山峰—减法聚类神经元模型及学习算法*

Clustering Neuron Model of Peak-subtraction and Learning Algorithm

周永权　　谢宁新
Zhou Yongquan　Xie Ningxin

(广西民族学院数学与计算机科学系　南宁　530006)
(Department of Mathematical and Computer Science, Guangxi University for Nationalities, Nanning, 530006)

摘要　将神经网络与数据集的密度指标结合起来提出一种山峰—减法聚类神经网络方法，利用数据集的密度指标对基类进行合并，并不断重复直至产生足够多的聚类中心，就可完成对聚类神经元的学习。给出该聚类的神经元模型和学习算法。该方法的主要优点是对于工程应用中的大样本集分类和重叠数据的模式分类问题，显得非常有效。

关键词　聚类法　激励函数　聚类神经元　学习算法
中图法分类号　TP391　　A

Abstract　A clustering neuron model with neuron activation function variable, which is peak-subtraction clustering neuron model, is proposed in the combination of neuron network and density indexes of data sets. The new model has more advantages in solving the problems of classifications of the big sample sets and overlap data modes in engineering application.

Key words　clustering, activation function, clustering neuron, learning algorithm

Yager 和 Filev[1]在 1994 年提出的山峰聚类方法是一种大致估计聚类中心的相对简单有效的方法. 利用该算法获得很多高级聚类算法，如：模糊 C 均值聚类算法[2]，K-medoid 方法[3]等. 也作为一种快捷独立的的近似聚类方法，在模糊建模的结构辨识中得到应用. 这种方法是基于人类视觉上的一个数据集形成聚类的原理，然后，在模拟人的大脑思维作出推理、判断. 山峰聚类法虽简单而有效，由于该方法必须计算所有格点上的山峰函数，因此该方法的计算量随着问题的维数增加而呈指数增长. 如：一个有 4 个变量且每一维有 10 条网格线的聚类问题，须计算 104 个网格点. 于是，Chiu[4]提出了改进方法，被称为减法聚类法. 在减法聚类算法中，聚

2002-01-05 收稿.
* 广西自然科学基金的资助项目(桂科基 0141034).

类中心的候选集为数据点，而非网格点，计算量与数据点成简单的线性关系，且与考虑的问题的维数无关. 虽然山峰—减法聚类有聚类速度快、简单等特点，但这种算法目前都存在着聚类不彻底，基类合并不完全，对数据输入顺序的依赖性较强等缺点.

针对目前聚类算法中普遍存在的问题，本文提出一种山峰—减法聚类神经网络的方法，该方法将神经网络与数据集的密度指标有机地结合起来，利用数据集的密度指标对基类进行合并，而神经网络的规模是由数据集的密度指标确定，它的主要优点是对于工程应用中的大样本集分类和重叠数据的模式分类问题，显得非常有效.

1　山峰—减法聚类法

设 M 维空间的 n 个数据点为 $(x_1, x_2, \cdots, x_n)$，不失一般性，假定数据点已归一化到一个超立方体中. 由于每个数据点都是聚类中心的候选者，因此，数据点 x_i 的密度指标可定义如下：

$$D_i = \sum_{j=1}^{n} \exp\left(-\frac{\| x_i - x_j \|^2}{(\delta_a/2)^2}\right), \tag{1}$$

其中 δ_a 是一正数. 特别地，如果一个数据点有多个邻近的数据点，则该数据点具有高密度值，半径 δ_a 定义了该点的一个邻域，半径以外的数据点对该点的密度指标贡献甚微. 在计算出每个数据点的密度指标后，选择具有最高密度指标的数据点为第一个聚类中心，令 x_{c1} 为选中的点，D_{c1} 为密度指标，其余每个数据点的密度指标可有下面公式进行修正：

$$D_i = D_i - D_{c1}\exp\left[-\frac{\| x_i - x_{c1} \|^2}{(\delta_b/2)^2}\right], \tag{2}$$

常数 δ_b 通常大于 δ_a，以避免出现距离相近的聚类中心，在文献[5]中建议，一般取：$\delta_b = 1.5\delta_a$. 修正了每个数据点的密度指标后，选定下一个聚类中心 x_{c2}，再次修正数据点的所有密度指标. 该过程不断重复，直至产生足够多的聚类中心.

由此可见，数据点的密度指标计算公式可用来表示数据集在数据空间中集中的地方，最高密度指标表示一个数据点有多个邻近的数据点.

2　聚类神经元新模型

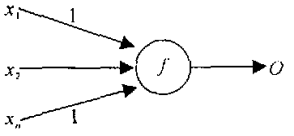

图 1　聚类神经元新模型

图 1 是本文提出的山峰—减法聚类神经元模型，它是一多输入单输出的非线性元件，其中：

$$O = f(x, \delta) \tag{3}$$

是神经元的输出，$f(\cdot)$ 是神经元的激励函数，x 是输入的矢量，δ 是一个可调节的参数，在对神经元进行训练时可以调节这个参数以改变激励函数 $f(\cdot)$，且权矢量恒取 1，使它与待解问题相适应，这是聚类神经元模型与一般 $M-P$ 模型的主要区别.

在图 1 的模型中，结合聚类法的特点，我们给出一种激励函数选取的方法：它由若干个简单的基函数 φ_j 的加权和来表示，记为：

$$f(x, \delta) = \sum_{j=1}^{n} a_j \varphi_j(x, \delta). \tag{4}$$

其中 $a_j = 1 (j = 1, 2, \cdots, n)$，$\varphi_j(x, \delta)$ 是聚类神经元的第 j 个基函数，δ 是参变数，使得激励函数 $f(x, \delta)$ 可调，一般根据先验知识确定. 在聚类神经元中，我们以任意一数据点处密度指标函数

为基函数：

$$\varphi_j(x,\delta_a)=\exp\left(-\frac{\|x-x_j\|^2}{(\delta_a/2)^2}\right), \tag{5}$$

那么，图 1 的神经元的输出：

$$O=f(x,\delta)=\sum_{j=1}^{n}\exp\left(-\frac{\|x-x_j\|^2}{(\delta_a/2)^2}\right), \tag{6}$$

很容易看出，在(4)中，若基函数只有一项，即 $j=1$，这时基函数选取为(5)式，此时，变成形如 M-P 模型中的激励函数的指数函数，所以，M-P 模型是激励函数可调的聚类神经元的特例.

3 聚类神经元学习算法

下面以图 1 的的神经元为例，介绍其自适应调整算法：

步骤 1：(归一化处理)

设 $x_1,x_2,\cdots,x_n$ 分别为输入聚类样本的 n 个坐标，对输入的样本归一化到一个超立方体中，每个数据点都是聚类中心的侯选者.

步骤 2：(计算数据点处的密度指标)

$i=1$

while($i\leqslant n$)

{对任意给定 $\delta_a>0$，计算第 i 个坐标处的数据密度指标：

$$O_i=\sum_{j=1}^{n}\exp\left(-\frac{\|x-x_j\|^2}{(\delta_a/2)^2}\right)$$

$i=i+1$;}

依次输出 $O_i(i=1,2,\cdots,n)$.

步骤 3：(调整激励函数的参数)

根据人们的先验知识，选取参数 δ_b 通常大于 δ_a，以避免出现距离相近的聚类中心，通常选取 $\delta_b=1.5\delta_a$.

步骤 4：(选取第一个聚类中心)

在步骤 2 结果基础上，选取：$O_{c1}=\max\{O_i|i=1,2,\cdots,n\}$. 为第一个聚类中心.

步骤 5：(调整激励函数的基值)

置 x_{c1} 为选中的坐标点，O_{c1} 为其密度指标. 那么每个坐标点的基函数值可用下式调整：

$$O_i=O_i-O_{c1}\exp\left(-\frac{\|x-x_{c1}\|^2}{(\delta_b/2)^2}\right).$$

步骤 6：修正了每个坐标点的密度指标后，选定下一个聚类中心坐标 x_{c2}，重复步骤 2 ～ 步骤 6，直至产生足够多的聚类中心，就可完成对聚类神经元的学习.

4 结束语

本文提出一种激励函数可调的聚类神经元模型，在这种模型中，激励函数中参数可调，由这种神经元构成的人工神经网络模型比目前常用的网络模型有更多的自由度；其激励函数的本质是数据密度指标函数，把神经网络与聚类分析法有机结合起来，本文的最大贡献给目前的聚类分析法提供了一神经网络新模型，它为作者进一步深入研究聚类神经网络的系统逼近理

(下转第 154 页)

组成特性，设计一种时间复杂度为$O(n+n\log_2 m)$的排序算法，m为原始输入数据序列中有序/逆有序的子序列个数，$1 \leqslant m \leqslant n/2$。此排序时间复杂性结果与输入数据的概率分布假设无关。在最坏情形下，本文的排序算法的时间复杂度与现有的排序方法相同。但是，由于在大多数情形下m是小于n的，所以本文给出的排序算法在工程实践中将更有优势。

本文算法的设计思想和冒泡排序、基数排序算法的设计思想有着异曲同工之妙——它们都来源于对客观事物和现实生活的观察与思考。

下一步的工作是将本文的算法并行化，并探讨其在各种并行计算环境下的实现。

参考文献

1 Knuth D E. The art of computer programming: sorting and searching, vol 3. 2nd ed. Reading, Mass.: Addison-Wesley, 1998.

2 苏德富，钟 诚. 计算机算法设计与分析. 北京：电子工业出版社，2001.

3 陈国良. 并行算法的设计与分析. 北京：高等教育出版社，1994.

（责任编辑：黎贞崇）

（上接第150页）

论打好基础，它将在数据挖掘、综合评判等方面有着重要的应用.

参考文献

1 Yager R R, Filve D P. Approximate clustering via the mountain method. IEEE Transactions on Systems Man and Cybernetics, 1994, (24): 1279～1284.

2 Dunn J C A. Fuzzy Relative of the ISODATE process and its use in detecting compact well-separated clusters. Cybern, 1973, 3(3): 32～57.

3 Ng R, Han J E. Efficient and effective clustering methods for spatial data Minning. Department of Computer Science, University of Birtish Columbia, 1994.

4 Chiu S L. Fuzzy model identification based on cluster estimation. Journal of Intelligent and Fuzzy Systems, 1994, 2(3): 54～58.

5 张智星，孙春在，水谷英二[日]著. 神经一模糊和软计算. 西安：西安交通大学出版社，2000.

（责任编辑：黎贞崇）