第47卷 第2期 计 算 机 工 程 2021年2月
Vol.47 No.2 Computer Engineering February 2021

·人工智能与模式识别· 文章编号:1000-3428(2021)02-0077-07 文献标志码:A 中图分类号:TP18

结合改进密度峰值聚类的LGC半监督学习方法优化

薛子晗,潘 迪,何 丽

(天津财经大学 理工学院,天津 300222)

摘 要: 基于图的局部与全局一致性(LGC)半监督学习方法具有较高的标注正确率,但时间复杂度较高,难以适用于数据规模较大的实际应用场景。从缩小图的规模入手,提出一种全局一致性优化方法。使用改进后的密度峰值聚类算法,迭代地从数据集中筛选出多个中心点,以每个中心点为簇中心进行局部聚类,并以中心点为顶点构建图,实现基于LGC的半监督学习。实验结果表明,优化后的LGC方法在D31、Aggregation等数据集上具有较好的鲁棒性,在标注正确率和算法执行时间上优势明显。

关键词: 半监督学习;密度峰值聚类;基于图方法;标签传递;迭代

开放科学(资源服务)标志码(OSID):

中文引用格式: 薛子晗,潘迪,何丽.结合改进密度峰值聚类的LGC半监督学习方法优化[J].计算机工程,2021,47(2):77-83,89.

英文引用格式: XUE Zihan, PAN Di, HE Li. Optimization of LGC semi-supervised learning method combined with improved density peaks clustering[J].Computer Engineering,2021,47(2):77-83,89.

Optimization of LGC Semi-Supervised Learning Method Combined with Improved Density Peaks Clustering

XUE Zihan,PAN Di,HE Li

(College of Science and Technology,Tianjin University of Finance and Economics,Tianjin 300222,China)

【Abstract】 The graph-based semi-supervised learning method with Local and Global Consistency(LGC) has excellent performance in labeling accuracy, but has a high time complexity and is difficult to apply to practical large-scale applications.To solve the problem, this paper proposes an LGC optimization method by reducing the size of the graph. This method uses the improved Density Peaks Clustering (DPC) algorithm, and iteratively selects multiple center points from the data set.Then local clustering is performed by taking each center point as the cluster center, and the center points are used as vertexes to construct a graph to perform LGC-based semi-supervised learning. Experimental results show that the optimized LGC method has good robustness on D31, Aggregation and other data sets, and has obvious advantages in label accuracy and algorithm execution time.

【Key words】 semi-supervised learning;Density Peaks Clustering (DPC);graph-based methods;label propagation;iteration

DOI: 10. 19678/j. issn. 1000-3428. 0057017

0 概述

强监督的机器学习方法需要大量有标签数据的支持,但随着大数据时代应用领域数据量的日益膨胀,通常获得的是大量的无标签数据。因此,半监督学习成为模式识别和机器学习领域的一个新的研究热点。半监督学习介于监督学习与无监督学习之间,是通过少量标记样本对大量未标记样本进行标注的一种学习方法[1]。基于图的半监督学习是该研究领域极具代表性的一种方法,在样本标注正确率上具有明显优势。

自文献[2]提出图分割最小割算法以来,基于图的半监督学习方法得到了广泛应用。文献[3]针对处于类边界区域的标记样本往往会降低标签传播有效性的问题,提出亲和力标签传播算法。文献[4]提出将标签传播和图卷积网络相结合的框架,扩展了

基金项目: 天津市自然科学基金(16JCYBJC42000,18JCYBJC85100);天津市教委科研计划项目(2017KJ237);教育部人文社会科学研究规划基金(19YJA630046)。

作者简介: 薛子晗(1995—).男,硕士研究生,主研方向为机器学习;潘 迪,硕士研究生;何 丽,教授。

收稿日期: 2019-12-25 **修回日期:** 2020-02-12 **E-mail:** 582834569@qq.com

建模能力，实现了标注效率的提升。文献[5]在LGC的基础上提出一种基于稀疏分解的l_0构图方法[6]，并将其结合到LGC算法中，提升了算法的分类精度和性能。文献[7]为LGC提供了一种新的归纳过程，诱导局部与全局一致性，提升了LGC算法的正确率。文献[8]在计算邻接矩阵时利用K-近邻图代替完全连接图，提升了时间效率，并在LGC开始迭代之前挑出噪声点，提高了LGC算法的准确率。文献[9]在计算邻接矩阵时利用K-近邻图代替完全连接图，在标签传递过程中，仅将未标记样本的标签根据相似度传递给其近邻，而将已标记样本的标签强制填回以确保标签传递源头的准确性。以上基于图的半监督学习方法虽然获得了较好的标注正确率，但是并没有考虑大规模数据集对算法执行时间的影响，忽略了算法的时间效率。针对上述问题，文献[10]提出了一个新的框架，将生成混合模型与基于图的正则化相结合；文献[11]使用顶点之间的线性组合关系来定义权重；文献[12]用生成树对图进行近似，以最小化总体切割大小的方式来标记树，并提出了一种新的方法，对生成树通过最小化目标函数，来预测未标记样本的标签[13]。

以上基于图的改进方法虽然能在一定程度上降低算法的时间复杂度，但标注正确率较低。为保证算法在标注正确率上的优势，降低图的规模，文献[14]提出了密度峰值聚类(Density Peaks Clustering，DPC)算法，随后研究人员在DPC算法的基础上进行优化与应用，取得了较好的效果[15-17]。但是这些方法都不适用于局部聚类。为使局部聚类方法能够在不同聚集形态的数据集上都能表现出较好的鲁棒性，本文基于DPC算法设计一种迭代选择中心点的密度峰值聚类(Iteration Density Peaks Clustering，IDPC)算法。利用该算法进行局部聚类，并运用每个簇的聚类中心为顶点构造图，通过迭代筛选出的聚类中心点表征原始数据的特征分布，以降低图的规模。

1 相关理论

1.1 局部与全局一致性算法

令数据集$\boldsymbol{D}=\{x_i|x_i\in\mathbb{R}^m, i=1,2,\cdots,n\}$，$n$为$\boldsymbol{D}$中的样本数。其中，$D_l=\{(x_1,y_1),\cdots,(x_l,y_l)\}$为已标记样本集合，$l<<n$，$D_u=\{x_{l+1},\cdots,x_n\}$表示未标记样本集合，$Y_l$为前$l$个已标记样本的标签集合，LGC的学习目标是利用$\boldsymbol{D}$与$Y_l$来计算$D_u$中样本的标签集合$Y_u$。用$\boldsymbol{Y}_{n*c}$表示$\boldsymbol{D}$中样本的初始化标签矩阵，其中，$c$为$\boldsymbol{D}$中样本的不同标签数。将$\boldsymbol{F}_{n*c}$定义为$\boldsymbol{D}$中样本对各个类的概率矩阵，$F_{ij}$表示$x_i$属于第$j$个类的概率。

$\boldsymbol{W}$为G中各个顶点之间的相似度矩阵，w_{ij}的计算方法如式(1)所示：

$$W_{ij}=\begin{cases}\exp\left(-\left\|x_i-x_j\right\|^2/2\sigma\right), & i\neq j\\ 0, & i=j\end{cases} \tag{1}$$

传播矩阵$\boldsymbol{S}$的计算方法如式(2)所示：

$$\boldsymbol{S}=\boldsymbol{D}^{(-1/2)}\boldsymbol{W}\boldsymbol{D}^{(-1/2)} \tag{2}$$

其中，$\boldsymbol{D}$是对角矩阵，D_{ii}为$\boldsymbol{W}$第i行的和。

获得传播矩阵$\boldsymbol{S}$后，迭代计算式(3)直到$\boldsymbol{F}$收敛，可以得到收敛状态下的最优$\boldsymbol{F}^*$。

$$\boldsymbol{F}^*=(1-\alpha)(I-\alpha\boldsymbol{S})^{-1}\boldsymbol{Y}$$

$$\boldsymbol{F}(t+1)=\alpha\boldsymbol{S}\boldsymbol{F}(t)+(1-\alpha)\boldsymbol{Y} \tag{3}$$

最后使用分类函数$y_i=\arg\max_j\leqslant_c\boldsymbol{F}^*_{ij}$来确定各个无标记样本的标签。

文献[5]在LGC算法中给出了LGC收敛性证明，并推导出$\boldsymbol{F}^*$是一个固定的值。因此，$\boldsymbol{F}^*$是LGC算法的唯一解而且与$\boldsymbol{F}$的初始值无关。

1.2 密度峰值聚类算法

传统DPC算法假设聚类中心比其临近点的局部密度更高，且与其他聚类中心的距离较远。在这种假设下，若要选取聚类中心，首先需要计算数据集$\boldsymbol{D}$中每个样本$x_i(x_i\in\boldsymbol{D}, 1\leqslant i\leqslant n)$的局部密度$\rho_i$和相对距离$\delta_i$。用$d_{ij}$表示样本$x_i$和$x_j$之间距离，且$d_{ij}=\mathrm{dist}(x_i,x_j)$是这两个样本之间的欧式距离，依此建立距离矩阵$\boldsymbol{D}_M$，即$\boldsymbol{D}_M=(d_{ij})_{n\times n}$。对于具有离散值的样本，在DPC算法中，$\rho_i$的定义为与$x_i$的距离小于$d_c$的样本个数。$x_i$的局部密度$\rho_i$的计算方法如式(4)所示：

$$\rho_i=\sum_j\chi(d_{ij}-d_c) \tag{4}$$

其中，d_{ij}为样本x_i和x_j之间的特征距离，d_c是截断距离，$\chi(\cdot)$为计数函数，定义如式(5)所示：

$$\chi(x)=\begin{cases}1, & x<0\\ 0, & x\geqslant 0\end{cases} \tag{5}$$

对数据集$\boldsymbol{D}$中的任一样本x_i计算其局部密度ρ_i后，若$\boldsymbol{D}$中存在x_j使$\rho_j>\rho_i$，则可以使用式(6)计算其距离δ_i：

$$\delta_i=\min_{j:\rho_i<\rho_j}(d_{ij}) \tag{6}$$

在式(6)中，若$\boldsymbol{D}$中存在点x_j使$\rho_j>\rho_i$，则将δ_i定义为与离x_i最近且局部密度更高的样本之间的距离；否则，将δ_i定义为与x_i相距最远的样本距x_i之间的距离。

对$\boldsymbol{D}$中的每个样本$x_i(1\leqslant i\leqslant n)$，得到其局部密度$\rho_i$与距离值$\delta_i$后，可使用式(7)来选择聚类中心：

$$\gamma_i=\rho_i\times\delta_i \tag{7}$$

其中，γ_i值越大，表示x_i为聚类中心的概率越大。对所有样本计算γ_i后，选择最大的若干个样本作为聚类中心进行聚类。

2 IDPC-LGC方法

传统的DPC方法只选择ρ与δ突出的极少数点作为聚类中心，而本文使用局部聚类的中心点作为顶点构造图，需要大量中心点来描述原始数据的特

征分布。因此,本文设计了一种迭代选取中心点的方法,并提出一种改进的DPC聚类方法IDPC。该方法使用迭代的方式选取多个中心点,并以中心点为聚类中心进行局部聚类,最后运用聚类生成簇中的已标记样本的标签对该簇的中心点进行标注。

IDPC-LGC算法实现的主要步骤如下:

1)对数据集$\boldsymbol{D}$中的所有样本,计算任意两个样本之间的欧式距离,并建立距离矩阵$\boldsymbol{D}_M$。

2)使用迭代的方法选取中心点,得到$\boldsymbol{D}$的中心点集合C。

3)以C中的每个中心点为聚类中心进行局部聚类,得到$\boldsymbol{D}$上的簇集合$C_{LS}=\{C_{L1},C_{L2},\cdots,C_{LP}\}$。

4)对C_{LS}中的每一个簇$C_{Li}(1\leqslant i\leqslant P)$,使用$C_{Li}$中已标记样本的标签对$C_{Li}$的中心点进行标注,得到中心点集合$C$的标签集合$Y_c$。

5)以中心点集合C中的每个样本为顶点构造图G,并按照式(1)计算G中的任意两个顶点之间的相似度,建立相似矩阵$\boldsymbol{W}$,然后利用Y_c完成基于LGC理论的样本标注过程,得到中心点集合C的预测标签集合Y_p。

6)利用Y_p中中心点的标签对各中心点所在簇中的所有未标注样本进行标注。

2.1 基于迭代的中心点选取方法

在IDPC-LGC算法中,中心点既是局部聚类的中心,也是基于LGC算法的样本标注的基础。为提升IDPC-LGC的标注准确率和算法执行的时间效率,选取的中心点应该能够描述原始数据集的样本分布形态,并使中心点的数量尽可能少。IDPC-LGC算法使用基于中心点的图结构实现LGC的标签传播过程。根据LGC的标签传递思想,建立图结构后,样本的标记信息不断向图中各个顶点的邻近样本传播,直至全局收敛稳定。因此,若属于不同类的中心点之间的距离太近,就可能导致本应属于不同类的中心点在LGC阶段被标注成相同的标签,导致中心点标注错误。

为保证LGC阶段中心点标注的准确率,本文在中心点选取时要求满足以下两个条件:

1)属于不同类的中心点之间的距离应尽可能远,使筛选出来的中心点尽量远离类边界。

2)应属于同一个类的中心点需尽量分布均匀,保持连贯,避免出现明显的间断情况。

对数据集$\boldsymbol{D}$中的每个样本$x_i(1\leqslant i\leqslant n)$,$n$为$\boldsymbol{D}$中的样本数。按照传统DPC算法计算其局部密度$\rho_i$与距离值$\delta_i$,并计算$\gamma_i=\rho_i\times\delta_i$。对$\boldsymbol{D}$中所有样本按$\gamma$值从大到小进行排序,将排序后的样本编号顺序加入到数组q中,即有$\gamma_{q[1]}\geqslant\gamma_{q[2]}\geqslant\cdots\geqslant\gamma_{q[n]}$。

根据DPC聚类算法的思想,样本的γ值越大,其成为簇中心的可能性越大,因此,该样本成为中心点的概率也越大。所以,可以按数组q中各个样本的出现顺序进行中心点筛选。为使筛选出的中心点能够远离分类边界,这里约定只有局部密度大于平均局部密度的样本才能参与迭代。若用$\rho_{_mean}$表示$\boldsymbol{D}$上所有样本的平均局部密度,对样本$x_{q[i]}$,当$\rho_{q[i]}>\rho_{_mean}$时,将样本$x_{q[i]}$添加到迭代训练数据集中,$\rho_{_mean}$的计算方法如式(8)所示:

$$\rho_{\_mean}=\frac{\sum_{i=1}^{n}\rho_i}{n} \tag{8}$$

若$\rho_{q[i]}>\rho_{_mean}$,从$\boldsymbol{D}_M$中选取$x_{q[i]}$的K个近邻,若$x_{q[i]}$的K个近邻均不是聚类中心,则将$x_{q[i]}$定义为一个新的中心点。这使得每个局部密度大于$\rho_{_mean}$的样本及其K邻域中至少有一个是中心点,这样可以更好地保证应属于同一个类别的中心点在形态分布上具有连贯性。若训练数据集用$\boldsymbol{D}$表示,距离矩阵为$\boldsymbol{D}_M$,中心点选取算法如算法1所示。

算法1 基于迭代的中心点选取算法

输入 数据集$\boldsymbol{D}$,距离矩阵$\boldsymbol{D}_M$,K

输出 $\boldsymbol{D}$的中心点集合C

1.初始化:令C=∅,q=∅;
2.for each $x_i\in$ D do
  使用式(4)、式(6)和式(7)分别计算ρ_i、δ_i和γ_i;
 end for
3.使用式(8)计算$\rho_{_mean}$;
4.对D中的每个样本x_i,按γ_i由大到小排序,并将其下标i加入数组q中;
5.for i=1 to n do
6. if $\rho_{q[i]}>\rho_{\_mean}$ then
7. 从D_M中选出点$x_{q[i]}$的K个近邻;
8. if($x_{q[i]}$的K个近邻均不在C中) then
9. $C>C\cup\{x_{q[i]}\}$;
10. end if
11. end if
12.end for
13.return C

算法1中K值的大小对算法的执行时间和中心点的分布有直接影响。K值越大,筛选出的中心点会越少,可能会导致中心点在分布形态上的不连贯,并使得标注准确率下降,但算法的执行时间会减少;反之,算法的标注准确率会提升,但过多的中心点会导致消耗额外的算法执行时间。K值的选取与训练数据集的规模、数据集中隐藏的类别数和数据集中样本的聚集形态有关,本文将在实验部分对K值的选取进行讨论。

算法1中的步骤4进行了由大到小的排序,对随机序列进行排序可以达到的最好时间复杂度为$O(n\log_a n)$,步骤5~步骤12为K近邻迭代过程,时间复杂度为$O(Kn^2)$,但在实际应用中,K值一般较小。因此,算法1的时间复杂度近似为$O(n^2)$。

为进一步说明本文提出的基于迭代的中心点选取方法对原始数据集特征描述的有效性,在其生成

的带有噪声的双月数据集上进行了中心点选取实验。实验中数据集的样本数为3 000,已标记样本数为16,噪声率设为0.16。数据集的原始图像和中心点选取结果如图1所示。其中,图1(a)为生成的原始数据图像,图1(b)为产生的中心点结果。从图1(a)可以看出,由于噪声的存在,两个双月之间存在比较明显的样本重叠。

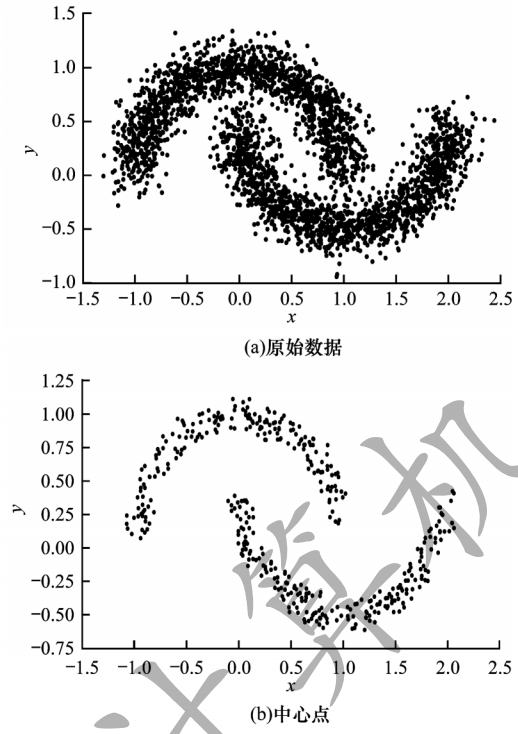

图1 原始数据与中心点的比较结果

Fig.1 Comparison result of raw data and central points

从图1(a)和图1(b)的对比可以看出,本文使用迭代选择出的中心点能够较好地描述原始数据集中两个类的特征,而在规模上,中心点的数量要明显少于原始数据集中的样本数。并且筛选出的中心点在同一分类上连贯性很强,且基本能够向类中心聚集。同时从图1(b)可以看出,两个类的中心点集群相距足够远,这为基于LGC的样本标注提供了很好的基础。

2.2 基于中心点的局部聚类方法

局部聚类的主要目的是利用同一聚类中的样本应该拥有相同类标签这一规则,来得到中心点集C的标签集合Y_c。这里的局部聚类是在已知中心点集合的情况下进行的,而且中心点理论上可以是每个聚类的中心或接近聚类中心的样本。根据DPC聚类对聚类中心的假设,中心点在局部应该拥有最高的局部密度。因此,可将非中心点归属到与其最近且密度更高的样本所在的簇,如此迭代,可以将数据集中的每个非中心点归属到其对应的中心点所在的簇。

为方便描述,本文引入聚类数组qc来记录在数据集$\boldsymbol{D}$中离当前样本最近且局部密度更高的样本的下标。对样本x_i,$qc[i]$表示$\boldsymbol{D}$中离x_i最近且局部密度更高的样本的下标,若$\boldsymbol{D}$中不存在比x_i密度更高的样本,则$qc[i]$中存储x_i的下标。

在聚类时,若$x_{qc[i]}$为中心点,则将点x_i归类$x_{qc[i]}$所在的簇,否则,将点x_i迭代归类到离$x_{qc[i]}$最近且密度更高的样本$x_{qc[qc[i]]}$,以此类推,直到将x_i归属到一个中心点所属的簇为止。基于中心点的局部聚类过程如算法2所示。

算法2 基于中心点的局部聚类算法

输入 数据集$\boldsymbol{D}$,聚类数组qc,中心点集合C

输出 簇集合$C_{LS}=\{C_{L1},C_{L2},\cdots,C_{LP}\}$

1.令下标数组s=∅;

2.对D中的每个样本x_i,按ρ_i进行由大到小排序,并将其下标i加入s中;

3.for i=1 to n do

4. if ($x_{s[i]}\notin C$) then

5. while $x_{qc[s[i]]}\notin C$ do

6. qc[s[i]]=qc[qc[s[i]]];

7. endwhile

8. 获得$x_{qc[s[i]]}$所在的簇号p;

9. $C_{LP}=C_{LP}\cup\{x_{s[i]}\}$;

10. end if

11.end for

12.return C_{LS}

在算法2中,步骤2对$\boldsymbol{D}$中的每个样本x_i按ρ_i进行由大到小排序可以达到的最好时间复杂度为$O(n\log_a n)$,对非中心点进行迭代聚类的最坏时间复杂度为$O((n-C)\times\max\rho)$,其中,C为中心点个数,$\max\rho$为$\boldsymbol{D}$中的各个样本局部密度的最大值,$\max\rho$远小于n,所以,算法2的时间复杂度为$O(n\log_a n)$。

3 实验与结果分析

3.1 实验设计

为分析不同数据规模和已标记样本比例下本文IDPC-LGC算法的有效性,首先在代码生成的有噪声的双月数据集上进行实验,以分析数据规模对标注正确率和运行时间的影响。同时,为验证IDPC-LGC算法在不同聚集形态数据集上的性能,选择4个拥有不同聚集形态和规模的公开数据集进行实验。在实验中,将本文算法与LGC、BB-LGC[9]、improved-LGC[8]、LGC-(l_0,K)[6]、KNN(K=1)、EEKNN[18]算法进行了比较。实验环境为Windows 7系统,8 GB内存,i5-4590处理器,实现语言为python,所有结果均为30次实验的平均值。

实验使用标注正确率和运行时间作为评价指标,标注正确率为标注正确样本数与数据集中的未标记样本总数的比值。

3.2　数据集规模对算法性能的影响

为分析数据集规模对算法性能的影响,首先使用代码生成的双月数据集进行实验,噪声率noise=0.16,标记样本数固定为16。不同数据规模下各个算法的标注正确率和运行时间对比如图2所示。

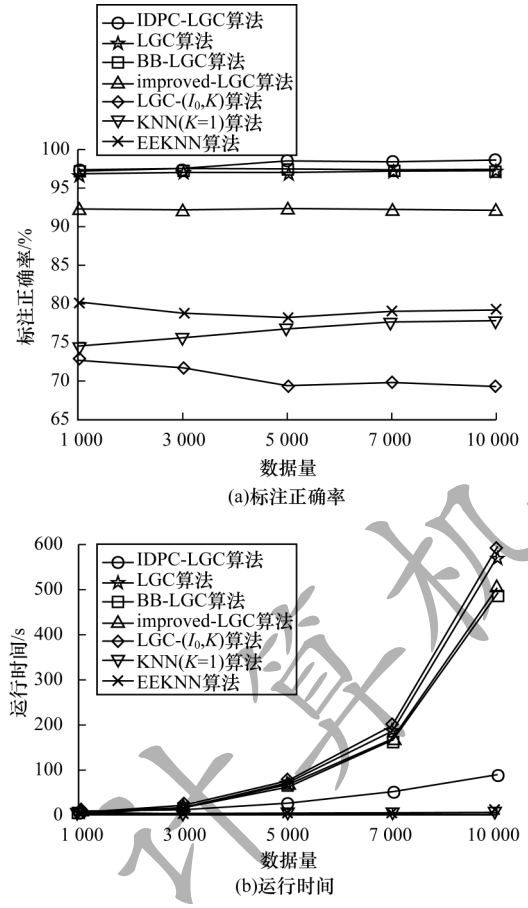

图2　数据集规模对算法性能的影响

Fig.2　Effect of dataset size on algorithm performance

从图2可以看出:随着数据量的增大,本文IDPC-LGC算法的标注正确率始终优于LGC算法与BB-LGC算法;在运行时间上,随着数据量的增大,LGC算法的运行时间增幅较快,而本文算法的增幅较小,且远低于LGC算法;相对于本文算法,BB-LGC与improved-LGC算法的时间效率优化并不明显;随着数据量的增大,本文算法在运行时间上的优势越来越明显,这主要是因为在同一特征分布下,数据规模越大,数据的密集程度就会越高,冗余性变强,这时利用中心点进行聚类可以获得更好的样本缩减比,能更有效地降低算法依赖的图的规模;LGC-(l_0,K)算法的准确率最低,是因为该算法使用k-means算法对原始数据集进行粗分类,但是k-means算法以计算各个点到聚类中心的距离为核心,在近似球状分布的数据集上有较好的表现,在双月数据集上表现不佳,因此,LGC-(l_0,K)算法的性能受数据集中样本聚集形态的影响;KNN算法与EEKNN算法的运行时间较短,但在标注正确率上表现较差。当数据集的规模为n时,LGC算法的时间复杂度为$O(n^3)$,而本文算法的时间复杂度为$O((n/t)^3)+O(n^2)$,t为局部聚类中各个簇的平均样本数,也即在局部聚类时构建图可以缩减的倍数。当n很大时,因为$(n/t)^3<<n^3$,所以本文方法在运行时间上的优势明显。

3.3　标记样本数对算法性能的影响

为进一步说明标记样本数对算法性能的影响,本文使用代码生成的双月数据集,并选择噪声率noise=0.16,样本规模n=3 000和多个不同的标记样本数进行实验,结果如图3所示。

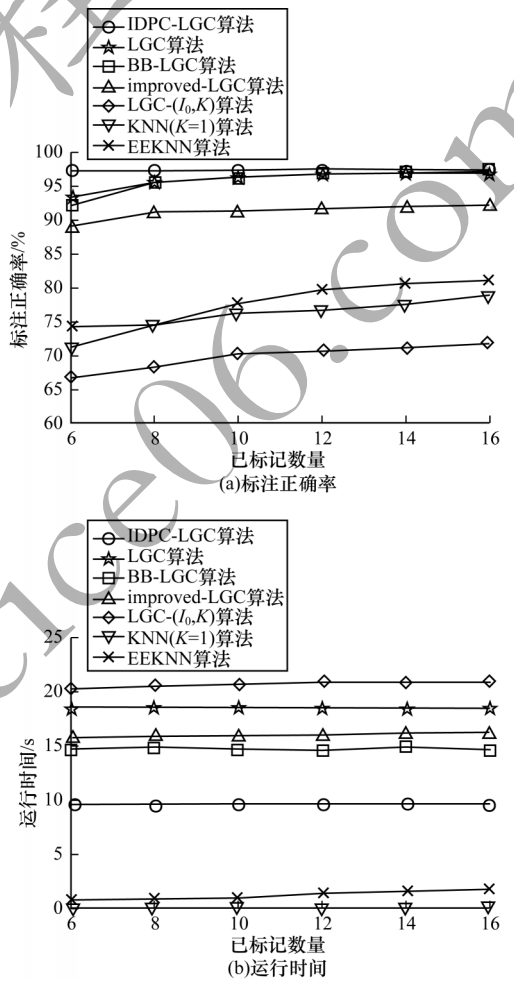

图3　标记样本数对算法性能的影响

Fig.3　Effect of labeled sample number on algorithm performance

从图3(a)可以看出,所有比较算法的标注正确率都会不同程度地受到标记样本数的影响,标记样本增加,标注的正确率也随之提升,而本文算法在较少标记样本数的情况下也能够获得较高的标记正确率,这是因为本文使用的迭代密度峰值局部聚类算法能够很好地解决类的边界重叠问题。从图3(b)可以看出,已标记样本数的变化对算法的运行时间影

响很小,EEKNN与KNN算法虽然在运行时间上优于本文算法,但标注正确率较低。总体上,本文算法在不同已标记样本数的情况下,在标注正确率和运行时间两个指标上优势明显。

3.4 数据集样本的聚集形态对算法性能的影响

为说明本文提出的IDPC-LGC算法在不同聚集形态和不同类别分布情况下的鲁棒性,在4个公开数据集上分别进行实验,并对不同算法在各个数据上的标注正确率和运行时间进行了比较,如表1所示。IDPC-LGC算法适用于大规模的数据集,并且数据集中各个类的边界越模糊,IDPC-LGC算法的优势将会越明显。为证明这一点,选择两个有边界重叠的近似球型数据集D31[19]和S2[20]。同时,为证明本文方法在小数据集和其他形态数据集上的有效性,选择了数据集Aggregation以及Flame。从表1可以看出,4个数据集的规模和类别数有较明显的变化。

表1 数据集属性

Table 1 Dataset attribute

数据集	数据量	类别数	维度
D31	3 100	31	2
S2	5 000	15	2
Aggretation	788	7	2
Flame	240	2	2

IDPC-LGC算法在各个数据集上使用的参数设置和产生的中心点数如表2所示。

表2 参数设置

Table 2 Parameter settings

数据集	*dc*	*K*	中心点数
D31	0.01	10	634
S2	0.01	50	601
Aggregation	0.01	11	134
Flame	0.01	1	92

表3和表4比较了各算法在4个数据集上的标注正确率和运行时间。

表3 标注正确率结果比较

Table 3 Comparison of labeling accuracy results %

算法	D31	S2	Aggregation	Flame
IDPC-LGC	95.58	96.94	99.49	99.16
LGC	95.00	93.46	98.85	85.83
BB-LGC	96.35	91.60	99.36	92.91
improved-LGC	94.03	91.04	98.73	91.66
LGC-(l_0,K)	97.67	96.98	93.90	83.75
KNN	67.70	89.34	91.23	84.58
EEKNN	76.72	84.04	93.90	81.30

表4 运行时间结果比较

Table 4 Comparison of running time results s

算法	D31	S2	Aggregation	Flame
IDPC-LGC	13.97	21.36	1.09	0.24
LGC	70.62	83.39	1.18	0.12
BB-LGC	20.91	60.76	0.56	0.10
improved-LGC	23.29	63.16	0.54	0.23
LGC-(l_0,K)	97.67	83.15	3.26	1.68
KNN	0.02	0.03	0.01	0.01
EEKNN	7.62	4.21	0.23	0.10

从表3和表4可以看出,在4个数据集上本文算法在标注正确率上均优于LGC、BB-LGC与improved-LGC算法,且LGC算法在数据集Flame上的标注正确率较低。LGC-(l_0,K)虽然在S2与D31两个数据集上具有最高的标注准确率,但在Flame上表现较差,因为该算法使用k-means进行粗分类,聚类结果与数据集中样本的聚集形态有关。表3的结果说明,本文算法对不同聚集形态和规模的数据集都具有较好的适应性,鲁棒性较好。在运行时间上,本文算法在规模较大的D31和S2数据集上明显优于在标注正确率上表现较好且稳定的LGC、BB-LGC与improved-LGC算法,虽然不及KNN和EEKNN算法,但是KNN和EEKNN的标注正确率相对较低,并且表现不稳定。与表现较好的LGC、BB-LGC与improved-LGC算法相比,本文算法在运行时间上的优势明显,并且数据集的规模越大,这种优势将更加明显,这主要是因为本文使用基于迭代的密度峰值局部聚类方法能够有效降低LGC算法依赖的图的规模。

实验结果显示,本文提出的IDPC-LGC算法在不同规模、不同标记样本数和不同聚集形态的数据集上,都能在标注正确率和运行时间两个评价指标上保持较好的优势。

3.5 参数讨论

IDPC-LGC算法涉及的参数较多,其中影响最大的是DPC聚类算法中的截断距离*dc*与迭代中*K*值的选取。因为*dc*值在各样本间距离值排列在前1%位置时,能够在各个数据集上获得最佳的聚类效果,而算法对*K*值的选取比较敏感,所以本节主要分析*K*值变化对算法性能的影响。*K*值的选取方法如式(9)所示:

$$K=\theta\sqrt{n/c} \tag{9}$$

其中,c为样本类别数,θ为调整系数,可以根据数据集中样本分布的特征及数据规模的大小进行调整,本文默认为1。若图像上各个聚类的形态类似球型,且数据量偏大,则表明可以用更少的中心点对原始数据的特征进行表征,这时θ值可以略大于1;若各个聚类的形态扁平或表现为各种不规则形状,这时

需要避免筛选出的中心点出现断层或分布不均匀的情况,因此需要将θ设置为小于1的数;在数据量极小且分类边界模糊的数据集上,如3.4节提到的Flame数据集,需要通过调整θ值使K值为1。

在数据集D31的实验中,将θ值设为1时,使用式(9)得到K=10。本节将观察K值变化对D31实验结果的影响,如图4所示。

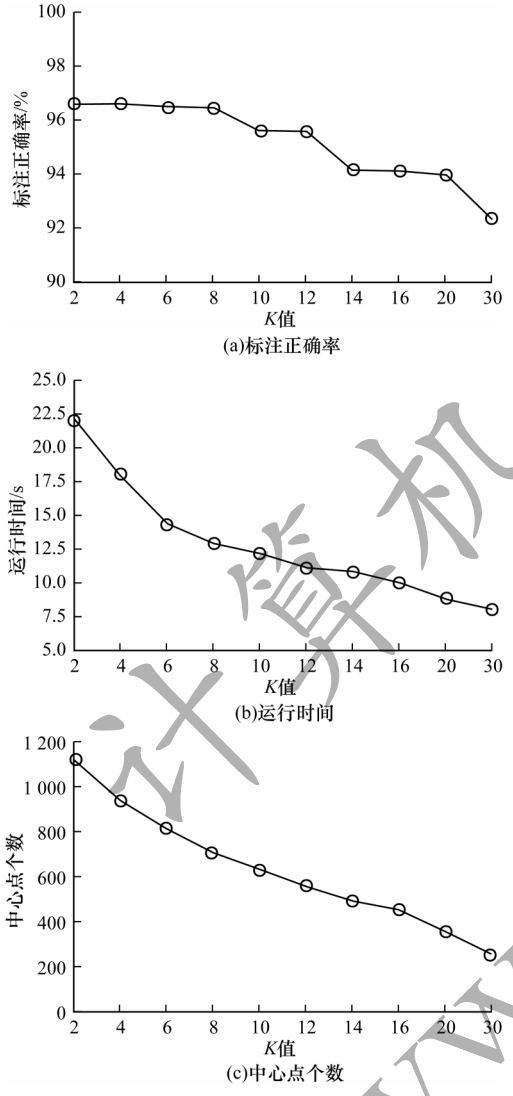

图4　K值变化对IDPC-LGC性能的影响

Fig.4　Effect of K value on IDPC-LGC performance

从图4(b)可以看出,当K值过小时,IDPC-LGC的运行时间偏高,因为K值越小,使用迭代筛选出的中心点数就越多,运用中心点建立的图的规模就越大,LGC运行所花费的时间也越多。同时,从图4可以发现,随着K值的增加,运行时间和中心点数下降较快,而标注正确率在一定范围内能够保持相对稳定。然而,当K值继续增加到30时,算法的标注正确率大幅下降,这是因为K值过大会导致中心点数量偏少,使得同一类别的中心点集出现断层或分布不均匀的情况,从而影响最终的标注正确率。

4　结束语

针对LGC半监督学习算法时间复杂度较高的问题,本文提出一种改进的半监督学习算法IDPC-LGC。通过迭代产生的少量中心点构建局部与全局一致性运行的图结构,实现基于LGC的半监督学习。实验结果表明,该算法能够有效降低LGC算法运行图的规模。同时,使用基于中心点的局部聚类方法能够较好地表达原始数据集的特征分布,适应不同聚集形态数据集的特征分布,有效降低噪声对标注准确率的影响,获得更优的标注准确率和运行时间。下一步将研究迭代过程中K值的自适应选取以及IDPC-LGC算法在大规模数据场景中的具体应用。

参考文献

[1] HADY M F A, SCHWENKER F. Semi-supervised learning[J]. Journal of the Royal Statistical Society, 2006,172(2):530-530.

[2] BLUM A, CHAWLA S. Learning from labeled and unlabeled data using graph mincuts[C]//Proceedings of the 18th International Conference on Machine Learning. San Francisco,USA:Morgan Kaufmann,2001:19-26.

[3] ZHANG Xiaoyan. Label propagation classification based on semi-supervised affinity propagation algorithm[C]// Proceedings of 2015 International Conference on Cyber Technology in Automation,Control,and Intelligent Systems. Shenyang,China:[s. n.],2015:476-481.

[4] LI Qimai,WU Xiaoming,LIU Han,et al. Label efficient semi-supervised learning via graph filtering[C]// Proceedings of IEEE Conference on Computer Vision and Pattern Recognition. Washington D. C.,USA:IEEE Press,2019:9582-9591.

[5] ZHOU D,BOUSQUET O,LAI T,et al. Learning with local and global consistency[C]//Proceedings of International Conference on Advances in Neural Information Processing Systems. Cambridge,USA:MIT Press,2003:321-328.

[6] BAI Bendu,FAN Jiulun. Learning with local and global consistency based on sparse representation[J]. Journal of Xi'an University of Posts and Telecommunications,2015,20(3):65-70. (in Chinese)
白本督,范九伦. 基于稀疏分解的局部全局一致性学习算法[J]. 西安邮电大学学报,2015,20(3):65-70.

[7] SOUSA C A R. An inductive semi-supervised learning approach for the local and global consistency algorithm[C]// Proceedings of 2016 International Joint Conference on Neural Networks. Vancouver,Canada:[s. n.],2016:4017-4024.

[8] LI Ming,ZHANG Xiaoli,WANG Xuesong. An improved learning with local and global consistency[C]//Proceedings of Chinese Control and Decision Conference. Xuzhou,China:[s. n.],2010:1152-1156.

[9] WANG Xuesong, ZHANG Xiaoli, CHENG Yuhu. Barebones learning with local and global consistency[J]. Control and Decision,2011,26(11):1726-1730.

(下转第89页)

（上接第83页）

[10] ZHU X, LAFFERTY J. Harmonic mixtures: combining mixture models and graph-based methods for inductive and scalable semi-supervised learning[C]//Proceedings of the 22nd International Conference on Machine Learning. Bonn, Germany: [s. n.], 2005: 1052-1059.

[11] WANG Fei, ZHANG Changshui. Label propagation through Linear neighborhoods[J]. IEEE Transactions on Knowledge & Data Engineering, 2007, 20(1): 55-67.

[12] ZHANG Yanming, HUANG Kaizhu, GENG Guanggang, et al. MTC: a fast and robust graph-based transductive learning method [J]. IEEE Transactions on Neural Networks and Learning Systems, 2015, 26(9): 1979-1991.

[13] ZHANG Yanming, ZHANG Xuyao, YUAN Xiaotong, et al. Large-scale graph-based semi-supervised learning via tree laplacian solver [C]//Proceedings of the 30th AAAI Conference on Artificial Intelligence. [S. 1.]: AAAI Press, 2016: 235-246.

[14] RODRIGUEZ A, LAIO A. Clustering by fast search and find of density peaks[J]. Science, 2014, 344(6191): 1492-1496.

[15] CHEN Yuhong, ZHANG Qinghua, YANG Jie. Density peak clustering algorithm based on interval shadowed sets[J]. Pattern Recognition and Artificial Intelligence, 2019, 32(6): 531-544. (in Chinese)
陈玉洪，张清华，杨洁. 基于区间阴影集的密度峰值聚类算法[J]. 模式识别与人工智能，2019，32(6)：531-544.

[16] QIAN Xuezhong, YAO Linya. Extended incremental fuzzy clustering algorithm for sparse high-dimensional big data[J]. Computer Engineering, 2019, 45(6): 75-81. (in Chinese)
钱雪忠，姚琳燕. 面向稀疏高维大数据的扩展增量模糊聚类算法[J]. 计算机工程，2019，45(6)：75-81.

[17] DONG Xiaojun, CHENG Chunling. K-CFSFDP Clustering algorithm based on kernel density estimation[J]. Computer Science, 2018, 45(11): 244-248. (in Chinese)
董晓君，程春玲. 基于核密度估计的K-CFSFDP聚类算法[J]. 计算机科学，2018，45(11)：244-248.

[18] LI Ni, KONG Haipeng, MA Yaofei, et al. Human performance modeling for manufacturing based on an improved KNN algorithm[J]. The International Journal of Advanced Manufacturing Technology, 2016, 84(4): 473-483.

[19] VEENMAN C J, REINDERS M J T, BACKER E. A maximum variance cluster algorithm[J]. IEEE Transactions on Pattern Analysis & Machine Intelligence, 2002, 24(9): 1273-1280.

[20] FRANTI P, VIRMAJOKI O. Iterative shrinking method for clustering problems[J]. Pattern Recognition, 2006, 39(5): 761-775.

编辑　索书志