基于摘要技术的混合模型流数据聚类算法

刘建伟[1] 李卫民[2]

(中国石油大学(北京)自动化研究所 北京 102249)[1] (上海大学计算机工程与科学学院 上海 200072)[2]

摘 要 传统的数据库管理系统和数据查询算法不能很好地支持对流数据的查询已经被广泛认识,因而需要研究新的流数据模式查询算法。提出了一种基于摘要技术的在线快速混合模型流数据聚类算法,该算法为分阶段混合模型聚类过程。算法首先对最初到达的流数据用多维网格结构进行划分,对划分形成的每一个单元进行数据摘要,提取足够的统计信息。对该摘要运行基于模型的贪心聚类算法,聚类形成的混合模型的摘要信息存储在永久摘要数据库中,从而形成初始聚类混合模型;在聚类模型的维持过程中,当不断有流数据到达时,对到达的数据块用多维网格结构进行划分,对划分形成的每一个单元提取足够的摘要信息。对该摘要运行基于模型的贪心聚类算法形成聚类混合模型。在判断是否可以把新到达的模型合并到现有的混合模型中去时,提出了三种合并标准。实验表明,该算法减少了分类误差,其速度也比传统的基于模型的贪心聚类算法大大加快。

关键词 流数据,混合模型,聚类,模式

Synopsis Data Structure Based Mixture Probabilistic Density Data Stream Clustering Approach

LIU Jian-wei[1] LI Wei-ming[2]

(Research Institute of Automation, China University of Petroleum, Beijing 102249, China)[1]

(School of Computer Engineering and Science, Shanghai University, Shanghai 200072, China)[2]

Abstract Many current and emerging applications require support for on-line analysis of rapidly changing data streams. Limitations of traditional DBMSs and data mining in supporting streaming applications have been recognized, prompting research to augment existing technologies and build new systems to manage streaming data and propose new algorithm for mining data stream. A synopsis data structure based mixture probabilistic density data stream clustering approach was proposed, which requires only the newly arrived data, not the entire historical data, to be saved in memory. This approach incrementally updates the density estimate taking only the newly arrived data and the previously estimated density. This method uses three distance metric criteria for judging if merging new arriving component into a component of existing Gaussian mixture model or as a new model is added existing Gaussian mixture model. The experimental results have demonstrated that the algorithm is feasible and fulfill high quality clustering results.

Keywords Data stream, Mixture model clustering, Patterns

最近几年,使用和管理的数据以流数据的形式大量涌现,而不是以前那种由DBMS处理的有限存储的静态数据。流数据典型应用有客户点击流、电信记录、多媒体流数据、股票交易记录、网络日志记录和商场销售事务记录等。

处理这类数据对数据库领域的研究者提出了新的挑战。设计流数据算法主要面对以下挑战:(a)对不断到达的流数据只能序列访问一次;(b)面对无限不断到达的流数据,只有有限的存储资源可供使用。这使得许多现有的算法不再适用。

数据聚类是极为重要的算法问题,最近几十年已被深入研究。聚类问题可定义为把数据对象分为组,又叫簇,使得在同一簇内数据尽可能相似,而在不同簇内的数据尽可能不同。在这里,数据对象之间的不相似性用对象之间的距离来表示。例如,当对象用数字属性来表示时,对象之间的距离自然地由它们之间的几何距离来表示,这种距离的定义可以用于评价聚类结果得到的簇的质量。在二范数的意义下,聚类过程变为怎样选择那些代表对象的点组合在一起,以至于每个点到簇的中心的均方距离之和为最小。

基于模型的聚类算法把数据对象看作从某个未知概率密度 $p(x)$ 分布中取出的样本 $x_1,\cdots,x_n$。现有两种基于模型的聚类方法:非参数的聚类方法[1],该聚类方法估计概率密度 $p(x)$ 的模式,并把数据样本安排到一个模式的吸引区域;而基于高斯混合模型(Gaussian Mixtures Model,以下简称GGM)的聚类算法[2]假定聚类簇可表示为属于某个参数簇如多变量正态簇密度函数 $p_g(x)$,整个数据集的概率密度为各个簇密度的混合密度,混合密度模型的个数及其参数可由数据样本通过统计估计得到。

基于模型的聚类算法已被证实能发现数据中不同形状和密度的簇,能有效地决定数据中的簇数,可检查出数据中的奇

到稿日期:2008-12-15 返修日期:2009-02-25

刘建伟(1966-),男,副研究员,主要研究方向为数据库、机器学习,E-mail:liujw@cup.edu.cn;李卫民(1974-),男,副教授,主要研究方向为数据库等。

异点，并且能有效地对各种不同属性的数据（如对分类的数据）进行聚类，而且可以证明 k-均值算法只是基于模型聚类算法的一个特例，因而研究基于模型的流数据聚类算法是非常重要的。

但是，基于模型的聚类算法使用期望-最大（Expectation-Maximization，以下简称 EM）迭代算法对混合模型中的参数进行极大似然估计[3]，其算法复杂性为 $O(n^2)$。对于大量不断到达的流数据，许多现有基于模型算法，由于数据存储和算法处理时间的限制，将不再适用于对流数据混合模型的参数进行极大似然估计，进而实现 GGM 聚类。如何找到新的适合的算法，目前还未有研究报道。

1 一种新的聚类算法研究分析

传统的 EM 混合密度估计算法由于其算法复杂性为 $O(n^2)$，因而不适宜对大量高速到达的流数据进行处理。在这里引入基于概率密度摘要的方法来聚类流数据。该方法类似于 BIRCH 聚类算法[4]，然而摘要的定义和产生簇树的过程却与之不同。该方法的思想是不存储所有的流数据或整个簇中的内容，而只存储足够的统计信息来描述流数据。

我们提出的聚类算法整个过程描述如下。

(1)形成初始聚类混合模型：对最初到达的流数据用多维网格结构进行划分，对划分形成的每一个单元进行数据摘要，提取足够的统计信息。对该摘要运行基于模型的贪心聚类算法，聚类形成的 k 个模型的摘要信息存储在永久摘要数据库中。

(2)聚类模型的维持过程：当不断有流数据到达时，对到达的数据块，用多维网格结构进行划分，对划分形成的每一个单元进行数据摘要，提取足够的统计信息。对该摘要运行基于模型的贪心聚类算法，把聚类形成的新的 k' 个模型合并到(1)步骤所形成的初始聚类混合模型当中去。不能合并时，则建立新的模型，并把该新建的模型和已有的聚类混合、模型混合合并，形成新的聚类混合模型。此过程随着流数据块的不断到达而反复执行。

1.1 概率密度摘要的构造

假定流数据由多维记录 $X_1, X_2, \cdots, X_K, \cdots$ 组成，其时间戳分别为 $T_1, T_2, \cdots, T_i, \cdots$。每一个 X_i 为 k-维的多维记录可记为 $X_i=\{x_i^1, \cdots, x_i^k\}$，其值域为 $D=\{d_i^1, \cdots, d_i^k\}$。每一个 k-维的多维记录 X_i 可看作一个 k 维空间中的点 x_i。以下用 x_i 代表 X_i。

摘要过程首先用多维网格结构对各个多维记录 $X_i=\{x_i^1, \cdots, x_i^k\}$ 的值域 $D_i=\{d_i^1, \cdots, d_i^k\}$ 进行划分，划分的结果，使得每个属性 x_i^j 的值域 d_i^j 等分为多段。划分形成的多维网格单元形成一个微簇。对于多维网格单元形成的一个微簇，增量计算落在单元 C_i 内的多维记录各属性值的平均值 $\overline{X}_{c_i}=\frac{1}{n_{c_i}}\sum_{C_i} x_i$，矩阵 $M_{c_i}=\frac{1}{n_{c_i}}\sum_{C_i} x_i \cdot x_i^T$ 的对角向量 $X_{C_i} \cdot X_{C_i}^T$ 及 C_i 单元内数据的个数，即形成微簇的概率密度摘要 ps。ps 可记为

$$ps(X_{C_i}, n_{c_i}, \overline{X}_{C_i}, \overline{X}_{C_i} \cdot \overline{X}_{C_I}^{(T)}) \tag{1}$$

1.2 初始聚类混合模型的算法

在文献[5]中，提出了使数据的对数似然估计 $\Gamma'(\theta)$ 的低边界函数 $\Psi(\theta, Q)$ 最大的迭代过程。同样利用该迭代过程，可以求解出数据的对数似然估计 $\Gamma'(\theta)$，这里 θ 为代表 π_i，m_k，C_k 的混合参数。因式分解分布 Q 为

$$Q=\prod_{i=1}^{n} q_i(s) \tag{2}$$

式中，$q_i(s)$ 为点 x_i 所形成的模型的责任函数。

根据文献[6]，低边界函数 $\Psi(\theta, Q)$ 可分解为以下形式：

$$\begin{aligned}\Psi(\theta, Q)&=\sum_{i=1}^{n}[\log f(x_i;\theta)-K-L(q_i(s)\,\|\,F(s|x_i;\theta))]\\&=\sum_{i=1}^{n}\sum_{s=1}^{k} q_i(s)[\log f(x_i, s;\theta)-\log q_i(s)]\end{aligned} \tag{3}$$

当把多维记录划分形成多维网格单元从而构成多个微簇后，假定每个微簇 c_i 内部的点具有相同的责任函数 $q_i(s)=q_{c_i}(s)$，则 $\Psi(\theta, Q)=\sum_{c_i \in D_i}\Psi_{c_i}(\theta, Q)$。$\Psi(\theta, Q)$ 相应于微簇 c_i 内部的 $\Psi_{c_i}(\theta, Q)$，可表示为

$$\begin{aligned}\Psi_{c_i}(\theta, Q)&=\sum_{x_i \in c_i}\sum_{s=1}^{k} q_{c_i}(s)[\log f(x_i|s)+\log f(s)-\log q_{c_i}(s)]\\&=n_{c_i}\sum_{s=1}^{k} q_{c_i}(s)[\log f(s)-\log q_{c_i}(s)+\frac{1}{n_{c_i}}\log f(x_i|s)]\end{aligned} \tag{4}$$

因此，只需设置 $\Psi_{c_i}(\theta, Q)$ 对 $q_{c_i}(s)$ 的微分为零，便可找到最优的分布 $q_{c_i}(s)$，使得 $\Psi_{c_i}(\theta, Q)$ 为全局最大，即

$$q_{c_i}(s) \propto f(s)\exp\langle \log f(x|s)\rangle_{c_i} \tag{5}$$

$\langle\rangle_{c_i}$ 代表对微簇 c_i 中的所有的点求平均值，因而只需要对在微簇 c_i 中的点求平均联合对数似然估计，便可得到最优分布。

由于已知概率密度摘要 $ps(X_{C_i}, n_{c_i}, \overline{X}_{C_i}, \overline{X}_{C_i} \cdot \overline{X}_{C_i}^{(T)})$，利用其参数 $\overline{X}_{c_i}$，$\overline{X}_{c_i} \cdot \overline{X}_{c_i}^{(T)}$ 完全可以求出式(5)中的平均值 $\langle \log f(x|s)\rangle_{c_i}$，因为

$$\begin{aligned}\langle \log f(x|s)\rangle_{c_i}&=\frac{1}{n_{c_i}}\log f(x_i|s)\\&=-\frac{1}{2}[\log|C_s|+\frac{1}{n_{c_i}}\sum_{x_i \in c_i}(x_i-m_s)C_s^{-1}(x_i-m_s)]\\&=-\frac{1}{2}[\log|C_s|+m_s^T C_s^{-1}m_s+\frac{1}{n_{c_i}}\langle x^T C_s^{-1}x\rangle_{c_i}-2m_s^T C_s^{-1}\langle x\rangle_{c_i}]\\&=-\frac{1}{2}[\log|C_s|+m_s^T C_s^{-1}m_s+Tr\{C_s^{-1}\langle x \cdot x^T\rangle_{c_i}\}-2m_s^T C_s^{-1}\langle x\rangle_{c_i}]\end{aligned}$$

式中，$tr()$ 为矩阵的迹。同样的参数能用于求解混合参数 θ，只需设置式(5)中的 $\Psi_{c_i}(\theta, Q)$ 对于 θ 的微分等于零，则可得到

$$f(s)=\frac{1}{n}\sum_{c_i \in D_i} n_{c_i} q_{c_i}(s) \tag{6}$$

$$m_s=\frac{1}{nf(s)}\sum_{c_i \in D_i} n_{c_i} q_{c_i}(s)\langle x\rangle_{c_i} \tag{7}$$

$$C_s=\frac{1}{nf(s)}\sum_{c_i \in D_i} n_{c_i} q_{c_i}(s)\langle xx^T\rangle_{c_i}-m_s m_s^T \tag{8}$$

使用文献[7]中提出的基于模型的贪心聚类算法对微簇的概率密度摘要进行聚类。基于模型的贪心聚类算法的本质是在具有已经收敛的 k 个高斯混合模型 $f_k(x)$ 情况下，贪心聚类算法发现一个新的具有均值 m_ϕ、方差 C_ϕ 混合权为 $a \in (0,1)$ 的模型 $\phi(x)$，以至于使得两个模型的混合对数似然估计 $\Gamma_{k+1}(\theta)$ 最大。

$$f_{k+1}(x)=(1-a)f_k(x)+a\phi(x;m_\phi, C_\phi) \tag{9}$$

贪心混合模型聚类算法的好处有：(1)不受混合模型的初

始化的影响；(2)不宜使搜寻过程进入局部最大，而可达到全局最大；(3)模型的个数选择变得简单，混合模型中的模型数 k 不需要预先确定，而可根据 BIC 由式(8)计算得到。

贪心混合模型聚类算法用微簇 c_i 中的概率密度摘要 ps 来使低边界函数 $\Gamma_{k+1}(\theta)$ 最大，从而更新式(9)中的参数 $\phi(a;m_\phi,C_\phi)$。假定 q_{c_i} 是新的参数 $\phi(a;m_\phi,C_\phi)$ 的责任函数，$1-q_{c_i}$ 为老的混合密度 $f_k(x)$ 的责任函数，则由式(9)中两个模型决定的微簇 c_i 的 $\Gamma^{c_i}_{k+1}(\theta)$ 为

$$\Gamma^{c_i}_{k+1}(\theta)=n_{c_i}q_{c_i}\left[\log\frac{a}{q_{c_i}}+\langle\log\phi(x)\rangle_{c_i}\right]+ n_{c_i}(1-q_{c_i})\left[\log\frac{1-a}{1-q_{c_i}}+\langle\log f_k(x)\rangle_{c_i}\right] \quad (10)$$

由于 $\Gamma^{c_i}_k(\theta)$ 是 $\sum_{x\in c_i}\log f_k(x)$ 的低边界，式(10)又可写为

$$\Gamma^{c_i}_{k+1}(\theta)\geqslant n_{c_i}q_{c_i}\left[\log\frac{a}{q_{c_i}}+\langle\log\phi(x)\rangle_{c_i}\right]+ n_{c_i}(1-q_{c_i})\left[\log\frac{1-a}{1-q_{c_i}}+\frac{\Gamma^{c_i}_k(\theta)}{n_{c_i}}\right] \quad (11)$$

由以上所述可知，在 E-步骤可设置式(11)的微分为零，进而可求出对应于每个微簇 c_i 的最优的 q_{c_i}，由此得到

$$q_{c_i}=\frac{a\exp\langle\log\phi(x)\rangle_{c_i}}{(1-a)\exp(\Gamma_{k+1}(\theta)/n_{c_i})+a\exp\langle\log\phi(x)\rangle_{c_i}} \quad (12)$$

相似地，在 M-步骤用式(12)求出的 q_{c_i} 使 $\Gamma^{c_i}_{D_i}(\theta)=\sum_{c_i\in D}\Gamma^{c_i}_{k+1}(\theta)$ 最大。此时，设置在每个微簇 c_i 之外的新的模型为零，则可得到以下的 M-步骤更新公式

$$a=\frac{\sum_{c_i\in D}n_{c_i}q_{c_i}}{n} \quad (13)$$

$$m_\phi=\frac{\sum_{c_i\in D}n_{c_i}q_{c_i}\langle x\rangle_{c_i}}{na} \quad (14)$$

$$C_\phi=\frac{\sum_{c_i\in D}n_{c_i}q_{c_i}\langle x\cdot x^T\rangle_{c_i}}{na}-m_\phi\cdot m_\phi^T \quad (15)$$

1.3 聚类混合模型的维持算法

当有新到达的流数据块时，首先对该流数据块用 2.1 节的划分算法对数据块进行划分，形成多维网格微簇，并计算各微簇概率密度摘要 ps 值，然后用 2.2 节提出的贪心混合模型聚类算法计算其各模型参数及混合权值。

决定哪两个模型合并的比较过程，实际上是判断模型之间的距离的过程。该过程使用以下 3 种方法计算模型之间的距离：

• 第一种方法使用假设检验来判定两个模型是否可以合并。假定在流数据块到达之前，贪心混合模型聚类算法得到的各模型参数及混合权值为

$\pi_i,m_i,C_i \quad i=1,\cdots,k_i$

新到达的流数据块的贪心混合模型聚类算法得到的各模型参数及混合权值为

$\pi_j,m_j,C_j \quad j=1,\cdots,k_j$

这里用 W 统计[8]无效假设测试来判断新到达的流数据块混合模型中的模型的协方差矩阵 C_i 是否与流数据块到达之前混合模型的各模型的协方差矩阵 C_j 统计相等。同时，用 Hotelling[8]提出的 T^2 概率分布统计测试新到达的流数据块混合模型中的模型的 m_i 是否与流数据块到达之前混合模型的各模型的 m_j 统计相等。

假定流数据块到达之前混合模型中的多维记录个数为 n_l，新到达的流数据块中的多维记录个数为 n_s，两个具有相同统计参数的模型拥有的多维记录个数分别为 n_{k_i} 和 n_{k_j}，则按以下公式合并两个具有相同统计参数的模型产生的新的模型参数为

$$m_{i,j}=\frac{n_{k_i}m_i+n_{k_i}m_j}{n_{k_i}+n_{k_j}} \quad (16)$$

$$C_{i,j}=\frac{n_{k_i}C_i+n_{k_j}C_j}{n_{k_i}+n_{k_j}}+\frac{n_{k_i}m_i\cdot m_i^T+n_{kj}m_j\cdot m_j^T}{n_{k_i}+n_{k_j}}-m_{i,j}m_{i,j}^T \quad (17)$$

$$\pi_i=\frac{n_{k_i}}{n_l+n_s} \quad (18)$$

否则如果统计不相等，则在现有混合模型中把新的流数据块混合模型中的模型加入到现有混合模型中，其平均值和方差不变，但其混合权为

$$\pi_{i,j}=\frac{n_{k_i}+n_{k_j}}{n_l+n_s} \quad (19)$$

• 第二种方法使用 Mahalanobis[9]或带权的协方差距离

$$d(x,y)=(x-y)^TC^{-1}(x-y) \quad (20)$$

来计算模型之间的距离。

• 第三种方法使用两个高斯模型合并后相应的对数似然估计的减小来度量两个高斯模型之间的距离[9]。仍假定贪心混合模型聚类算法得到的各模型参数及混合权值为

$\pi_i,m_i,C_i \quad i=1,\cdots,k_i$

新到达的流数据块的贪心混合模型聚类算法得到的各模型参数及混合权值为

$\pi_j,m_j,C_j \quad j=1,\cdots,k_j$

这里定义两个模型之间的距离为

$$d_{i,j}=-n_i\cdot\log C_i-n_j\cdot\log C_j+(n_i+n_j)\cdot\log C_{i,j} \quad (21)$$

第二和第三种方法均按式(16)、式(17)和式(18)计算两个符合合并条件的模型计算合并产生的新的模型参数。

2 实验结果

这一节实现了我们提出的算法，进行了一系列实验来评价提出的算法的有效性。实验是在奔腾 IV、256M 字节内存台式机、操作系统为 Window2000 上进行的。提出的算法用 Matlab6.5 编程实现。

实验首先针对不同的网格划分标准，使用 3 种不同的合并标准(基于假设检验的合并模型标准、基于 Mahalanobis 距离的合并模型标准和基于似然估计距离的合并模型标准)比较聚类算法的精度。得出结论：基于似然估计距离的合并模型标准的聚类算法的精度最高，基于假设检验的合并模型标准的聚类算法的精度最低。当使用 40×40 网格划分标准时，基于模型的贪心聚类算法得到的精度已经很高，再使用更小的网格划分时，聚类算法得到的精度就提高不了多少了。

第二个实验是使用人工合成的数据来检验随着模型的增加，基于假设检验的合并模型标准、基于 Mahalanobis 距离的合并模型标准和基于似然估计距离的合并模型标准的聚类算法的学习误差改变的情况。试验得出结论：随着高斯模型数的增加，使用合并模型的方法比不使用合并算法的基于模型的贪心聚类算法的聚类结果误差率反而变小，算法的误差减小，而合并模型的方法的运行速度要比不使用合并算法的基于模型的贪心聚类算法高得多。

第三个实验是比较在滑动窗口模型下不使用合并算法的

基于模型的贪心聚类算法(以下简称 BMGC)和基于似然距离合并模型的流数据聚类算法(以下简称 BLGC)误差随着滑动窗口数的增加聚类结果误差率变化的情况。我们用累加相对误差(Cumulative Relative Error,以下简称 CRE)来衡量。CRE 可定义为

$$CRE=\left|\frac{\sum_{j=1}^{s}Error_{BLGC}(w_j)-\sum_{j=1}^{s}Error_{BMGC}(w_j)}{\sum_{j=1}^{s}Error_{BMGC}(w_j)}\right|\times 100$$

式中,s 是逝去的滑动窗口数。图 1 显示了滑动窗口数为 1000,5000,10000 时实验得到的 CRE 值。从图中可以看到,随着滑动窗口数的增加,实验得到的 CRE 值也缓慢地增加。

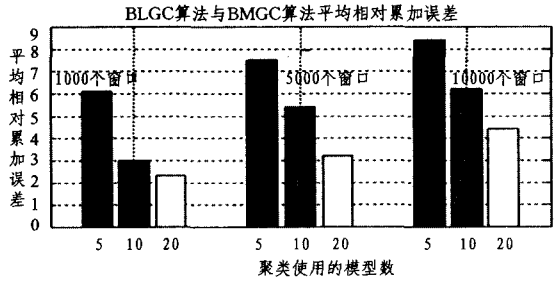

图 1 使用 40×40 网格划分 BMGC 算法和 BLGC 算法随着滑动窗口数的增加聚类结果累加相对误差

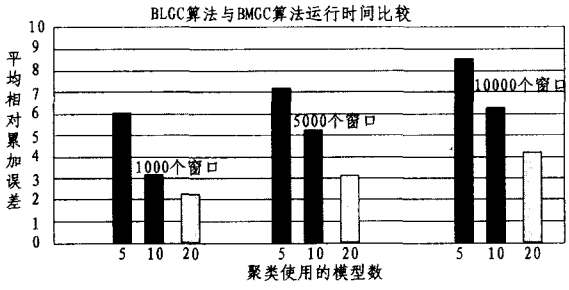

图 2 使用 40×40 网格划分 BMGC 算法和 BLGC 算法随着滑动窗口数的增加运行时间比较

第四个实验是比较在滑动窗口模型下不使用合并算法的基于模型的贪心聚类算法和基于似然距离合并模型的流数据聚类算法随着滑动窗口数的增加聚类算法运行时间。假定这两个算法的运行时间之比为 k,则

$$k=\frac{\sum_{j=1}^{s}time_{BLGC}(w_j)}{\sum_{j=1}^{s}time_{BMGC}(w_j)}$$

图 2 显示了滑动窗口数为 100,500,1000 时实验得到的 k 值。从图中可以看到,随着滑动窗口数的增加,BLGC 算法比 BMGC 算法快了将近 100 倍。且随着窗口数的增加,实验得到的运行时间之比 k 更大。这是由于对于 BMGC 算法来说,随着窗口数的增加,所要处理的数据量会显著地增加,而对于 BLGC 算法所要处理的数据量几乎不变。

3 相关工作

流数据聚类算法方面,文献[10] 提出分而治之的 k-均值方法聚类流数据,该算法只需扫描数据一次。文献[11]提出了聚类不断演化的流数据算法框架,该框架定义了金字塔时间帧的概念,整个聚类过程分为微聚类和宏聚类两个阶段,只保存微聚类产生的簇统计信息,大大缩小了存储量。同一作者在文献[12]提出了按需要分类流数据的概念,该算法定义了几何时间帧概念,微聚类的结果分成不同的时间段。按用户提出的时间段和聚类中的簇数聚类流数据。这些流数据聚类算法均使用 k-均值聚类算法。

结束语 本文提出了一种在线基于摘要技术的混合模型流数据快速聚类算法。在算法中提出了分阶段混合模型聚类过程。首先对最初到达的流数据用多维网格结构进行划分,对划分形成的每一个单元进行数据摘要,提取足够的统计信息。对该摘要运行基于模型的贪心聚类算法,聚类形成的混合模型的摘要信息存储在永久摘要数据库中,从而形成初始聚类混合模型;在聚类模型的维持过程中,当不断有流数据到达时,对到达的数据块用多维网格结构进行划分,对划分形成的每一个单元提取足够的摘要信息,对该摘要运行基于模型的贪心聚类算法,形成聚类混合模型。本文提出了 3 种合并标准,基于假设检验的合并模型标准,基于 Mahalanobis 距离的合并模型标准和基于似然估计距离的合并模型标准,来判断是否可以把新到达的模型合并到现有的混合模型中去。

当不能合并到初始聚类混合模型,则建立新的模型,并把该新建的模型和已有的聚类混合模型合并,形成新的聚类混合模型。

我们的实验验证了算法相比于传统的基于模型的贪心聚类算法,既提高了聚类的质量,减少了分类误差,同时算法速度也比传统的基于模型的贪心聚类算法大大加快。

参 考 文 献

[1] Hartigan J. Statistical theory in clustering[J]. Journal of Classification, 1985(2): 63-76

[2] Ester M, Kriegel H, Sander J, et al. A density-based algorithm for discovering clusters in large spatial databases with noise[C]// Proceedings of the 2nd International Conference on Knowledge Discovery and Data Mining (KDD-96). 1996: 226-231

[3] Dempster A P, Laird N M, Rubin D B. Maximum Likelihood from Incomplete Data via the EM Algorithm[J]. Journal of the Royal Statistical Society, Series B (Methodological), 1977, 39(1): 1-38

[4] Zhang T, Ramakrishnan R, Livny M. BIRCH: An efficient Data Clustering Method for Very Large Databases[C]// Proceedings of the International Conference on Management of Data (SIGMOD). Motreal, QB, Canada: ACM Press, 1996: 103-114

[5] Neal R M, Hinton G E. A view of the EM algorithm that justifies incremental, sparse, and other variants[M]// Jordan M I, ed. Learning in graphical models. Dordrecht, The Netherlands: Kluwer Academic Publishers, 1998: 355-368

[6] Verbeek J J, Vlassis N, Krose B. Efficient greedy learning of Gaussian mixture models[J]. Neural Computation, 2003, 15(2): 469-48

[7] Ledoit O, Wolf M. Some hypothesis tests for the covariance matrix when the dimension is large compared to the sample size [J]. The Annals of Statistics, 2002, 30(4): 1081-1102

[8] Duda R O, Hart P E. Pattern Classification and Scene Anylysis [M]. John Wiley & Sons, 1973

[9] Guha S, Meyerson A, Mishra N, et al. Clustering Data Streams: Theory and Practice[J]. IEEE TKDE, 2003, 15(3): 515-528

[10] Aggarwal C C, Han J, Wang J, et al. A Framework for Clustering Evolving Data Streams[C]// Proceedings of the VLDB Conference. 2003

[11] Aggarwal C C, Han J, Wang J, et al. On Demand Classification of Data Streams[C]// Proceedings of the ACM KDD Conference. 2004