基于数据集压缩的聚类算法性能优化研究

赵延龙，滑 楠
（空军工程大学 信息与导航学院，西安 710077）

摘 要：针对目前聚类算法对大数据集的聚类分析中存在时间花费过大的问题，提出了一种基于最近邻相似性的数据集压缩算法。通过将若干个相似性最近邻的数据点划分成一个数据簇并随机选择簇头构成新的数据集，大大缩减了数据的规模。然后分别采用 K-means 算法和 AP 算法对压缩后的数据集进行聚类分析。实验结果表明，压缩后的数据集与原始数据集的聚类分析相比，在保证聚类准确率基本一致的前提下，有效降低了聚类的花费时长，提高了算法的聚类性能，证明了该数据集压缩算法在聚类分析中的有效性和可靠性。

关键词：聚类；数据压缩；聚类性能
中图分类号：TP301.6 **文献标志码**：A **文章编号**：1001-3695(2018)05-1450-04
doi:10.3969/j.issn.1001-3695.2018.05.037

Research on optimization of clustering algorithm performance based on dataset compression

Zhao Yanlong, Hua Nan
(*College of Information & Navigation, Air Force Engineering University, Xi'an* 710077, *China*)

Abstract: This paper proposed a data set compression algorithm based on nearest neighbor similarity to solve the problem that the clustering algorithm is too expensive in the large data clustering analysis. It greatly reduced the size of the data set by dividing several data points nearest to each other into a data cluster and forming new data set with randomly selecting cluster heads. Then it used the K-means algorithm and the AP algorithm to cluster the compressed datasets respectively. The experimental results show that compared with the original data set clustering analysis, the compressed dataset can reduce the time of clustering and improve the clustering performance of the algorithm in the case of the clustering accuracy is basically the same, which proves that the validity and reliability of data set compression algorithm in cluster analysis.

Key words: clustering; data compression; clustering performance

0 引言

聚类(clustering)是指把本身并没有类别标签的样本数据按照“物以类聚”的思想划分成不同的组别，每一组样本数据所组成的集合叫做簇。它是指依据一定的规则，计算各数据点之间的相似程度，通过将相似程度较高的数据聚集起来划归为一类来实现的。典型的聚类算法有：a)基于划分的聚类(如 K-means 算法[1~3])，该类算法需要预先确定聚类数目，对初值较敏感，不适合发现非凸面形状的数据集；b)基于密度的聚类(如 DBSCAN 算法[4~6])，该算法当数据点空间密度分布不均匀、聚类间距离相差比较大时，聚类效果较差；c)基于层次的聚类(如 BIRCH[7,8]算法)，该类算法数据的输入顺序对聚类结果有一定的影响，而且必须界定聚类停止的具体时间，从而得到某个数量的分类。AP(affinity propagation)算法[9~13]是 2007 年由 Frey 等人在 Science 杂志上提出的一种新的聚类算法。该类算法不需要预先确定聚类的数目，而是将所有的数据点视为潜在的聚类中心，通过不同数据点之间的信息(归属度和吸引度)传递，迭代计算出最佳的聚类中心，最终实现聚类。然而利用这些算法在对大数据集进行聚类分析时，时间花费往往过大，降低了算法的聚类性能，限制了聚类算法的适用范围，已远远满足不了大数据环境下数据挖掘[14]的需求。

因此，本文从原始数据的角度出发，通过提出基于最近邻相似性的数据集压缩算法，对聚类算法的性能进行优化研究。

1 聚类算法简介

1.1 K-means 算法

K-means 算法的基本思想是给定 N 个数据点，将其分成 k 个簇。随机初始化簇的质心，通过数据点到最近邻质心的移动，最终使得簇内的相似性尽可能高，而簇间的相似性尽可能低，因此该算法需要预先指定聚类簇的个数。现给出 N 个数据点组成一个数据集 $X=\{x_1,x_2,\cdots,x_N\}$，其中 $x_i=\{x_{i1},x_{i2},\cdots,x_{id}\}$ 代表一个数据对象，d 表示数据对象的维度。

K-means 算法的基本步骤如下：

a)确定初始质心。指定聚类簇的个数 k，并在数据集 X 中随机地选择 k 个种子点分别作为 k 个簇的初始质心(聚类中心)。

b)簇的划分。计算数据集 X 中每个元素到质心点的距离(通常计算欧氏距离)，假如数据点 x_i 距离质心点 x_j 的距离最近，那么 x_i 属于 x_j 点群，最终构成 k 个簇。

收稿日期：2017-01-10；**修回日期**：2017-02-22

作者简介：赵延龙(1992-)，男，河北邢台人，硕士研究生，主要研究方向为现代通信理论与技术(1241492516@qq.com)；滑楠(1974-)，男，教授，硕导，主要研究方向为现代通信理论与技术.

c)更新质心。计算各个簇平均值作为新的质心,重复步骤 b),直到每个簇中的数据点不再移动为止。

d)算法结束。每个簇中的数据点表示一类点集,最终完成聚类。

1.2 AP 算法

AP 算法的基本思想是给定 N 个数据点,将每个数据点看做潜在的聚类中心,设置相同的初始偏向度取值,通过两类信息即归属度和吸引度的信息传递,确定最终的聚类中心,完成聚类。

AP 聚类算法的基本步骤如下:

a)计算相似度矩阵 S。各数据点之间相似度值 $s(i,k)$ 构成一个 $N\times N$ 的相似度矩阵 S,$s(i,k)$ 按如下方法计算。

$$s(i,k)=\begin{cases}-\|x_i-x_k\| & i\neq k\\ p(k) & i=k\end{cases} \tag{1}$$

其中:当 $i=k$ 时,$p(k)$ 表示初始偏向度取值,即数据点 k 最终作为类代表点的偏向程度大小。

b)信息的相互传递。AP 聚类算法依据两类信息 $r(i,k)$ 和 $a(i,k)$ 的相互传递来完成聚类。其中:$r(i,k)$ 表示数据点 x_k 适合作为数据点 x_i 类代表点程度的大小;$a(i,k)$ 表示数据点 x_k 把数据点 x_i 当做类代表点程度的大小。两类信息传递分别按如下方法。

$$r(i,k)=s(i,k)-\max_{k'\neq k}\{a(i,k')+s(i,k')\} \tag{2}$$

$$a(i,k)=\begin{cases}\min\{0,r(k,k)+\sum\limits_{i'\notin\{i,k\}}\max\{0,r(i',k)\} & i\neq k\\ a(k,k)=\sum\limits_{i'\neq k}\max\{0,r(i',k)\} & i=k\end{cases} \tag{3}$$

经验证,式(3)过程会存在一定的振荡,收敛速度较慢,因此通常需要引入阻尼因子 $\lambda(\lambda\in(0,1))$,即满足

$$\begin{aligned}r(i,k)^{t+1}&=(1-\lambda)\times r(i,k)^{t+1}+\lambda\times r(i,k)^{t}\\ a(i,k)^{t+1}&=(1-\lambda)\times a(i,k)^{t+1}+\lambda\times a(i,k)^{t}\end{aligned} \tag{4}$$

c)确定聚类中心。如果 x_k 要作为 x_i 的类代表点,k 要满足

$$\arg\max\{a(i,k)+r(i,k)\} \tag{5}$$

式(5)的含义是,当 i 的值一定时,使得 $a(i,k)+r(i,k)$ 取得最大值的 k 的取值。

d)终止迭代。当达到规定最大迭代次数或者经过多次迭代聚类中心未发生变化,此时完成聚类。

2 最近邻相似性对聚类的影响分析

本文所提出的基于最近邻相似性的数据集压缩算法的基本思想是对于给定的数据集 X,先将两两最相似的数据点归为一类构成数据簇,然后随机选取簇头并参加接下来的聚类过程,从而有效地降低原始数据集的规模,进而减少聚类分析的时间开销,从某一意义上提高了算法的聚类性能,在大数据处理领域具有一定的理论研究和工程应用价值。

2.1 影响指标

2.1.1 有效性指标

为了验证本文所提数据集压缩算法对聚类分析的有效性,按如下方法设置有效性评价指标。

$$\text{effect}_X=\frac{|f(X')|}{|X'|} \tag{6}$$

其中:X'表示由数据集 X 中最近邻相似性数据簇$(x_i,x_j)$$(i,j\in\{1,2,\cdots,N\}$且$i\neq j)$构成的簇集合;$f(X')$表示集合 X'中的同类数据簇(簇中元素具有相同的类标签)所构成的集合;$|X'|$和$|f(X')|$分别表示集合中元素的个数;effect_X 表示对数据集 X 的有效性度量指标,该指标越大表示数据集 X 运用最近邻相似性的数据压缩方法的有效性和可靠性越高。

2.1.2 准确率指标

本文依据目前数据挖掘领域中广泛承认的聚类性能评价指标,选择准确率(prec$\in(0,1)$)作为聚类算法对数据集聚类分析优劣的评价指标:

$$\text{prec}=\frac{2\times(TP+FN)}{N\times(N-1)} \tag{7}$$

其中:TP 表示同一类别的数据点划分到同一簇中的个数;FN 表示不同类别的数据点划分到不同簇中的个数;N 表示数据集的规模大小;prec 指标反映正确聚类的数据点对数占所有数据点对数的比率,该指标越大,表示聚类准确率越高,聚类性能越好。

2.1.3 效率指标

聚类算法花费时长(time)的多少直接反映该算法的效率大小。

Time 指标表示算法结束时所花费的时长,反映算法的时间复杂度。该指标越小,表示算法时间复杂度越低,聚类性能越好。

2.2 仿真与分析

本文选取 UCI 机器学习数据库中四种标准测试数据集 4k2-far[15]、wine[16]、iris[17] 和 leuk72-3k[18] 分别进行研究。各数据集的基本特征如表 1 所示。

表 1　数据集基本特征

数据集	样本数	维数	类别数
4k2-far	400	2	4
wine	178	13	3
iris	150	4	3
leuk72-3k	72	39	3

仿真实验硬件环境:Intel 3.2 GHz,内存 4 GB;操作系统:Microsoft Windows XP。

表 1 中列出的四种测试数据集,无论样本数、维数以及类别数均存在一定的差别,从而使得数据集更具有代表性。利用 MATLAB 2015b 软件对上述四种数据集分别进行仿真研究,并计算有效性指标;分别采用 K-means 算法和 AP 算法对各原始数据集进行聚类分析,并计算准确率指标。最终得到实验结果如表 2、3 所示。

表 2　数据集有效性

数据集	4k2-far	wine	iris	leuk72-3k
$\lvert X'\rvert$	400	178	150	72
$\lvert f(X')\rvert$	400	169	143	69
effect_x	1	0.949	0.953	0.958

表 3　聚类准确率

数据集	4k2-far	wine	iris	leuk72-3k
K-means	1	0.946	0.874	0.947
AP	1	0.890	0.899	0.931

从表 2、3 中可分析得出,对于上述给定的任意数据集,其有效性都大于或等于该数据集通过 K-means 算法和 AP 算法进行聚类分析最终得到的准确率,并且经验证其他标准数据集同样符合这个规律。由此分析得出,基于最近邻相似性的数据集压缩算法对于聚类性能的贡献率大于聚类算法本身对于聚类性能的贡献率。因此本文所提出的压缩算法在对于大数据

聚类分析中具有一定的潜力,是一个非常有价值的研究课题。

3 基于最近邻相似性的数据压缩算法

3.1 基本思想

从 2.2 节的仿真实验结果分析得出,对于数据集中相似性程度较大的数据点,拥有相同的类别标签的概率也相对较大。因此本文所提出的基于最近邻相似性的数据集压缩算法的基本思想是将数据集中相似性最近邻的若干个数据点划归为一个数据簇,并随机选择簇头,即数据集的类代表点,然后再利用其他聚类算法对由这些簇头所构成的新的数据集进行聚类分析,从而有效减小原始数据集的规模,提高聚类分析的效率。

3.2 实现流程

基于最近邻相似性的数据集压缩算法的实现流程如图 1 所示。

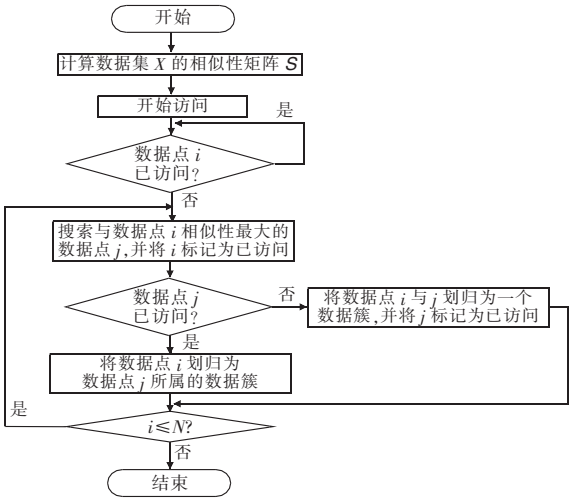

图 1 基于最近邻相似性的数据集压缩算法流程

上述数据压缩算法的关键步骤是对数据集相似性矩阵 S 的计算。数据点之间相似性的度量一般采取基于距离的度量方法,如欧氏距离。当数据集的维度较低时,基于欧氏距离的相似性度量方法在数据集聚类分析中可以达到理想的效果,然而随着维度的逐渐增加,数据点之间的对比性将被逐渐削弱,因此在对较高维的数据集进行聚类分析时需要采用合适的相似性度量方法。

为消除不同维度的量纲大小对相似性度量的影响,需要对数据集进行标准化处理。

对于给定的由 N 个数据点构成的数据集 $X=\{x_1,x_2,\cdots,x_N\}$,其中的数据点 $x_i=\{x_{i1},x_{i2},\cdots,x_{id}\}$($d$ 表示数据点的维度)按如下方法进行标准化。

$$x'_{ij}=\frac{x_{ij}-\min(X_j)}{\max(X_j)-\min(X_j)} \tag{7}$$

其中:x_{ij} 和 x'_{ij} 分别表示数据标准化前后第 i 个数据点的第 j 维上的度量值;X_j 表示数据集 X 中 N 个数据点的第 j 维数据度量值所构成的集合;$\max(X_j)$ 和 $\min(X_j)$ 分别表示数据集 X 中第 j 维数据度量的最大值和最小值。通过式(7)可以将数据集 X 中的所有数据值转换到[0,1]内。

当数据点的维度逐渐增大时,基于欧氏距离的相似性度量方法存在较大的偏差,即对于欧氏距离相同的若干个点不一定隶属于同一分类,亦即相似性不一定较大,这就是文献[19~21]中提到的维度灾难。因此本文在计算相似性矩阵时引入衰减系数 $e^{-|\Delta|}$,如下:

$$S_{ij}=-\sqrt{\sum_{k=1}^{d}e^{-|x_{ik}-x_{jk}|}\times(x_{ik}-x_{jk})^2} \tag{8}$$

其中:$\Delta=-|x_{ik}-x_{jk}|(k\in\{1,2,\cdots,d\})$ 表示数据集 X 中数据点 x_i 与 x_j 的第 k 维度的差值。当 Δ 较大时,相应的衰减系数 $e^{-|\Delta|}$ 迅速减小,从而降低该维度的数据度量值过大时对整个相似性度量的影响。

3.3 算法实现

依据上述的基本思想和实现流程,基于最近邻相似性数据集压缩算法的实现如下所示。

1)数据集 X 归一化处理

```
for(i=1;i<N;i++)
    for(j=1;j<d;j++)
        x'_ij=(x_ij-min(X_j))/(max(X_j)-min(X_j));
    end
end
```

2)计算相似性矩阵 S

```
for(i=1;i<N;i++)
    for(j=1;j<N;j++)
      S_ij=-[sum(e^(-|x_ik-x_jk|)(x_ik-x_jk)^2)]^0.5;
    end
end
```

3)基于最近邻相似性的数据集压缩算法

```
int result,count=0;
for(i=1;i<N;i++)
   if(flag(i)==0)数据点 i 未访问//
     t=find(S(i,:)==max(S(i,:)));搜索最近邻相似点//
     if(flag(t)==0)
        count++;
        result(count)=[i,t];将 i 和 t 归为一类//
        flag(t)=1;标记已访问//
     else
        locate=find(result==t);搜索 t 的所属类//
        result(locate)=i->{t};将 i 归为 t 所属的类//
     end
     flag(i)=1;标记 i 已访问//
   end
end
```

4 仿真验证

本章中利用 MATLAB 2015b 仿真软件对 2.2 节中的四种标准数据集采用第 3 章中提出的数据集压缩算法进行仿真实验,结果如表 4 所示。

表 4 数据集压缩率

数据集	4k2-far	wine	iris	leuk72-3k
压缩前规模	400	178	150	72
压缩后规模	151	54	51	24
压缩率	0.378	0.303	0.340	0.333

其中:压缩前(后)规模表示数据集压缩前(后)元素的个数;压缩率表示压缩后的规模与压缩前相比。

利用 K-means 算法和 AP 算法对压缩后的数据集进行聚类分析,最终得到仿真结果如图 2~5 所示。

从图 2、3 的实验结果分析得出,在利用 K-means 算法对数据集 4k2-far 和 leuk72-3k 进行仿真分析时,在数据集压缩前后,聚类准确率都保持不变;在利用 AP 算法对数据集 4k2-far 和 wine 进行仿真分析时,前者的聚类准确率在数据集压缩前后保持不变,后者的聚类准确率有一定程度的提高。

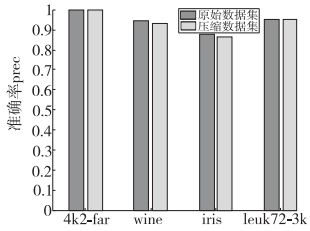

图2　K-means算法对四种数据集的聚类准确率分析

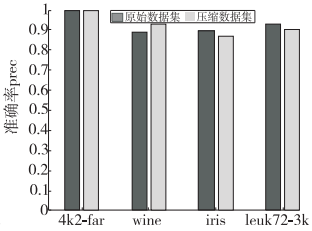

图3　AP算法对四种数据集的聚类准确率分析

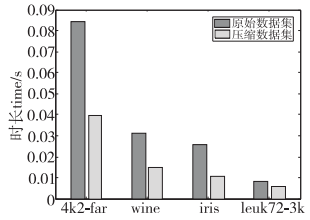

图4　K-means算法对四种数据集的花费时长分析

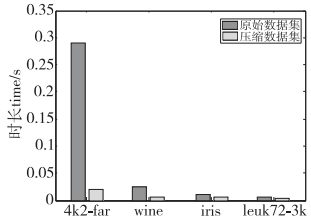

图5　AP算法对四种数据集的花费时长分析

从图 4、5 的实验结果分析得出,利用 K-means 算法和 AP 算法对四种数据集压缩前后的聚类效率均有很大的优化,表现在花费时长有所减少,如表 5 所示。

表 5　压缩前后聚类效率优化程度

数据集	4k2-far	wine	iris	leuk72-3k
K-means	0.468	0.481	0.420	0.670
AP	0.069	0.276	0.532	0.651

从表 5 可分析得出,不同的聚类算法对不同的数据集压缩前后聚类花费时长的优化程度各不相同:对于 K-means 算法,优化程度最高的是数据集 iris,最低的是数据集 leuk72-3k,其优化程度分别为 0.420 和 0.670;对于 AP 算法,优化程度最高的是数据集 4k2-far,最低的是数据集 leuk72-3k,其优化程度分别为 0.069 和 0.651。

通过对上述实验结果的分析,在一定程度上验证了基于最近邻相似性的数据压缩算法对于聚类分析的有效性和可靠性,然而也存在一些不足。从图 2、3 的结果中同样可以得出,在利用 K-means 算法对数据集 wine 和 iris 进行仿真分析以及利用 AP 算法对数据集 iris 和 leuk72-3k 进行仿真分析时,压缩后的聚类准确率与压缩前相比有一定程度的降低。经验证这是因为 wine、iris 以及 leuk72-3k 三种数据集的规模较小,分别为 178、150 和 72,当采用本文的压缩算法对其进行压缩时,改变了这三种数据集的空间结构,最终导致在利用 K-means、AP 两种算法进行聚类分析时准确率有一定程度的波动,表现为降低。当数据集的规模较大时(如本文数据集 4k2-far 的规模为 400)压缩前后的聚类准确率相对稳定保持一致。

5　结束语

本文通过利用 K-means 和 AP 两种聚类算法对数据集 4k2-far、wine、iris 和 leuk72-3k 中的最近邻相似性数据点与聚类准确率及聚类效率之间的关系进行研究,提出了一种基于最近邻相似性的数据集压缩算法。利用该压缩算法对上述四种数据集进行仿真实验,并采用 K-means 和 AP 两种聚类算法对压缩后的数据集进行验证。实验结果表明,当数据集的规模较大时,该数据集压缩算法可以有效降低原始数据集的规模,大大地提高了聚类分析的效率,并且聚类准确率相对稳定保持一致,从而证明了该压缩算法的有效性和可靠性。然而当数据集的规模较小时,利用压缩算法对数据集进行分析会在一定程度上改变原始数据集的空间结构,使得最终的聚类准确率有一定的偏差,因此基于最近邻相似性的数据集压缩算法更适合于针对大数据集的研究分析。

参考文献:

[1] Lee S S, Lin J C. An accelerated K-means clustering algorithm using selection and erasure rules[J]. Journal of Zhejiang University-SCIENCE C(Computers & Electronics),2012,13(10):761-768.

[2] 李洪成,吴晓平,陈燕. Map Reduce 框架下支持差分隐私保护的 K-means 聚类方法[J]. 通信学报,2016,37(2):124-130.

[3] 杜辉,王宇平,董晓盼. 采用万有引力定律自动确定类数的 K 均值算法[J]. 西安交通大学学报,2014,48(10):115-119.

[4] Xie Yonghong, Ma Yanhui, Zhou Fang, *et al*. PDBSCAN: parallel DBSCAN for large-scale clustering applications[J]. Journal of Donghua University,2012,29(1):76-79.

[5] 刘淑芬,孟冬雪,王晓燕. 基于网格单元的 DBSCAN 算法[J]. 吉林大学学报:工学版,2014,44(4):1135-1139.

[6] 张晓倩,杨波,王琳,等. 使用 DBSCAN 的 FCM 神经网络分类器[J]. 模式识别与人工智能,2016,29(2):185-192.

[7] 张蓉,钟艳. 基于 BIRCH 算法的模糊集数据库挖掘算法[J]. 科技通报,2014,30(4):47-49.

[8] 张宇. 基于极值特征的雷达侦察数据 BIRCH 聚类方法[J]. 电子设计工程,2016,24(9):15-18.

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

[10] Xu Xinzheng, Ding Shifei, Shi Zhongzhi, *et al*. Optimizing radial basis function neural network based on rough sets and affinity propagation clustering algorithm[J]. Journal of Zhejiang University-SCIENCE C(Computers & Electronics),2012,13(2):131-138.

[11] Jin Ran, Liu Ruijian, Li Yefeng, *et al*. Improved semi-supervised clustering algorithm based on affinity propagation[J]. Journal of Donghua University,2015,32(1):125-131.

[12] 江颉,王卓芳,陈铁明,等. 自适应 AP 聚类算法及其在入侵检测中的应用[J]. 通信学报,2015,36(11):118-126.

[13] 计华,张化祥,孙晓燕. 基于最近邻原则的半监督聚类算法[J]. 计算机工程与设计,2011,32(7):2455-2459.

[14] Niu Dongxiao, Wang Yongli, Ma Xiaoyong. Optimization of support vector machine power load forecasting model based on data mining and lyapunov exponents [J]. Journal of Central South University, 2010,17(2):406-412.

[15] 4k2-far 数据集[DB/OL]. http://download. csdn. net/ detail/u010459260/ 7099691/.

[16] Wine 数据集[DB/OL]. http://archive. ics. uci. edu/ml/datasets/Wine/.

[17] Iris 数据集[DB/OL]. http://archive. ics. uci. edu/ml/datasets/Iris/.

[18] leuk72-3k 数据集[DB/OL]. http://download. csdn. net/ detail/u010459260/ 7099691/.

[19] Lee S H, Yan Sun, Jeong Y S, *et al*. Similarity measure design for high dimensional data[J]. Journal of Central South University, 2014, 21(9):3534-3540.

[20] Yu Guangzhu, Zeng Xianhui, Shao Shihuang. Mining frequent closed itemsets in large high dimension data[J]. Journal of Donghua University,2008,25(4):416-424.

[21] 周勇,卢晓伟,程春田. 非规则流中高维数据流典型相关性分析并行计算方法[J]. 软件学报,2012,23(5):1053-1072.