ISSN 1000-9825, CODEN RUXUEW
Journal of Software, Vol.19, No.12, December 2008, pp.3147−3160
DOI: 10.3724/SP.J.1001.2008.03147

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

基于多项式核的结构化有向树数据聚类算法*

丁军娣[1,2], 马儒宁[3], 陈松灿[2+]

[1](南京理工大学 计算机科学与技术学院,江苏 南京 210094)

[2](南京航空航天大学 信息科学与技术学院,江苏 南京 210016)

[3](南京航空航天大学 理学院,江苏 南京 210016)

Polynomial Kernel Based Structural Clustering Algorithm by Building Directed Trees

DING Jun-Di[1,2], MA Ru-Ning[3], CHEN Song-Can[2+]

[1](School of Computer Science and Technology, Nanjing University of Computer Science and Technology, Nanjing 210094, China)

[2](Department of Computer Science and Engineering, Nanjing University of Aeronautics and Astronautics, Nanjing 210016, China)

[3](Department of Science, Nanjing University of Aeronautics & Astronautics, Nanjing 210016, China)

+ Corresponding author: E-mail: s.chen@nuaa.edu.cn

Ding JD, Ma RN, Chen SC. Polynomial kernel based structural clustering algorithm by building directed trees. *Journal of Software*, 2008,19(12):3147−3160. http://www.jos.org.cn/1000-9825/19/3147.htm

Abstract: Within the internal organization of the data, the data points respectively play three different structural roles: the hub, centroid and outlier. The neighborhood-based density factor (NDF) used in the neighborhood based clustering (NBC) algorithm has the ability of identifying which points act as hubs, centriods or outliers in separated-well data set. However, NDF often works poorly in the circumstances of noise and overlapping. This paper introduces a polynomial kernel based neighborhood density factor (PKNDF) to address this issue. Relying on the PKNDF, a structural data clustering algorithm is further presented which can find all salient clusters with arbitrary shapes and unbalanced sizes in a noisy or overlapping data set. It builds clusters into the framework of directed trees in graph theory and thereby each point is scanned only once in the process of clustering. Hence, its computational complexity is nearly linear in the size of the input data. Experimental results on both synthetic and real-world datasets have demonstrated its effectiveness and efficiency.

Key words: data clustering; polynomial kernel; neighborhood-based density factor; directed tree; graph theory; overlapping data; structural role; structural clustering

摘 要: 各个点在数据内部的组织结构中自然地扮演着3种不同的结构性角色,分别是毂、质心和野值.在基于邻域的聚类算法中,邻域密度因子能够识别分离数据集中的毂、质心和野值.但是,邻域密度因子对有噪声和重叠的数据往往失效.为了解决该问题,引入了基于多项式核的邻域密度因子,并在有向树框架下,提出了一种结构化的数据聚类算法,其计算复杂度线性于输入数据的大小.对带有噪声和重叠的数据集,该算法能够找到所有显著的、任意形状的不均衡聚类.在人工和真实数据集上的实验结果都证实了该算法的有效性和快速性.

关键词: 数据聚类;多项式核;邻域密度因子;有向树;图论;重叠数据;结构性作用;结构化聚类

* Supported by the National Natural Science Foundation of China under Grant No.60632050 (国家自然科学基金)
Received 2007-10-20; Accepted 2008-03-28

中图法分类号: TP311 文献标识码: A

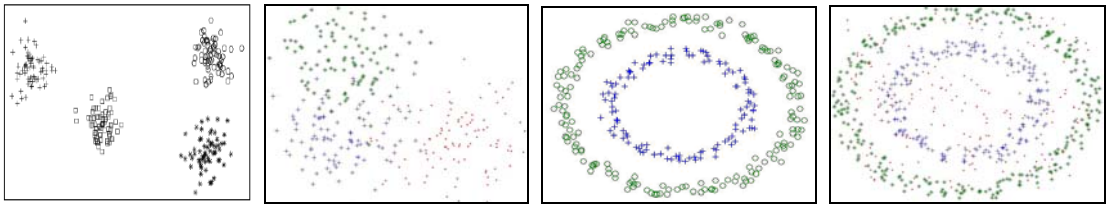
在机器学习、模式识别和计算机视觉等领域中,数据聚类研究一直是一个非常活跃且重要的课题[1−4],其目的是通过某种相似性度量,将数据集合划分为若干个有意义的子集从而发现隐藏的数据内部结构.一般而言,数据呈两种分布形式[5−7],即云状(clouds)分布和流形状(manifold)分布.现有的大多数聚类方法都是基于紧性准则,典型的如 *k*-均值[1]、模糊 *c*-均值[3]、谱(spectral)聚类[4]、逐对(pairwise)聚类[6]等.它们都能够有效地发现云状数据中的聚类,特别是分离性很好的云状数据,如图 1(a)所示[7].但是,它们对流形状的数据往往聚类失败[7−9],即使是分离性很好的简单流形数据,如图 1(c)所示.因为它们通常直接或间接地将两两数据间的欧氏(Euclidean)距离作为相似性度量,并且寻找相互距离最近的点使其成为同一类以达到类内相似性最大,或者类内相异性最小的目的.然而,对于流形状数据来说,类间距离往往小于类内距离,即类内相似性小于类间相似性,这恰恰与紧性准则相悖.因此,流形状数据的聚类问题近年来受到了极大的关注,并因此而涌现出了许多新颖的聚类算法,例如基于路径的逐对聚类方法[7,8],鲁棒的基于路径的谱聚类方法[9],DBSCAN[10]和基于邻域的聚类(NBC)算法[11].前两者分别是对原来只强调紧性的逐对聚类算法[6]和谱聚类算法[4]的改进.正如所期望的那样,它们能够处理一些不相交的流形状数据,如图 1(c)所示,但对如图 1(d)所示的噪声和野值仍然相当敏感.后两者,即 DBSCAN 和 NBC,都是基于密度的方法.除了利用原有数据间的几何距离之外,它们还利用了数据点的局部密度信息.它们独立于某个给定的相似性度量,并认为一个聚类是由一些局部稠密点组成的一个集合,因此,它们能够发现任意形状的不同密度的聚类,有着更广泛的适用性.相比而言,NBC 的聚类能力比 DBSCAN 更突出[11].因为在聚类过程中,通过邻域密度因子(NDF),NBC 隐含地抓住了数据点本身所体现出的一些结构性信息部分.

事实上,各个点在数据的内部组织结构中都自然地扮演着 3 种不同的结构性角色,分别是毂(hub)、质心和野值(outlier).毂一般连接两个或两个以上的聚类,质心位于一簇稠密点集的中心,而野值远离所有的聚类.当然,并非所有数据都同时包含 3 种不同的结构性数据点,例如干净(clean)且分离性好(separated-well)的数据集,如图 1(a)和图 1(c)所示,即可能不含野值点.但在重叠和含噪的数据集中,如图 1(b)和图 1(d)所示,这一点就体现得特别明显.另外,真实数据大多是重叠的或者是含噪或含野值的,如人体基因表达数据[13]、结构化的网络数据[14]、非规范(non-metric)的非向量接近(proximity)数据[15].准确地获取这些结构性信息对重叠或含噪数据进行聚类非常关键.

在基于邻域的聚类(NBC)算法中[11],一个点的邻域密度因子(NDF)被定义为该点反 *k*-邻域里的元素个数与其 *k*-邻域里的元素个数的一个比值.一般地,NDF 值有 3 种可能,即大于 1、等于 1 和小于 1.在 NBC 中[11],根据这 3 种不同的 NDF 值大小,所有的数据点被相应地称为稠密点、平坦点和稀疏点.由 NDF 的定义可知,稠密点往往被许多点(指它自身的反 *k*-邻域中的元素)包围,直观上它看起来像这些数据点的质心.另外,如果聚类之间相互离得很近,接近聚类边界的点很有可能是平坦点;它们将属于几个不同的聚类,即连接这几个不同的聚类,扮演着一种类似于毂的结构性角色.而几乎没有反 *k*-邻居的稀疏点肯定是一个野值,因为“没有反 *k*-邻居”即意味着所有点都离它很远,反过来,它离所有的聚类也很远,从而是一个野值.于是,我们说 NDF 完全有能力识别出数据分布中,无论是云状分布还是流形状分布,也无论哪个点是质心,哪个点是毂和哪个点是野值.借助并依赖于 NDF、NBC 有一定程度的抗噪性,并且能够发现非重叠数据中任意形状、任意大小和任意密度的聚类[11].

显然,*k*-邻域和反 *k*-邻域的定义涉及到两两数据点之间的一个距离度量问题.NBC[11]单纯地用欧氏距离来衡量两两数据点之间的一个几何距离关系,这并不适用于重叠数据和含噪(noisy)数据.因为这种距离度量极其容易将位于聚类之间的毂点或噪声点混淆为质心点,如图 1(b)和图 1(d)所示.视觉上,这些毂点和噪声点同样也被许多点包围.这也直接导致了 NBC 对重叠数据和带有一定噪声和野值的数据无能为力.同时,NBC 还敏感于参数 *k* 的选择;其算法结构也比较繁杂,并且计算复杂度相对来说也比较高.改进的基于邻域的聚类(modified NBC,简称 MNBC)算法[12]通过构建有向树(directed tree,简称 DT)来寻找聚类,有几棵有向树就有几个聚类.这在很大程度上简化和加速了 NBC,并且很好地保持了 NBC 算法的优点,即能够处理简单数据中任意形状、任意大小和任意密度的聚类.由于在构建有向树的过程中,每个数据点只需被遍历一次,从而 MNBC 的计算复杂度线性

于输入数据的大小.正因为如此,该算法也被应用于一些大型的现实数据分析上,如图像分割[12,16].但同样地,MNBC 也不能很好地聚类重叠数据和带有一定噪声和野值的数据,也敏感于参数 k 的选择.



(a) Clouds of points (b) Clouds of points (c) Manifold-Shaped points (d) Manifold-Shaped points

(a) 云状数据 (b)云状数据 (c) 流形状数据 (d) 流形状数据

Fig.1 Toy datasets with two different distributions

图 1 两种分布形式的数据

针对这些问题,本文充分利用核方法的思想,即通过某个核函数将输入数据映入一个高维的特征空间,并在特征空间中完成相应的聚类过程[17−20].事实上,核方法就是以某个核函数表示的非线性度量代替原始数据之间的欧氏度量.径向型和内积型核函数是最常用的两个非线性核度量.径向型核度量的特点是对欧氏度量进行一个统一的径向伸缩,并不会改变数据之间的相对距离大小,因此不会对数据点的 k 邻域与反 k 邻域产生任何影响.所以本文不考虑径向型核函数,而是引入内积型核度量.最典型的内积型核为多项式核.很显然,2 阶以上的多项式核度量并不具备对原始欧氏空间的平移不变性;同时,1 阶多项式核度量即为原始的欧氏度量.这些都说明了引入多项式核度量的可行性和合理性.于是,本文给出了一个基于多项式核的邻域密度因子(polynomial kernel based neighborhood density factor,简称 PKNDF),它能很好地确定映射后的多项式核空间中,各个特征点所扮演的不同的结构性角色,质心、毅还是野值.利用 MNBC[12]中构建有向树的算法结构,本文进一步提出了一种依赖于 PKNDF 的结构化的有向树数据聚类(SDTC)算法,其中,NBC[11]和 MNBC[12]为 1 阶多项式核度量下的 SDC 特例.SDTC 能够有效、快速地聚类重叠数据和噪声数据,并且不敏感于参数 k 的选择.

1 基于多项式核的邻域密度因子

首先回顾一下 NBC 算法[11]中邻域密度因子的定义:

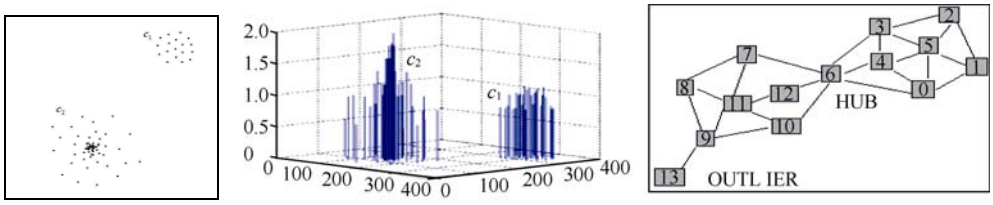
定义 1(邻域密度因子). 给定 $X=\{x_1,x_2,\ldots,x_N\}$,则任意点 x 的反 k-邻域和 k-邻域中元素个数的比值称作该点的邻域密度因子:

$$NDF(x)=\frac{|R\_kNB(x)|}{|kNB(x)|},$$

其中,$|\cdot|$为集合的势,$kNB(x)$ 及 $R\_kNB(x)$ 分别为 x 的 k-邻域和反 k-邻域.记 $knn(x)$ 为 x 的 k 近邻集合,$d(\cdot,\cdot)$ 为欧氏距离度量,并且令 $r=\max_{o\in knn(x)} d(x,o)$,则

$$kNB(x)=\{y\in X\,|\,d(y,x)\le r, y\ne x\},\ R\_kNB(x)=\{y\in X\,|\,x\in kNB(y)\}.$$

文献[11]中指出,不同数据点的 NDF 值是互有差别的,如图 2(a)和图 2(b)所示(即文献[11]中的图 1),最稠密的点 NDF 值最大(大于 1),它们接近于聚类中心;最稀疏的点 NDF 值最小(小于 1),它们离聚类中心最远;而平坦点的 NDF 值等于或接近于 1,它们往往在聚类边界的附近.如果聚类之间离得比较近,如图 2(c)所示(原图见文献[14]中的图 1),点 6 位于两个聚类(0~5,7~12)的公共边界上,因此点 6 就是一个毅.显然,点 13 的周围最稀疏,故它是一个野值(outlier);而接近于聚类中心的点(11 和 5)最稠密.这也说明了一个点在数据内部组织结构中扮演怎样一个结构性角色可以通过 NDF 的值来反映.然而,如果数据重叠或噪声野值点很多,如图 1(b)和图 1(d)所示,那么毅点和野值点的 NDF 值也将因为大于 1 而被误认为是稠密点,如图 5(a)所示,这就导致 NBC[11]具有明显的缺陷:1) 无法处理重叠的数据,如 Iris 数据[20];2) 对数据的噪声比较敏感,加入噪声后往往聚类失败;3) 严重依赖参数 k,参数的灵敏度太高.



(a) A simple data (b) NDF values of (a) (c) 3 structure points in a toy network

(a) 一个简单数据集 (b) 简单数据集内各个点的 NDF 值 (c) 一个人工网络数据集的 3 种结构性数据点

Fig.2 Distinguished NDFs of different points are useful for discriminating different structural roles in the data

图 2 不同数据点间互有差别的 NDF 值可用于判别不同的结构性角色

1.1 多项式核度量的引入

为此,考虑用多项式核度量来代替原始数据之间的欧氏度量.通常,核度量 $d_k(x,y)$的引入可使欧氏空间中均匀的度量变为特征空间中不均匀的度量,从相对意义上讲,使得原来距离较近的数据变得更近,原来距离较远的数据变得更远,从而使相似数据之间的联系增强,不相似数据之间的联系变弱[17−20],其中,

$$d_k(x,y)=\sqrt{k(x,x)+k(y,y)-2k(x,y)},$$

$k(x,y)$为某个核函数.径向型和内积型核函数是经常用的两种核函数:

1) 径向型核函数 $k(x,y)=f\left(\|x-y\|\right)$ ($\|\cdot\|$ 表示欧氏范数);

2) 内积型核函数 $k(x,y)=f\left(dot(x,y)\right)$ ($dot(\cdot,\cdot)$ 表示内积).

Gauss 核函数和 Sigmoid 核函数均为径向型,多项式核函数则为内积型.径向型核度量的特点是对欧氏度量进行了径向的伸缩(近的更近,远的更远),可以显著地增强数据之间的相似性,其在模糊 C 均值等方法中得到了很好的应用[3],但是径向型核度量不会改变数据之间距离大小的次序,不会对点的 k-邻域与反 k-邻域产生任何影响,所以,本文不考虑径向型核而引入内积型核.最典型的内积型核为多项式核

$$\text{Poly\_}k(x,y)=\left(dot(x,y)+a\right)^n,$$

一般地,取 a=1,于是 n 阶多项式核度量可定义为:

定义 2(n 阶多项式核度量).

$$d_n(x,y)=\sqrt{\left(1+dot(x,x)\right)^n+\left(1+dot(y,y)\right)^n-2\left(1+dot(x,y)\right)^n}.$$

注意,当 n=1 时,多项式核度量即等价于欧氏度量.这也从某种意义上说明了利用多项式核度量取代欧氏距离度量的可行性和合理性.

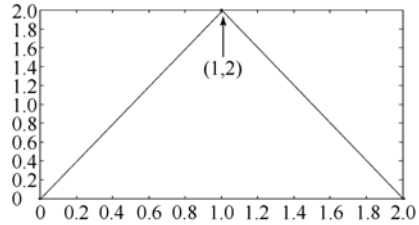
多项式核度量与径向型核度量具有很大的不同,径向型核度量在原始欧氏空间中具有明显的平移不变性,但是 2 阶以上的多项式核度量没有这种平移不变性.例如,取 $x=(0,0)$, $y=(1,0)$, $x_1=(0,10)$, $y_1=(1,10)$,则有

$$d_3(x,y)=2.6458,\quad d_3(x_1,y_1)=175.8,$$

$$d_5(x,y)=5.5678,\quad d_5(x_1,y_1)=23037.$$

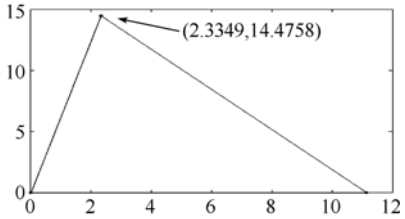
显然,欧氏空间中平移后两点的多项式核度量与平移前两点的多项式核度量有很大不同,而且这种不同随着多项式阶数的增加变得更剧烈.

为了形象地说明多项式核度量对样本空间内数据间相对距离的改变,取输入样本空间中的两个基准点(0,0)和(2,0),对于任意数据点,这里取(1,2),它与两个基准点组成的三角形(图 3(a))通过多项式核映射仍然映为一个三角形(图 3(b)),其中,两个基准点分别映射为(0,0)和 x 轴正半轴上的一点,数据点(1,2)则通过 3 阶多项式核度量被映射为特征空间中的点(2.3349,14.4758);而多项式核度量下 3 个数据间的两两距离即对应 3 条边的边长.



(a) Three data (0,0),(2,0) and (1,2) constitutes a triangle in the input space

(a) 输入空间中由 3 个数据点(0,0),(2,0)和(1,2)组成的三角形



(b) Corresponding ones mapped by the polynomial kernel with rank 3

(b) 3 阶多项式核映射下的三角形

Fig.3 Change of pairwise distances in the sample space measured by the polynomial kernel induced metric

图 3 多项式核度量对样本空间内数据间相对距离的改变

进一步地,考虑如图 4 所示中的数据集(第 1 行、第 1 列):内环用点表示,外环用加号表示.由于两个圆环有一定的重叠,一般的聚类方法很难将二者成功分开.利用不同阶的多项式核,数据间相对距离值的变化如图 4(第 1 行第 2 列至第 3 行)所示,从上到下,从左到右,多项式核的阶数分别为 n=3,4,5,6 和 7.可以发现,随着多项式核阶数的增加,内环点越来越集中在原点附近,外环点则相对分散;命题 1 对其给出了数学上的近似描述.显然,这一点非常有利于对重叠数据进行聚类.

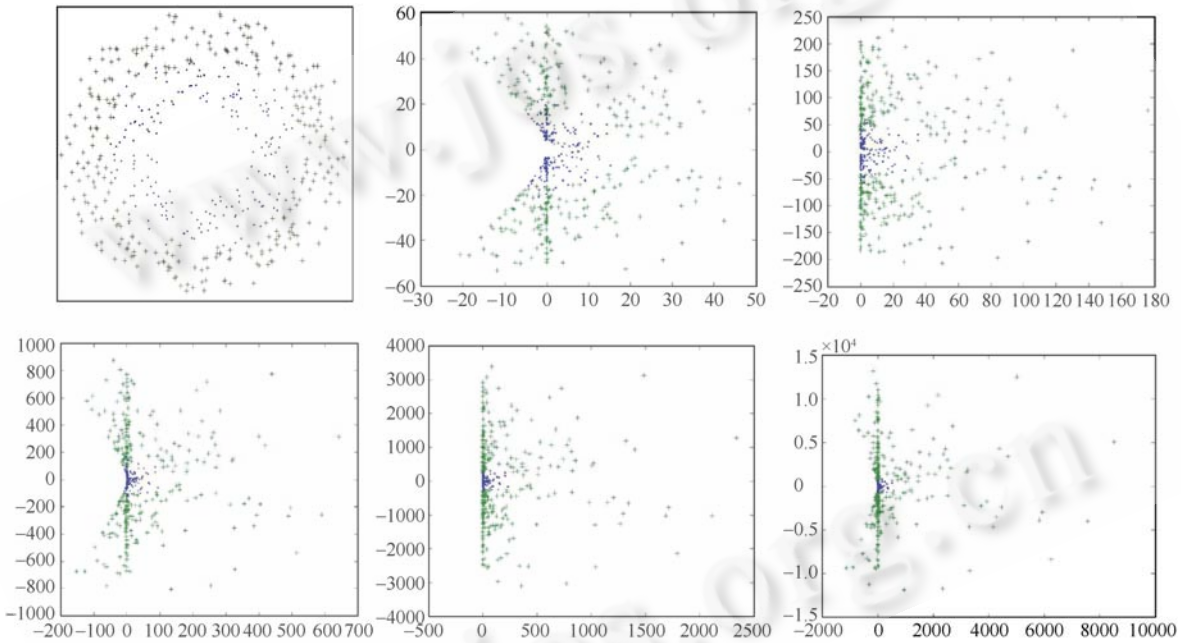


Fig.4 Inter-Distances between overlapping points vary with the rank of used polynomial kernel, where points on the inner and outer circle are denoted by "•" and "+" respectively

图 4 不同阶多项式核下重叠环数据间的相对距离示意图(点表示内环数据点,加号表示外环数据点)

命题 1. n 越大,内侧点相对地越靠近原心,外侧点相对地越分散.

证明:假设 $x_1 = x, y_1 = (1+r)x (r>0)$; $x_2 = cx (c>1), y_2 = (c+r)x$ 为同一条直线上的 4 个点;并且可视 x_1, y_1 为内侧的两个点, x_2, y_2 为外侧的两个点(因为 $c>1$);这样,内侧上的两点 x_1, y_1 之间的欧氏距离显然等于外侧上的两点 x_2, y_2 之间的欧氏距离,即

$$\frac{d(x_1, y_1)}{d(x_2, y_2)} = \frac{\|rx\|}{\|rx\|} = 1,$$

但是由定义 2 可知,它们的多项式核距离显然不同.具体地,根据式(1.2)有,

$$\frac{d_n(x_1,y_1)}{d_n(x_2,y_2)}=\sqrt{\frac{\left(1+\|x\|^2\right)^n+\left(1+(1+r)^2\|x\|^2\right)^n-2\left(1+(1+r)\|x\|^2\right)^n}{\left(1+c^2\|x\|^2\right)^n+\left(1+(c+r)^2\|x\|^2\right)^n-2\left(1+c(c+r)\|x\|^2\right)^n}},$$

分子分母同除以 $1+(c+r)^2\|x\|^2$ 得

$$\frac{d_n(x_1,y_1)}{d_n(x_2,y_2)}=\sqrt{\frac{(a_1)^n+(a_2)^n-2(a_3)^n}{(a_4)^n+1-2(a_5)^n}},$$

其中,

$$a_1=(1+\|x\|^2)/(1+(c+r)^2\|x\|^2),$$
$$a_2=(1+(1+r)^2\|x\|^2)/(1+(c+r)^2\|x\|^2),$$
$$a_3=(1+(1+r)\|x\|^2)/(1+(c+r)^2\|x\|^2),$$
$$a_4=(1+c^2\|x\|^2)/(1+(c+r)^2\|x\|^2),$$
$$a_5=(1+c(c+r)\|x\|^2)/(1+(c+r)^2\|x\|^2).$$

由于 $a_i\in(0,1)(i=1,2,3,4,5)$,因此 $\lim_{n\to\infty}(a_i)^n=0$,即 $\lim_{n\to\infty}\frac{d_n(x_1,y_1)}{d_n(x_2,y_2)}=0$,这意味着当 n 比较大时,内侧点间的距离比外侧点间的距离要小得多.因此,从相对意义上讲,n 越大,内侧点越靠近原心,外侧点则越分散. □

1.2 基于多项式核的邻域密度因子

于是,定义 1 中的欧氏距离度量 $d(\cdot,\cdot)$ 被取代为 $d_n(x,y)$,即为基于多项式核的邻域密度因子(PKNDF).

定义 3(基于多项式核的邻域密度因子). 在多项式核特征空间中,任意一个特征点 x 的反 k-邻域和 k-邻域中元素个数的比值称为该特征点的一个基于多项式核的邻域密度因子,即

$$\mathrm{PKNDF}(x)=\frac{|R\_kNB_{PK}(x)|}{|kNB_{PK}(x)|}.$$

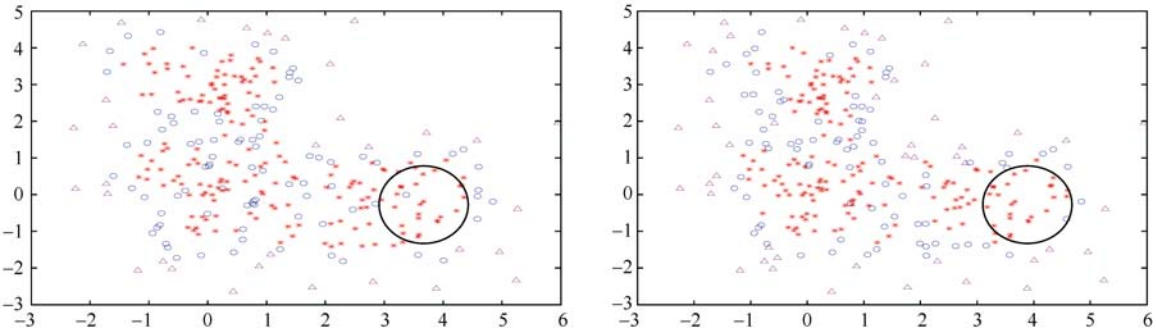
相应地,记 $knn_{PK}(x)$ 为核特征空间(记 $X_{\ker}$)中 x 的 k 近邻集合,则多项式核特征空间 $X_{\ker}$ 中 x 的 k-邻域为

$$kNB_{PK}(x)=\{y\in X\,|\,d_n(y,x)\le r, y\ne x\},$$

反 k-邻域为 $R\_kNB_{PK}(x)=\{y\in X\,|\,x\in kNB(y)\}$,其中 $r=\max_{o\in knn_{PK}(x)}d_n(x,o)$.

同样,PKNDF 的值也有 3 种可能,即大于 1、等于 1 和小于 1,也能反映出每个特征点所扮演的不同结构性角色.PKNDF 值大于或等于 1 的点充当某些聚类的质心,PKNDF 值接近于 0 的点即为野值,而 PKNDF 值小于 1 但接近于 1(一般大于 0.5)的点是毅.直观地,质心点可用以生成聚类,而毅点用以终止某个聚类的继续扩张,野值点被排除在任何聚类之外.

然而,如前所述,对于重叠或含噪数据集而言,定义 1 中的 NDF 并不能正确地反映出各个点在数据内部组织中所扮演的“真正”结构性角色,如图 5(a)所示,椭圆标记处的一些毅点(○表示)或噪音点(△表示)都成了质心点(+表示),因此这将直接导致 NBC 算法不易将重叠聚类分开.但是,如图 5(b)所示,PKNDF 却对这些数据点所扮演的结构性角色作出了正确的判断,这对处理重叠或含噪聚类非常关键,因此可以说引入多项式核度量并进一步提出 PKNDF 对这些类型数据的识别具有积极意义.于是,借助于这个 PKNDF,下面将给出一个结构化的数据聚类算法,该算法的聚类过程实质上是一个生成有向树的过程.



(a) NDF (k=10) (b) PKNDF (n=5,k=10)

Fig.5 Discriminated structure points in the same dataset by NDF (left: n=1, k=10) and PKNDF (right: n=5, k=10)

图 5 相同数据集内分别由 NDF(左:n=1,k=10)及 PKNDF(右:n=5,k=10)判别的不同结构性角色

2 结构化的有向树数据聚类算法

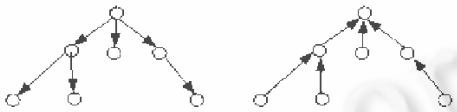
首先给出有向树等相关的图论概念.

定义 4(有向图和有向路径)[12,21]. 有向图 $G(V,E)$ 为具有结点集 $V=X=\{x_1,x_2,...,x_N\}$ 和有向边集 $E=\{e(i,j)\}$ 的连通图(始点为 x_i,终点为 x_j,且 x_j 称为 x_i 的子结点).另外,对于有向边集 $\{e_1,...,e_n\}$,若 e_1 的始点为 A,e_n 的终点为 A',并且 e_k 的终点为 e_{k+1} 的始点 $(k=1,...,n-1)$,则称 $\{e_1,...,e_n\}$ 为 A 到 A' 的有向路径.

基于定义 4,有向树大致可分成两种:自下而上(B-U)和自上而下(T-D),分别定义如下:

定义 5(有向树)[12,21]. 有向图称为有向树,若不存在循环路径(即节点 A 到自身的有向路径)并且存在唯一的节点 R 满足:(1) 节点 R 不是任何有向边的始点(终点);(2) 任意一个不是 R 的结点 A 恰好为一条有向边的始点(终点);满足上述条件的结点 R 分别称为 B-U 和 T-D 有向树的根结点.

相应地,构建有向树的方法也可以分成两种:“寻根”和“繁殖”,如图 6 所示.对于“寻根”方法来说,从任意结点出发,顺着每条边的始点到终点,寻找其父亲结点,自下而上,一直到根结点为止,这样有着共同根结点 R 的所有结点就形成了一棵有向树,即生成一个聚类.对于“繁殖”方法来说,事先选定一个具有繁殖能力的结点作为根结点 R,从 R 出发,不断地繁殖其孩子结点,顺着每条边的始点到终点,自上而下,一直到所有新的孩子结点都不再具有繁殖能力为止,这样,根结点 R 及其所有的子结点、子结点的子结点等等,就构成一棵有向树,即一个聚类.



(a) Top-Down (b) Bottom-Up
(a) 自上而下 (b) 自下而上

Fig.6 Two ways of building directed tree

图 6 生成有向树的两种方式

本文采用与文献[12]中 MNBC 算法一样的“繁殖”方法.以样本集作为结点集,规定只有质心点(PKNDF(x)≥1)才具有繁殖能力,其余两种结构性结点均无繁殖能力.如此,所有结点大致可以分为两类:具有繁殖能力的结点和不具有繁殖能力的结点.具有繁殖能力的质心结点可以有子结点;不具有繁殖能力的毂结点不能有子结点,只能作为边的终止结点;而剩下没有繁殖能力的野值点不能成为任何边的结点,其算法如下:

算法 1. 结构化的有向树数据聚类(SDTC)算法.

(1) 给定样本集 $X=\{x_1,x_2,...,x_N\}$,输入 k,利用多项式核度量计算每个样本点的 PKNDF(x_i),$i=1,2,...,N$.

(2) 初始化 V=X,Num_trees=0, $Itrees=\varnothing$.

(3) 如果$\forall x\in V$, PKNDF(x)<1,则算法终止;否则,转(4).

(4) 任意选择$x\in V$满足PKNDF(x)≥1为根节点,其kNB_{PK}内除自身外的所有结点作为其子结点.即x与其kNB_{PK}内所有结点间建立有向边,形成一棵初始有向树,记$Itrees=\{x\}\cup kNB_{PK}(x)$,并令$I=kNB_{PK}(x)$.

(5) 遍历集合I中满足PKNDF(y)≥1的结点y作为内部结点(具有繁殖能力的子结点)以扩展初始有向树.即建立y到其kNB_{PK}内所有结点的有向边(除生成有向树上已存在的结点).记

$$I=\{t|t\in kNB_{PK}(y),t\notin Itrees,\text{PKNDF}(y)\geq 1,y\in Itrees\},$$

令$Itrees=Itrees\cup I$.

(6) 若$\{y\in I|\text{PKDND}(y)\geq 1\}\neq\varnothing$,则重复(5);否则,$Final\_trees=Itrees$,$Num\_trees=Num\_trees+1$.

(7) 输出有向树$Final_trees$;令$V=V\setminus Final_trees$,返回(3).

(8) 每棵有向树$Final_trees$为一个聚类,Num_trees为聚类个数;未出现在任何一棵有向树上的结点称为野值(0类).

该算法中,计算每个样本点的PKNDF值即步骤(1)复杂度最高,涉及到数据间两两距离的计算,复杂度为$O(N^2)$,其中N为结点总数,即样本点总数.事实上,几乎所有算法都要涉及数据之间距离的计算.因此,若不考虑步骤1的计算复杂度,则该算法的计算复杂度线性于样本点个数,即$O(N)$[12],因为每个样本点只需被遍历一次.正因为如此,该算法被广泛地应用于大规模的数据分析问题,如图像分割[12,16].进一步地,由文献[16]可知,每棵有向树实际上都对应于一个等价类(质心点),因此,该算法对初始质心结点的选取鲁棒,即无论从哪个质心结点开始生成有向树,聚类结果都相同.如图7所示,X={x1,x2,x3,x4,x5},当$n=1,k=2$时,分别以质心点$x1,x2,x3$为根结点,算法1生成3棵不同的有向树,但生长在有向树上的结点相同,均为{x1,x2,x3,x4}并且x5均被判为野值.另外,由于1阶多项式核度量等价于欧氏度量,因而MNBC算法为SDTC算法的一个特例,等价于1阶SDTC算法.

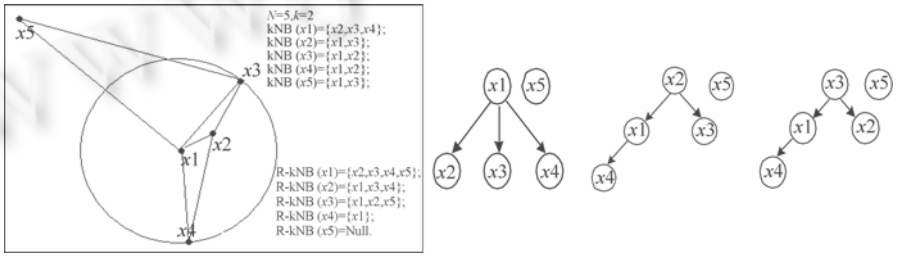


Fig.7 Simple illustration of the generation of directed tree

图7 有向树生成的简单示意图

3 实验与分析

首先比较本文算法SDTC与文献[3]中的NBC算法,其次比较SDTC与:模糊C均值(FCM)[1]和谱聚类(SC)[4]这两种经典算法.

3.1 与NBC算法的比较

如前所述,NBC算法有3点不足:(1) 无法处理重叠数据集;(2) 对噪声比较敏感,甚至常导致聚类失败;(3) 严重依赖参数k,对其敏感度过高.本节试图通过数据实验来说明SDTC算法能克服NBC算法的上述不足.

3.1.1 重叠数据与含噪数据上的实验

首先考虑由3个环组成的数据(如图8所示),内环、中环、外环分别具有100,200,300个数据点,随着环的宽度从左到右由0.2,0.3,0.35逐渐地增加为0.4,3个环之间的重叠性越来越大.实验结果显示,仅对数据a(w=0.2),NBC可以成功聚类,而对于数据b,c,d,NBC的聚类均告失败.而本文SDTC算法对于数据a,b,c,d则都能聚类成功(详见图7),说明SDTC能够处理重叠数据的聚类问题.

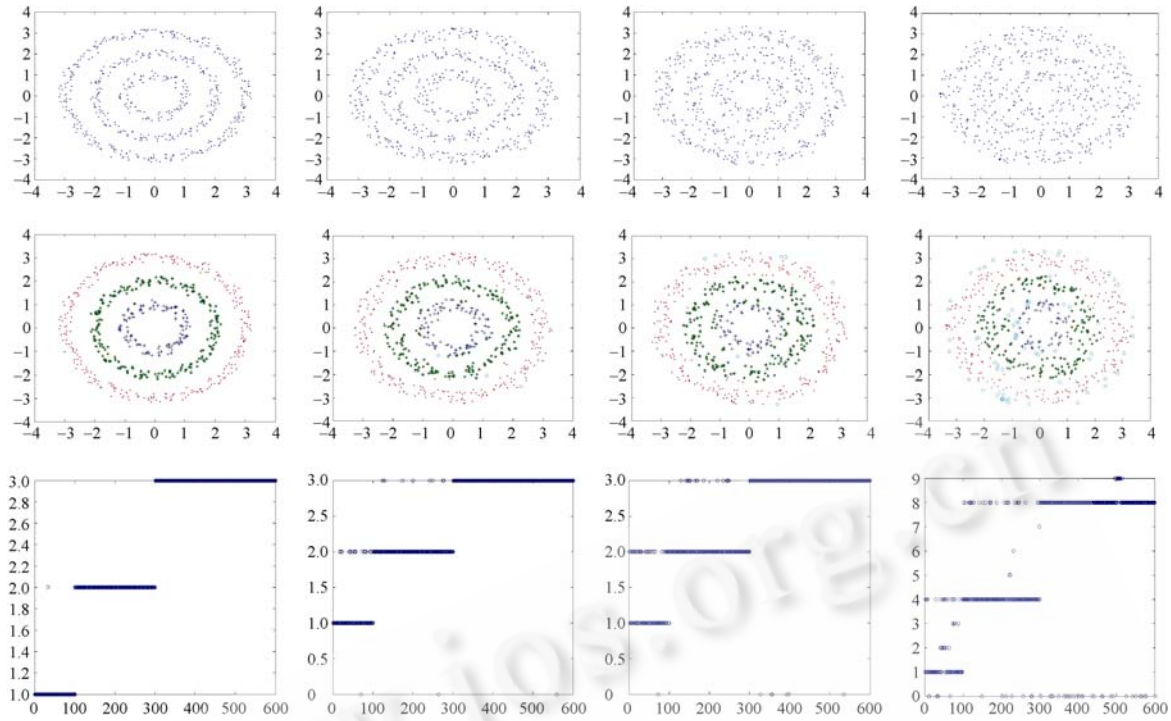


Fig.8 Experimental results on three concentric circles with different widths. From up to down, the three rows are input data, clustering results and cluster labels. From left to right, the parameter settings are n=1, 3, 3 and 7, k=18, 16, 17 and 10 respectively

图 8 4 个不同宽度的 3 个同心圆数据集上的实验结果.自上而下,1~3 行分别为输入数据,聚类结果及类标号;从左往右,1~4 列分别是 NBC,三阶、三阶和七阶 SDTC 的聚类结果,其中 k=18,16,17,10

随后考虑由两个环组成的数据,并加入随机噪声,内环为 200 个点(加号表示),外环为 300 个点(星号表示),同时加入了 200 个随机噪声点(点表示),详见图 1(d),分别使用 NBC 算法和五阶多项式核的 SDTC 算法进行实验,最优结果如图 9 所示.显然,NBC 聚类完全失败,而 SDTC 则自然地将内环和外环聚为不同的两类(虽然有些噪声点被错分进了内环).

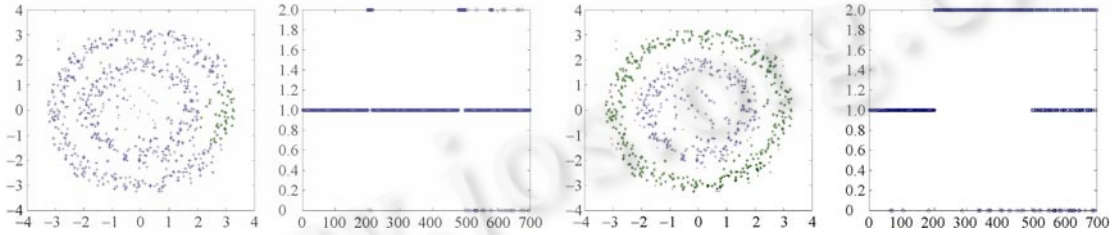


Fig.9 Experimental results on a two concentric circles contaminated by noise. The first two columns are ones by NBC (n=1, k=21); and the last two columns are ones by STDC (n=5, k=17)

图 9 加入噪声的两个环的实验结果.前两列:NBC(n=1,k=21);后两列: SDTC(n=5,k=17)

3.1.2 参数的灵敏度

NBC 算法的聚类结果对参数 k 非常敏感,限制了 NBC 的应用.为了探讨 SDTC 算法对参数 k 的敏感度,考虑图 1(c)中简单的流形状数据集,它包含两个分离的同心圆.具体实验结果见表 1,黑体表示能够将数据聚为两类,且内环与外环所属的类不同(即将两类分开)时的结果.表 1 中,k 从 6~25 取值,计算 NBC 算法和 3~6 阶 SDTC

算法的错分率与聚类个数;并且只对能够将两类分开的聚类结果计算错分率,否则认为聚类失败.如果前 120 个点所属的最大聚类 c1 和后 180 个点所属的最大聚类 c2 不同,则认为将两类分开,否则认为没有将两类分开,聚类失败.错分率的计算公式为:前 120 个数据点的类标号不等于 c1 的个数/120+后 180 个点的类标号不等于 c2 的个数/180.

Tabel 1 Test results of parameter sensitivity

表 1 参数敏感度测试结果

Error rate/cluster number	NBC	SDTC of rank 3	SDTC of rank 4	SDTC of rank 5	SDTC of rank 6
k=6	83.67%/23	59.33%/11	67.00%/10	57.00%/7	43.33%/6
k=7	68.00%/17	34.67%/5	35.00%/5	34.33%/8	39.67%/7
k=8	39.67%/7	34.67%/4	27.00%/3	27.67%/3	29.33%/4
k=9	37.67%/4	26.00%/3	35.67%/4	36.67%/4	**4.67%/2**
k=10	10.33%/3	10.33%/3	25.33%/3	**2.33%/2**	**3.67%/2**
k=11	**0 / 2**	10.33%/3	**0.67%/2**	**2.33%/2**	**3.00%/2**
k=12	19.33%/4	**0/2**	**0.67%/2**	**1.33%/2**	**2.33%/2**
k=13	11.00%/4	**0/2**	**0.67%/2**	**2.33%/2**	**4.67%/2**
k=14	Failure/1	**0/2**	**0.67%/2**	**2.33%/2**	**7.67%/2**
k=15	Failure/1	**0/2**	**2.00%/2**	**5.00%/2**	**13.00%/2**
k=16	Failure/1	**0.67%/2**	**4.33%/2**	**7.33%/2**	**14.67%/2**
k=17	Failure/1	**2.67%/2**	**7.33%/2**	**11.00%/2**	**18.33%/2**
k=18	Failure/1	**4.67%/2**	**8.33%/2**	**12.33%/2**	20.67% (F)/2
k=19	Failure/1	**6.67%/2**	**10.33%/2**	**15.67%/2**	23.33% (F)/2
k=20	Failure/1	**9.67%/2**	**13.33%/2**	**19.00%/2**	25.33% (F)/2
k=21	Failure/1	**11.67%/2**	**16.00%/2**	20.33% (F)/2	27.00% (F)/2
k=22	Failure/1	**14.33%/2**	**18.33/2**	23.33% (F)/2	28.00% (F)/2
k=23	Failure/1	**14.67%/2**	20.33% (F)/2	24.67% (F)/2	30.33% (F)/2
k=24	Failure/1	**18.33%/2**	23.33% (F)/2	27.00% (F)/2	32.00% (F)/2
k=25	Failure/1	20.33% (F)/2	24.67% (F)/2	29.67% (F)/2	36.33% (F)/2

可以看出,NBC 算法仅当参数 *k*=11 时获得完全成功,而 SDTC 算法则显示出了对参数 *k* 的不敏感性.对于 SDTC 而言,多项式核的阶数从 3 阶~6 阶,至少存在 10 个 *k* 的参数值能够使数据聚为基本的内环和外环两类.图 10 分别给出了不同参数的 NBC 与 SDTC 对于数据错分率和聚类数目的变化趋势.由图 10 的左边可以看出,NBC 算法的错分率非常剧烈地随 *k* 而发生变化,相反,SDTC 算法的错分率变化则要平缓得多;图 10 的右边则显示了聚类数目的变化情况,NBC 仅在 *k*=11 时可以聚为所期望的两类,而 SDTC 对于大部分 *k* 都能够将数据聚为基本的两类.这表明参数 *k* 的选取对 SDTC 的影响要远小于对 NBC 的影响.

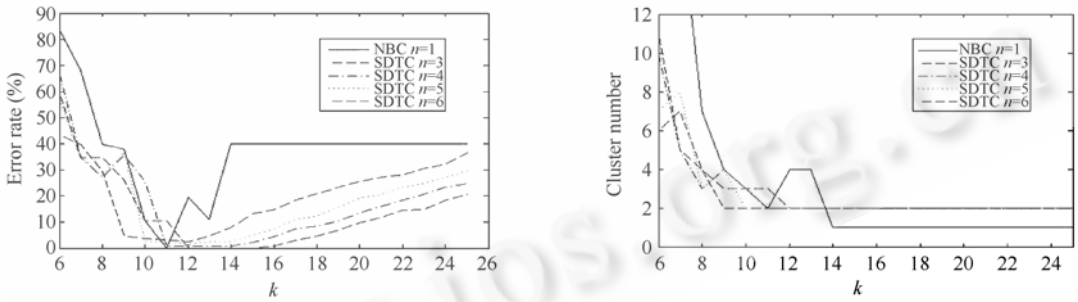


Fig.10 Comparisons of NBC and SDTC on error rate and cluster numbers varied with the parameter *k*

图 10 参数 *k* 取不同值时 NBC 和 SDTC 在错误率和聚类数目上的比较

3.1.3 云状数据和 Iris 数据上的实验

NBC 算法的另一个比较明显的不足是对云状数据的聚类能力不够,通过多项式核的映射则能够获得改善.图 1(b)为一个云状数据的例子,有 3 类组成,每一类均为平面上的一个正态分布,各有 100 个数据点(分别用加号、星号和点号表示).图 11 显示了 NBC 和三阶、四阶、五阶核 SDTC 的最佳实验结果.从图中可以看出,NBC 无法将 3 类分开,第 1 类和第 3 类被分成了一类,而不同阶核的 SDTC 均能将 3 类分开.由此可见,对于云状重叠数据集,SDTC 同样也能获得令人满意的聚类结果.

Iris 数据是常用的聚类测试数据,共有 150 个数据点组成 3 类,第 2 类和第 3 类有一定重叠.NBC 算法无法成功聚类 Iris 数据,总是将第 2 类和第 3 类聚为一类,而多项式核的 SDTC 算法比较成功地将 Iris 数据聚为 3 类,结果见表 2(聚类结果中的 0 类即为野值).

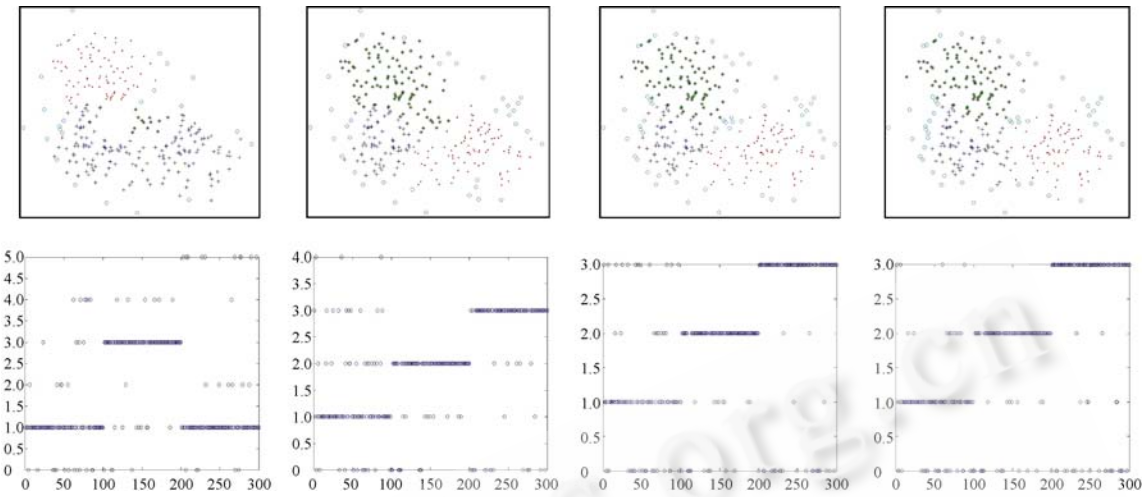


Fig.11 Experimental results on the overlapping data with clouds of points. From left to right, the four columns are results clustered by NBC (k=9), SDTC (k=11, 14 and 14) respectively, where points denoted by "○" are outliers

图 11 云状重叠数据集的聚类结果.第 1~4 列分别为 NBC,3 阶、4 阶、5 阶 SDTC 的聚类结果,参数分别为 k=9,11,14 和 14,其中圆圈表示 0 类(被判为野值)

Table 2 Clustering results on Iris by the polynomial kernel induced metric

表 2 利用多项式核度量的 Iris 数据聚类结果

SDTC (k=12)	Rank 5		Rank 6		Rank 7		Rank 8	
	Error number	Outlier number	Error number	Outlier number	Error number	Outlier number	Error number	Outlier number
Class I	0	6	0	6	0	6	0	6
Class II	2	4	2	7	5	4	5	4
Class III	10	4	10	5	10	6	10	6

3.2 与FCM和SC算法的比较

在第 3.1 节中,通过实验已经表明 SDTC 算法能够克服 NBC 算法的几个不足,能够聚类多种类型的数据,同时具有对噪声、野值以及参数选择的鲁棒性.本节将 SDTC 算法和两种经典的算法——模糊 C 均值(FCM)和谱聚类(SC)算法进行比较.相对于 FCM 和 SC,SDTC 最大的区别在于不需要事先给定聚类的数目,也就是说,本文算法不需要对聚类个数有先验的了解,只要选定参数 k,就可以自动地将数据进行聚类.本节中,首先对一些人工数据进行实验,然后对加入野值和噪声的 Iris 数据进行实验.

3.2.1 人工数据上的实验

考虑在两个人工数据(如图 12 所示)上对 3 种算法进行比较:嵌入两个随机分布方形(各有 100 个点)的圆环(200 个点)数据集及 3 个螺旋臂(各有 200 个点)的数据集.从图 12 的结果可以看出,本文算法对这两个数据集都有很好的聚类结果.FCM 作为最常用的聚类方法之一,注重数据空间上的联系,但不能充分挖掘数据的内部结构,因而对第 1 个数据集只能将两个方形数据分开,而不能聚类圆环;对第 2 个数据集尽管能从空间上分为了 3 类,但并不能正确地对应于 3 个螺旋臂.谱聚类是目前研究较多的聚类方法,与 FCM 相比,谱聚类更能够发现数据的内部结构,但当数据的结构较复杂时,谱聚类并非都能成功.从图中可以看出,SC 虽然比 FCM 的聚类结果相对要好些,但仍不能完全将数据聚为正确的 3 类,仅能部分地发现数据的内部结构.本文算法 SDTC 继承并加强

了 NBC 的聚类能力,能够发现不同形状、不同类型的数据结构,通过适当调整参数,获得了对两个数据均为满意的结果.

3.2.2 加入野值和噪声的 Iris 数据实验

模糊 C 均值和谱聚类都可以成功地对 Iris 数据进行聚类,由第 3.1.3 节可知,五阶以上的核 SDTC 也能成功聚类 Iris 数据.为了比较 SDTC,FCM 和 SC 对噪声和野值的鲁棒性,利用这 3 种算法对加入一些野值与噪声样本的 Iris 数据进行分别聚类.首先在 Iris 数据中加入 4 个野值$(a,0,0,0),(0,a,0,0),(0,0,a,0)$和$(0,0,0,a)$,并将本文的算法与模糊 C 均值和谱聚类方法进行比较,结果见表 3.

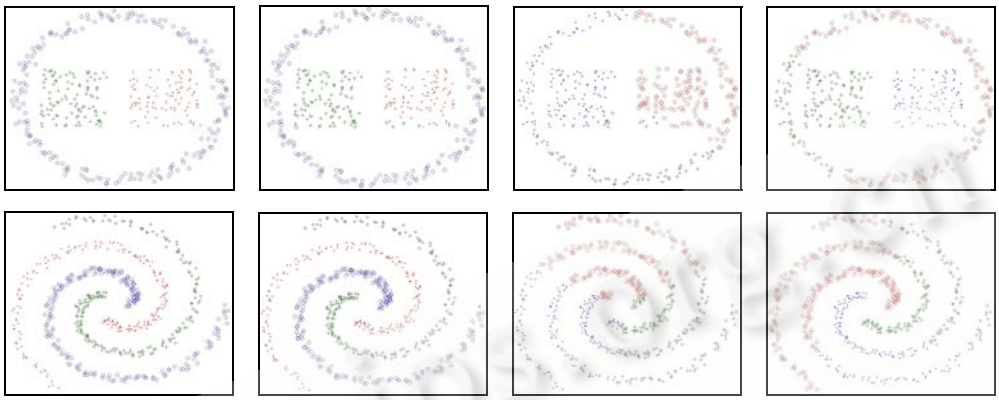


Fig.12 Experimental comparisons on two synthetic datasets. From left to right, the four columns are the ideal clusters and results by 3-rank SDTC (k=15, 17), FCM (c=3) and SC (num=3)

图 12 两个人工数据的聚类比较结果.从左往右,一至四列分别为理想分类图(三类)、3 阶 SDTC(k =15,17)、FCM(c=3)和 SC(num=3)

Table 3 Comparison results with the added outliers

表 3 加入野值后的聚类结果和比较

Methods	Error number				
	Iris	a=10	a=20	a=50	a=60
FCM (c=3)	I	0	0	0	0
	II	5	2	1	46
	III	10	14	26	0
SC (num=3)	I	0	0	0	0
	II	1	50	47	46
	III	13	0	0	0
SDTC (k=12, n=5)	I	0	0	0	0
	II	2	2	2	2
	III	10	10	10	10

其次,在 Iris 数据中加入 10 个随机噪声数据,每个噪声数据的 4 个坐标均在区间$[0,a]$内随机取值,采用与上面相同的参数,3 种算法的结果如图 13 所示.图 13 中,横坐标为野值向量中 a 的值,纵坐标为错分个数.显然,从表 3 和图 13 可以看出,噪声和野值对本文方法的聚类结果几乎没有影响,但对模糊 C 均值和谱聚类的影响却相对较大,从而使聚类失败.

4 结束语

本文提出了一种结构化的有向树聚类(SDTC)算法,其中引入的基于多项式核的邻域密度因子能够抓住数据点本身所体现的结构性信息.在人工和真实数据上的实验显示了 SDTC 对一些复杂的流形或云状数据都具有良好的聚类能力,并且对噪声、野值及参数的选择具有一定程度的鲁棒性.与 NBC[11]及经典的聚类算法 FCM

和 SC 相比,SDTC 显示出了较强的聚类优势.

另外,由于邻域密度因子仅仅与每个点 *k*-邻域及反 *k*-邻域内点的个数有关,因此针对不同类型的数据集,只要能够合理地定义出相应的相似性度量,SDTC 即可推广到多种类型的数据集,而不仅仅局限于本文给出的一些数值型数据集,如符号值数据集、人脸数据集、图像像素集或网络数据集等.实际上,已有文献[12,16]在做这方面的工作.同时,这也是本文后续将要进一步考虑的研究问题和主攻方向.

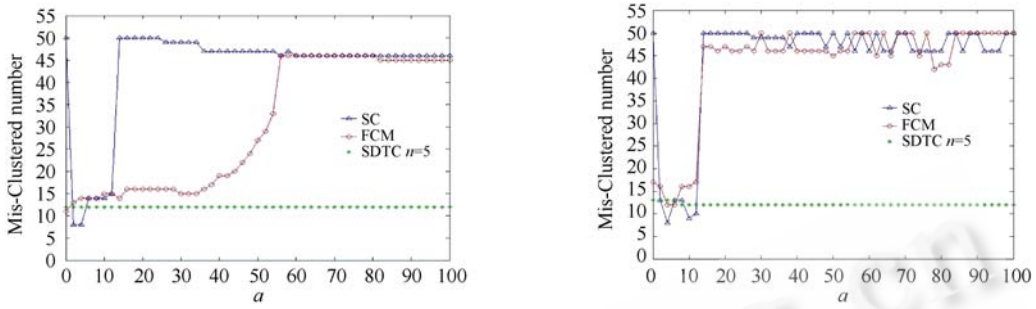


Fig.13 Experimental comparisons on Iris data with different outliers (left) and noise (right)

图 13 加入野值后的(左侧)和加入随机噪声后(右侧)的 Iris 数据聚类结果和比较

References:

[1] Theodoridis S, Koutroumbas K. Pattern Recognition. 2nd ed., New York: Academic Press, 1999.

[2] Han J, Kamber M. Data Mining: Concepts and Techniques. Morgan Kaufmann Publishers, 2000.

[3] Chen SC, Zhang DQ. Robust image segmentation using FCM with spatial constraints based on new kernel-induced distance measure. IEEE Trans. on Systems, Man, and Cybernetics—Part B: Cybernetics, 2004,34(4):1907−1916.

[4] Shi J, Malik J. Normalized cuts and image segmentation. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2000,26(8): 888−905.

[5] Breitenbach M, Grudic GZ. Clustering through ranking on manifolds. In: Proc. of the 22nd Int'l Conf. on Machine Learning (ICML 2005), Vol.119. New York: ACM, 2005. 73−80.

[6] Hofmann T, Buhmann JM. Pairwise data clustering by deterministic annealing. IEEE Trans. on Pattern Analysis and Machine Intelligence, 1997,19(1):1−14.

[7] Fischer B, Zoller T, Buhmann JM. Path based pairwise data clustering with application to texture segmentation. Energy Minimization Methods in Computer Vision and Pattern Recognition. 2001. 235−250.

[8] Fischer B, Buhmann JM. Bagging for path-based clustering. IEEE Trans. on Pattern Analysis and Machine Intelligence, 2003, 25(11):1411−1415.

[9] Chang H, Yeung DY. Robust path-based spectral clustering with application to image segmentation. In: Proc. of the 10th IEEE Int'l Conf. on Computer Vision (ICCV). Beijing: IEEE Computer Society, 2005. 278−285.

[10] Ester M, Kriegel HP, Sander J, Xu X. A density-based algorithm for discovering clusters in large spatial databases with noise. In: Simoudis E, Han J, Fayyad U, eds. Proc. of the 2nd Int'l Conf. on Knowledge Discovery and Data Mining. Beijing: The AAAI Press, 1996. 221−226.

[11] Zhou S, Zhao Y, Guan J, Huang J. A neighborhood-based clustering algorithm. In: Proc. of the 9th Pacific-Asia Conf. on Knowledge Discovery and Data Mining (PAKDD 2005). LNAI 3518, Berlin/Heidelberg: Springer-Verlag, 2005. 361−371.

[12] Ding J, Ma R, Chen S, Wang B. A fast directed tree based neighborhood clustering for image segmentation. In: Proc. of the 13th Int'l Conf. on Neural Information, Part II. LNCS 4233, Berlin, Heidelberg: Springer-Verlag, 2006. 369−378.

[13] Gibbons FD, Roth FP. Judging the quality of gene expression-based clustering methods using gene annotations. Genome Research, 2002,12:1574−1581.

[14] Xu X, Yuruk N, Feng Z, Schweiger TAJ. SCAN: A structural clustering algorithm for networks. In: Proc. of the 13th Int'l Conf. on Knowledge Discovery and Data Mining. 2007.

[15] Puzicha J, Hofmann T, Buhmann JM. A theory of proximity based clustering: Structure detection by optimization. Pattern Recognition, 2000,33:617−634.

[16] Ding J, Ma R, Chen S. A scale-based coherence connected tree algorithm for image segmentation. IEEE Trans. on Image Processing, 2008,17(2):204−216.

[17] Muller KR, Mika S, Ratsch G, Tsuda K, Scholkipf B. An introduction to kernel-based learning algorithms. IEEE Trans. on Neural Networks, 2001,12(2):181−201.

[18] Shawe-Taylor J, Cristianini N. Kernel Methods for Pattern Analysis. London: Cambridge University Press, 2004.

[19] Hur AB, Horn D, Siegelmann HT, Vapnik V. Support vector clustering. Journal of Machine Learning Research, 2001,2:125−137.

[20] Wang Z, Chen S, Liu J, Zhang D. Pattern representation in feature extraction and classifier design: Matrix versus vector. IEEE Trans. on Neural Networks, 2008,19(4):758−769.

[21] Koontz W, Narendra P, Fukunaga K. A graph-theoretic approach to non-parametric cluster analysis. IEEE Trans. on Computers, 1976,25(9):936−944.

丁军娣(1978－),女,江苏宜兴人,博士,主要研究领域为数据聚类,图像分割,聚类学习.

马儒宁(1976－),男,博士,副教授,主要研究领域为神经网络,图像处理,聚类分析.

陈松灿(1962－),男,博士,教授,博士生导师,CCF 高级会员,主要研究领域为模式识别,机器学习.