳的*NMI*和*ARI*,在在Segmentation和Optdigit数据集上,$\rho=0.5$时聚类性能有一定程度的降低.可能是这两个数据集的特征维数较低,较少的特征无法捕捉数据的内在结构.另一方面,图6显示了高于500维的四个数据集的实验结果.$\rho=0.3$时WSCEC在COIL20等数据集上取得了较好的聚类结果,而对于高维数据集11_Tumors和Leukemia2,$\rho=0.5$时聚类性能有明显下降.再次证明对高维数据进行适当的降维是必要的,维数灾难影响实例间的距离度量的准确性,进而影响算法的聚类效果.

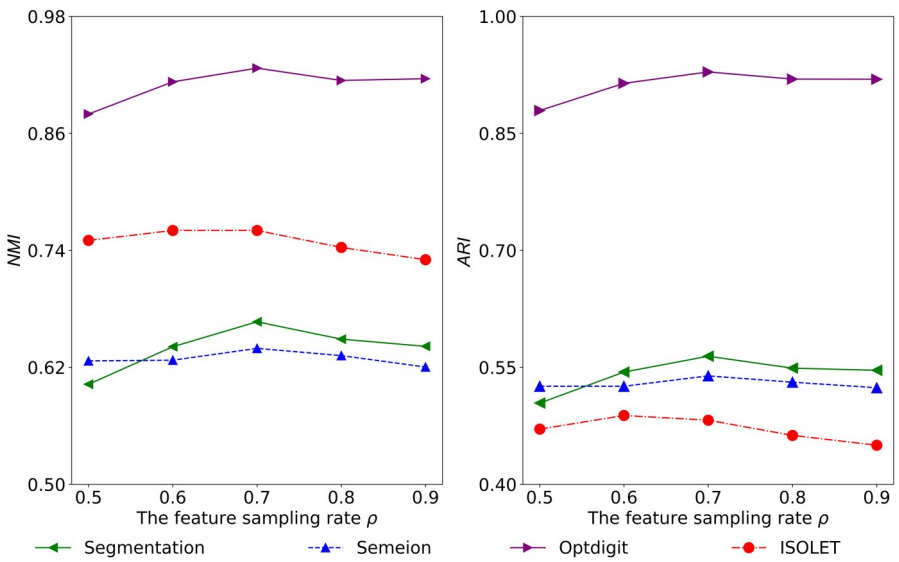

图5　在低维数据集上参数ρ对WSCEC算法的影响

Fig. 5　The influence of parameter ρ of WSCEC algorithm on low-dimensional dataset

4　结论

本文提出一种基于拓展约束的加权半监督聚类集成算法WSCEC,在面对少量的先验信息和高维数据时有较好的聚类精度和更强的鲁棒性.首先,通过引入融合聚类的随机子空间来尽可能地选择具有代表性的特征,同时降低冗余特征的影响.其次,拓展的约束投影不仅提高了成对约束的利用率,还可以在更低维数据空间中进行可靠的聚类.最后,设计的聚类解权重选择策略对不同重要性的聚类解进行区分,保证最终聚类划分的一致性.实验结果显示,提出的算法在大部分实验数据集上都取得最佳的聚类效果且具有稳

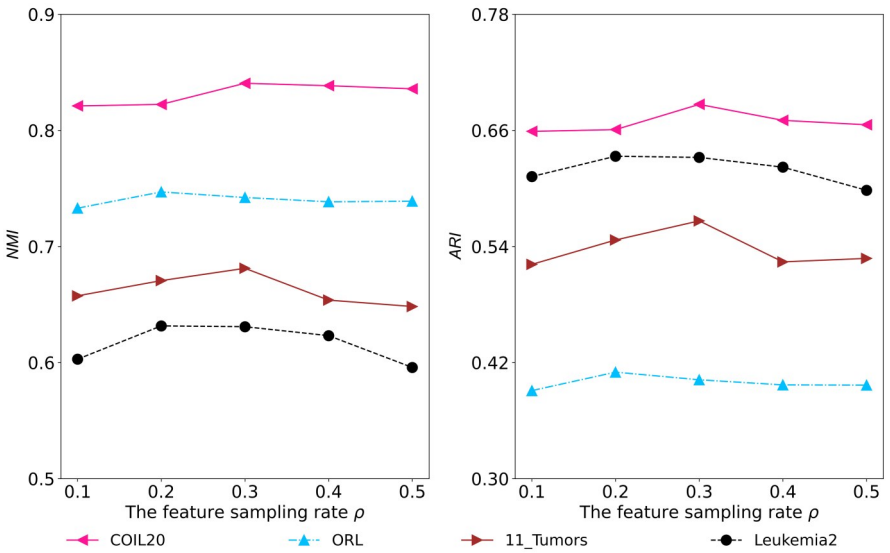

图 6　在高维数据集上参数 ρ 对 WSCEC 算法的影响

Fig. 6　The influence of parameter ρ of WSCEC algorithm on high-dimensional dataset

定性. 尤其在高维数据集上,WSCEC 算法和其他算法相比,有一定的优势.

不是所有的成对约束都是有用的,希望可以找到一些对聚类结果影响最大的成对约束. 未来的工作将考虑结合主动学习[30]进行半监督聚类集成,这样做的优势在于:第一,即使在更少的成对约束情况下依然可以取得相当的聚类性能;第二,面对大数据集,可以减少聚类的迭代次数从而降低时间复杂度,有效地提升聚类效率.

参考文献

[1] Jain A K. Data clustering:50 years beyond K-means. Pattern Recognition Letters,2010,31(8):651－666.

[2] Wu J J, Liu H F, Xiong H, et al. K-means-based consensus clustering: A unified view. IEEE Transactions on Knowledge and Data Engineering, 2015,27(1):155－169.

[3] Yu Z W, Luo P N, Liu J M, et al. Semi-supervised ensemble clustering based on selected constraint projection. IEEE Transactions on Knowledge and Data Engineering,2018,30(12):2394－2407.

[4] Fern X Z, Brodley C E. Random projection for high dimensional data clustering: A cluster ensemble approach//Proceedings of the 20th International Conference on Machine Learning. Washington, DC, USA:ACM,2003:186－193.

[5] Fred A L N, Jain A K. Combining multiple clusterings using evidence accumulation. IEEE Transactions on Pattern Analysis and Machine Intelligence,2005,27(6):835－850.

[6] Iam-On N, Boongoen T, Garrett S, et al. A link-based approach to the cluster ensemble problem. IEEE Transactions on Pattern Analysis and Machine Intelligence,2011,33(12):2396－2409.

[7] Huang D, Wang C D, Lai J H, et al. Locally weighted ensemble clustering. IEEE Transactions on Cybernetics,2018,48(5):1460－1473.

[8] Strehl A, Ghosh J. Cluster ensembles: A knowledge reuse framework for combining multiple partitions. The Journal of Machine Learning Research, 2002, 3(3):583－617.

[9] Mimaroglu S, Erdil E. Combining multiple clusterings using similarity graph. Pattern Recognition,2011,44(3):694－703.

[10] Li T, Ding C, Jordan M I. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization//Proceeding of the 21th International Conference on Machine Learning. Omaha,NE,USA:IEEE,2007:577－582.

[11] Huang D, Lai J H, Wang C D. Ensemble clustering using factor graph. Pattern Recognition, 2016(50): 131－142.

[12] Tian J L, Ren Y Z, Cheng X. Stratified feature

sampling for semi-supervised ensemble clustering. IEEE Access, 2019(7): 128669－128675. DOI: 10.1109/ACCESS.2019.2939581.

[13] Yu Z W, Luo P N, You J E, et al. Incremental semi-supervised clustering ensemble for high dimensional data clustering. IEEE Transactions on Knowledge and Data Engineering, 2016, 28(3): 701－714.

[14] Lai Y X, He S Y, Lin Z J, et al. An adaptive robust semi-supervised clustering framework using weighted consensus of random K-means ensemble. IEEE Transactions on Knowledge and Data Engineering, 2021, 33(5): 1877－1890. DOI: 10.1109/TKDE.2019.2952596.

[15] Iqbal A M, Moh'd A, Khan Z. Semi-supervised clustering ensemble by voting. 2009, arXiv: 1208.4138.

[16] Wei S T, Li Z X, Zhang C L. Combined constraint-based with metric-based in semi-supervised clustering ensemble. International Journal of Machine Learning and Cybernetics, 2018, 9(7): 1085－1100.

[17] Yang Y, Jiang J M. Bi-weighted ensemble via HMM-based approaches for temporal data clustering. Pattern Recognition, 2018(76): 391－403.

[18] Yu Z W, Chen H S, You J, et al. Double selection based semi-supervised clustering ensemble for tumor clustering from gene expression profiles. IEEE/ACM Transactions on Computational Biology and Bioinformatics, 2014, 11(4): 727－740.

[19] Yu Z W, Kuang Z Q, Liu J M, et al. Adaptive ensembling of semi-supervised clustering solutions. IEEE Transactions on Knowledge and Data Engineering, 2017, 29(8): 1577－1590.

[20] Wang H J, Qi J H, Zheng W F, et al. Semi-supervised cluster ensemble based on binary similarity matrix//2010 2nd IEEE International Conference on Information Management and Engineering. Chengdu, China: IEEE, 2010: 251－254.

[21] Yang F, Li T, Zhou Q F, et al. Cluster ensemble selection with constraints. Neurocomputing, 2017(235): 59－70.

[22] Xiao W C, Yang Y, Wang H J, et al. Semi-supervised hierarchical clustering ensemble and its application. Neurocomputing, 2016(173): 1362－1376.

[23] Ho T K. The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20(8): 832－844.

[24] Zhang D Q, Chen S C, Zhou Z H, et al. Constraint projections for ensemble learning//Proceeding of the 23rd National Conference on Artificial Intelligence. Chicago, IL, USA: AAAI Press, 2008: 758－763.

[25] Lu Z W, Peng Y X. Exhaustive and efficient constraint propagation: A graph-based learning approach and its applications. International Journal of Computer Vision, 2013, 103(3): 306－325.

[26] Wagstaff K, Cardie C, Rogers S, et al. Constrained k-means clustering with background knowledge// Proceedings of the 18th International Conference on Machine Learning. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., 2001: 557－584.

[27] Wang H J, Li T, Li T R, et al. Constraint neighborhood projections for semi-supervised clustering. IEEE Transactions on Cybernetics, 2014, 44(5): 636－643.

[28] Liu H F, Wu J J, Liu T L, et al. Spectral ensemble clustering via weighted K-means: Theoretical and practical evidence. IEEE Transactions on Knowledge and Data Engineering, 2017, 29(5): 1129－1143.

[29] Huang D, Wang C D, Wu J S, et al. Ultra-scalable spectral clustering and ensemble clustering. IEEE Transactions on Knowledge and Data Engineering, 2020, 32(6): 1212－1226.

[30] Xiong S C, Azimi J, Fern X Z. Active learning of constraints for semi-supervised clustering. IEEE Transactions on Knowledge and Data Engineering, 2014, 26(1): 43－54.

（责任编辑　杨可盛）