ISSN 1000-9825, CODEN RUXUEW
*Journal of Software*, Vol.21, No.4, April 2010, pp.644−658
doi: 10.3724/SP.J.1001.2010.03570

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel/Fax: +86-10-62562563

# 基于小波概要的并行数据流聚类*

陈华辉[1,2+], 施伯乐[1], 钱江波[2], 陈叶芳[2]

[1](复旦大学 计算机科学技术学院,上海 200433)

[2](宁波大学 信息科学与工程学院,浙江 宁波 315211)

## Wavelet Synopsis Based Clustering of Parallel Data Streams

CHEN Hua-Hui[1,2+], SHI Bai-Le[1], QIAN Jiang-Bo[2], CHEN Ye-Fang[2]

[1](School of Computer Science and Technology, Fudan University, Shanghai 200433, China)

[2](School of Information Science and Engineering, Ningbo University, Ningbo 315211, China)

+ Corresponding author: E-mail: chenhuahui@nbu.edu.cn

**Chen HH, Shi BL, Qian JB, Chen YF. Wavelet synopsis based clustering of parallel data streams. *Journal of Software*, 2010,21(4):644−658.** http://www.jos.org.cn/1000-9825/3570.htm

**Abstract**: In many real-life applications, such as stock markets, network monitoring, and sensor networks, data are modeled as dynamic evolving time series which is continuous and unbounded in nature, and many such data streams concur usually. Clustering is useful in analyzing such paralleled data streams. This paper is interested in grouping these evolving data streams. For this purpose, a synopsis is maintained dynamically for each data stream. The construction of the synopsis is based on Discrete Wavelet Transform and utilizes the amnesic feature of data stream. By using the synopsis, a fast computation of approximate distances between streams and the cluster center can be implemented, and an efficient online version of the classical *K*-means clustering algorithm is developed. Experiments have proved the effectiveness of the proposed method.

**Key words**: clustering; synopsis; amnesic feature; discrete wavelet transform; data stream

**摘 要**: 许多应用中会连续不断产生大量随时间演变的序列型数据,构成时间序列数据流,如传感器网络、实时股票行情、网络及通信监控等场合.聚类是分析这类并行多数据流的一种有力工具.但数据流长度无限、随时间演变和大数据量的特点,使得传统的聚类方法无法直接应用.利用数据流的遗忘特性,应用离散小波变换,分层、动态地维护每个数据流的概要结构.基于该概要结构,快速计算数据流与聚类中心之间的近似距离,实现了一种适合并行多数据流的 *K*-means 聚类方法.所进行的实验验证了该聚类方法的有效性.

**关键词**: 聚类;概要;遗忘特性;离散小波变换;数据流

**中图法分类号**: TP311 **文献标识码**: A

许多应用场合会连续不断地产生大量数据.一个典型的场景是多传感器系统.设想在系统中有上万个传感

* Supported by the National Natural Science Foundation of China under Grant Nos.60803021, 60973047 (国家自然科学基金); the Zhejiang Provincial Natural Science Foundation of China under Grant No.Y1091189 (浙江省自然科学基金); the Ningbo Municipal Natural Science Foundation of China under Grant Nos.2007A610007, 2009A610072 (宁波市自然科学基金)

Received 2008-05-15; Revised 2008-08-07; Accepted 2008-11-28

器在同时工作,每个传感器每秒钟获取一个测量值,每个传感器就会产生一个随时更新变化的时间序列数据流,整个系统形成一个多数据流并行的数据环境.类似的场景还有证券交易中各股票实时行情形成的并行数据流等.对这类并行数据流进行实时的分析和监控提出了一类有挑战的数据处理工作.本文主要关心如何对其进行聚类分析.

聚类是一项经典的数据挖掘工作,特别是其中的时间序列聚类[1]与本文的工作有相似之处,两者的聚类对象均是序列型的数据.与时间序列不同,数据流连续产生,长度可能无限,数据量巨大,无法保存全部数据重复处理.因而虽然已有针对时间序列聚类的研究,但仍不能直接应用在数据流环境中.

针对数据流环境的聚类方法已有较多研究[2−4],但在多数研究中,聚类的对象是数据流中不断到来的数据,而本文方法将动态演变的数据流本身作为聚类对象.文献[5]进行了与本文相类似的数据流聚类工作.但文献[5]中的方法对每个数据流取一个活动窗口,并假设窗口中的数据是可重复存取的,因而需要较大的存储和计算代价.

本文利用数据流的遗忘特性来对数据流进行压缩,建立一个比整个数据流的数据规模小得多的概要数据结构来保存数据流的主要特征,本文称这种结构为分层遗忘概要(hierarchical amnesic synopses,简称 HAS).所谓遗忘特性是指数据流应用中这样的一种情形:一般对数据流中的近期数据比久远的数据更关注,对近期的数据会更多地关注其细节,而对较远的过去的数据,需要的主要是其大略的概况.如对股票交易价格,对近几天可能关心其具体的价格,对一年前的可能主要关心其大致的波动范围.即在数据流中,数据的影响是随时间衰减的(time-decaying)或者说流中的数据是被逐步遗忘的(amnesic).利用数据流的遗忘特性,HAS 结构将数据流动态地维护为一组分层的数据节点.

离散小波变换(discrete wavelet transform,简称 DWT)是一种重要的数据压缩方法,通过对原始数据集进行小波变换,保存部分重要的小波系数,能够近似地还原出原始数据集合.本文用一组小波系数表示 HAS 中的数据节点,将这种基于小波的 HAS 结构称为 W-HAS(wavelet based HAS).对用 W-HAS 近似表示的一组并行数据流,用改进的 *K*-means 方法进行聚类.将这种并行数据流聚类方法称为 W-HAS-clustering.

本文的主要贡献在于:(1) 利用数据流的遗忘特性和 DWT,构造了数据流的分层概要结构 W-HAS;(2) 基于 W-HAS,讨论了数据流之间的近似距离和聚类中心的动态计算,进而提出了一种适合并行多数据流的 *K*-means 聚类方法;(3) 实验验证了本文方法的有效性.

本文第 1 节介绍背景知识和相关工作.第 2 节介绍 W-HAS-clustering 的基本思想.第 3 节讨论 W-HAS 的动态维护.第 4 节讨论具有误差控制的 W-HAS.第 5 节介绍规范化后数据流的 W-HAS 结构.第 6 节给出聚类中心小波系数的计算及 W-HAS-clustering 聚类的主要步骤.第 7 节用实验验证本文方法的有效性.最后是总结.

## 1 背景知识和相关工作

### 1.1 基于Haar小波的数据压缩

Haar 小波是 DWT 中最简单的一种,因其简单、易实现且有效而在数据压缩等领域得到广泛应用.一维 Haar 小波分解将向量 $D=(x_1,x_2,...,x_n)$ 变换为 $n$ 个小波系数 $(c_1,c_2,...,c_n)$.例:设 $D$=(8,4,5,3,3,3,4,6)($n$=8),表 1 演示了该序列的 Haar 小波变换.

**Table 1** Computing the Haar wavelet transform for serial *D*

**表 1** 序列 *D* 的 Haar 小波变换

| Resolution *l* | Averages | Detail coefficients |
|---|---|---|
| *l*=3 | (8,4,5,3,3,3,4,6) | – |
| *l*=2 | (6,4,3,5) | (2,1,0,−1) |
| *l*=1 | (5,4) | (1,−1) |
| *l*=0 | (4.5) | (0.5) |

具体计算如下:对 Resolution 列中层次 *l*=3,Averages 列中是原始序列.对原始序列数据两两分对求其均值,得到层次 *l*=2 中的 Averages,即((8+4)/2,(5+3)/2,(3+3)/2,(4+6)/2)=(6,4,3,5).显然,在求平均的过程中将原始序列中

的某些信息丢失掉了.为了能够重构原始序列,将每对数据中的均值和第 2 个数据的差也保存在 Detail coefficients 列中,即(6–4,4–3,3–3,5–6)=(2,1,0,–1).如此反复,直到层次 $l$=0.该序列的小波系数由最后的均值和全部 Detail coefficients 列中的数组成,即 $D$ 的小波系数为(4.5,0.5,1,–1,2,1,0, –1).

Haar 小波分解的过程可直观地表示成一种称为误差树(error tree)的树形结构[6].图 1 表示了序列 $D$ 分解的误差树.树中节点 $c_i$($i$=1,2,...,8)对应小波系数,叶节点 $x_i$($i$=1,2,…,8)对应原始数据.对一给定的误差树 $T$ 和 $T$ 中的内节点 $c_k$,令 $leaves_k$ 表示以 $c_k$ 为根的子树的叶节点集合,$leftleaves_k$ 表示 $c_k$ 的左子树的叶节点集合,$rightleaves_k$ 表示 $c_k$ 的右子树的叶节点集合,$path_k$ 为 $T$ 中从 $c_k$(或 $x_k$)到根的路径上全体非零系数的集合.对 $k$=2,3,…,$n$,设 $a_k$ 是 $leftleaves_k$ 中数据的均值,$b_k$ 是 $rightleaves_k$ 中数据的均值,则 $c_k=(a_k-b_k)/2$.而 $c_1$ 是全部数据的均值.从误差树上易知原始数据 $x_k$ 的重构只与 $path_k$ 上的系数有关,即: $x_i=\sum_{c_j\in path_i}\delta_{ij}\cdot c_j$ ,其中 $\delta_{ij}$=+1,如果 $x_i\in leftleaves_j$ 或 $j$=1,否则,$\delta_{ij}$=–1.如 $x_7$= +4.5–0.5–(–1)+( –1)=4.

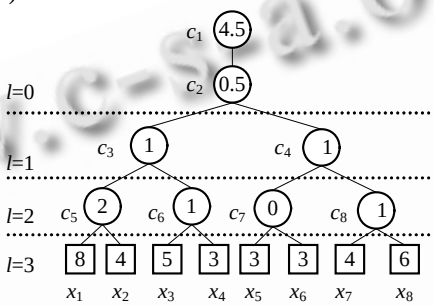


Fig.1 Error tree for serial $D$

图 1 序列 $D$ 的误差树

基于小波的数据压缩利用了小波分解的一个良好性质:只要保留 $r<n$ 个最重要的小波系数(将其他不保留的系数看作是 0),就能重构出原始序列的一个较好的近似.将这 $r$ 个系数称为原始序列的小波概要.小波系数对重构的重要性取决于两个方面:一是绝对值大的系数的缺失会对相关数据值的重构产生更大的影响;二是从误差树上易知,越接近根的系数,用于重构时影响的数据越多,因而具有更高的重要性.为了平衡各系数这两方面的影响,可对系数进行规范化处理.一种常用的规范化策略是将各系数乘以 $\sqrt{n/2^l}$ ,该过程将使 Haar 小波基变换成正交基,设 $c_i^*$ 是对应 $c_i$ 的规范化后的系数,则根据 Parseval 定理[7],有 $\sum_i x_i^2=\sum_i (c_i^*)^2$ .

### 1.2 相关工作

部分相关工作已在前言中加以介绍,本节介绍其他相关工作.

如何在动态数据流上维持小波概要,近年得到了很多关注.Gilbert 等人[8]的方法能够获得误差平方和最小化的数据流小波概要,文献[9]进一步讨论了多维数据流的情形.文献[10,11]等提出了最大绝对误差和最大相对误差度量最优的小波概要构造方法,并将其扩展到流数据的情形.

但这些研究没有考虑数据流的遗忘特性.Palpanas 等人[12]提出了一种处理数据流遗忘特性的方法,本文与他们的方法的主要区别在于,他们的工作着重于遗忘函数,而本文工作在数据表达的动态抽取的同时结合了数据的遗忘特性.Zhao 等人[13]提出了一种框架结合了数据的表达方法和数据重要性的可变性,但这一方法是以数据流的全部数据可多次读取为前提的.Bulut 等人[14]设计了一种称为 SWAT 的基于小波的树形结构,具有表达遗忘特性的能力,但其数据遗忘的速度是不可控制的.Potamias 等人[15]设计了一种类似的称为 AmTree 的树形结构.Aggarwal 等人[16]采用 pyramidal time frame 的方式来保存数据流的概要信息,这种方式对较远的数据采用更粗的粒度,同样具有数据遗忘的特性.与他们的方法相比,本文的 W-HAS 结构具有更好的控制遗忘速度的能力.

文献[5]中的方法对每个数据流取一长度为 $w$ 的滑动窗口,用离散傅里叶变换 DFT(discrete Fourier

transform)对该窗口中的数据进行压缩,利用保留的 DFT 系数进行数据流距离的计算并进而完成聚类.但该方法存在下面几个问题:(1) 所考虑的数据流滑动窗口长度 $w$ 是固定的,而用户对数据流进行分析时可能希望尝试不同长度的数据.(2) 当流中新数据到来时,滑动窗口中数据的 DFT 系数的计算不但需要新到来的数据,而且需要即将移出窗口的数据,即窗口中的全部原始数据均需保存.当窗口长度 $w$ 较大时,需要较大的存储空间.(3) DFT 计算相对本文下面讨论的 DWT 有较大的计算量.本文的方法对如上几个方面作了改进.

## 2 W-HAS-clustering 基本思想

W-HAS-clustering 动态地维护每个数据流的 W-HAS 结构,再利用 W-HAS 对并行多数据流进行在线聚类.本节介绍其基本思想.

### 2.1 W-HAS概要结构

W-HAS-clustering 首先利用数据流的遗忘特性来动态提取数据流的概要信息,形成 W-HAS 概要结构,其主要思想如下:

将数据流中不断到来的数据作为第 0 层,第 0 层中每新到来由 $m$ 个数据组成的子序列,就对其进行浓缩,提炼成第 1 层中的一个数据节点.设该子序列为 $D=(x_1,x_2,\ldots,x_m)$,提取得到的数据节点 $P$,表示成五元组 $(ts,m,\overline{X},\overline{XX},\Gamma)$,其中 $ts=ts(x_m)$为该数据节点的时间戳,表示 $D$ 中最后一个数据的到达时刻,$m=|D|$为 $D$ 中数据个数,$\overline{X}=\frac{1}{m}\sum_{i=1}^{m}x_i$ 为 $D$ 中数据的均值,$\overline{XX}=\frac{1}{m}\sum_{i=1}^{m}x_i^2$ 为 $D$ 中数据平方和的均值.对子序列 $D$ 进行 Haar 小波变换,对小波系数进行规范化,$\Gamma$分量中保存得到的 $m$ 个系数中 $r$ 个最前面的系数.

随着新数据的不断到来,第 1 层上的数据节点不断增加,当达到一定数量时,将最老的 $M$ 个数据节点进行归并,合并成第 2 层上的 1 个数据节点,以此逐层向上,从而使得该数据流总是被压缩成一组分层次的数据节点.其中,层越低,数据节点所对应的数据流子序列越短,同样大小的概要信息对原数据序列的近似程度就越好.反之,较高层次的数据节点对应较长的子序列,其概要信息对原数据序列的表示就较粗略,即随着时间的推移,越久远的数据被浓缩到越高的层次,而越高层次的数据节点对原始数据的遗忘程度越大.

要在数据流上动态地维护 W-HAS 结构的一个基础操作是数据节点的归并,本文定义如下的加法操作来加以实现.

**定义 1(数据节点的加法)**. 设 $D_1$ 和 $D_2$ 为一个数据流上的两个相邻子序列,对应数据节点 $P_1$ 和 $P_2$. $D_1\cup D_2$ 表示由 $D_1$ 和 $D_2$ 串接起来的子序列,从 $D_1\cup D_2$ 提取的数据节点为 $P$.由 $P_1$ 和 $P_2$ 直接计算 $D_1\cup D_2$ 对应的数据节点的操作,称为数据节点的加法,记为 $P_1\oplus P_2$.若经节点加法得到的节点与直接从子序列 $D_1\cup D_2$ 提取得到的节点一致,即 $P=P_1\oplus P_2$,则称满足节点的可加性.

节点的可加性保证利用加法操作归并得到的数据流序列的概要信息与直接从原始数据中提取得到的概要信息是一致的,从而保证了 W-HAS 能够被分层地动态维护.W-HAS 结构动态维护的具体算法在第 3 节中给出.

### 2.2 W-HAS-clustering的聚类距离计算

W-HAS-clustering 采用基于距离的聚类方法进行数据流的聚类,但直接基于数据流的欧氏距离进行聚类是无法实现的,因为:(1) 数据流的原始数据没有保存;(2) 即使在某些数据流环境中能够保存数据本身,但针对大量的数据,其计算量必然巨大.本文利用小波变换的距离保持特性定义如下基于 W-HAS 的近似距离.

**定义 2(数据节点之间的近似距离)**. 对任意的两数据节点 $P_1$ 和 $P_2$,设其 W-HAS 表示的$\Gamma$分量分别为 $\Gamma_1=(c_{1,1}^*,c_{1,2}^*,\ldots,c_{1,r}^*)$ 和 $\Gamma_2=(c_{2,1}^*,c_{2,2}^*,\ldots,c_{2,r}^*)$,则其数据节点近似距离定义为 $d_A(P_1,P_2)=\sqrt{\sum_{i=1}^{r}(c_{1,i}^*-c_{2,i}^*)^2}$.

**定义 3(数据流之间的近似距离)**. 设两数据流 $S1$ 和 $S2$ 的 W-HAS 表示分别由 $q$ 个数据节点构成,按时间顺序表示成 $(P_{1,1},P_{1,2},\ldots,P_{1,q})$ 和 $(P_{2,1},P_{2,2},\ldots,P_{2,q})$,定义 $S1$ 和 $S2$ 的近似距离为 $d_A(S_1,S_2)=\sqrt{\sum_{i=1}^{q}\left(d_A(P_{1,i},P_{2,i})\right)^2}$.

### 2.3 数据流的规范化

因为聚类所关心的是各数据流的相对位置,因而需对原始数据流进行规范化处理,将各数据流统一变换成均值为 0,标准差为 1.

**定义 4(数据流的规范化)**. 设数据流 $S=(x_1,x_2,\ldots,x_n)$,对应的规范化后的数据流为 $\hat{S}=(\hat{x}_1,\hat{x}_2,...,\hat{x}_n)$,则定义数据流的规范化为 $\hat{x}_i=\frac{x_i-\bar{X}}{\sigma},i=1,2,...,n$,其中 $\bar{X}=\frac{1}{n}\sum_{i=1}^{n}x_i$ 为数据流 $S$ 的均值,$\sigma=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\bar{X})^2}$ 为 $S$ 的标准差.

W-HAS-clustering 中的聚类和距离计算均以规范化后的数据流进行.但数据流中的数据是动态到来的,数据流的均值和标准差也是动态变化的,因而这种规范化也应动态地进行.我们采用的方法是在 W-HAS 概要结构中保存的是未规范化的数据流的概要信息,当需要聚类和距离计算时,直接对概要信息进行变换,得到相应的规范化后的数据流的概要信息,具体见第 5 节的讨论.

## 3 数据节点加法及 W-HAS 的动态维护

### 3.1 Haar小波分解的归并

根据 Haar 小波分解过程,本文提出下面的引理 1 和引理 2,给出两个数据序列的 Haar 小波分解如何归并的思想,为数据节点的加法提供基础.

**引理 1(Haar 小波分解的归并)**. 设 $D_1$ 和 $D_2$ 是长度为 $n$ 的数据序列,且 $n$ 为 2 的幂次.$D_1$ 和 $D_2$ 的 Haar 小波分解是 $(c_{1,1},c_{1,2},...,c_{1,n})$ 和 $(c_{2,1},c_{2,2},...,c_{2,n})$,由 $D_1$ 和 $D_2$ 串接起来的子序列 $D_1\cup D_2$ 的 Haar 小波分解是$(c_1,c_2,\ldots,c_{2n})$,则有:

$$c_1=(c_{1,1}+c_{2,1})/2,c_2=(c_{1,1}-c_{2,1})/2,$$

且对 $k$=3,…,2$n$,有:

$$c_k=\begin{cases}c_{1,i} & \text{for}\quad 2^l<k\le 2^l+2^{l-1},\ i=k-2^{l-1}\\ c_{2,i} & \text{for}\quad 2^l+2^{l-1}<k\le 2^{l+1},\ i=k-2^l\end{cases},$$

其中,$l=1,2,...,\log_2^{2n}-1$ 对应 $D_1\cup D_2$ 的 Haar 小波分解误差树的层号.

证明:从 Haar 小波分解的误差树表示易知,任一小波系数的计算只与以该系数为根节点的子树的叶节点上的数据有关,即 $c_1$ 是全部数据的均值,因而 $c_1=(c_{1,1}+c_{2,1})/2$. 对除 $c_1$ 以外的任一系数 $c_j$($j$=2,3,…,2$n$),设 $a_j$ 是 $leftleaves_j$ 中数据的均值,$b_j$ 是 $rightleaves_j$ 中数据的均值,则 $c_j=(a_j-b_j)/2$,因而 $c_2=(c_{1,1}-c_{2,1})/2$, $c_2$ 左子树上的系数仍保持 $(c_{1,2},c_{1,3},...,c_{1,n})$, $c_2$ 右子树上的系数仍保持 $(c_{2,2},c_{2,3},...,c_{2,n})$. 因而对 $k=3,...,2n$,有:

$$c_k=\begin{cases}c_{1,i} & \text{for}\quad 2^l<k\le 2^l+2^{l-1},\ i=k-2^{l-1}\\ c_{2,i} & \text{for}\quad 2^l+2^{l-1}<k\le 2^{l+1},\ i=k-2^l\end{cases}.$$ □

引理 1 说明,若已知两序列 $D_1$ 和 $D_2$ 的 Haar 小波分解 $(c_{1,1},c_{1,2},...,c_{1,n})$ 和 $(c_{2,1},c_{2,2},...,c_{2,n})$,则只需计算前两个系数,即可得到合并后的序列 $D_1\cup D_2$ 的小波分解 $(c_1,c_2,...,c_{2n})$.图 1 中例子的 Haar 小波分解的归并可如图 2 所示.图 2 中左边两棵误差树 $T1$ 和 $T2$ 表示的是子序列(8,4,5,3)和(3,3,4,6)的分解过程,而最右边误差树 $T$ 是序列(8,4,5,3,3,3,4,6)的分解过程.$T$ 可由 $T1$ 和 $T2$ 的归并得到: $c_1=(c_{1,1}+c_{2,1})/2$, $c_2=(c_{1,1}-c_{2,1})/2$, 且 $T$ 中的子树 $A$ 来自 $T1$ 中的子树 $A$,$T$ 中的子树 $B$ 来自 $T2$ 中的子树 $B$.

若考虑小波系数的规范化,对子序列 $D_1\cup D_2$ 的 2$n$ 个小波系数用因子 $\sqrt{2n/2^l}$ 进行规范化,则引理 2 给出了规范化后 Haar 小波系数的归并.

**引理 2(规范化后 Haar 小波分解的归并)**. 设 $D_1$ 和 $D_2$ 的规范化后 Haar 小波分解是 $(c_{1,1}^*,c_{1,2}^*,...,c_{1,n}^*)$ 和 $(c_{2,1}^*,c_{2,2}^*,...,c_{2,n}^*)$,由 $D_1$ 和 $D_2$ 串接的序列 $D_1\cup D_2$ 的规范化后 Haar 小波分解是 $(c_1^*,c_2^*,...,c_{2n}^*)$,则:

$$c_1^*=(c_{1,1}^*+c_{2,1}^*)/\sqrt{2},\ c_2^*=(c_{1,1}^*-c_{2,1}^*)/\sqrt{2},$$

且对 $k$=3,…,2$n$,有:

$$c_k^* = \begin{cases} c_{1,i}^* & \text{for} \quad 2^l < k \le 2^l + 2^{l-1},\ i = k - 2^{l-1} \\ c_{2,i}^* & \text{for} \quad 2^l + 2^{l-1} < k \le 2^{l+1},\ i = k - 2^l \end{cases},$$

其中, $l = 1,2,...,\log_2^{2n} - 1$.

证明:可由引理 1 推出,具体过程略. □

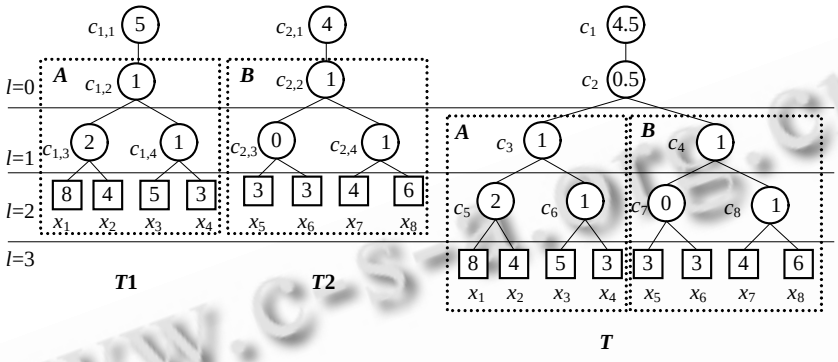


Fig.2 Mergence of Haar wavelet decomposition

图 2 Haar 小波分解的归并

### 3.2 W-HAS结构中数据节点的加法

下面的讨论中设子序列 $D$ 由子序列 $D_1$ 和子序列 $D_2$ 串接而成,$D_1$ 和 $D_2$ 不相交, $|D_1| = |D_2| = |D|/2 = n/2$, $P_1$ 和 $P_2$ 是对应 $D_1$ 和 $D_2$ 的数据节点.$P$,$P_1$ 和 $P_2$ 分别表示为 $(ts, n, \bar{X}, \overline{XX}, \Gamma)$, $(ts_1, n_1, \bar{X}_1, \overline{XX}_1, \Gamma_1)$, $(ts_2, n_2, \bar{X}_2, \overline{XX}_2, \Gamma_2)$. 数据节点 $P'$是 $P_1$ 和 $P_2$ 经数据节点的加法运算得到的,即 $P' = P_1 \oplus P_2$, $P'$ 表示为 $(ts', n', \bar{X}', \overline{XX}', \Gamma')$.

**定义 5(W-HAS 结构中数据节点的加法).** W-HAS 结构中数据节点的加法操作 $P_1 \oplus P_2 \Rightarrow P'$ 定义为

(1) $ts' = \max(ts_1, ts_2)$;

(2) $n' = n_1 + n_2$;

(3) $\bar{X}' = (\bar{X}_1 + \bar{X}_2)/2$;

(4) $\overline{XX}' = (\overline{XX}_1 + \overline{XX}_2)/2$;

(5) 若 $\Gamma_1 = (c_{1,1}^*, c_{1,2}^*, ..., c_{1,r}^*)$, $\Gamma_2 = (c_{2,1}^*, c_{2,2}^*, ..., c_{2,r}^*)$, 且设 $\Gamma' = (c_1^*, c_2^*, ..., c_r^*)$, 则: $c_1^* = (c_{1,1}^* + c_{2,1}^*)/\sqrt{2}$, $c_2^* = (c_{1,1}^* - c_{2,1}^*)/\sqrt{2}$, 且对 $k = 3,...,r$, 有 $c_k^* = \begin{cases} c_{1,i}^* & \text{for} \quad 2^l < k \le 2^l + 2^{l-1},\ i = k - 2^{l-1} \\ c_{2,i}^* & \text{for} \quad 2^l + 2^{l-1} < k \le 2^{l+1},\ i = k - 2^l \end{cases}$, 其中, $l = 1,2,...,\lceil \log_2^r \rceil - 1$.

**定理 3.** 定义 5 给出的 W-HAS 数据节点的加法操作满足可加性.

证明:根据定义 5,易知 $ts = ts'$, $n = n'$, $\bar{X} = \bar{X}'$, $\overline{XX} = \overline{XX}'$.

又 $\Gamma = first_r\left(c_1^*, c_2^*, ..., c_n^*\right)$,其中 $first_r(\bullet)$表示取前 $r$ 项.根据引理 2,

$$\begin{aligned} \Gamma &= first_r\left((c_{1,1}^* + c_{2,1}^*)/\sqrt{2}, (c_{1,1}^* - c_{2,1}^*)/\sqrt{2}, c_{1,2}^*, c_{1,3}^*, ..., c_{1,n_1}^*, c_{2,2}^*, c_{2,3}^*, ..., c_{2,n_2}^*\right) \\ &= first_r\left((c_{1,1}^* + c_{2,1}^*)/\sqrt{2}, (c_{1,1}^* - c_{2,1}^*)/\sqrt{2}, c_{1,2}^*, c_{1,3}^*, ..., c_{1,r}^*, c_{2,2}^*, c_{2,3}^*, ..., c_{2,r}^*\right) \\ &= \Gamma'. \end{aligned}$$

因而 $P$=$P'$,即满足可加性. □

定理 3 保证无须原始数据即可将多个下层节点相加得到上层节点.

### 3.3 W-HAS结构的动态维护

设到当前时刻为止,数据流长度为 $n$,以第 3.2 节讨论的数据节点的加法为基础,下面的算法 1 分层动态地维护了数据流的 W-HAS 结构.

**算法 1**. 数据流上 W-HAS 结构的动态维护.

Input: 流 $D,\alpha,m,M$.

Output: $D$ 的 W-HAS.

/* $i_0,i_1,i_2,\ldots$分别表示第 0,1,2,…层上数据或节点的个数, $P_i^k$ 表示第 $k$ 层上的第 $i$ 个节点,$j$ 表示当前最大层号,各层上的数据节点构成 W-HAS*/

```
begin
  i0 =0; j=0;
  for each new coming data x in D
  {i0=i0+1; x_{i0} =x;
    if i0=m then
     {if j=0 then {i1=1; j=1;}
            else i1=i1+1;
      i0=0; P^1_{i1} =node(i1,i2,…,im);
      /*node()按第 2.1 节所述形成数据节点*/
      for k=1 to j
      {if ik=(α+1)M then
          {if k=j then {ik+1=1; j=j+1;}/*层增长*/
                else ik+1=ik+1+1;
           ik=αM; P^{k+1}_{ik+1} = P^k_1 ⊕ P^k_2 ⊕…⊕ P^k_M ;
           /*将 k 层上最老的 k 个节点合并到 k+1 层*/
           P^k_{M+1}, P^k_{M+2},..., P^k_{(α+1)M} ⇒ P^k_1, P^k_2,..., P^k_{αM} ;
           /*将 k 层上余下节点重编号成 1,2,…,αM*/
          }
          else break;
      }
    }
  }
end
```

算法 1 的一个示意性说明可如图 3 所示.其中空白的圆圈表示数据流中到来的数据,灰色的圆圈表示数据节点,0,1,...,$j$ 为层号.

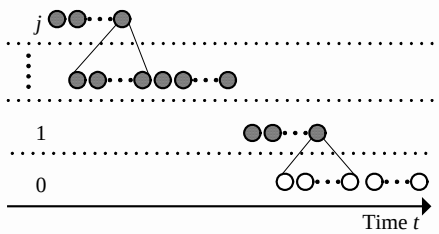


Fig.3 Maintaining W-HAS over data stream

图 3 数据流上 W-HAS 结构的动态维护

**定理 4**. 算法 1 的空间复杂性为 $O(\log(n))$,流中每个数据处理的时间复杂性为 $O(\log(n))$.

证明:根据算法 1,第 $j$ 层上的一个数据节点对应第 0 层上的 $M^{j-1}\times m$ 个原始数据.又因为每层至少有 $\alpha\times M$ 个数据节点,因而 1 至 $j$ 层全部数据节点共对应原始数据至少为 $\alpha\times M\times(M^{j-1}+M^{j-2}+\ldots+M^0)\times m$, 则对数据流中的 $n$ 个数据至多需要的层号 $j=\log[n\times(M-1)/(\alpha\times M\times m)+1]/\log M$.

空间复杂性:

每层最多保存 $(\alpha+1)\times M$ 个数据节点,因而需存储空间 $\log[n\times(M-1)/(\alpha\times M\times m)+1]/\log M\times(\alpha+1)\times M$, 其中 $M,\alpha$为小的常数,因而其空间复杂性为 $O(\log(n))$.

每个数据处理的时间复杂性:

最大的层数为 $j$,因而每个数据参与的数据节点加法次数最多为 $j$ 次,而根据引理 1 和引理 2,每次数据节点加法只需对前两个小波系数进行算术计算,其时间复杂性是 $O(1)$,所以流中每个数据处理的时间复杂性为 $O(\log(n))$. □

该定理表明,W-HAS 在空间复杂性和时间复杂性上是一种可行的数据流概要结构.适当地控制参数 $\alpha,m,M$, 可控制层的生长速度,从而控制数据的遗忘和衰减速度.根据需要,各层的 $\alpha$ 和 $M$ 值也可设置成不一样,实现更灵活的控制.参数 $\alpha,m,M$ 的取值可根据数据流的情况人为地设定.例如,可将当前 1 小时内产生的数据作为第 0 层,每满 1 小时归并成 1 个第 1 层节点,而每 24 小时产生的第 1 层节点归并成 1 个第 2 层节点,每周的数据归并成 1 个第 3 层节点,每月的数据归并成 1 个第 4 层节点,等等.

算法 1 的做法完全以主观的方式设定 W-HAS 中数据归并的速度,没有考虑归并后的概要和原始数据相比

误差的大小.W-HAS 也可以用控制误差的方式控制其归并的速度,下一节给出这种方式的过程.

## 4 具有误差控制的 W-HAS

总体来说,当用 W-HAS 近似表示数据流时,若要根据近似表示重构原始序列,则序列中越久远的数据,重构误差会越大.但同时,越久远的数据,其影响的衰减也越大,因而这些数据上大的重构误差对整个数据流的重构误差的影响不至于太大.上一节中给出的 W-HAS 动态维护算法在总体上保持了这种数据的遗忘特性,但没有给出对这种数据流重构误差的数量控制,本节对此进行定量分析,讨论如何具体控制数据流的重构误差在一个给定的范围内,给出具有误差控制的 W-HAS 动态维护算法(见算法 2).与算法 1 比较,算法 2 将不控制每层上的数据节点数目,而是控制整个数据流的重构误差.

**算法 2**. 具有误差控制的 W-HAS 的动态维护.

Input: 流 $D,\alpha,m,M$,重构误差$\varepsilon$,遗忘函数 $b$;

Output: $D$ 的 W-HAS.

```
begin
  i0=0; j=0;
  for each new coming data x in D
  {i0=i0+1; x_{i0} =x;
    if i0≥m then
     {P=node(x1,x2,…,im);
       if error(P,b)<ε then
       /* error(P,b):节点 P 的重构误差*/
       {if j=0 then {i1=1; j=1;}
              else i1= i1+1;
        i0=i0−m; P^1_{i1} =P;
        x_{m+1}, x_{m+2},... ⇒ x1, x2,... ;
        /*将 0 层上余下数据重编号成 1,2,…*/
     }
  for k=1 to j
       {if ik≥M then
         { P = P1^k ⊕ P2^k ⊕...⊕ PM^k ;
           if error(P,b)<ε then
           {if k=j then {ik+1=1; j=j+1;}
                  else ik+1= ik+1+1;
            ik = ik − M ; P^{k+1}_{ik+1} = P ;
            P^k_{M+1}, P^k_{M+2},... ⇒ P1^k, P2^k,... ;
           }
         }
       }
    }
  }
end
```

定义以下的相对重构误差作为误差控制的目标.

**定义 6(数据序列的相对重构误差)**. 设 $D'$为数据序列 $D$ 基于 W-HAS 近似表示的重构,则其相对重构误差定义为 $E=\dfrac{\|D-D'\|^2}{\|D\|^2}$,其中,符号||x||表示向量 $x$ 的 2 范数.

定义中的数据序列既可以是整个数据流,也可以是某一数据节点对应的子序列.另外,若考虑数据流中数据的影响随时间衰减和遗忘,引入遗忘函数 $b$,遗忘函数要能使越久远的数据,其影响的衰减也越大.

**定义 7(遗忘函数)**. 对数据流 $D$,$D=(x_1,x_2,\ldots,x_n)$为从起始到当前时刻的 $n$ 个数据,则其相应的遗忘函数 $b$ 应满足:对 $i_1,i_2\in[1,n]$,若$i_1<i_2$,则$b(i_1)\le b(i_2)$.

**定义 8(基于遗忘函数的数据序列相对重构误差)**. 设 $D'$ 为数据序列 $D$ 基于 W-HAS 近似表示的重构,则其基于遗忘函数的重构误差定义为 $E_{\text{Amnesic}}=\dfrac{\|(D-D')\bullet b\|^2}{\|D\|^2}$,其中•表示内积.

数据流误差控制的目标是对一给定的小阈值$\varepsilon$,控制整个数据流的相对重构误差 $E$,使 $E\le\varepsilon$.基本思想是控制每个数据节点的相对重构误差,从而控制整个数据流的相对重构误差.我们有如下定理:

**定理 5**. 设有 $q$ 个数据节点表示的数据流 $D=(P_1,P_2,\ldots,P_q)$,若对 $i=1,\ldots,q$,数据节点 $P_i$ 的相对重构误差 $E^{(i)}\le\varepsilon$,则该数据流的相对重构误差 $E\le\varepsilon$.

证明:设对数据流 $D=(P_1,P_2,\ldots,P_q)$,对 $i=1,\ldots,q$,数据节点 $P_i$ 对应的子序列为 $D_i$,相应的重构子序列为 $D_i'$,则 $E^{(i)}=\dfrac{\|D_i-D_i'\|^2}{\|D_i\|^2}\le\varepsilon$, 即 $\|D_i-D_i'\|^2\le\varepsilon\|D_i\|^2$,因而,对数据流的相对重构误差 $E$ 有:

$$E=\frac{\|D-D'\|^2}{\|D\|^2}=\frac{\sum_{i=1}^{q}\|D_i-D_i'\|^2}{\|D\|^2}\le\frac{\sum_{i=1}^{q}\varepsilon\|D_i\|^2}{\|D\|^2}=\frac{\varepsilon\sum_{i=1}^{q}\|D_i\|^2}{\|D\|^2}=\varepsilon.$$ □

若考虑遗忘函数 $b$,类似地有如下定理 6 成立.

**定理 6**. 设有 $q$ 个数据节点表示的数据流 $D=(P_1,P_2,\ldots,P_q)$,若对 $i=1,\ldots,q$,数据节点 $P_i$ 的基于遗忘函数的相对重构误差 $E_{\text{Amnesic}}{}^{(i)}\le\varepsilon$,则数据流的基于遗忘函数的相对重构误差 $E_{\text{Amnesic}}\le\varepsilon$.

证明:类似定理 5,略. □

为了在数据流上动态地维护具有误差控制的 W-HAS 结构,必须能够根据数据节点的概要信息计算数据节点的相对重构误差,定理 7 给出了该计算公式.

**定理 7**. 设对子序列 $D$,$D=(x_1,x_2,\ldots,x_n)$,提取得到的数据节点 $P$ 以 W-HAS 表示,相应的 $r$ 个小波系数由 $\Gamma=(c_1,c_2,\ldots,c_r)$给出,则数据节点 $P$ 的相对重构误差为 $E=\left(n\overline{XX}-\sum_{i=1}^{r}(c_i)^2\right)\Big/\left(n\overline{XX}\right)$.

证明:对序列 $D$ 进行 DWT 分解,设 $n$ 个小波系数为 $(c_1,c_2,\ldots,c_n)$,$(u_1,u_2,\ldots,u_n)$ 是相应的正交基向量,则 $D=\sum_{i=1}^{n}c_iu_i$,根据 Parseval 定理[7],$\|D\|^2=\sum_{i=1}^{n}(c_i)^2$.根据 $r$ 个系数重构 $D$,重构向量 $D'=\sum_{i=1}^{r}c_iu_i$.因而误差向量 $e=D-D'=\sum_{i=r+1}^{n}c_iu_i$,$\|e\|^2=\sum_{i=r+1}^{n}(c_i)^2=\sum_{i=1}^{n}(c_i)^2-\sum_{i=1}^{r}(c_i)^2=\|D\|^2-\sum_{i=1}^{r}(c_i)^2$,亦即相对重构误差为

$$E=\|e\|^2\big/\|D\|^2=\left(\sum_{i=1}^{n}x_i^2-\sum_{i=1}^{r}(c_i)^2\right)\Big/\sum_{i=1}^{n}x_i^2=\left(n\overline{XX}-\sum_{i=1}^{r}(c_i)^2\right)\Big/\left(n\overline{XX}\right).$$ □

有了上述定义和定理,可将算法 1 改写成以下具有误差控制的 W-HAS 动态维护算法(见算法 2).设最小数据节点长度为 $m$ 个数据,每次数据节点归并将 M 个下层数据节点归并成 1 个上层数据节点.

在并行多数据流的环境中,对不同的数据流,算法 2 维护的 W-HAS 结构一般是不同步的.所谓不同步是指,对两数据流 $S_1$ 和 $S_2$ 同一时刻 $t_i$ 出现的两个数据 $x_{1,i}$ 和 $x_{2,i}$,在各自的 W-HAS 结构中被归并到不同层次的数据节点中.显然,对这样的 W-HAS 无法计算定义 2 和定义 3 中给出的数据流之间的近似距离,也无法进行下一节给出的数据流聚类中心的计算.解决的办法是在距离和聚类中心计算之间将这些数据流的 W-HAS 同步化,即统一将层次低的数据节点归并到较高的层次,使不同数据流中同一时刻的数据抽象到相同层次的数据节点中.

## 5 规范化后数据流的 W-HAS 结构

第 2.3 节已经说明 W-HAS-clustering 中的聚类和距离计算均以规范化后的数据流进行.第 3 节和第 4 节讨论的均是未规范化的数据流的小波系数和 W-HAS 结构的动态维护,本节说明如何直接从该 W-HAS 结构中获得相应规范化后数据流的小波系数.

设数据流 $S=(x_1,x_2,\ldots,x_n)$的 W-HAS 结构由 $q$ 个数据节点构成: $P_j=(ts_j,n_j,\overline{X}_j,\overline{XX}_j,\Gamma_j),j=1,\ldots,q$,则 $S$ 的数据个数 $n=\sum_{j=1}^{q}n_j$,均值 $\overline{X}=\frac{1}{n}\sum_{j=1}^{q}n_j\overline{X}_j$,$\overline{XX}=\frac{1}{n}\sum_{j=1}^{q}n_j\overline{XX}_j$,因而 $S$ 的标准差 $\sigma=\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i-\overline{X})^2}=\sqrt{\overline{XX}-\overline{X}^2}$.

设 $S$ 对应的规范化后数据流为 $\hat{S}=(\hat{x}_1,\hat{x}_2,\ldots,\hat{x}_n)$,则 $\hat{x}_i=\frac{x_i-\overline{X}}{\sigma},i=1,2,\ldots,n$.关于 $\hat{S}$ 的 W-HAS 有如下定理 8:

**定理 8**. 若数据流 $S$ 的 W-HAS 结构中任一数据节点 $P$ 的 $\Gamma$ 分量为 $\Gamma=(c_1,c_2,\ldots,c_r)$,则 $S$ 规范化后的数据流 $\hat{S}$ 的 W-HAS 结构中对应数据节点的 $\Gamma$ 分量为 $\hat{\Gamma}=\frac{1}{\sigma}(c_1-\overline{X}\sqrt{n_P},c_2,\ldots,c_r)$,其中,$\sigma$ 为 $S$ 的标准差,$\overline{X}$ 为 $S$ 的均值,$n_P$ 是数据节点 $P$ 对应的原始数据个数.

证明:设数据节点 $P$ 对应的数据子序列为 $D$.记 $D$ 的小波系数为 $(c_1,c_2,\ldots,c_{n_p})$,$D$ 的小波分解的误差树为 $T$,$D$

规范化后的序列为 $\hat{D}$, $\hat{D}$ 的小波系数为 $(\hat{c}_1,\hat{c}_2,...,\hat{c}_{n_P})$, $\hat{D}$ 的误差树为 $\hat{T}$.

对 $T$ 中任一节点 $c_j$,设 $a_j$ 是 $leftleaves_j$ 中数据的均值,$b_j$ 是 $rightleaves_j$ 中数据的均值,则 $c_1=\frac{a_1+b_1}{2}\sqrt{n_P}$,对除 $c_1$ 以外的任一系数 $c_j(j=2,3,\ldots,n)$, $c_j=\frac{a_j-b_j}{2}\sqrt{n_P/2^l}$,其中,$l$ 是 $c_j$ 在误差树 $T$ 中的层号.

对 $\hat{T}$ 中任一节点 $\hat{c}_j$,相应地设 $\hat{a}_j$ 是 $\hat{T}$ 中 $leftleaves_j$ 中数据的均值,$\hat{b}_j$ 是 $\hat{T}$ 中 $rightleaves_j$ 中数据的均值,则 $\hat{a}_j=\frac{a_j-\bar{X}}{\sigma}$, $\hat{b}_j=\frac{b_j-\bar{X}}{\sigma}$.因而, $\hat{c}_1=\frac{\hat{a}_1+\hat{b}_1}{2}\sqrt{n_P}=\frac{1}{\sigma}(c_1-\bar{X}\sqrt{n_P})$, $\hat{c}_j=\frac{\hat{a}_j-\hat{b}_j}{2}\sqrt{n_P/2^l}=\frac{1}{\sigma}c_j, j=2,3,...,n_P$.

所以, $\hat{\Gamma}=(\hat{c}_1,\hat{c}_2,...,\hat{c}_r)=\frac{1}{\sigma}(c_1-\bar{X}\sqrt{n_P},c_2,...,c_r)$. □

## 6 W-HAS-Clustering 聚类方法

根据定义 2 和定义 3 给出的 W-HAS 结构下的数据节点和数据流近似距离,可用 *K*-means 的思想进行数据流的聚类.下面首先介绍聚类中心的小波系数的计算,然后再具体给出 W-HAS-clustering 聚类方法.

### 6.1 聚类中心小波系数的计算

*K*-means 中的聚类中心采用每一类中全部向量的均值,下面的定理 9 给出已知一组数据流的 W-HAS 结构,如何计算这组数据流的均值的小波系数,即如何得到聚类中心的小波系数,从而可以利用定义 3 快速计算各数据流和聚类中心的距离.

**定理 9**. 设有 $N$ 个数据流 $S_i,i=1,\ldots,N$,构成一个聚类,记聚类的中心为数据流 $\tilde{S}$, $S_i$ 由 $q$ 个数据节点 $P^{(ij)}(j=1,\ldots,q)$,组成, $\tilde{S}$ 由 $q$ 个数据节点 $\tilde{P}^{(j)}(j=1,...,q)$,组成.记数据节点 $P^{(ij)}$的$\Gamma$分量为 $\Gamma^{(ij)}=(c_1^{(ij)},c_2^{(ij)},...,c_r^{(ij)})$, $\tilde{P}^{(j)}$ 的$\Gamma$分量为 $\tilde{\Gamma}^{(j)}=(\tilde{c}_1^{(j)},\tilde{c}_2^{(j)},...,\tilde{c}_r^{(j)})$,则 $\tilde{c}_k^{(j)}=\frac{1}{N}\sum_{i=1}^{N}c_k^{(ij)}, k=1,...,r; j=1,...,q$.

证明:略. □

根据定理 9,聚类中心的小波系数可直接通过对各数据流的小波系数求均值得到.另外,参与聚类的各数据流已经过规范化处理,各数据流中数据的均值为 0,因而聚类中心数据流中各数据的均值也为 0.

### 6.2 W-HAS-Clustering聚类

W-HAS-Clustering 聚类的基本思想是对各数据流动态地维持各自的 W-HAS,再利用各数据节点的小波系数进行 *K*-means.W-HAS-Clustering 聚类的主要步骤如下:

(1) 维护数据流的 W-HAS 结构,根据用户对数据流关心的长度 $N$,取相应的最近 $q$ 个数据节点,使这 $q$ 个节点所代表的原始数据大于等于 $N$,用这 $q$ 个节点近似原数据流.

(2) 取这 $q$ 个节点的小波系数,计算相应的规范化后数据流的小波系数,构成数据流的长度为 $q\times r$ 的小波系数向量,根据人工估计的聚类个数 $K$ 和随机选取的聚类中心进行 *K*-means 聚类.

(3) 若数据流中有新数据节点到来和老数据节点的删去,则更新数据流的小波系数向量,并以上次聚类为基础(即以上次数据流的聚类结果作为本次 *K*-means 聚类的初始划分),以各数据流更新后的小波系数重新计算聚类中心,继续进行 *K*-means 聚类.考虑到数据流的演化很可能仍保留原有的聚类,以上次聚类为基础继续 *K*-means 可减少迭代次数.

(4) 数据流的演化可能会引起聚类个数 $K$ 的增减.与文献[5]类似,假定聚类是渐变的,$K$ 的可能变化是±1,加 1 表示有一个聚类分裂成了 2 个,而减 1 表示有 2 个聚类进行了融合.已有研究提出了度量聚类个数 $K$ 合适程度的指标[5],记该指标为 $Q(K)$.为了计算 $Q(K-1)$,W-HAS-clustering 从原有 $K$ 个聚类中去掉一个,将该聚类的数据流重新分配给另外 $K-1$ 个聚类中心的最近一个,得到一个 $Q(K-1)$.重复 $K$ 次,每次从 $K$ 个聚类中去掉不同的一个,以确定最佳的 $Q(K-1)$.为了计算 $Q(K+1)$,W-HAS-clustering 在原有 $K$ 个聚类的基础上增加一个新类,得到最佳的

$Q(K+1)$.取

$$K = \arg\max\{Q(K-1), Q(K), Q(K+1)\}$$

作为当前最佳的聚类个数.

讨论:与文献[5]中方法比较,W-HAS-clustering 方法在以下几个方面有所改进:(1) 采用基于 Haar 小波的 W-HAS 结构对数据流进行压缩,相比 DFT 数据压缩有更小的计算量;(2) 利用 W-HAS 结构中数据节点的可加性,避免保存原始数据本身,有更小的空间复杂度,更适合数据流环境;(3) W-HAS 中保存了整个数据流的概要结构,只要取适当个数的数据节点,即能对感兴趣长度的数据流进行分析,而不局限在固定长度的滑动窗口.

## 7 实 验

我们的实验对 W-HAS-Clustering 的有效性进行了验证.采用的设计工具为 Matlab 7,用人工和实际数据集进行了实验.着重验证该方法的聚类质量.

为了在实验中比较不同距离度量和聚类方法的聚类结果,本文采用 Gavrilov 等人[17]提出的聚类结果评估方法,该方法也在 Keogh 等人[18]的时间序列聚类结果中得以采用.该方法将聚类结果与数据集的真实类别相比较,当然,该方法的前提是能够获知数据的真实类别.设数据集的真实类别为 $C=C_1,\ldots,C_k$,聚类结果为 $C'=C'_1..C'_k$,则用下式表示类之间的相似度:

$$Sim(C_i, C'_j) = 2\frac{|C_i \cap C'_j|}{|C_i| + |C'_j|},$$

即两类之间的相似度定义为两者一致的数据个数和数据总数的比,则聚类结果 $C'$和真实类别 $C$ 的相似度为

$$Sim(C, C') = \left(\sum_i \max_j (Sim(c_i, C'_j))\right) \Big/ K.$$

根据该相似度度量,若聚类结果与实际类别完全一致,则相似度为 1;与实际类别完全不一致,则为 0.

实验中将本文的方法(下面的图表中标识为 W-HAS)与其他两种方法进行了对比:一是和直接 *K*-means 聚类比较,下面的图表中标识为 *K*-means,该方法直接对全部原始数据进行 *K*-means 聚类;二是与文献[5]中将序列用离散傅里叶压缩的方法相比较,下面的图表中标识为 DFT.实验中,*K* 均取数据集中真实的类别数.

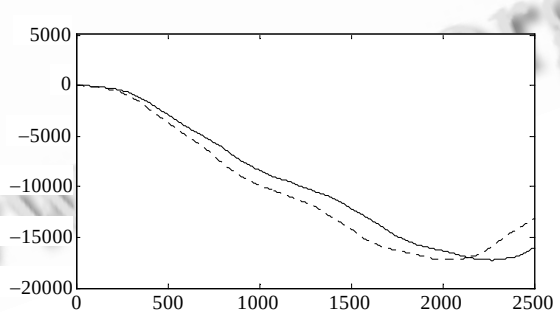


Fig.4 Example of a prototypical data stream (solid line) and a distorted stream (dashed line)

图 4 一个原型数据流(实线)及相应的变异后的数据流(虚线)的例子

第 1 个实验的人工数据集采用与文献[5]类似的生成方法:首先,对每个类产生一个原型 $p(\cdot)$.$p(\cdot)$是一个由下面二阶差分方程定义的随机过程:

$$p(t+\Delta t) = p(t) + p'(t+\Delta t),$$
$$p'(t+\Delta t) = p'(t) + u(t),$$

其中,$t = 0, \Delta t, 2\Delta t, \ldots, u(t)$是在区间$[-a,a]$上均匀分布的独立随机变量,常量 $a$ 的值越小,随机过程 $p(\cdot)$就越平滑.同一类别的序列通过对同一原型 $p(\cdot)$进行水平方向(时间轴上)的压缩和垂直方向(数据值)上添加噪声而得到,即数据流 $x(\cdot)$定义为

$$x(t) = p(t + h(t)) + g(t),$$

其中,$g(\cdot)$是和 $p(\cdot)$同样产生的随机过程,但常量 $a$ 的取值可不一样.$h(\cdot)$由下式产生:

$$h(t + \Delta t) = h(t) + v(t),$$

其中,$t = 0, \Delta t, 2\Delta t, \ldots, v(t)$ 是在区间$[0,b]$上均匀分布的独立随机变量.图 4 是一个可能的原型和由该原型形成的一个数据流例子.

实验中数据集生成的参数设置:$p(0)=0$,$g(0)=0$,对 $p(\cdot)$取 $a=0.5$,对 $g(\cdot)$取 $a=0.1$,$h(\cdot)$中取 $b=0.2$,共生成 4 个不同的类,每个类包含 100 条数据流.取数据节点个数 $q=4$,每个数据节点中保留的小波系数个数 $r=4$.DFT 方法划分数据流成 8 个块,并保留 16 个 DFT 系数.为了解决聚类结果的随机性,实验中对同一数据重复运行 10 次,取最好的一个结果,再将 20 个这样的结果求平均作为最终的实验结果.图 5 所示是对 4 个不同数据流长度的数据集的聚类结果.W-HAS 的结果与 *K*-means 类似或好于后者,而且好于 DFT 方法.

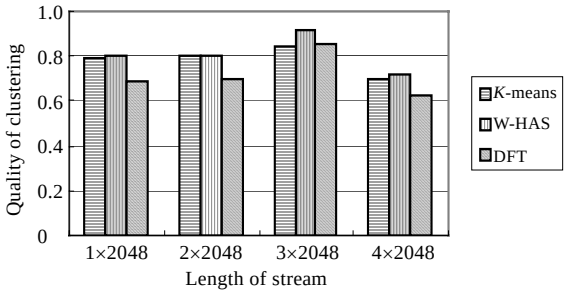


Fig.5 Comparing the clustering quality of *K*-means,W-HAS and DFT

图 5 比较 *K*-means,W-HAS 和 DFT 方法的聚类质量

表 2 比较了 3 种方法的运行时间(含进行 DWT 或 DFT 变换对数据流进行处理的时间和聚类的时间,单位:秒.W-HAS 要大大快于 *K*-means,同时也快于 DFT.

**Table 2** Comparing the clustering time and space

**表 2** 聚类时间和空间比较

| Length of stream | Clustering time | | | Clustering space | | |
|---|---|---|---|---|---|---|
| | *K*-means | DFT | W-HAS | *K*-means | DFT | W-HAS |
| 1×2048 | 524.4 | 10.1 | 7.8 | 2 048 | 2 080 | 32 |
| 2×2048 | 833.3 | 10.3 | 8.9 | 4 096 | 4 128 | 32 |
| 3×2048 | 1 199.1 | 11.2 | 9.3 | 6 144 | 6 176 | 32 |
| 4×2048 | 2 510.7 | 15.8 | 11.0 | 8 192 | 8 224 | 32 |

表 2 还比较了 3 种方法所需的存储空间.所需空间大小以存储的数据量来衡量,没有考虑大数据量引起的聚类算法中临时存储空间的增大.*K*-means 方法中需要保存全部原始数据,因而空间和数据集大小一致.DFT 方法除保存原始数据以外,还保存选中的 DFT 系数和每个数据块中两个用于数据规范化的参数.W-HAS 保存数据节点.

第 2 个实验的数据集取自 UCR 时间序列数据集[19],共包含 16 个数据集,既有人工数据也有实际数据,测试结果见表 3.其中,Name 列是数据集名称,#列是数据集中聚类的个数,Size 是数据集中序列数目,Length 是数据集中序列的长度,UCR 列是文献[19]中采用普通 *K*-means 的聚类结果,DFT 列和 W-HAS 列分别是采用 DFT 压缩和本文方法的结果.W-HAS 方法将序列分成 3 段,每段子序列压缩成 4 个小波系数.DFT 方法保留了 12 个 DFT 系数.从表中可以看出,本文方法虽然大大压缩了序列长度,但聚类结果仍与普通 *K*-means 相当或较其为好.

**Table 3** Comparing the clustering quality for UCR time series data set

**表 3** 比较 UCR 时间序列数据集上的聚类质量

| Name | # | Size | Length | UCR | DFT | W-HAS |
|---|---|---|---|---|---|---|
| Synthetic control | 6 | 600 | 60 | 0.679 | 0.722 | 0.791 |
| Gun-Point | 2 | 200 | 150 | 0.500 | 0.584 | 0.5 |
| CBF | 3 | 930 | 128 | 0.626 | 0.593 | 0.745 |
| Face (all) | 14 | 2 250 | 131 | 0.36 | 0.264 | 0.392 |
| OSU leaf | 6 | 442 | 427 | 0.378 | 0.317 | 0.378 |
| Swedish leaf | 15 | 1 125 | 128 | 0.406 | 0.433 | 0.465 |
| 50Words | 50 | 905 | 270 | 0.420 | 0.248 | 0.417 |
| Trace | 4 | 200 | 275 | 0.485 | 0.567 | 0.529 |
| Two patterns | 4 | 5 000 | 128 | 0.322 | 0.334 | 0.323 |
| Wafer | 2 | 7174 | 152 | 0.625 | 0.626 | 0.625 |
| Face (four) | 4 | 112 | 350 | 0.669 | 0.676 | 0.675 |
| Lightning-2 | 2 | 121 | 637 | 0.611 | 0.635 | 0.591 |
| Lightning-7 | 7 | 143 | 319 | 0.484 | 0.501 | 0.562 |
| ECG | 2 | 200 | 96 | 0.698 | 0.719 | 0.690 |
| Adiac | 37 | 781 | 176 | 0.384 | 0.339 | 0.350 |
| Yoga | 2 | 3 300 | 426 | 0.517 | 0.559 | 0.512 |

第 3 个实验的数据集是 Tickwise[19],这是一个相对较大的数据集,包含 1985 年 05 月 20 日~1991 年 04 月 12 日美元对瑞士法郎的实时汇率,共 329 112 个数据.本实验为了小波分解计算的方便,取其前 262 144($2^{18}$)个数据构成的序列.

Tickwise 数据集上首先进行了具有重构误差控制的 W-HAS 的实验,图 6 中横轴是控制的误差$\varepsilon$,纵轴是控制序列的重构误差$\le\varepsilon$所需的数据节点的数目.每 128 个原始数据生成 1 个第 1 层的数据节点,2 个下层数据节点归并成 1 个上层数据节点.每个数据节点的衰减函数为该数据节点离序列流的最终位置的数据节点的数目的倒数.

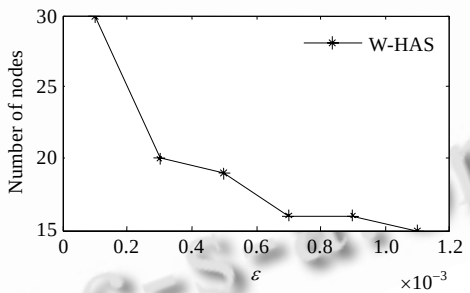


Fig.6 W-HAS structure with controlled error

图 6 控制重构误差的 W-HAS

为了进行并行数据流的聚类实验,在 Tickwise 数据序列基础上生成 4 个并行数据流类.与第 1 个实验类似,首先对每个类产生一个原型 $p(\cdot)$:

$$p(t)=T(t)+p'(t),$$

$$p'(t+\Delta t)=p'(t)+u(t),$$

其中,$T(t)$是 Tickwise 数据序列,$u(t)$ 是在区间$[-a,a]$上均匀分布的独立随机变量,在本实验中,$a$=0.01.与实验 1 一样,在原型数据流上生成该类的数据流,每个类包含 100 条数据流.对每个数据流自近及远共划分成:2 个 1 层数据节点(2×256 个原始数据),3 个 2 层数据节点(3×512 个原始数据),以后各层均 2 个数据节点,共 9 层,19 个数据节点,上层的数据节点由 2 个下层数据节点合并而成.因原始数据量太大,无法直接进行 *K*-means 聚类,所以本实验仅比较 W-HAS 和 DFT.图 7 是实验结果,横轴 $r$ 是 W-HAS 方法中数据节点保留的小波系数个数,相应地,DFT 方法中保留 19×$r$ 个 DFT 系数.

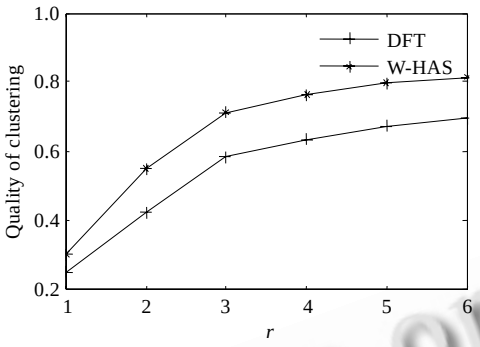


Fig.7 Comparing the clustering quality of W-HAS and DFT (Tickwise)

图 7 比较 W-HAS 和 DFT 方法的聚类质量(Tickwise 数据集)

从以上实验结果可以发现,W-HAS-clustering 因为使用小波对数据流进行了压缩,使得聚类的时间大大加快.同时,由于小波表示的良好性能,压缩后仍保留了原始数据的主要特征,而且去除了某些噪音,这种压缩不仅没有引起聚类性能的下降,而且在某些数据集上还有所提高.

## 8 总 结

本文讨论了一种基于分层遗忘小波概要的并行多数据流聚类分析方法,介绍了该概要结构的动态维护方法,给出了利用该概要结构快速计算数据流之间的近似距离和聚类中心的方法,提出了一种适用于并行多数据流的改进的 *K*-means 聚类方法.实验验证了本文方法的有效性.我们下一步的工作将从以下几方面来进行:一是开展基于该概要结构的数据流的相似性检索、异常检测等其他处理的研究;二是将进一步研究数据流的其他概要结构构造方法下的聚类方法;三是将 W-HAS-Clustering 进一步应用到实际数据流场景,解决实际问题,如应用到传感器网络中.

**References**:

[1] Keogh E, Kasetty S. On the need for time series data mining benchmarks: A survey and empirical demonstration. Data Mining and Knowledge Discovery, 2003,7(4):349−371. [doi: 10.1023/A:1024988512476]

[2] Guha S, Meyerson A, Mishra N, Motwani R, O'Callaghan L. Clustering data streams: Theory and practice. IEEE Trans. on Knowledge and Data Engineering, 2003,15(3):515−528. [doi: 10.1109/TKDE.2003.1198387]

[3] Aggarwal CC, Han J, Wang J, Yu PS. A framework for clustering evolving data streams. In: Johann CF, Peter CL, Serge A, Michael JC, Patricia GS, Andreas H, eds. Proc. of the 29th Int'l Conf on Very Large Data Base. San Francisco: Morgan Kaufmann Publishers, 2003. 81−92.

[4] Charikar M, O'Callaghan L, Panigrahy R. Better streaming algorithms for clustering problems. In: Proc. of 35th ACM Symp. on Theory of Computing. New York: ACM Press, 2003. 30−39. http://doi.acm.org/10.1145/780542.780548

[5] Beringer J, Hullermeier E. Online clustering of parallel data streams. Data & Knowledge Engineering, 2006,58(2):180−204. [doi: 10.1016/j.datak.2005.05.009]

[6] Matias Y, Vitter JS, Wang M. Wavelet-Based histograms for selectivity estimation. In: Tiwary A, Franklin M, eds. Proc. of the 1998 ACM SIGMOD Int'l Conf. on Management of Data. New York: ACM Press, 1998. 448−459.

[7] Boggess A, Narcowich FJ, Wrote; Rui GS, *et al*., Trans. A First Course in Wavelets with Fourier Analysis. Beijing: Publishing House of Electronics Industry, 2004 (in Chinese).

[8] Gilbert AC, Kotidis Y, Muthukrishnan S, Strauss M. One-Pass wavelet decompositions of data streams. IEEE Trans. on Knowledge and Data Engineering, 2003,15(3):541−554. [doi: 10.1109/TKDE.2003.1198389]

[9] Guha S, Kim C, Shim K. XWAVE: Approximate extended wavelets for streaming data. In: Nascimento MA, Özsu MT, Kossmann D, Miller RJ, Blakeley JA, Schiefer KB, eds. Proc. of the 30th Int'l Conf. on Very Large Data Bases. Toronto: Morgan Kaufmann Publishers, 2004. 288−299.

[10] Guha S, Harb B. Wavelet synopsis for data streams: Minimizing non-euclidean error. In: Grossman RL, Bayardo R, Bennett K, Vaidya J, eds. Proc. of the 11th ACM SIGKDD Int'l Conf. on Knowledge Discovery in Data Mining. New York: ACM Press, 2005. 88−97.

[11] Karras P, Mamoulis N. One-Pass wavelet synopses for maximum-error metrics. In: Böhm K, Jensen CS, eds. Proc. of the 31st Int'l Conf. on Very Large Data Bases. Trondheim: VLDB Endowment, 2005. 421−432.

[12] Palpanas T, Vlachos M, Keogh E, Gunopulos D, Truppel W. Online amnesic approximation of streaming time series. In: Proc. of the 20th Int'l Conf. on Data Engineering. Los Alamitos: IEEE Computer Society, 2004. 339−349. http://doi.ieeecomputersociety.org/10.1109/ICDE.2004.1320009

[13] Zhao YC, Zhang SC. Generalized dimension-reduction framework for recent-biased time series analysis. IEEE Trans. on Knowledge and Data Engineering, 2006,18(2):231−244. [doi: 10.1109/TKDE.2006.30]

[14] Bulut A, Singh A K. SWAT: Hierarchical stream summarization in large networks. In: Dayal U, Ramamritham K, Vijayaraman TM, eds. Proc. of the 19th Int'l Conf. on Data Engineering. Los Alamitos: IEEE Computer Society, 2003. 303−314.

[15] Potamias M, Patroumpas K, Sellis T. Amnesic online synopses for moving objects. In: Proc. of the 15th ACM Int'l Conf. on Information and Knowledge Management. New York: ACM Press, 2006. 784−785. http://doi.acm.org/10.1145/1183614.1183729

[16] Aggarwal CC, Han J, Wang J, Yu PS. On demand classification of data streams. In: Kohavi R, Gehrke J, DuMouchel W, Ghosh J, eds. Proc. of the 10th ACM SIGKDD Int'l Conf. on Knowledge Discovery in Data Mining. New York: ACM Press, 2004. 503−508.

[17] Gavrilov M, Anguelov D, Indyk P, Motwani R. Mining the stock market: Which measure is best? In: Proc. of the 6th ACM SIGKDD Int'l Conf. on Knowledge Discovery in Data Mining. New York: ACM Press, 2000. 487−496. http://doi.acm.org/10.1145/347090.347189

[18] Keogh E, Xi X, Wei L, Ratanamahatana CA. The UCR time series classification/Clustering homepage. 2007. http://www.cs.ucr.edu/~eamonn/time_series_data/

[19] SFR-USD tickwise stock data set. 2001. http://www-psych.stanford.edu/~andreas/Time-Series/Data/

## 附中文参考文献:

[7] Boggess A,Narcowich FJ,著;芮国胜,等,译.小波与傅里叶分析基础.北京:电子工业出版社,2004.

**陈华辉**(1964－),男,浙江鄞县人,博士,副教授,主要研究领域为数据流,数据挖掘.

**钱江波**(1974－),男,博士,副教授,CCF 会员,主要研究领域为数据库和数据流管理技术,数据挖掘,逻辑电路设计.

**施伯乐**(1936－),男,教授,博士生导师,CCF 高级会员,主要研究领域为数据库,知识库.

**陈叶芳**(1973－),女,讲师,CCF 会员,主要研究领域为数据挖掘,计算机辅助教学.