指定类数下仿射传播聚类的快速算法①

王开军 郑 捷 (福建师范大学 数学与计算机学院 福建 福州 350108)

摘 要: 针对 Science 杂志上提出的仿射传播（Affinity propagation）聚类产生指定类数的聚类结果时效率较低的问题，提出了基于多网格策略的快速算法。该算法采用多网格搜索策略来减少调用仿射传播算法的次数，改进偏向参数的上界以缩小搜索范围。新方法大幅度地提高了仿射传播聚类在指定类数下的速度性能。实验结果表明新方法十分有效，在运行时间上比现有方法减少了 22%－90%。

关键词: 快速聚类; 指定类数的聚类; affinity propagation

Fast Algorithm of Affinity Propagation Clustering under Given Number of Clusters

WANG Kai-Jun, ZHENG Jie

(School of Mathematics and Computer Science, Fujian Normal University, Fuzhou 350108, China)

Abstract: To enhance the low efficiency of producing the clustering result with given number of clusters by Affinity propagation clustering (AP), the fast multi-grid searching method is proposed. The proposed method uses multi-grid searching to reduce the calling times of AP, and improves the upper bound of preference parameter to reduce the searching scope, so that it can largely enhance the speed performance of AP under given number of clusters. The experimental results show that the proposed method is very effective and reduces the running time by 22%-90%, compared with the existing method.

Keywords: fast clustering; clustering under given number of clusters; affinity propagation

聚类方法在工业、农业、金融、科技等诸多行业中有着广泛的应用。与 k-中心点聚类算法相比，仿射传播聚类(Affinity propagation clustering, AP)的长处体现在处理大类数和大数据集的聚类问题时聚类质量高且运算速度快[1,2]。AP 算法的基本思路是：初始时将数据集的所有 N 个样本点都视为候选的类代表(聚类中心)，为每个样本点 x_i 建立与其他样本点 x_j 之间的吸引程度和归属程度的信息，例如可以将负的两个点之间距离 $-\|x_i-x_j\|^2$ 设想为点 x_i 对点 x_j 的吸引度或点 x_j 对点 x_i 归属度；再通过一个迭代循环为每个候选类代表从其它数据点搜集证据(吸引度和归属度)，不断地进行证据的搜集和传递(亦称为消息传递)；这样，处于聚类中心处的数据点对其它数据点的吸引力之和较大，成为类代表的可能性也大；而处于聚类边缘处的数据点对其它数据点的吸引力之和比较小，成为类代表的可能性也小；当迭代过程收敛时，K 个类代表随之确定，再将每个样本点分配给最近的类代表所属的类，产生 K 个聚类。

另一方面，AP 算法不能将指定类数 K 作为算法的输入参数，以产生 K 个聚类的聚类结果。要获得指定类数(K 个类)的聚类结果，一般是采用搜索的方法，多次调用 AP 算法以检查是否获得 K 个聚类。不幸的是，仿射传播聚类方法在指定类数下的工作效率较低，搜索过程中调用 AP 算法的次数往往较多(耗时较多)，从而其单次运算速度快的效果被严重削弱。这种低效率会严重制约 AP 方法的推广和应用。因此，提高 AP 算法产生指定类数的效率，对于发展和完善 AP 算法在指定类数情况下的速度优势具有十分重要的意义。本

① 基金项目:福建省教育厅项目(JA09043)
收稿时间:2009-11-03 收到修改稿时间:2009-12-06

文针对此问题，提出了基于多网格策略的快速算法，以提高仿射传播聚类产生指定类数的聚类结果的速度性能。本文方法的 Matlab 程序可由文献[3]获得。

1　多网格快速方法

先分析 AP 算法产生指定 K 个聚类的机制，讨论现有搜索方法的不足，再设计产生指定类数聚类结果的快速方法。

AP 算法以 N 个数据点之间的相似度组成的 N×N 相似度矩阵 S 为工作基础，例如当使用欧式距离时，数据点 x_i 和 x_j 之间的相似度是 $S(i,j)=-\|x_i-x_j\|^2$；所有的数据点在算法开始时都被视为潜在的类代表。矩阵 S 对角线上的 S(i,i)被看成是偏向性参数 p(i)，表示数据点 xi 被选作聚类中心的倾向性。给 p(i)赋予较大的值将增大点 xi 被选作聚类中心的可能性，并影响最终类代表的产生。可以看出，参数 p 影响 AP 算法输出的聚类数目。然而，对给定的数据集，p 取何值能产生指定类数的聚类结果却是未知的，这是由于 p 与输出的聚类个数之间没有一一对应关系。

因此，要获得指定类数(例如 K 个聚类)的聚类结果，一般只能采用搜索的方法。现有的搜索方法由粗搜索(缩小参数 p 的取值范围)和精搜索(找到 K 个聚类)阶段组成，其步骤是：首先依据数据集的相似度矩阵 S 确定参数 p 的取值范围(即对应最大类数 N 的 p 值上界 p_{up} 和对应最小类数 1 的 p 值下界 p_{low})，建立 p 值的搜索网格(即$\{p_{up}-dp\times10^{-4}, p_{up}-dp\times10^{-3}, p_{up}-dp\times10^{-2}, p_{up}-dp\times10^{-1}\}$，其中 $dp=p_{up}-p_{low}$)；然后，令参数 p 依次取搜索网格中的值并调用 AP 算法产生相应的 m 个聚类的聚类结果；同时对 m 进行检查，若 m 等于 K 则搜索过程结束，若 m 小于 K 则改进 p 值下界(plow 提高到搜索网格中的值)，粗搜索阶段结束；再进入精搜索阶段，在 p 值的上下界内采用二分查找法(折半查找法)进行搜索，找出指定类数的聚类结果。

现有搜索方法的不足有：粗搜索阶段在缩小 p 的取值范围时没有进行 p_{up} 的改进工作；对搜索网格进行全搜索在许多情况下有浪费现象，例如对于指定类数很小的情况，网格点 $p_{up}-dp\times10_{-1}$ 对应于较小的类数，从而对其他网格点(对应于较大的类数)的搜索多有浪费；对于指定类数很大的情况，网格点 $p_{up}-dp\times10^{-4}$ 对应于较大的类数，从而对 $p_{up}-dp\times10^{-1}$ 网格点的搜索多有浪费。

针对上述不足，对粗搜索阶段的搜索策略进行重新设计。在缩小 p 取值范围的粗搜索过程中，同时进行 p_{low} 和 p_{up} 的改进工作：在每一次 p 值网格点的搜索时检查 m 是否小于 K，若是则改进 p 值下界，否则若 m 大于 K 则改进 p 值上界。可以看出，p_{up} 的改进是利用寻找更好的 p_{low} 过程实现的，并没有增加调用 AP 算法或搜索的次数。因此，p_{up} 的改进有利于提高精搜索阶段的搜索效率。针对指定类数大小的不同情况，分别制订 p 值的搜索网格，在指定类数不很大时减小搜索网格的大小。这种多网格搜索策略可以减少搜索次数，提高搜索效率。

在现有的搜索方法的基础上，获得指定类数的聚类结果的快速方法主要在多网格搜索策略和改进 p 值上界方面进行设计，其具体内容设计如下：

(1)粗搜索阶段建立多网格搜索策略(替代单一网格搜索策略)

对于指定类数较小(K≤9)的情况，建立 p 值的搜索网格为 $w=\{p_{up}-dp\times10^{-1}\}$；

对于指定类数较大(10≤K≤25)的情况，建立 p 值的搜索网格为 $w=\{p_{up}-dp\times10^{-2}, p_{up}-dp\times10^{-1}\}$；

对于指定类数很大(26≤K≤100)的情况，建立 p 值的搜索网格为 $w=\{p_{up}-dp\times10^{-3}, p_{up}-dp\times10^{-2}, p_{up}-dp\times10^{-1}\}$；

对于指定类数超大(K≥101)的情况，建立 p 值的搜索网格为 $w=\{p_{up}-dp\times10^{-4}, p_{up}-dp\times10^{-3}, p_{up}-dp\times10^{-2}, p_{up}-dp\times10^{-1}\}$。

(2)粗搜索阶段新增加对 p 值上界的改进

在参数 p 依次取网格中的值 w_i 并调用 AP 算法产生相应 m_i 个聚类的聚类结果的过程中，对 m_i 进行检查，若 m_i 等于 K 则搜索过程结束，若 m_i 小于 K 则改进 p 值下界(即 plow=wi)，否则改进 p 值上界，即 pup=wi(i=1, 2)(因 w3 和 w4 中 $dp\times10^{-3}$ 和 $dp\times10^{-4}$ 的贡献值太小，忽略不用)。

2　实验结果

本节对多网格快速方法(fastAP)与现有的 AP 方法[1](existAP)进行对比实验，以说明新方法的性能。实验中 AP 算法的参数设置为最大循环次数 2000、收敛次数 200 及阻尼因子 0.9[1]。关于样本(数据点)x_i 和 x_j 之间的相似性测度，对一般数据采用欧式距离，

对基因表达数据Colon采用Pearson相关系数P(i,j)，并转换P(i,j) ∈ [-1,1]为正的P(i,j) ∈ [0,1](值越大表示两个样本相距越远)[4]。实验中采用的5个数据集是Iris[5]、Colon[6]、5k8close[7]、22k10far[4]和FaceImage[1]，其中5k8close和22k10far是模拟数据集，其余的是真实数据集。

表1是fastAP和existAP获得指定类数(已知的正确类数)的聚类结果的运行情况，其中次数表示fastAP或existAP调用AP算法的次数，时间表示从开始搜索指定类数到搜索结束的时间（不包括建立相似度矩阵的时间），是搜索过程在同一PC机(CPU Intel Pentium Dual 2.20GHz，1GB内存)上的运行时间(秒)，时间节省表示fastAP比existAP少花费的搜索时间占existAP搜索时间的百分比。从中可以看出，在调用AP算法的次数上fastAP比existAP节省1-3次，在运行时间上fastAP比existAP减少22% - 90%。这些结果验证了本文方法能大幅度提高仿射传播聚类在指定类数下的速度性能。

表1 fastAP和existAP获得指定类数的聚类结果的运行情况

数据集	类数	样本数	维数	existAP		fastAP		时间
				次数	时间	次数	时间	节省%
Iris	3	150	4	4	4.80	1	0.77	83.9
Colon	4	2000	62	4	2652.76	1	261.21	90.1
5k8close	5	1000	8	4	312.44	1	79.74	74.8
22k10far	22	790	10	2	307.98	1	46.31	84.9
FaceImage	100	900	50′50	14	1669.32	12	1298.91	22.1

3 结论

为了改进仿射传播聚类产生指定类数的聚类结果时较低的工作效率，提出了基于多网格策略的快速算法。新方法通过多网格搜索策略和p值上界的改进大幅度提高了仿射传播聚类在指定类数下的速度性能。此外，需注意的是当指定类数大于100时，本文方法所提高的速度性能比较有限。

参考文献

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

2 王开军,李健,张军英,涂重阳. 半监督的仿射传播聚类.计算机工程, 2007,33(23):197－198,201.

3 fastAP:[2009-12-8]. http://www.mathworks.com/matlabcentral/fileexchange/authors/24811.

4 王开军,张军英,李丹,张新娜,郭涛.自适应仿射传播聚类.自动化学报, 2007,33(12):1242－1246.

5 Blake CL, Merz CJ. UCI repository of machine learning databases (University of California, Irvine, 1998). [2009-12-8].http://mlearn.ics.uci.edu/MLRepository. html

6 Alon U, Barkai N, Notterman DA, Gish K, Ybarra S, Mack D, Levine AJ. Broad patterns of gene expression revealed by clustering of tumor and normal colon tissues probed by oligonucleotide arrays. Proc. Natl. Acad. Sci. USA, 1999,96(12):6745－6750.

7 Strehl A. Relationship-based Clustering and Cluster Ensembles for High- dimensional Data Mining [Ph.D. Thesis]. Austin: University of Texas at Austin, 2002.