文章编号:1001 - 9081(2007)04 - 0881 - 03

一种基于概率密度的数据流聚类算法

张 伟,陈春燕
(江南大学 信息工程学院,江苏 无锡 214122)
(ccyljl@126.com)

摘 要:数据流具有数据量无限且流速快等特点,使得传统的聚类算法不能直接应用于数据流聚类问题。针对该问题,提出了一种基于概率密度的数据流聚类算法。此方法不需要存储全部的历史数据,只需要存储新到达的数据并对其应用 EM 算法,利用高斯混合模型增量式地更新概率密度函数。实验表明,该算法对于解决数据流聚类问题非常有效。

关键词:数据流;聚类;高斯混合模型;概率密度
中图分类号: TP311.13 **文献标识码**:A

Data stream clustering algorithm based on probability density

ZHANG Wei, CHEN Chun-yan
(*Department of Information Engineering, Southern Yangtze University, Wuxi Jiangsu* 214122, *China*)

Abstract: Data stream is characterized by infinite data and quick stream speed, so traditional clustering algorithm cannot be applied to data stream clustering directly. In view of above questions, a probability-density-based data stream clustering algorithm was proposed. It requires only newly arrived data, not the entire historical data, to be saved in memory. It applies EM algorithm on the newly arrived data and updates probability-density function by incremental Gaussian mixture model. Experimental results show that the algorithm is very effective to solve data stream clustering.

Key words: data stream; clustering; Gaussian mixture model; probability-density

0 引言

随着科学技术的飞速发展,许多技术领域都存在源源不断的规模庞大的数据流。所谓数据流就是大量连续到达的潜在无限的数据的有序序列,这些数据或其摘要信息只能按照顺序存取并被读取一次或有限次。典型的数据流有高速公路传感器网络的监测信息数据,电信公司大型交换机上的通话记录数据以及气象、环境的监测数据等。由于数据流的特殊性,在短时间内有大量的数据到达,使得传统的数据查询、分析、挖掘等算法不能直接应用于数据流,促使人们设计新的算法来适应数据流模型。

数据流聚类是数据流挖掘中一个重要研究内容。聚类[1]就是将数据对象集合划分为多个类或簇,在同一个簇中的对象之间有较高的相似度,而不同簇中的对象差别较大。传统的聚类算法需要多遍扫描数据集合,不能直接应用到数据流聚类上。因为数据流的速度很快,对算法的响应要求很高,所以数据流算法经常采用用精度换时间的方法,尽量在对数据的一次访问中获得较优的解。因此,与传统的聚类算法相比,数据流聚类要满足以下三个要求:1)在满足聚类要求的情况下,尽可能少地扫描数据集,最好是一遍扫描[3];2)快速增长地处理新到达的数据,对数据流进行概化或有选择的舍弃;3)每一次记录的处理时间尽可能短,要能够跟上数据流的速度。

1 数据流聚类的框架

Clustream 算法[4]把数据流聚类分为两个部分:在线的 micro-cluster 过程和离线的 macro-cluster 过程。在线的 micro-cluster 过程对数据流进行初级聚类,阶段性地存储数据流详细的摘要信息,对数据采用增量式的处理和更新。离线的 macro-cluster 过程通过用户输入参数来对在线过程存储的摘要信息进行聚类。通常用户感兴趣的是最近的数据而不是全部历史数据,因此微聚类按金字塔时间框架所产生的时间序列及时以快照的形式存储。

本文采用了基于概率密度的数据流聚类算法。该算法仅需要存储新到达的数据,依靠新到达的数据和历史数据的概率密度,增量式地更新概率密度函数[5]。本算法沿用了 Clustream 算法解决数据流聚类问题的思想,把聚类过程分为两个阶段:在线的进程和离线的进程。记录当前数据流摘要信息的进程为在线的进程,记为 pdmicro-cluster(probability-density micro-cluster);而离线的查询响应进程称为 pdmacro-cluster(probability-density macro-cluster)。

2 pdmicro-cluster 过程

本文算法是一种确定性和随机性的增量式聚类算法,它分为一个合并阶段和一个更新阶段。在合并阶段算法减少了簇的数目;在更新阶段,算法接受新的更新,并试图在不增大簇半径的情况下保持最多的 k 个簇。

2.1 算法的理论基础

下面首先给出本文中使用的一些符号

T:表示离散时间。

X^T:表示随机向量。

$g^N(x)$:表示 $P_0(x)$ 基于历史数据 $X_1,\cdots,X_N$ 的概率密度

收稿日期:2006 - 10 - 08;**修订日期**:2006 - 12 - 07

作者简介:张伟(1956 -),男,吉林长春人,副教授,主要研究方向:聚类分析、数据挖掘、机器学习; 陈春燕(1981 -),女,山西霍州人,硕士研究生,主要研究方向:数据挖掘、聚类分析.

函数。

$g^{N+M}(x)$:表示 $P_0(x)$ 基于历史数据 $X_1,\cdots,X_N$ 和新到达的数据 $X_{N+1},\cdots,X_{N+M}$ 的概率密度函数。

$t(x)$:表示 $P_0(x)$ 基于 $X_{L+1},\cdots,X_{N+M}$ 的概率密度函数。

$s(x)$:表示 $P_0(x)$ 基于 $X_1,\cdots,X_L$ 的概率密度函数。

$h(x)$:表示 $P_0(x)$ 基于 $X_{N+1},\cdots,X_{N+M}$ 的概率密度函数,其中 $X_{N+1},\cdots,X_{N+M}$ 是新到达的数据。

为了方便,把 $g^N(x)$ 写成 $g(x)$。假设每一个数据点的聚类可以由 $g(x)$ 唯一的确定,那么可以利用 $g^N(x)$ 和新到达的数据 $X_{N+1},\cdots,X_{N+M}$ 来得到 $g^{N+M}(x)$。

定理 1 概率密度函数 $s(x),g(x),h(x),t(x)$ 定义在 $[1,N+M]$ 的四个时间段上,它们之间有如下的关系:

$$t(x)=\frac{Ng(x)-Ls(x)+Mh(x)}{N-L+M} \tag{1}$$

证明 令 $f(x)$ 是 $p_0(x)$ 基于 $X_i(i=1,2\cdots,N+M)$ 的概率密度函数。假设 i 在 $[1,N+M]$ 上服从几何分布,其概率分布函数 $P(i)=\frac{1}{N+M},i\in[1,N+M]$。

根据联合概率的定义,可以得到:

$$f(x)=f(x\mid 1\leqslant i\leqslant L)P(1\leqslant i\leqslant L)+f(x\mid L+1\leqslant i\leqslant N+M)P(L+1\leqslant i\leqslant N+M)$$

因为,$s(x)$ 是 $X_1,\cdots,X_L$ 的概率密度函数。所以有:

$$f(x\mid 1\leqslant i\leqslant L)=s(x)$$

同理:

$$f(x\mid L+1\leqslant i\leqslant N+M)=t(x)$$

因为,i 在 $[1,N+M]$ 上服从几何分布,所以:

$$P(1\leqslant i\leqslant L)=\frac{L}{N+M}$$

$$P(L+1\leqslant i\leqslant N+M)=\frac{N+M-L}{N+M}$$

则有:

$$f(x)=s(x)\cdot\frac{L}{N+M}+t(x)\cdot\frac{N+M-L}{N+M} \tag{2}$$

同理,可以得到:

$$f(x)=g(x)\cdot\frac{N}{N+M}+h(x)\cdot\frac{M}{N+M} \tag{3}$$

根据式(2)和(3)可以得到:$L\cdot s(x)+(N+M-L)\cdot t(x)=N\cdot g(x)+M\cdot h(x)$,即是等式(1)。

定理 2 概率密度函数 $g^N(x),g^{N+M}(x)$ 和 $h(x)$ 定义在 $[1,N+M]$ 的三个时间段上,它们之间有如下关系:

$$g^{N+M}(x)=\frac{N}{N+M}\cdot g^N(x)+\frac{M}{N+M}\cdot h(x) \tag{4}$$

证明 令等式(1)中 $L=0$,则:

$$t(x)=\frac{Ng(x)+Mh(x)}{N+M}$$

因为 $L=0$,则 $t(x)$ 是 $p_0(x)$ 基于 $X_1,\cdots,X_{N+M}$ 的概率密度函数,即全部历史数据的概率密度函数,所以 $t(x)$ 即 $g^{N+M}(x)$。而 $g(x)$ 实际上是 $g^N(x)$ 的简写。因而得到:

$$g^{N+M}(x)=\frac{N}{N+M}\cdot g^N(x)+\frac{M}{N+M}\cdot h(x)$$

根据定理1和定理2已经证明,利用 $g^N(x)$ 和新到达的数据 $X_{N+1},\cdots,X_{N+M}$ 可以得到 $g^{N+M}(x)$。

2.2 根据新到达的数据更新高斯混合模型

定理2说明,只利用新到达的数据,可以获得最新的概率密度,这样可以明显减少数据的存储空间,提高聚类的效率。通过式(4),可以根据新到达的数据得到 $h(x)$,并根据 $h(x)$ 和 $g(x)$ 来得到 $g^{N+M}(x)$。因此需要合并概率相等的分量。对于新到达的数据增量式地应用 EM 算法,以得到最佳的概率密度函数 $h(x)$。首先需要根据新到达的数据 $X_{N+1},\cdots,X_{N+M}$ 得到 $h(x)$ 的高斯混合模型。$h(x)$ 中分量的个数 K_a 使用贝叶斯信息标准定义为:$-2logL(X_{N+1},X_{N+2},\cdots,X_{N+M})+\upsilon\log M$。其中,$L$ 是似然函数,υ 是参数的自由度。对于 $h(x)$ 而言,有 K_a 个分量的 d 维高斯混合模型,$\upsilon=[K_a(d+1)(d+2)/2]-1$。根据 $h(x)$ 的高斯混合模型,把新到达的数据分为 K_a 个簇。令 D^K 是簇 K 中数据的集合。M^K 是 D^K 中数据的个数。对于 $h(x)$ 中的每个簇 K 和 $g^N(x)$ 中的每一个分量比较,使用 W 统计量来决定它们是否有概率相等的协方差,使用霍特林 T^2 统计量决定他们是否有概率相等的平均数。如果发现 $g^N(x)$ 中的分量 j 与 $h(x)$ 中的分量 k 相等,则合并 $g^N(x)$ 中的分量 j 与 $h(x)$ 中的分量 k,作为 $g^{N+M}(x)$ 新的分量。如果没有相等的部分,则将 $h(x)$ 中的分量 k 作为一个新的部分加入到 $g^{N+M}(x)$ 中。这样一直持续到 $h(x)$ 中的所有分量添加到 $g^{N+M}(x)$ 中为止。最后,用类似的方法合并 $g^{N+M}(x)$ 中概率相等的分量。

2.2.1 如何确定协方差矩阵相等

为了确定样本 $x_1,x_2\cdots x_n$ 的协方差矩阵与给定的协方差矩阵 Σ_0 是否相等,假设1:$\Sigma_x=\Sigma_0$,并认为样本是多变量正态分布的,Σ_0 是确定的。首先把数据做如下的变换:$y_i=L_0^{-1}x_i,i=1,\cdots,n$。其中 L_0 是由 Σ_0 的乔里斯基分解得到的下三角矩阵,即 $\Sigma_0=L_0L^T$。与假设1等价的假设2是:$\Sigma_y=I$,其中 $\boldsymbol{I}$ 是 d 维的同型矩阵。那么 W 统计量为:$W=\frac{1}{d}tr[(S_y-I)^2]-\frac{d}{n}[\frac{1}{d}tr(S_y)]^2+\frac{d}{n}$,其中 S_y 是 y 的样本协方差,$tr(\cdot)$ 矩阵的迹。基于上述假设,统计量 $\frac{(nW-d)d}{2}$ 有自由度为 $\frac{d(d+1)}{2}$ 的近似 χ^2 分布,即:

$$\frac{(nW-d)d}{2}+d\sim\chi^2_{d(d+1)/2}$$

2.2.2 如何确定平均数向量相等

为了确定样本 $x_1,x_2,\cdots,x_n$ 的平均数向量是否与给定的向量 μ_0 相等,假设 $\mu=\mu_0$,T^2 统计量定义为 $n(\bar{x}-\mu_0)^TS^{-1}(\bar{x}-\mu_0)$,其中 $\boldsymbol{S}$ 为样本协方差矩阵。根据以上假设,$\frac{n-d}{d(n-1)}T^2$ 服从分子自由度为 d,分母自由度为 n-d 的 F- 分布。即:

$$\frac{n-d}{d(n-1)}T^2\sim F_{d,n-d}。$$

使用 T^2 统计量确定平均数向量是否相等时,必须求样本协方差的逆矩阵。不过在样本空间比较小的情况下,可以用 Σ_0 代替 S。

2.2.3 合并或创建分量

如果 D^K 通过了与 $g^N(x)$ 中分量 j 的协方差和平均数的测试,则认为 $h(x)$ 中的分量 k 和 $g^N(x)$ 中的分量 j 有概率相等的分布。因此根据平均数 μ,协方差矩阵 Σ,和权 π 合并它们,在 $g^{N+M}(x)$ 创建一个分量。

根据平均数和协方差的定义可以得出:

$$\mu=\frac{N\pi_j\mu_j+M_k\mu_k}{N\pi_j+M_k}$$

$$\Sigma=\frac{N\pi_j\Sigma_j+M_k\Sigma_k}{N\pi_j+M_k}+\frac{N\pi_j\mu_j\mu_j^T+M_k\mu_k\mu_k^T}{N\pi_j+M_k}-\mu\mu^T$$

$$\pi = \frac{N\pi_j + M_k}{N + M}$$

对于 $g^N(x)$ 中的分量 j,期望在簇中点的数目为 $N\pi_j$。我们需要的统计量有:样本的平均数向量 μ_j、μ_k,平均协方差矩阵 Σ_j、Σ_k,权 π_j 和样本空间 N、M、M_K。因此,应用高斯混合模型避免了查询全部历史数据,也没有丢失任何信息。当 D^K 已经通过了协方差和平均数的测试时,在 $g^N(x)$ 中有两个或更多与其不同的分量,则为 D^K 选择有最大相似度的分量。

对于 $h(x)$ 中剩余的分量 k,在 $g^N(x)$ 中没有与其概率相等的分量,则在 $g^{N+M}(x)$ 中创建一个新的分量,其平均数 $\mu = \mu_k$,协方差 $\Sigma = \Sigma_K$,权 $\pi = \frac{M_K}{N + M}$。

对于 $g^N(x)$ 中剩余的分量 j,在 $h(x)$ 中没有与其概率相等的分量,则在 $g^{N+M}(x)$ 中创建一个新的分量,其平均数 $\mu = \mu_j$,协方差 $\Sigma = \Sigma_j$,权 $\pi = \frac{N\pi_j}{N + M}$。

3　pdmacro-cluster 过程

在线的 pdmicro-cluster 过程快速有效地保存了数据流实时的摘要统计信息,而离线的 pdmacro-cluster 过程只需要在线过程存储的摘要信息。因而在此过程中,用户可以根据自己的需要输入参数:时间范围 h 和聚类的数目 k。使用 K-means[8] 方法作为算法的离线聚类过程。K-means 算法选择 k 个点作为随机种子,并且为了更新簇的划分,反复地为每一个种子分配数据集中的点。在每一次反复过程中,原来的种子被每一个划分的中心所代替。在本文算法中,K-means 方法需要做如下修改:

1)在初始化阶段,种子不再是随机的选择,而是根据概率按照已给微聚的点的数目成比例的抽样,相应的种子是微聚类的中心。

2)在划分阶段,种子到微聚类的距离认为是种子到相应的微聚类中心的距离。

3)在种子调整阶段,一个给定划分的新的种子被认定为在此划分中微聚类加权的中心。

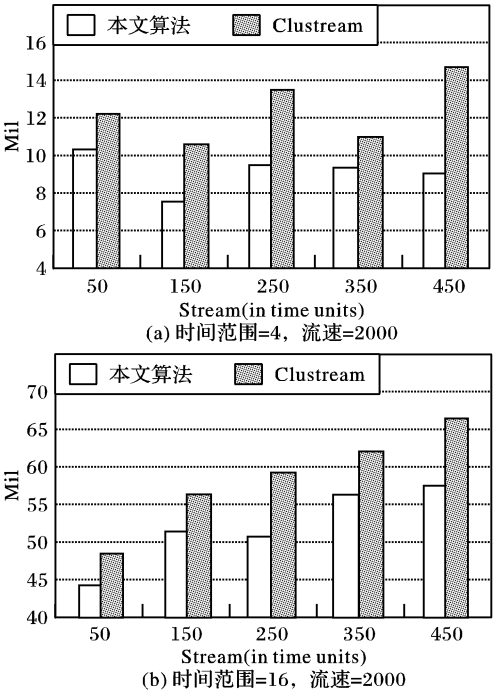
另外,由于 pdmacro-cluster 过程的相对独立,可采取任意一种支持加权聚类的算法来进行。

4　聚类质量比较

使用 KDD－CUP'98 Charitable Donation 数据[4] 来进行算法的测试。此数据曾经被用来测试一次扫描的聚类算法,包含了 95 412 条给予直接邮件求助的人的捐助记录,并且把有类似捐助行为的捐助者进行聚类。我们只用每条记录中的 56 个信息进行测试,根据数据输入的次序作为流的次序,并且假定它们有速度。

所有的实验都在 Pentium 4,256M 内存,Windows XP 环境下进行,聚类质量用距离平方和 SSQ[9] 来衡量。SSQ 是一种比较 k-划分聚类质量的方法,它通过计算所有点到各自的聚类中心的距离来衡量算法所给出的 k-划分的质量。SSQ 值越小,说明算法聚类质量越好。使用 k-means 方法作为本文算法和 Clustream 算法的离线聚类过程来比较两个程序的初始聚类质量,然后比较两个程序算出的 SSQ 值。

所有的实验结果表明,本文算法比 Clustream 更稳定并有较好的聚类质量。图 1 给出了部分实验结果,其中流速 = 2 000是指每个时间单元有 2000 个数据流入。每一个算法运行 10 次来计算它们平均的 SSQ 值。根据 SSQ 值的大小来看,本文算法优于 Clustream。在不同时间的平均 SSQ 值,都比 Clustream 的要小。



(a) 时间范围=4，流速=2000

(b) 时间范围=16，流速=2000

图 1　聚类质量比较

5　结语

针对数据流数据量大且流速快的特点,本文讨论了基于概率密度的数据流聚类算法。此算法的理论基础源于高斯混合模型,合并概率相等的高斯分量,利用数据流的统计结构,仅需要存储新到达的数据,离线的过程使用 k-means 方法,实验表明此算法对于解决数据流聚类问题非常有效。

参考文献:

[1] GOLAB L, ÖZOM M. Issues in Data Stream Management[J]. SIGMOD Record, 2003, 32(2): 5－14.

[2] HAN J, KAMBER M. Data Mining Concepts and Techniques[M]. Beijing: Higher Education Press, 2001. 223－262.

[3] BABCOCK B, BABU S, DATAR M, *et al*. Model and Issues in Data Stream Systems[A]. Proc of ACM SIGMOD/SIGACT Conf on Princ of data Syst[C]. Madison: ACM Press, 2002. 1－16.

[4] AGGARWAL C, HAN J, WANG J, *et al*. A Framework for Clustering Evolving Data Streams[A]. Conference on Very Large Data Bases[C]. Berlin: VLDB conference, 2003. 312－323.

[5] SONG M, WANG H. Highly Efficient Incremental Estimation of Gaussian Mixture Models for online Data Stream Clustering[A]. Proceedings of SPIE: Intelligent Computing-Theory and Application Ⅲ[C]. Florida, 2005, 5803: 174－183.

[6] SONG M, WANG H. Detecting Low Complexity Clusters by Skewness and Kurtosis in Data Stream Clustering[A]. Proceedings of the Ninth International Symposium on Artificial Intelligence and Mathematics[C]. Florida: Proceedings of AIM, 2006. 1－8.

[7] DANIEL B. Requirements for Clustering Data Streams[J]. SIGKDD Exploration, 2003, 3(2): 23－27.

[8] GUHA S, MISHRA N, MOTWANI R, *et al*. Clustering Data Stream [A]. The 41st Annual Symp. on Fundations of Computer Science, FOCS 2000[C]. Redondo Beach: IEEE Computer Science, 2000. 359－366.

[9] AGGARWAL C, HAN J, WANG J, *et al*. A Framework for Projected Clustering of High Dimensional Data Streams[A]. Proceeding of the 30th very large data bases conference[C]. Toronto: VLDB conference, 2004. 852－863.