文章编号: 1001-0920(2017)10-1796-07 DOI: 10.13195/j.kzyjc.2016.0861

基于概率无向图模型的近邻传播聚类算法

覃 华, 詹娟娟†, 苏一丹

(广西大学 计算机与电子信息学院，南宁 530004)

摘 要: 针对近邻传播聚类算法偏向参数难选定、生成的簇数目偏多等问题,提出一种概率无向图模型的近邻传播聚类算法. 首先为样本数据构建概率无向图模型,利用极大团和势函数计算无向图中数据样本的概率密度,将此概率密度作为一种聚类先验知识注入近邻传播算法的偏向参数中,提高算法的聚类效率;并用高斯降噪和簇归并方法进一步提升算法的聚类精度. 在UCI数据集上的实验结果表明,所提出算法的聚类效率和精度均优于相比较的同类算法.

关键词: 近邻传播聚类算法；偏向参数；概率无向图模型；高斯平滑；簇归并

中图分类号: TP301.6 **文献标志码:** A

Affinity propagation clustering algorithm based on probabilistic undirected graphical model

QIN Hua, ZHAN Juan-juan†, SU Yi-dan

(College of Computer and Electronic Information，Guangxi University，Nanning 530004，China)

Abstract: In order to solve the problem that the preference of the traditional affinity propagation clustering algorithm is difficult to choose and the number of generate clusters is likely to be overmuch, an affinity propagation clustering method based on the probabilistic undirected graph model is proposed in this paper. Firstly, the probabilistic undirected graph model is constructed for sample data, while the probability density is calculated for each sample data by maximum clique and potential function. Then the probability density as a priori clustering knowledge is put into the preference of the affinity propagation algorithm to improve its efficiency. The clustering accuracy of the algorithm is further promoted by using the Gauss noise reduction and cluster merging method. Experimental results on the UCI data sets show better clustering efficiency and accuracy of the proposed algorithm against several other similar algorithms.

Keywords: affinity propagation clustering algorithm；preference；probabilistic undirected graphical model；gaussian smooth；cluster merging

0 引 言

近邻传播聚类算法(AP)是一种新型聚类算法[1-2],与传统的K-means等聚类算法相比,它事先不需要知道类别个数,根据输入的数据集,通过反复迭代自动找出聚类中心,具有很强的通用性,目前已被应用于文本挖掘、图像识别、基因数据处理等领域[3-6]. 但在实际应用中,AP算法存在以下问题: 1)对偏向参数取值极为敏感,取值不当易引发迭代震荡,影响聚类效果; 2)产生的簇数目比实际的簇数目多; 3)处理复杂数据集耗时长且聚类效果欠佳. 针对AP算法存在的这些问题,有学者提出了相关改进. 文献[7]通过对偏向参数及阻尼因子自适应取值,使聚类结果更加精准; 文献[8]引入半监督思想,利用已知标签作为先验知识对相似度矩阵进行调整,以提高了AP聚类准确度; 文献[9]结合CVM压缩与合并方法,对大型数据集进行压缩分类预处理,提高算法的准确率; 文献[10-11]的K-AP算法中,允许用户预先设定样本聚类簇数目K,减少确定簇数目的计算开销,提高了计算效率; 文献[12]提出惩罚权重和奖励权重,减少了算法的迭代次数; 文献[13]提出高斯核函数的自适应AP聚类算法,在一定程度上克服了原有算法对数据集敏感的问题; 文献[14]将概率无向图模型与服务语义链网络相结合,用概率无向图模型的联合分布概率优化服务语义链网络.

收稿日期: 2016-07-04；**修回日期:** 2016-12-11.

基金项目: 国家自然科学基金项目(61363027)；教育部人文社会科学研究规划基金项目(11YJAZH080).

作者简介: 覃华(1972−),男,教授,从事量子计算理论、近似动态规划最优化方法、数据挖掘等研究;詹娟娟(1993−),女,硕士生,从事数据挖掘的研究.

†通讯作者. E-mail: cherryzhan1993@163.com

本文提出一种基于概率无向图模型的近邻传播聚类算法(PUGM-AP),用概率无向图模型计算AP算法的偏向参数,在偏向参数中注入数据样本相互依赖的先验知识,提高算法的聚类效率和精度,以及对数据集的自适应性. 主要工作有: 1) 根据AP算法特点,为数据集建立一个概率无向图模型,利用概率无向图的最大团和势能函数计算数据样本的概率密度,并用此概率密度计算AP算法的偏向参数; 2) 引入高斯平滑算子对数据集作降噪处理,减少噪声数据对聚类精度的影响; 3) 对算法产生的初始簇进行必要归并,使算法产生的簇数目与真实的簇数目一致. 最后在UCI数据集上验证本文算法的可行性,并与传统AP算法、K-AP、M-AP算法进行比较.

1　用概率无向图模型计算偏向参数

AP聚类算法的基本思想是: 给定数据集$X = \{X_1, X_2, \cdots, X_N\}$,利用距离公式计算相似度矩阵$S$,然后构造循环,迭代更新支持度矩阵($R$)和归属度矩阵($A$),迭代过程中根据$R$和$A$逐步筛选出聚类中心点. 相似度矩阵$S$对角线上的元素$S(i,i)$称为偏向参数,它表示数据样本$X_i$作为聚类中心的适合程度,直接影响$R$和$A$的计算,从而影响聚类中心的产生. 传统AP算法假定初始时每一个数据样本作为聚类中心的概率相同,故令对角线元素$S(i,i)$取相同的值,即$S(i,i) = p$. 但由图1的数据分布特征可知,X_i比X_j成为聚类中心的概率更大,故$S(i,i)$取相同的初值会导致AP算法花费更多的迭代次数和计算时间去发现真正的聚类中心. 如果能事先估计数据样本X_i的概率密度$P(X_i)$,并用$P(X_i)$计算偏向参数p,则相当于在$S(i,i)$中引入了聚类先验知识,区分各数据样本成为聚类中心的概率,有利于减少算法的迭代次数和计算时间,并提高聚类精度. 本文采用概率无向图模型计算数据样本X_i的概率密度$P(X_i)$.

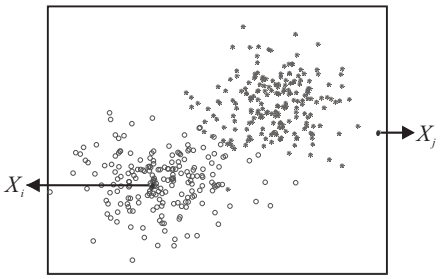


图1　数据样本分布图

对于数据集$X = \{X_1, X_2, \cdots, X_N\}$,将数据样本$X_i$看作无向图的一个顶点$v_i$,由此可得无向图的顶点集$V = \{v_1, v_2, \cdots, v_N\}$. 定义顶点的边集为$E$,并用矩阵形式表示,矩阵中元素$E(i,j)$表示顶点$v_i$与$v_j$有依赖关系,其取值的确定方法是: 引进一个阈值$\eta$,顶点$v_i$和$v_j$在相似度矩阵$S$中对应的值为$S(i,j)$,则有

$$E(i,j) = \begin{cases} 1, & S(i,j) > \eta; \\ 0, & (i,j) \leqslant \eta. \end{cases} \tag{1}$$

其中: 1表示顶点v_i与v_j间有边; 0表示顶点v_i与v_j之间无边.

定义顶点v_i对应的随机变量为Y_i,得到随机变量的集合$Y = \{Y_1, Y_2, \cdots, Y_N\}$. 下面证明所建立的无向图$G = (V, E)$具有成对马尔科夫性[15]:

在AP算法的相似度矩阵S中,数据样本X_i与X_j的距离$S(i,j)$不会随其他任意数据样本的存在或消失而发生变化,即X_i与X_j之间的距离关系与其他任意数据样本$k(k \in [1,N]$,且$k \neq \{i,j\})$间的距离相互独立,因此S中的各元素之间也是相互独立的. 同理可得: 由相似度矩阵S构建的边集E的各元素之间也是相互独立的. 对于数据样本X_i与X_j,边集E中相对应的元素为$E(i,j)$,在顶点集V中相对应的顶点为v_i和v_j,在Y中所对应的随机事件分别为Y_i和Y_j; 除v_i和v_j外的其他任意顶点记为v_k,v_k所对应的随机事件记为Y_k. 由已知条件可知: 无向图中顶点v_i、v_j与E中所对应随机事件Y_i和Y_j相互独立,即事件Y_i、Y_j对应的联合分布概率满足

$$P(Y_i, Y_j | Y_k) = P(Y_i | Y_k) P(Y_j | Y_k). \tag{2}$$

式(2)证明本文构建的无向图$G = (V, E)$满足成对马尔科夫性,由概率无向图模型(PUG)的定义可知[16-17]: 所建立的无向图$G = (V, E)$是一个概率无向图模型,也称为马尔科夫随机场(MRF).

利用概率无向图的最大团和势能函数来计算随机变量Y_i的概率$P(Y_i)$. 团是无向图G顶点集V的一个子集,记为C,团中任意两个顶点均有边连接. 如果C中不能再加入G中任何一个顶点,使之成为更大的团,则称此C为最大团. 包含有顶点v_i的最大团的集合记为C_i,根据Hammersley-Clifford定理[18],有

$$P(Y_i) = \prod_{C_i} \psi_{C_i}(Y_{Ci}), \tag{3}$$

$$\psi_{C_i}(Y_{Ci}) = \exp\{-E(Y_{Ci})\}. \tag{4}$$

其中: $\psi_{C_i}(Y_{C_i})$为势能函数,在C_i上是严格为正的指数函数,Y_{C_i}为C_i对应的随机变量. 根据X_i、v_i、Y_i之间的对应关系,得到数据样本X_i的概率密度

$$P(X_i) = P(Y_i). \tag{5}$$

用式(5)计算AP算法的偏向参数

$$S(i,i)=P(X_i)S_{\min}, \tag{6}$$

其中$S_{\min}$为矩阵S中的最小值. 式(6)实现了用概率无向图模型计算AP偏向参数的目标,由于概率无向图模型能反映顶点之间的依赖关系,相当于在式(6)的偏向参数中加入了数据样本间的相互依赖先验知识,有利于加快AP聚类中心的出现.

2 基于概率无向图模型的AP聚类算法

本文在传统AP算法基础上,引入概率无向图模型、高斯平滑算子、簇归并等方法对其进行改进,算法步骤如下.

输入: 数据集X;

输出: 簇中心点的位置及个数.

Step 1: 初始化算法参数. 阻尼因子λ,最大迭代次数maxits,连续收敛次数convits,高斯算子参数σ,初始化吸引度矩阵R和归属度矩阵A.

Step 2: 数据集预处理. 读入数据集X,用高斯算子对X降噪,再对X进行归一化.

Step 3: 将文献[19]的组合核函数作为距离函数,读入预处理后的X,计算相似度矩阵S.

Step 4: 根据X和S构造概率无向图模型$G=(V,E)$,利用第1节中的概率无向图模型计算各随机变量Y_i的概率$P(Y_i)$,利用式(6)计算相应的偏向参数p.

Step 5: 构造循环,迭代以下子步骤:

Step 5.1: 计算并更新吸引度矩阵R,即

$$r(i,k)=s(i,k)-\max_{k'\neq k}\{a(i,k')+s(i,k')\}, \tag{7}$$

$$r_i(i,k)=\lambda r_{i-1}(i,k)+(1-\lambda)r_i(i,k). \tag{8}$$

其中

$$r(k,k)=p(k)-\max\{a(k,j)+s(k,j)\}, \tag{9}$$

$$a(k,k)=\sum_{i'\neq k}\max\{0,r(i',k)\}. \tag{10}$$

Step 5.2: 计算并更新归属度矩阵A,即

$$a(i,k)=\min\Big\{0,r(k,k)+\sum_{i'^1 i}\max\{0,r(i',k)\}\Big\}, \tag{11}$$

$$a_i(i,k)=\lambda a_{i-1}(i,k)+(1-\lambda)a_i(i,k). \tag{12}$$

Step 5.3: 计算$E=R+A$,从E中找出每行中值最大的数据样本,作为本次迭代所得的聚类中心点.

Step 5.4: 若已达到最大迭代次数maxits,或最近连续convits次迭代的聚类中心点均保持不变,则迭代结束,转Step 6;否则转至Step 5.1.

Step 6: 将Step5产生的簇进行适当簇归并.

Step 7: 输出最终的聚类结果.

上述算法步骤中的若干要点说明如下.

2.1 Step 2中的高斯平滑预处理

原始数据集X中可能包含有噪声点,在聚类过程中会引发算法误判,从而影响聚类效果. 根据高斯算子对图像平滑降噪的作用[20-21],本文引入高斯平滑算子对数据集X进行平滑降噪预处理[22],所用高斯算子函数如下:

$$G(X)=\frac{1}{\sqrt{2\pi}\sigma}\mathrm{e}^{-\frac{X^2}{2\sigma^2}}, \tag{13}$$

其中σ为高斯算子参数. 用高斯平滑算子预处理数据的主要步骤为: 1) 在高斯分布下导入样本数据集X; 2) 根据高斯分布的概率密度函数确定样本点的权重; 3) 根据各样本的权重确定最终参与聚类的数据样本,并归一化.

2.2 Step 3中的组合核函数与过拟合

对于给定的一个数据集X,本文利用支持向量机的核函数计算数据样本的相似度矩阵S,再根据S建立概率无向图模型G. 若核函数选择不当或数据样本较少,则核函数有可能过拟合数据的特征,所得相似度矩阵中的特征与实际情况偏差较大,用该矩阵生成的无向图计算数据样本的概率密度与样本的实际概率密度会有较大偏差,又因AP算法对偏向参数较敏感,故将失真的样本概率密度注入AP的偏向参数后,可能会导致AP聚类失效. 为方便判别核函数的选择对数据集X是否有效,本文Step3中使用文献[19]的组合核函数,其形式如下:

$$\widetilde{K}(X_i,X_j)=\alpha_1K_1(X_i,X_j)+\alpha_2K_2(X_i,X_j)+\alpha_3K_3(X_i,X_j). \tag{14}$$

其中K_1、K_2、K_3分别代表径向基核函数、多项式核函数与高斯核函数,核函数的定义分别为

$$K_1(X_i,X_j)=\exp(-\gamma||X_i-X_j||^2), \tag{15}$$

$$K_2(X_i,X_j)=[(X_i\cdot X_j)+1]^d, \tag{16}$$

$$K_3(X_i,X_j)=\exp\Big(\frac{-||X_i-X_j||^2}{2\sigma^2}\Big). \tag{17}$$

利用文献[19]的半定规划法解出最优组合系数α_1、α_2、α_3,如果某个组合系数接近于零,则表示与之对应的核函数在数据集X上接近无效,需要重新设计,从而避免核函数过拟合数据特征.

2.3 Step 6中的簇归并

因传统AP算法产生的簇个数常比实际簇的个数多,导致聚类中心不准确,聚类精度下降. 本文对Step 5产生的簇进行必要归并,设Step 5所得的聚类簇中心为$M=\{M_1,M_2,\cdots,M_K\}$,K为簇的总个数,M_i为第i个簇中心点. 记D_i为第i个簇内数据样

本之间的距离和,NUM_i为第i个簇内数据样本的总数,T_i为第i个簇内数据样本间的平均距离,有

$$T_i = \frac{D_i}{\mathrm{NUM}_i(\mathrm{NUM}_i - 1)}. \tag{18}$$

各簇数据样本平均距离之和

$$T = \sum_{i=1}^{K} T_i. \tag{19}$$

数据集X的所有数据样本之间的平均距离为

$$\bar{d} = \frac{\mathrm{dist}(X')}{N(N-1)}, \tag{20}$$

其中$\mathrm{dist}(X, X')$为加权欧氏距离.

令M中任意两个簇M_i、M_j的簇间距离为$D'_{i,j}$. 对M进行簇归并的子算法步骤如下.

Step 1: 采用加权欧氏距离公式,$i = 1, 2, \cdots, K$,分别计算各簇内数据样本间的距离D_i.

Step 2: $i = 1, 2, \cdots, K$,利用式(18)计算各簇数据样本平均距离T_i,并利用式(19)计算各个簇内平均距离之和T.

Step 3: 根据式(20)计算整个样本数据集X的平均距离$\bar{d}$.

Step 4: 构造循环,归并M中的簇.

Step 4.1: 从M中取两个簇M_i和M_j,计算两簇间距离$D'_{i,j}$.

Step 4.2: 根据$D'_{i,j}$进行判断,若同时满足以下两个条件:

$$\min(\min(D'_{i,j})) < \frac{T^2}{\bar{d}}, \tag{21}$$

$$\max(\max(D'_{i,j})) < T, \tag{22}$$

则认为M_i与M_j两个簇较为相似,可合并为同一簇;否则,认为M_i与M_j两个簇有较大差异,是两个不同的簇,不执行合并操作.

Step 5: 输出归并、更新后的簇集合M.

3　实验与分析

实验环境: PC机处理器为Intel Pentium G460 (2.8 GHz), 内存8 GB, Windows764 bit, 用Matlab实现算法. 取7个已知类别(簇)的UCI数据集测试算法,如表1所示.

表1　实验数据集

数据集[23]	样本数目	属性数目	簇的数目
Iris	150	4	3
Wine	178	13	3
Ionosphere	351	34	2
vote	435	16	2
Pima	768	8	2
Yeast	1 484	8	10
Soybean	307	35	15

算法参数中阻尼因子λ取0.8,最大迭代次数maxits为1 000,连续收敛次数convits取50,高斯算子参数σ取$\sqrt{2}$. 将本文算法与传统AP算法、K-AP算法、M-AP算法比较聚类结果. 采用准确率、正则化互信息、芮氏指标[24]对算法的聚类结果进行评估.

1) 准确率

$$\mathrm{ACC} = \sum_{i=1}^{K} \frac{L_i}{\mathrm{NUM}_i}. \tag{23}$$

其中: K为聚类后所得簇的总数; L_i为第i个簇内的数据样本中聚类正确的样本数目. ACC将聚类后一个簇内的样本与相应真实簇中的样本相比较来判断正确聚类的样本数L_i.

2) 正则化互信息

$$\mathrm{NMI} = \frac{\sum_{i=1}^{K}\sum_{j=1}^{K} N_{i,j} \log \frac{N N_{i,j}}{N_i N_j}}{\sqrt{\sum_{i=1}^{K} N_i \log \frac{N_i}{N} \sum_{j=1}^{K} N_j \log \frac{N_j}{N}}}. \tag{24}$$

其中: K为簇的总数目, $N_{i,j}$为聚类结果的第i簇中属于第j类(簇)的样本数目, N_i、N_j为第i、j簇中数据样本的数目, N为数据样本总数. NMI指标为聚类个数与真实类别个数之间不匹配的容忍度,取值范围为[0,1],指标值越接近1,表示聚类结果与真实类别的匹配度越高,聚类效果越好.

3) 芮氏指标

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}}. \tag{25}$$

其中: TP为同一类的数据样本被分到同一个簇内的数目, TN为不同类的数据样本被分到不同簇中的数目, FP为不同类的数据样本被分到同一簇中, FN为同一类的数据样本被分到不同簇中, N为整个数据集样本的总数量. 芮氏指标与ACC指标相似,但它是在不知道聚类所得的一个簇与真实簇之间的对应关系下,评价聚类结果的准确性. 例如: 聚类得到的簇用1、2、3表示,但真实簇却用A、B、C表示,不知道簇1对应A,还是B或C的情况下,评价聚类的准确性.

表2是在5个数据集上,4种算法聚类结果的ACC、NMI、RI指标以及计算时间TIME的比较. 由表2的数据可知: 在7个数据集中,本文的PUGM-AP算法聚类效果优于传统AP算法、K-AP算法和M-AP算法. 以Ionosphere数据集为例,在ACC指标上,传统AP算法与K-AP算法的ACC指标值相当,但本文算法的ACC值比AP、K-AP算法均高出1.8倍,同时也比M-AP算法高16.9 %; 在NMI、RI指标上,本文算法比AP算法高出1.3倍,比K-AP、M-AP算法均高22 %.

表 2 聚类结果的比较

算 法	比较指标	数据集						
		Iris	Wine	Ionosphere	Vote	Pima	Yeast	Soybean
AP	ACC	0.886	0.702	0.290	0.813	0.674	0.135	0.365
	NMI	0.730	0.428	0.132	0.409	0.051	0.264	0.638
	RI	0.832	0.718	0.586	0.742	0.559	0.751	0.902
	t/s	7.04	9.94	36.29	171.06	234.02	606.83	13.80
K-AP	ACC	0.500	0.404	0.290	0.103	0.649	0.181	0.322
	NMI	0.672	0.428	0.132	0.474	0.063	0.257	0.505
	RI	0.812	0.678	0.586	0.778	.546	0.735	0.867
	t/s	2.60	3.43	5.08	8.05	20.56	101.40	2.99
M-AP	ACC	0.840	0.707	0.709	0.848	0.630	0.388	0.528
	NMI	0.752	0.432	0.132	0.410	0.042	0.266	0.646
	RI	0.865	0.720	0.586	0.742	0.533	0.745	0.903
	t/s	2.81	2.89	7.34	9.93	14.17	223.36	5.43
PUGM-AP	ACC	0.960	0.747	0.829	0.880	0.662	0.498	0.588
	NMI	0.864	0.492	0.314	0.415	0.069	0.352	0.625
	NMI	0.949	0.744	0.715	0.765	0.574	0.727	0.881
	t/s	2.06	2.06	3.68	5.86	8.14	15.33	2.55

在Yeast、Soybean数据集上,本文算法的实验结果均优于AP、K-AP、M-AP算法. 在Pima数据集上,虽然本文算法的ACC指标较AP算法差1.7 %,但偏差较小,可认为此指标上两算法性能相当;在NMI指标上,本文算法比AP算法高33 %,偏差显著. 综合考量,在Pima数据集上本文算法的结果优于AP算法. Pima数据集的比较结果说明:评价算法聚类效果时,因不同指标的侧重点和计算方法不同,需要综合多个指标从多个角度综合考量,使评价结果更客观、全面.

图2直观地展示了3种算法在5个数据集上计算时间的比较. 表3是本文算法与AP算法迭代次数的比较.

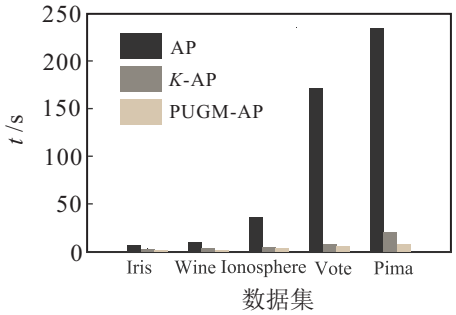


图 2 计算时间比较

表 3 PUGM-AP算法和AP算法的迭代次数

迭代次数	算法	数据集				
		Iris	Wine	Ionosphere	Vote	Pima
Iterations	AP	113	156	219	313	316
	PUGM-AP	77	79	71	129	109

由表2、表3和图2可知:本文算法的计算效率优于AP、K-AP、M-AP算法. 在简单的数据集上,例如Iris,PUGM-AP比AP算法快2.4倍;在复杂的数据集上,例如Pima、Yeast,PUGM-AP算法比AP快27、38倍之多.

表4的数据是用传统AP算法对Pima数据集聚类时,需要列表记录偏向参数的系数a对聚类精度的影响,然后根据列表筛选出最优的系数a来计算偏向参数值$p = a\text{median}(S)$[1]. 表4说明,传统的AP算法对于不同数据集,需要手工调整偏向参数值以获取好的聚类效果. 相比之下,本文采用概率无向图模型实现对偏向参数的自动计算,避免了传统AP算法手动调整偏向参数的弊端,新算法具有较强的自适应性.

表 4 偏向参数系数a的选择

数据集	真实簇	系数a	簇数目	ACC	NMI	RI	t/s
Pima	2	55	3	0.413	0.031	0.505	201.73
		56	3	0.186	0.056	0.501	191.59
		57	3	0.186	0.056	0.507	192.59
		58	3	0.486	0.045	0.523	227.33
		59	2	0.675	0.518	0.559	227.33
		60	2	0.675	0.052	0.559	251.23
		61	2	0.675	0.051	0.559	251.23
		62	3	0.500	0.038	0.522	328.44
		63	3	0.500	0.038	0.522	321.38
		64	3	0.500	0.038	0.521	241.67
		65	2	0.665	0.079	0.559	210.42

综上所述,本文利用概率无向图模型在AP算法的偏向参数中注入数据样本的依赖先验知识,对各样本成为聚类中心的概率作预先评估,有效地避免了算法对聚类中心的误判,提高了聚类精度,同时也减少了算法的迭代次数,提高了计算效率;用概率无向图模型计算偏向参数,有效地避免了传统AP算法需要手工调参的弊端,提高了算法的自适应性. 故引入概率无向图模型改进AP算法是可行的.

图3是本文算法计算复杂度特性曲线图. 将5个测试数据集分别以2、4、8、16倍数复制扩展,生成更大规模的数据集,然后采用本文算法聚类这些新数据集,绘制算法的计算时间随数据集规模变化而变化的趋势图. 由图3可看出,当5个数据集的规模呈指数级增长时,本文算法的计算时间呈多项式曲线的变化趋势,所以本文算法具有多项式时间的计算复杂度.

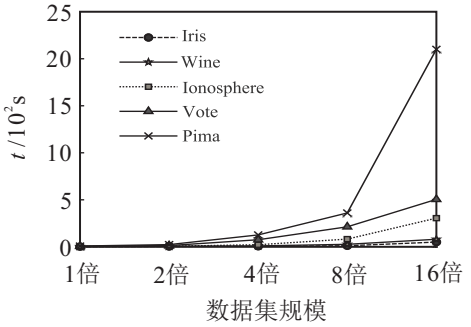


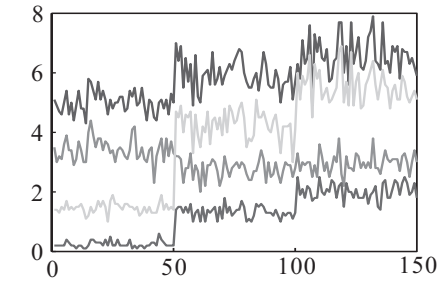
图 3　计算复杂度特征曲线

表5是簇归并前后簇数目的对比,归并前指算法Step 5产生的初始簇. 由表5可知,初始簇的数目可能会多于真实簇的数目,例如Vote、Pima、Yeast、Soybean数据集,但归并后,簇的数目与真实簇的数目完全相同,说明本文的簇归并操作是有效的.

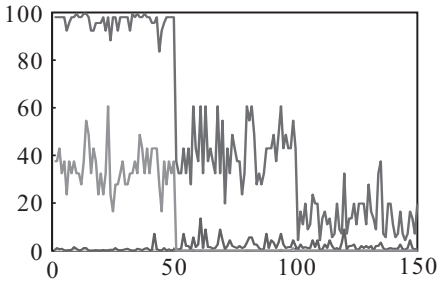
表 5　归并前后簇数目的对比

数据集	归并前	归并后	真实簇数目
Iris	3	3	3
Wine	4	3	3
Ionosphere	2	2	2
vote	4	2	2
Pima	4	2	2
Yeast	12	10	10
Soybean	19	15	15

图4通过Iris数据集描述高斯平滑算子的降噪功能. 图4(a)是原始的Iris数据集属性特性,4根曲线代表Iris数据集的数据分布特性呈现出4个类别的特性. 图4(b)是对Iris数据集进行高斯降噪后,数据分布特性呈现出3个类别的特性,与该数据集的实际类别数接近.



(a)　原始Iris数据集的属性特性



(b)　高斯平滑后的Iris数据集属性特性

图 4　高斯平滑算子的降噪效果

表6是研究高斯平滑算子是否能有效处理野值数据, 在表6中:以Wine和Vote数据集为例,分别在数据集中随机选取1%、2%、5%、10%、15%、20%的样本改造成野值样本;被选中样本,随机取其一或两个属性,并将被选中的属性取值改为−1,改造成野值样本. 由表6可知:当数据集中的野值样本率在5%以内时,算法的ACC精度指标下降百分率小于5%;但当野值率在10%以上时,算法的ACC精度指标下降较为明显. 这说明:高斯平滑算子具有一定的抗野值数据能力,但效果欠佳,需要引入其他方法提高AP算法的抗野值数据能力.

表 6　本文算法抗野值数据的效果

野值率/%	数据集			
	Wine		Vote	
	ACC	ACC变化百分率/%	ACC	ACC变化百分率/%
1	0.726	−0.134	0.857	−2.614
2	0.742	−0.669	0.86	−2.273
5	0.73	−2.276	0.853	−3.068
10	0.629	−15.797	0.85	−3.409
15	0.601	−19.545	0.715	−18.75
20	0.623	−16.6	0.702	−20.23

4　结　论

本文提出了一种概率无向图模型的AP聚类算法,用以解决传统AP算法需要手工筛选偏向参数的问题,在UCI数据集上与传统AP算法、K-AP算法进行聚类效果对比,验证了所提出算法的可行性,得到以下结论:

1) 采用概率无向图模型计算偏向参数,能在偏向参数中注入数据样本,依赖先验知识加快聚类中心的出现时机,减少迭代次数,提高计算效率.

2) 用概率无向图模型计算偏向参数,克服了传统AP算法需要手工筛选最优偏向参数系数的问题,算法对不同复杂度的数据集有较强的自适应性,扩展了算法的适应范围.

3) 用高斯算子作数据预处理,对算法产生的初始簇进行归并操作,能进一步提高算法的聚类精度.

参考文献(References)

[1] Frey B J, Dueck D. Clustering by passing messages between data points[J]. Science, 2007, 315(5814): 972-976.

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

[3] Kang J H, Lerman K, Plangprasopchok A. Analyzing microblogs with affinity propagation[C]. Proc of the First Workshop on Social Media Analytics. Washington DC, 2010: 67-70.

[4] Yang C, Bruzzone L, Sun F, et al. A fuzzy-statistics-based affinity propagation technique for clustering in multispectral images[J]. IEEE Trans on Geoscience & Remote Sensing, 2010, 48(6): 2647-2659.

[5] 章涛, 吴仁彪. 近邻传播观测聚类的多扩展目标跟踪算法[J]. 控制与决策, 2015, 31(4): 764-768.
(Zhang T, Wu R B. Multiple extended target tracking using AP clustering[J]. Control and Decision, 2015, 31(4): 764-768.)

[6] Vlasblom J, Wodak S J. Markov clustering versus affinity propagation for the partitioning of protein interaction graphs[J]. Bmc Bioinformatics, 2009, 10(4): 1-14.

[7] 王开军, 张军英, 李丹, 等. 自适应仿射传播聚类[J]. 自动化学报, 2007, 33(12): 1242-1246.
(Wang K J, Zhang J Y, Li D, et al. Adaptive affinity propagation clustering[J]. Acta Automatica Sinica, 2007, 33(12): 1242-1246.)

[8] 肖宇, 于剑. 基于近邻传播算法的半监督聚类[J]. 软件学报, 2008, 19(11): 2803-2813.
(Xiao Y, Yu J. Semi-supervised clustering based on affinity propagation algorithm[J]. J of Software, 2008, 19(11): 2803-2813.)

[9] 甘月松, 陈秀宏, 陈晓晖. 一种AP算法的改进: M-AP聚类算法[J]. 计算机科学, 2015, 42(1): 232-235.
(Gan Y S, Chen X H, Chen X H. Improved AP Algorithm: M-AP Clustering Algorithm[J]. Comoputer Science, 2015, 42(1): 232-235.)

[10] Zhang X, Wang W, Norvag K, et al. K-AP: Generating specified K clusters by efficient affinity propagation[C]. Int Conf on Data Mining. Sydney, 2010: 1187-1192.

[11] Wang Y, Chen L. K-MEAP: Generating specified K clusters with multiple exemplars by efficient affinity propagation[C]. IEEE Int Conf on Data Mining. Shenzhen, 2014: 1091-1096.

[12] Zhang J, He M, Dai Y. Modified affinity propagation clustering[C]. 2014 IEEE China Summit Int Conf on Signal and Information Processing. Xi'an, 2014.

[13] 付迎丁, 兰巨龙. 基于核偏向式的近邻传播聚类算法[J]. 计算机应用研究, 2012, 29(5): 1644-1647.
(Fu Y D, Lan J L. Kernel-based adaptation for affinity propagation clustering algorithm[J]. Application Research of Computers, 2012, 29(5): 1644-1647.)

[14] Zhao A, Huang Z, Qiu Y. Service semantic link network discovery based on Markov structure[C]. The 6th Int Conf on Semantics Knowledge and Grid (SKG). Beijing: IEEE, 2010: 9-16.

[15] Koller D, Friedman N. Probabilistic graphical models: Principles and techniques-adaptive computation and machine learning[J]. 2009, 42(2): 161-168.

[16] Pearl J. Probabilistic reasoning in intelligent systems: Networks of plausible inference[J]. Computer Science Artificial Intelligence, 1988, 70(2): 1022-1027.

[17] Besag J. Spatial interaction and the statistical analysis of lattice systems[J]. J of the Royal Statistical Society, 1974, 36(2): 192-236.

[18] Chandgotia N. Generalisation of the hammersley-clifford theorem on bipartite graphs[EB/OL]. (2014-06-07). https://arxiv.org/abs/1406.1849.

[19] 张敏, 覃华, 苏一丹. 半定规划支持向量机模型的研究[J]. 计算机工程与设计, 2011, 32(5): 1785-1788.
(Zhang M, Qin H, Su Y D. Research of semi-definite programming SVM model[J]. Computer Engineering and Design, 2011, 32(5): 1785-1788.)

[20] Ito K, Xiong K. Gaussian filters for nonlinear filtering problems[J]. IEEE Trans on Automatic Control, 2000, 45(5): 910-927.

[21] Shui P L, Zhang W C. Noise-robust edge detector combining isotropic and anisotropic Gaussian kernels[J]. Pattern Recognition, 2012, 45(2): 806-820.

[22] 刘江, 王玉金, 段建雷, 等. 基于高斯分布的多层无迹卡尔曼滤波算法[J]. 控制与决策, 2016, 31(4): 609-615.
(Liu J, Wang Y J, Duang J L, et al. Multi-layer unscented Kalman filtering algorithm based on Gaussian distribution[J]. Control and Decision, 2016, 31(4): 609-615.)

[23] Blake C L, Merz C J. UCI repository of machine learning database[EB/OL]. [2016-12-28]. http://archive.ics.uci.edu/ml/index.html.

[24] Schutze H. Introduction toInformation retrieval[M]. New York: Cambridge University Press, 2008: 102-115.

(责任编辑: 孙艺红)