基于粒子群的粗糙核聚类算法*

姚丽娟，罗 可
（长沙理工大学 计算机与通信工程学院，长沙 410114）

摘 要：针对 K-means 聚类算法容易陷入局部最优、不能处理边界对象及线性不可分的缺点，提出一种基于粒子群的粗糙核聚类算法。该算法通过 Mercer 核将输入样本空间中的样本映射到高维空间，使样本变得线性可分，并结合粗糙集的思想，通过动态改变上下近似集的权重因子对边界对象进行有效处理，同时采用 reliefF 方法对样本属性进行加权处理，以解决混合数据的聚类问题，最后利用粒子群算法防止算法陷入局部最优。仿真实验表明，相对于其他改进算法，该算法具有较高的正确率和较短的收敛时间，并进一步验证了该算法的鲁棒性和稳定性，具有一定的实用价值。

关键词：聚类；核函数；粗糙集；粒子群算法；属性加权
中图分类号：TP391.4 **文献标志码**：A **文章编号**：1001-3695(2012)08-2854-04
doi:10.3969/j.issn.1001-3695.2012.08.014

Rough kernel clustering algorithm based on particle swarm optimization

YAO Li-juan, LUO Ke
(*School of Computer & Communication Engineering, Changsha University of Science & Technology, Changsha* 410014, *China*)

Abstract: According to the disadvantages of the K-means clustering algorithm such as easing to fall into local optimum, can not handle data of boundary objects and non-linear, this paper proposed a rough kernel clustering algorithm based on particle swarm. The sample in the input space was mapped to high-dimensional space by Mercer kernel, so that the sample characteristics which were not shown in the sample space would be appear in the high-dimensional space, and combined with the idea of rough set, by changing the weighting factors of upper and lower approximation dynamically, to cope with the boundary objects efficiently. And reliefF method weighted samples' properties to solve the problem of mixed data clustering. Finally, used an improved particle swarm optimization algorithm to prevent the algorithm into a local optimum. Simulation results show that the algorithm has higher accuracy and shorter convergence time compared with the others improved algorithms, and is verified robustness and stability furtherly, and has some practical value.

Key words: clustering; kernel function; rough set; PSO; property weighted

聚类分析现已成为数据挖掘研究领域中一个非常活跃的研究课题[1]。现有的 K-means 聚类算法具有简单、抗噪能力和局部搜索能力强的优点；但是 K-means 算法是基于硬计算的划分原理，所以对边界对象不能有效处理，对线性不可分的数据聚类效果比较差，且具有易陷入局部最优的缺点[2]。

为了解决这些问题，目前有很多学者将 Mercer 核引入到聚类的方法中[3~6]。例如文献[7]采用核函数对样本进行预处理，增大了样本差异，提高了聚类精度，但是对边界对象不能进行有效的聚类，而通过引入粗糙集能有效处理边界对象[8,9]。文献[9]提出了一种粗糙 K-means 的聚类方法，通过上下近似集中样本的比例共同决定新的聚类中心来提高聚类效果，但是对于具有混合属性的数据聚类效果较差。针对这一缺点，文献[10]提出了一种基于特征加权的模糊聚类方法，它采用 reliefF 方法对样本属性进行加权处理，平衡样本属性对聚类的贡献程度，提高了聚类的精度，但是没有对数据进行预处理，对于线性不可分的数据聚类效果较差，且容易陷入局部最优。而由文献[11]可知粒子群算法对提高收敛速度和全局最优具有显著的效果。

目前大多数算法都仅仅从单一方面来考虑怎样提高聚类的质量，算法的综合性能较差。针对这些算法的不足，从综合方面考虑，本文提出一种基于粒子群的粗糙核聚类算法。该算法采用高斯核函数将样本映射到特征空间，放大样本差异；结合粗糙集来有效处理边界对象，并动态地改变上下近似集权重 ω_l 和 ω_{bnr} 在每一次迭代中的比重；并采用 reliefF 方法对样本属性进行加权，最后结合改进的粒子群算法，以聚类中心作为问题的解来进行优化，将聚类问题从易陷入局部最优的问题转变为连续优化的问题。

1 相关知识介绍

1.1 核函数

核是一个函数 $K(x,y)$，它描述了在某个特征空间 H 中的一个内积，对所有 X 中的 x、y，满足 $K(x,y)=\varphi(x)\varphi(y)$，其中 φ 是从 X 到特征空间 H 的映射[12]，如图 1 所示。

定义 1 $K(x,y)$ 表示一个连续的对称核，$x,y\in X$，核 $K(x,y)$ 可以展开为

收稿日期：2011-12-31；**修回日期**：2012-02-20 **基金项目**：国家自然科学基金资助项目(10926189,10871031)；湖南省自然科学衡阳联合基金资助项目(10JJ8008)；湖南省教育厅重点资助项目(10A015)

作者简介：姚丽娟(1986-)，女，硕士，主要研究方向为数据挖掘、计算机应用(yaolijuan1986@163.com)；罗可(1961-)，男，教授，博士，主要研究方向为数据挖掘、计算机应用等.

$$K(x,y)=\sum_{i=1}^{\infty}\lambda_i\varphi_i(x)\varphi_i(y)\quad \lambda_i>0 \tag{1}$$

式(1)绝对一致收敛的充要条件是$\int_a^b\int_a^b K(x,y)g(x)g(y)\,dxdy\geqslant 0$,对于所有满足条件的$g(x)$($\int g(x)\,dx<\infty$,$g(x)\neq 0$)都成立。其中$\varphi_i(x)$称为展开的特征函数,$\lambda_i$称为特征值。

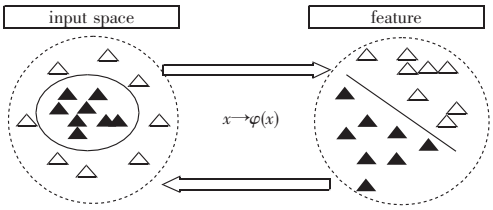

图 1　核函数示意

任何一个函数只要满足 Mercer 定理的条件就可以作为一个 Mercer 核。对输入空间的样本集$x^{(q)}\in R^N(q=1,2,\cdots,Q)$。用非线性函数$\varphi$把所有样本映射到高维空间$H$中,可以得到新的样本集$\varphi_1(x),\varphi_2(x),\cdots,\varphi_N(x)$,则由定义 1 可得

$$\begin{aligned}\|\varphi(x_i)-\varphi(x_j)\|^2&=(\varphi(x_i)-\varphi(x_j))^{\mathrm{T}}(\varphi(x_i)-\varphi(x_j))=\\&\varphi(x_i)^{\mathrm{T}}\varphi(x_i)-\varphi(x_j)^{\mathrm{T}}\varphi(x_i)-\varphi(x_i)^{\mathrm{T}}\varphi(x_j)+\varphi(x_j)^{\mathrm{T}}\varphi(x_j)=\\&K(x_i,x_i)+K(x_j,x_j)-2K(x_i,x_j)\end{aligned} \tag{2}$$

特征空间中的点积可用输入空间的核来表示,则有

$$d(x,y)=\|\varphi(x)-\varphi(y)\| \tag{3}$$

$d(x,y)$是特征空间中的欧式距离,核代入技巧使得在原输入空间中诱导出一种依赖于核的新的距离度量。引入核函数之后,特征空间的内积运算转换为样本空间中核函数的计算,而不用知道$\varphi(x)$的具体形式。

1.2　粗糙集理论

粗糙集理论[13]理论是波兰数学家 Pawlak 在 1982 年提出的一种分析理论,常用于处理模糊和不精确的问题。下面给出粗糙集中与本文相关的一些基本定义。

定义 2　知识库。知识库$K=(U,R)$,R是非空有限集合论域U上的等价关系族集。

定义 3　不可区分关系。R的非空子集P上的不可区分关系为$\mathrm{ind}(P)$。称$U/\mathrm{ind}(P)$为$K=(U,R)$关于论域U的P基本知识。称$[x]_{\mathrm{ind}(P)}$为P的基本概念。

定义 4　上近似、下近似及边界域。给定知识库$K=(U,R)$,对于每个子集$X\subseteq U$和一个等价关系$R\in\mathrm{ind}(K)$。称$\underline{RX}=\cup\{Y\in U/R\mid Y\subseteq X\}$为$X$关于$R$的下近似(lower approximation)。称$\overline{RX}=\cup\{Y\in U/R\mid Y\cap X\neq\varnothing\}$为$X$关于$R$的上近似(upper approximation),而$BNR(X)=\overline{RX}-\underline{RX}$则称为$X$的$R$边界区域(boundary)。

定义 5　粗糙集。若$\underline{RX}\neq\overline{RX}$则$X$为$R$的粗糙集,否则称$X$为$R$精确集。$pos_R(X)=\underline{RX}$称为$X$的$R$正域,即由确定属于$X$的$U$中元素组成的集合;$\overline{RX}$是由可能属于$X$的$U$中元素组成的集合;$BNR(X)$称为$R$的边界域,即既不能判断肯定属于$X$,又不能判断肯定不属于$\sim X(=u-x)$的$U$中的元素组成的集合。

定义 6　近似精度。集合范畴的不确定性是由于边界域的存在而引起的。集合的边界域越大,精确性则越低。则引入精度$a_R(X)=|\underline{RX}|(|\overline{RX}|)^{-1}$,其中:$X\neq\varnothing$;$|X|$表示集合$X$的基数;精度$a_R(X)$反映集合$X$的知识的完整程度。

1.3　粒子群算法

PSO 算法[14]是 1995 年由美国社会心理学家 Kennedy 和 Eberhart 提出的一种进化计算方法。PSO 初始化为一群随机粒子,每个粒子通过跟踪两个极值来更新自己:个体极值P_{id}和全局极值P_g。在找到上述两个最优值时,粒子根据式(4)(5)更新自己的速度和新的位置:

$$\begin{aligned}v(t)=&v(t-1)+c_1r_1(p_{id}(t)-x(t-1))+\\&c_2r_2(p_g(t)-x(t-1))\end{aligned} \tag{4}$$

$$x(t)=x(t-1)+v(t) \tag{5}$$

其中:c_1和c_2为加速常数,通常c_1和c_2在[0,2]之间,一般取$c_1=c_2=1$;r_1和r_2为两个在[0,1]之间的随机数。

当前改进的 PSO 算法大多加入了惯性因子$\omega(0<\omega<1)$。当ω取值较大时,算法具有较好的全局收敛能力;当ω取值较小时,算法则具有较强的局部搜索能力。为了保证 PSO 算法能达到有效的收敛效果,惯性因子ω必须保持一定的关系,但是惯性因子不能完全地线性减小,因此采用如下的改进[15,16]方法:

a)$\omega\in[a,b]$,一般取[0.4,1];

b)$F_{gd}(t)$表示第t代所有粒子的全局最优适应度;

c)$F_{id}(t)$表示第t代第i个粒子局部最优状态。

$$F_i(t)=k\times F_{gd}(t)/F_{id}(t) \tag{6}$$

$$\omega(t)=b-(b-a)\times e^{(-F_i(t)/t)} \tag{7}$$

其中:$k>0$,t为当前迭代次数。

经过分析,速度和位置的公式可以改为

$$\begin{aligned}v(t)=&\omega(t)v(t-1)+c_1r_1(p_{id}(t)-x(t-1))+\\&c_2r_2(p_g(t)-x(t-1))\end{aligned} \tag{8}$$

$$x(t)=x(t-1)+v(t) \tag{9}$$

2　基于粒子群的粗糙核聚类算法

2.1　粗糙核聚类算法

粗糙核聚类(rough kernel-Kmeans)算法的思想是:首先利用一个非线性映射将样本映射到一个高维的特征空间H中,使样本差异加大,那么样本就变得线性可分(或近似可分);然后在高维空间中进行粗糙 K-means 聚类。

虽然样本空间通过 Mercer 核函数非线性映射到高维特征空间,放大了样本间的特征差异,改善了聚类效果,但算法仍采用基于核空间的欧式距离。欧式距离假设样本的每个属性对最终聚类的贡献程度一样,但在实际情况中,某些属性在聚类过程中发挥的作用巨大,而某些属性的作用却很小甚至可以忽略。因此,为了使作用强的属性的利用程度也最大,于是采用 reliefF 方法对样本属性进行加权处理,则粗糙核聚类的目标函数为

$$\begin{aligned}J_w=\sum_{i=1}^{k}\Big(&(\omega_l{}^{*}\sum_{\varphi(x_j)\in C_l}\sum_{l=1}^{d}w_l\|\varphi(x_{jl})-\varphi(v_{il})\|^2+\\&\omega_{bnr}{}^{*}\sum_{\varphi(x_j)\in C_{bnr}}\sum_{l=1}^{d}w_l\|\varphi(x_{jl})-\varphi(v_{il})\|^2\Big)\end{aligned} \tag{10}$$

其中:$\varphi(v_{il},v_{i2},v_{i3},\cdots,v_{id})$为第$c_i$类的聚类中心;$w_l$为特征加权系数,且$\sum_{l=1}^{d}w_l=1$;$\omega_l$、$\omega_{bnr}$分别为第$c_i$类的上下近似集的权重,前一项为下近似集中的对象到所属簇中心距离的加权和,后一项为边界集中的对象到所属簇中心距离的加权和。

近似集范围判定方法:$\forall x_m\in U$,有$d(x_m,c_i)=\min\{d(x_m,c_j),j=1,2,\cdots,k\}$;若$\exists c_j$,使得$d(x_m,c_i)-d(x_m,c_j)<\gamma\times d(c_i,c_j)$,则令$x_m\in\overline{C}_j$,否则令$x_m\in\underline{C}_i$,其中$\gamma=0.07$。

算法的具体步骤为:

a)给出n个聚类样本$X=\{x_1,x_2,\cdots,x_n\}$,$x_i\in R^d$,经过核

函数映射到 Hilbert 空间的样本为 $\{\varphi(x_1),\varphi(x_2),\cdots,\varphi(x_n)\}\subset F$。

b)确定初始聚类中心 $\varphi(v_1,v_2,v_3,\cdots,v_k)$,其中 k 是聚类数目。

c)在核空间中,将每个样本 $\varphi(x)$ 根据近邻原则分配给最近的各类上下近似集。

d)根据式(10)计算 J_w 值,如果 $|J_w(t)-J_w(t-1)|\leqslant\varepsilon$ 或者 $t\geqslant t_{\max}$,则结束;否则,根据式(11)更新聚类中心,继续转步骤 c)。

在核空间中已经把所有的样本分配到每个聚类中心的上下近似集中,对应样本空间中反映的是不同类别样本对聚类的贡献,因此采用 lingras 算法[17] 进行聚类中心的更新,如式(11)所示:

$$v_k=\begin{cases}w_l\sum\limits_{\varphi(x_n)\in|\underline{c_k}|}\dfrac{\varphi(x_n)}{|\underline{c_k}|}+w_{bnr}\sum\limits_{\varphi(x_n)\in|c_kB|}\dfrac{\varphi(x_n)}{|\overline{c_k}|} & \text{for } \underline{c_k}=\varnothing\\ w_l\sum\limits_{\varphi(x_n)\in\overline{|c_k|}}\dfrac{x_n}{|\overline{c_k}|} & \text{otherwise}\end{cases}\tag{11}$$

e)否则 $t=t+1$,继续转步骤 c)。

2.2 基于粒子群的粗糙核聚类算法

2.2.1 本文算法的编码方案

PSO 算法采用的是实数编码,它不必像遗传算法那样采用二进制编码,一个编码对应一个可行解,这种方法有利于改善粒子群算法的计算复杂性,提高运算效率。在本文中,对数据采用基于聚类中心的编码格式,即每个粒子的位置由 k 个聚类中心组成。由于样本的维数是 d,所以粒子的位置是 $k\times d$ 维变量,速度也是 $k\times d$ 维变量。粒子编码结果如下:

位置:$v_{11},v_{12},\cdots,v_{1d},v_{21},v_{22},\cdots,v_{2d},\cdots,v_{k1},v_{k2},\cdots,v_{kd}$

速度:$V_{11}V_{12}\cdots V_{1d}\cdots V_{k1}V_{k2}\cdots V_{kd}$

2.2.2 粒子适应度设置

因为每个粒子是对整个数据集的一种划分,且每个粒子对应 k 个聚类中心,为了对聚类划分有一个整体的认识以及对聚类结果有一个正确的判断,结合粗糙核聚类算法,本文采用式(12)作为算法的适应度函数:

$$F=RJ_w+\frac{1}{k}\sum_{i,j=1}^{k}\|\varphi(v_i)-\varphi(v_j)\|^2\tag{12}$$

其中:

$$RJ_w=1/J_w\tag{13}$$

J_w 是基于粗糙核聚类的目标函数,用来评价聚类的内聚程度;而 $\frac{1}{k}\sum_{i,j=1}^{k}\|\varphi(v_i)-\varphi(v_j)\|^2$ 用来评价类间的分离程度。进行聚类的目的是要使类中散布尽可能小,同时类间分离尽可能大。当 k 增加时,类内的内聚程度单调下降,而类间的分离程度呈增加趋势。为了限制类间分离程度的单调增加,为类间分离程度设置 $1/k$ 的权重,使类间距离和内类距离达到一个平衡的状态,进而使得聚类更加有效。

2.2.3 ω_l、ω_{bnr}的动态调整

通过多次实验发现,随着迭代次数的不断增加,如果使得 ω_l 的比重不断增加,相应地 ω_{bnr}的比重会不断减小,算法的效率会有所增加,而且在前几次粗糙聚类时,上近似集(边界集)中的数据量较大。因此在利用式(12)计算粒子适应度时,给予边界集权重因子 ω_{bnr}一个相对较大的数;同样地,在算法运行到后期时,大多数样本已经被归结为每类的下近似集,这时把 ω_l 的值相对提高,自然会使算法的效率有所提高。于是,本文将动态调整 ω_l、ω_{bnr}的值为

$$\omega_l(\text{iter})=0.4+0.4\times\frac{\text{iter}}{\text{iterMax}}\tag{14}$$

$$\omega_{\text{bnr}}(\text{iter})=0.6-0.4\times\frac{\text{iter}}{\text{iterMax}}\tag{15}$$

3 本文算法的步骤及应用

本文改进的粗糙核聚类算法虽然使样本空间从线性不可分变为线性可分,某种程度上方便了聚类,但本质上还是一种基于梯度下降的局部搜索算法,因此不可避免地会陷入局部最优,而且它对初值较敏感。而粒子群算法是通过模拟自然界粒子运动且具有全局搜索能力的算法,因此,它可以用于解决聚类寻优的问题。

本文提出的基于优化粒子群的粗糙核聚类算法思想是:将优化粒子群算法与改进的粗糙核聚类算法相结合,通过动态调整 ω_{bnr}和 ω_l 的值,以权衡其在不同代中的比重,提高聚类效果。

为了避免初始聚类中心分布集中,在读入初始数据后,经过高斯核处理,并记录每维上的最大值 $x_{\max i}$和最小值 $x_{\min i}$,粒子 i 的位置可表述为 $Z_i(v_{11},v_{12},\cdots,v_{1d},v_{21},v_{22},\cdots,v_{2d},\cdots,v_{k1},v_{k2},\cdots,v_{kd})$,其中 $\varphi(v_{jl})\in[x_{\min l},x_{\max l}]$ $(j=1,2,\cdots,k;l=1,2,\cdots,d)$,故在 $x_{\max l}-x_{\min l}$中随机选取一个值作为 $\varphi(v_{jl})$的初值。重复该操作 N 次,就形成随机性较大的初始粒子群 $A_0(Z_1,Z_2,\cdots,Z_N)$。

算法的具体步骤为:

a)给出 n 个聚类样本 $X=\{x_1,x_2,\cdots,x_n\}$,$x_i\in R^d$,记录在高斯空间下每一维上数据的最大值 $x_{\max i}$和最小值 $x_{\min i}$。

b)粒子群的初始化。设一初始聚类数 $k(2\leqslant k\leqslant n)$,将每个样本随机指派给某一类作为初始的划分矩阵。再从每一类中随机取出一个样本作为该类的初始聚类中心,各个聚类中心按照编码规则生成粒子的初始位置,同时将该初始位置作为粒子的个体最优位置 P_{id}。随机生成粒子的初始速度,根据式(12)计算粒子的适应值 F。反复执行 N 次,生成 N 个初始粒子 $Z_1,Z_2,\cdots,Z_N$,组成初始种群 A_o,计算全局最优位置 P_g,并初始化 w_l 系数。设置当前迭代次数 $t=0$。

c)根据式(12)计算种群 A_t 中每个粒子的适应值 $F(Z_i)$ $(i=1,2,\cdots,N)$,根据 $F(Z_i)$值更新 $P_{id}(t)$和 $P_g(t)$,判断 $t>T_m$ 或者 $|F(t)-F(t-1)|\leqslant\varepsilon$ 是否成立。若不成立,则继续计算;否则,输出全局最优粒子 $P_g(t)$及其聚类结果。

d)用式(8)(9)更新粒子的位置和速度,形成粒子群 Q_t,并对 Q_t 中粒子 Z_i 的 $\varphi(v_{jl})$进行如下处理:

$$\varphi(v_{jl})=\begin{cases}\varphi(v_{jl}) & x_{\min l}<\varphi(v_{jl})<x_{\max l}\\ x_{\max l} & \varphi(v_{jl})>x_{\max l}\\ x_{\min l} & \varphi(v_{jl})<x_{\min l}\end{cases}\tag{16}$$

得到粒子群 R_t。

e)在 R_t 中利用粒子的位置编码对数据集进行粗糙核聚类处理,根据得到的聚类中心按照本文提出的上下集划分的方法将所有样本分配到每一类的上下近似集中并更新聚类中心,取代原来粒子的编码值,从而得到粒子群 S_t。

f)计算 S_t 中每个粒子的适应度值 $F(Z_i)$ $(i=1,2,\cdots,N)$,其中惯性因子 ω 根据式(7)进行取值,得到新一代粒子群 A_{t+1}。

g)将得到的每个粒子的最新位置作为新的聚类中心,并根据式(14)(15)调整 ω_l、ω_{bnr}的值,同时采用 reliefF 算法加权优化 w_l 矩阵。

h)$t = t + 1$,并转至步骤 c)。

4　算法仿真与性能分析

仿真环境:软件为操作系统 Windows 7,编译软件 MATLAB 7.0.1;硬件为 Pentium® Dual-Core CPU T4200 @ 2.00 GHz,内存 2 GB。

参数分析:本文采用高斯核函数[18]:

$$k(x,y) = \exp(-\beta \| x - y \|^2 / 2\sigma^2) \tag{17}$$

其中:参数 σ^2 设为 0.4,参数 β 为 1。设置粒子总群数量为 30,最大迭代次数为 30。加权系数矩阵由 reliefF 算法通过样本动态变化生成并归一化处理。

为了测试本文算法的性能,分别采用 Rough K-means、Kernel-Kmeans、Rough Kernel-Kmeans 及本文算法对人工数据和标准数据集进行聚类分析,并比较各算法的正确率和收敛时间,如表 1 ~3 所示。

表 1　各算法在人工数据集的聚类结果比较

比较项	算法			
	Rough K-means	Kernel-K-means	Rough Kernel-K-means	本文算法
类别数	2	2	2	2
实验次数	30	30	30	30
最好正确率/%	80.5	95.5	99.5	100
最差解正确率/%	58.5	88.9	94.0	98.5
平均解正确率/%	69.4	90.6	96.2	99.6
算法收敛时间/s	12.5	10.3	7.9	3.5

表 2　各算法在 Iris 数据集的聚类结果比较

比较项	算法			
	Rough K-means	Kernel-K-means	Rough Kernel-K-means	本文算法
类别数	3	3	3	3
实验次数	30	30	30	30
最好正确率/%	83.5	94.8	95.8	100
最差解正确率/%	63.5	84.9	87.6	97.5
平均解正确率/%	72.7	88.6	98.4	99.2
算法收敛时间/s	11.5	9.6	7.5	2.36

表 3　各算法在 Wine 数据集的聚类结果比较

比较项	算法			
	Rough K-means	Kernel-K-means	Rough Kernel-K-means	本文算法
类别数	3	3	3	3
实验次数	30	30	30	30
最好正确率/%	72.4	76.8	82.5	87.5
最差解正确率/%	54.5	65.9	70.6	83.5
平均解正确率/%	60.7	69.6	76. 4	85.7
算法收敛时间/s	20.5	15.6	12.5	6.25

a)人工数据。样本数据如图 2 所示,共有 220 个样本点,将其分成两类,一类 120 个,另一类 100 个,其分布整体上呈两个嵌套的圆环。如果利用非核方法来聚类,很难把这两个圆环中的样本区分开来,本文采用基于优化粒子群的粗糙核进行聚类,其聚类结果如图 3 所示。

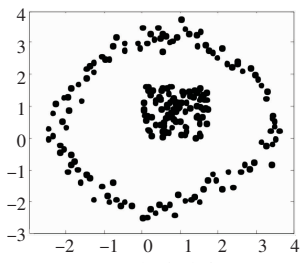

图 2　样本分布

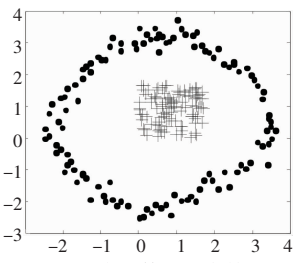

图 3　本文算法聚类结果

从图 3 可以得出,样本数据被聚为两类,分别用不同标志表示。说明采用核空间能对非凸形状的数据准确地进行聚类,且能较好地反映现实情况。

通过比较表 1 的各项数据可以得出,本文算法相比其他三种算法在聚类性能上有较大的改进。由于所选样本的特殊性,导致 Rough K-means 算法的聚类性能很差,最差解为 62%。对于线性不可分的数据,通过引入核函数不管从聚类正确率还是从收敛时间上都得到了提高,最差解都至少在 90% 以上。Kernel-Kmeans 和 Rough Kerel-Kmeand 算法容易受初值的影响,且采用传统的欧氏距离度量方法导致聚类正确率在 5% -10% 范围内浮动,而本文算法聚类结果比较稳定,平均正确率可以达到 99% 以上。且本文算法的收敛时间为 3.5 s,与其他三种算法相比至少提高了 2.5 倍。

b)标准数据集。iris 数据集共有三类,每类 50 个样本,每个样本具有 4 个属性;wine 数据集共有三类,每类样本个数分别为 59、71、48,每个样本具有 13 个属性。

从表 2、3 可知,本文算法在标准数据集中的聚类性能相比其他几种算法都有所提高。由于属性加权的设计能平衡样本中各个属性在聚类中的重要程度,而且能较好地融入到粗糙集、核空间及粒子群算法中,使聚类更加符合实际情况。与其他算法相比,本文算法不仅提高了聚类性能,也减少了收敛时间,算法的综合性能得到了提高。

5　结束语

本文算法是 Rough Kernel-K-means 算法的推广,在 Rough Kernel-K-means 算法的基础上增加了属性加权矩阵、动态上下集权重及优化粒子群算法,使得聚类的整体性能得到了很大程度上的提高。该算法通过核函数把这些样本映射到高维的 Hilbert 空间,在一定程度上放大了样本的差异,再加入粗糙集的思想,并利用粒子群算法进行寻优操作。实验结果也表明本文算法在性能上比目前大多数算法都有较大的改进,算法正确率和收敛速度都有明显的提高,尤其在其他算法失效的情况下,本文算法也能得到正确的聚类结果。但该算法在聚类过程中没有考虑怎样动态确定 k 值,这将作为下一步的研究方向。

参考文献:

[1] HAN Jia-wei, MICHELINE K, PEI Jian. Data mining: concepts and techniques[M]. 2nd ed. San Francisco: Morgan Kaufmann Publishers, 2005: 70-131.

[2] SARIEL H P, AKASH K. Smaller coresets for K-median and K-means clustering[C]//Proc of the 21st Annual Symposium on Computational Geometry. 2005: 3-19.

[3] CANASTRA F, VERRI A. A novel kernel method for clustering[J]. Pattern Analysis and Machine Intelligence, 2005, 27(5): 801-805.

[4] 于进,前锋. 基于粒子群优化的高斯核函数聚类算法[J]. 计算机工程,2010,36(14):22-24.

[5] CHIANG J H, HAO Pei-yi. A new kernel-based fuzzy clustering approach: support vector clustering with cell growing[J]. IEEE Trans on Fuzzy Systems, 2003, 11(4): 518-527.

[6] WU Wen-li, LIU Yu-shu, ZHAO Ji-hai. New mixed clustering algorithm[J]. Journal of System Simulation, 2007, 19(1): 16-19.

[7] 丁军娣,马儒宁,陈松灿. 基于多项式核的结构化有向树数据聚类算法[J]. 软件学报,2008,19(12):3147-3160.

[8] GEORG P. Some refinements of rough K-means clustering[J]. Patterm Recognition, 2006, 39(8): 1481-1491.　(下转第 2902 页)

(上接第 2857 页)

[9] LIN Liang-dong, QU Wei, YU Xiang. A semi-supervised clustering algorithm based on rough reduction[C]//Proc of the 21st Annual International Conference on Chinese Control and Decision Conference. 2009:5463-5467.

[10] 李洁,高新波,焦李成. 基于特征加权的模糊聚类新算法[J]. 电子学报,2006,34(1):89-92.

[11] KENNEDY J, EBERHART R. Particle swarm optimization [C]//Proc of the 4th IEEE International Conference on Neural Networks. Piscataway: IEEE Press, 1995:1942-1948.

[12] CONG Rong, WANG Xiu-kun, LI Jin-jun, *et al*. Plot-track association algorithm based on hierachical and density clustering analysis[J]. Journal of System Simulation, 2005, 17(4):841-843.

[13] MASAHIRO I, LI Bing-jun. Rough set approach to information tables with imprecise decisions[C]//Proc of the 6th International Conference on Rough Sets and Current Trends in Computing. Berlin: Springer-Verlag, 2008:121-130.

[14] SURESH C H, KUMAR E V, REDDYY P P, *et al*. PSO clustering with preprocessing of data using artificial immune system[J]. Computer and Information Science, 2011, 4(1):163-164.

[15] 廖子贞,罗可,周飞红,等. 一种自适应惯性权重的并行粒子群聚类算法[J]. 计算机工程与应用,2007,43(28):166-168.

[16] 雷秀娟,付阿利,孙晶晶. 改进 PSO 算法的性能分析与研究[J]. 计算机应用研究,2010,27(2):453-458.

[17] LINGRAS P, WEST J. Interval set clustering of Web user with rough K-means[J]. Joumal of Intelligent Information Systems, 2004, 23(1):5-16.

[18] 皱立颖,郝冰,沙丽娟. 基于快速高斯核函数模糊聚类算法的图像分割[J]. 化工自动化及仪表,2010,37(11):81-84.