

# 基于函数的时间序列分段线性表示方法

谢福鼎<sup>1</sup> 王赫楠<sup>2</sup> 张永<sup>2</sup> 孙岩<sup>2</sup>

(辽宁师范大学城市与环境学院 大连 116029)<sup>1</sup> (辽宁师范大学计算机系 大连 116081)<sup>2</sup>

**摘要** 考虑到时间序列的时间特性对不同区段的影响以及时间序列数据动态增长的实际情况,在 RPAA(Reversed Piecewise Aggregate Approximation)和 PAA(Piecewise Aggregate Approximation)方法的基础上,提出了一种新的时间序列分段线性表示方法 FPAA(Function Piecewise Aggregate Approximation)。FPAA 方法通过定义函数影响因子,克服了 RPAA 和 PAA 方法的不足。该方法具有线性时间复杂度,满足下界定理,并且支持时间序列的在线划分。实验表明,与 PAA 方法和 RPAA 方法相比,所提出的方法可以较有效地进行时间序列的在线查询。

**关键词** 时间序列,分段线性表示,时间特性,影响因子,在线划分

中图分类号 TP311 文献标识码 A

## Method of Time Series Piecewise Linear Representation Based on the Function

XIE Fu-ding<sup>1</sup> WANG He-nan<sup>2</sup> ZHANG Yong<sup>2</sup> SUN Yan<sup>2</sup>

(Institute of City and Environment, Liaoning Normal University, Dalian 116029, China)<sup>1</sup>

(Department of Computer Science, Liaoning Normal University, Dalian 116081, China)<sup>2</sup>

**Abstract** Considering the actual situation of the different influence on the different segments in terms of time property of time series and the dynamic growth data of time series, a new method FPAA(Function Piecewise Aggregate Approximation) of piecewise linear representation was proposed based on the method of RPAA(Reverse Piecewise Aggregate Approximate) and PAA(Piecewise Aggregate Approximate). The proposed method overcomes the disadvantages of RPAA and PAA by defining the influence factor of function. FPAA has the linear complexity, satisfies lower bounding lemma and supports online segmentation of time series. Compared with the methods of PAA and RPAA, the FPAA method can effectively query time series online.

**Keywords** Time series, Piecewise linear representation, Time property, Influence factor, Online segmentation

## 1 引言

一般来说,时间序列是一组按照时间或空间顺序观察所得的有序数据,在金融<sup>[1-3]</sup>、生物医药<sup>[4,5]</sup>和地质<sup>[6]</sup>等诸多领域都有广泛应用,如金融证券市场中股票价格的波动、生物医学中某一症状病人在每个时刻的心跳变化、石油工业生产领域中某一井口在某个深度的录井读数等等。作为这样一种重要的高维数据类型,直接分析和处理的难度较大。因此,时间序列的降维问题已经成为数据挖掘领域广泛关注的热点问题。

线性分段技术作为一种简单直观的降维方法,具有下列优点:1)支持快速相似性搜索;2)支持时间序列新的距离度量,包括模糊查找、加权序列、DTW 距离、信息反馈等等;3)同时支持文本和数据序列;4)支持新的聚类、分类算法;5)支持奇异点检测。目前该方法已得到了广泛的应用<sup>[7]</sup>。线性分段算法的关键问题在于如何选择合适的线段数目以及如何选择合适的分段点。根据这两个问题的不同解决方法,我们将线性分段算法分为以下两种类型:1)限制分段误差,通过控制分

段误差来选择合适的分段点。根据分段误差控制的方法不同,文献[8]将这类分段算法分为以下3种:滑动窗口技术(Sliding Window,简称 SW)、自顶向下(Top-Down,简称 TD)和自底向上(Bottom-Up,简称 BU)。2)限制分段数  $K$ ,给定的时间序列产生的 PLR(Piecewise linear Representation)表示只包括  $K$  条直线。

文献[9,10]分别独立提出了时间序列分段聚集近似的 PAA 方法。这种方法简单直观,能够支持任意长度的相似性查询、所有的 Minkowski 度量和加权欧氏距离,而且能够用于索引提高查询的效率。Keogh 等人的实验表明将 PAA 方法用于时间序列的索引,使得相似性查询的效率比 DFT 表示方法提高了 1~2 个数量级<sup>[9]</sup>,而且能够用于加权欧氏距离计算。虽然 Keogh 和 Yi 等人提出的 PAA 方法能够有效地划分时间序列,对时间序列进行降维,但是在时间序列的演变过程中,对于不同时间区段的数据,其对未来时间区段数据的影响是不同的,越接近时间序列当前时间的区段影响越大,其值的对时间序列的预测的参考价值也越大。而远离的区段则影响较小,其值的参考价值也越小,所以有必要在计算时间序列

到稿日期:2010-12-23 返修日期:2011-05-09 本文受国家自然科学基金(10771092),辽宁省博士启动基金(20081079)资助。

谢福鼎(1965—),男,博士,教授,主要研究方向为人工智能、数据挖掘;王赫楠(1986—),女,硕士,主要研究方向为人工智能、数据挖掘, E-mail: wanghenan09@126.com(通信作者);张永(1975—),男,博士,副教授,主要研究方向为机器学习、智能计算、可信计算。

间距离时将不同时间区段的数据影响区分开来,但 PAA 对于不同区段的数据是用均值来表示的,在计算不同区段数据的距离时则相应地同等对待,没有考虑到时间序列的时间特性。王等<sup>[1]</sup>考虑到时间序列的时间特性,将影响因子应用到时间序列的分段中,提出一种 PAA 的改进表示方法,记为 RPAA 方法。但 RPAA 方法将滑动窗口首先作用在时间序列的尾端,然后沿着时间轴的逆方向移动,计算各区段的均值,以及相应的影响因子,这使得 RPAA 不能实现在线实时划分时间序列,对于时间序列数据动态增长的实际情况是不利的。

本文考虑了时间序列的时间特性,提出了函数影响因子的概念。该方法不仅继承了 RPAA 的优点,而且实现了时间序列的在线实时划分,并能够用于加权欧氏距离计算。同时此方法支持双向划分,既可以沿时间轴正向划分,也可以沿时间轴逆向划分,与 PAA 方法和 RPAA 方法相比,此方法有较好的优越性。

本文第 2 节介绍时间序列的 PAA 和 RPAA 方法及相关内容;第 3 节介绍基于函数的分段线性表示方法 FPAA;第 4 节通过实验验证 FPAA 方法的可行性;最后总结全文。

## 2 PAA 以及 RPAA 表示方法

### 2.1 相关定义

**定义 1(时间序列)** 给定一个属性  $P$ , 时间序列是一个具有  $n$  个值的集合  $\{\langle p_1, a_1 \rangle, \langle p_2, a_2 \rangle, \dots, \langle p_n, a_n \rangle\}$ 。这里有  $n$  个时间值, 每个时间值与一个属性  $P$  的值相关联。通常这些值由特殊的经过明确周期定义的时间点来标识。在这种情况下, 可以认为这些取值是一个向量  $\langle p_1, p_2, \dots, p_n \rangle$ 。

**定义 2(时间序列长度)** 对于有限长时间序列  $P = p_1, p_2, \dots, p_n$ ,  $P$  的长度为组成  $P$  的实数个数, 记为  $|P|$ , 即  $|P| = n$ 。对于无限长时间序列,  $P$  的长度定义为  $|P| = \infty$ 。

**定义 3(时间序列的子序列)** 给定长度为  $n$  的时间序列  $P$  的序列区段  $S$ , 其是在  $P$  中点  $p_i$  开始, 数量为  $\omega (1 \leq \omega \leq n)$  个的连续位置点组成  $P$  的一个抽样, 即:  $S_i = p_i, p_{i+1}, \dots, p_{i+\omega-1}$ , 其中  $1 \leq i \leq n - \omega + 1$ 。

**定义 4(滑动窗口)** 给定一个长度为  $n$  的时间序列  $P$  和一个用户给定的区段长度  $\omega (1 \leq \omega \leq n)$ ,  $P$  的所有子序列的矩阵  $S$  可以通过在  $P$  上滑动一个宽度为  $\omega$  的窗口, 将每一个区段  $S_i$  放入  $S$  的第  $S$  行得到,  $S$  的大小为  $(n - \omega + 1) * \omega$ ,  $\omega$  为滑动窗口的大小。

### 2.2 PAA 方法简介

Keogh 和 Yi 等人分别独立提出了时间序列的 PAA 表示方法, 用等宽度窗口分割时间序列, 每个窗口内的时间序列用窗口平均值来表示, 就得到了时间序列的一种分段线性表示, 它的输入参数为窗口大小, 记为  $\omega$ 。

为了将长度为  $n$  的时间序列降维, 首先将其分为  $N$  维个大小相同的帧, 在每一帧中计算落入帧中的数据均值, 以该向量作为原时间序列的表示。长度为  $n$  的时间序列  $P$  在  $N (1 \leq N \leq n)$  维空间中表示为向量  $\langle P_1', P_2', \dots, P_N' \rangle$ , 其中第  $i (1 \leq i \leq N)$  个元素  $P_i'$  的计算式为:

$$P_i' = 1/\omega \sum_{j=\omega(i-1)+1}^{\omega i} p_j (\omega = \lceil n/N \rceil) \quad (1)$$

图 1 给出了 PAA 表示示例。

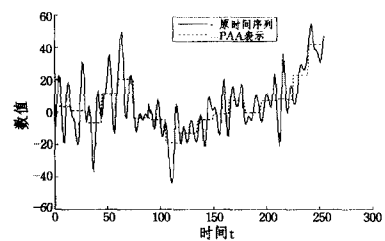


图 1 PAA 表示图

### 2.3 RPAA 表示方法

针对时间序列的时间特性, 王元珍等人提出时间序列的 RPAA 表示法, 在 PAA 表示方法的基础上引入了影响因子  $\rho (0 \leq \rho \leq 1)$  的概念。在 PAA 中计算索引时将滑动窗口作用在原始时间序列  $P$  的前端, 然后沿着时间轴移动, 计算各区段的均值, 按照原时间轴的方向组成索引向量。但加入影响因子后, 这种按照时间轴方向的运算将必须每次计算当前区段与最后区段的位置距离  $(j-i)$ , 这势必会导致  $\rho^{j-i}$  多次重复计算。为了解决这个问题, RPAA 方法将滑动窗口首先作用在时间序列  $P$  的尾部, 利用大小为  $\omega$  的滑动窗口沿着时间轴逆向滑动, 每滑动一次, 计算均值和影响系数, 从而计算出 RPAA 中该区段的索引量。计算公式为:

$$P' = \rho^{i-1} / \omega \sum_{j=n-\omega i+1}^{n-(i-1)\omega} p_j (i=1, 2, \dots, N; \omega = \lceil n/N \rceil) \quad (2)$$

由于 RPAA 方法是将滑动窗口作用在时间轴的尾部, 对于动态增长的数据流是不适用的, 因此这种方法是不支持在线划分的, 而且影响因子的取值也是要根据数据的不同而调整的, 这样势必影响计算的准确度。

### 2.4 欧氏距离

欧氏距离作为衡量时间序列距离的方法被广泛使用。给定长度为  $n$  的时间序列  $P = p_1, p_2, \dots, p_n$  和  $Q = q_1, q_2, \dots, q_n$ , 则  $P$  和  $Q$  之间的欧氏距离为:

$$D(P, Q) = \sqrt{\sum_{i=1}^n (p_i - q_i)^2} \quad (3)$$

为了防止递加数过多而导致欧氏距离过大, 可以计算加权的欧氏距离:

$$D(P, Q) = \sqrt{1/n \sum_{i=1}^n (p_i - q_i)^2} \quad (4)$$

由于函数  $f(x) = x^2$  为单调函数, 因此在实际计算中可以计算平方后的加权欧氏距离:

$$D(P, Q) = 1/n \sum_{i=1}^n (p_i - q_i)^2 \quad (5)$$

PAA 表示支持对于平方后的加权欧氏距离的计算, 将一个查询  $Q$  也降维  $Q' = (Q_1', Q_2', \dots, Q_N')$ , 其中

$$Q_i' = 1/\omega \sum_{j=\omega(i-1)+1}^{\omega i} q_j (\omega = \lceil n/N \rceil) \quad (6)$$

则此时计算 PAA 表示平方后的加权欧氏距离为:

$$\begin{aligned} D(P, Q) &= 1/N \sum_{i=1}^N (P_i' - Q_i')^2 \\ &= 1/N \sum_{i=1}^N \left[ \frac{1}{\omega} \sum_{j=\omega(i-1)+1}^{\omega i} (p_j - q_j) \right]^2 \end{aligned} \quad (7)$$

式中,  $\omega = \lceil n/N \rceil$ 。

## 3 时间序列的 FPAA 表示方法

### 3.1 时间序列的 FPAA 表示方法

考虑到时间序列的时间特性, 以及 RPAA 方法不适用于在线划分的不足, 我们提出了一种时间序列的函数表示方法

FPAA, 其将函数引入到时间序列的划分中, 从而实现在线划分的同时兼顾了时间序列的时间特性。

**定义 5 (函数影响因子)** 函数影响因子  $f(x)$  满足两个条件: (1)  $0 < f(x) < 1$ ; (2)  $f(x)$  是单调递增的函数。

这里, 我们定义了  $f(x) = 1 - [1/(x+1.1)] (0 \leq x)$ , 作为函数影响因子, 此函数满足函数影响因子的条件: 在  $0 \leq x$  时,  $0 < f(x) < 1$ , 且是单调递增的。

FPAA 方法计算索引时将滑动窗口作用在原始时间序列  $P$  的前端, 然后沿着时间轴的方向移动, 计算各区段的均值和函数影响因子的乘积, 按照时间轴的方向组成了索引向量, 此时:

$$P_i' = 1/\omega f(i) \sum_{j=n-\omega i+1}^{n-(i-1)\omega} p_j \quad (i=0, 1, \dots, N-1; \omega = \lceil n/N \rceil) \quad (8)$$

同时也可以将函数影响因子作用在原始时间序列的尾端, 按照时间轴的逆方向组成索引向量, 此时有:

$$P_i' = 1/\omega f(i) \sum_{j=n-\omega i+1}^{n-(i-1)\omega} p_j \quad (i=N-1, \dots, 0, 1; \omega = \lceil n/N \rceil) \quad (9)$$

所以 FPAA 方法支持双向时间序列划分。算法如下:

算法: 构造时间序列的 RPAA 表示

输入:  $P = p_1, p_2, \dots, p_n$  // 长度为  $n$  的时间序列

$\omega (1 \leq \omega \leq n)$  // 滑动窗口大小

输出: 时间序列  $P$  的 FPAA 表示

处理:

- (1)  $N = \lceil n/\omega \rceil$ ; // 计算总的区段数 (最后一段数据量不足  $\omega$  则全部数据作为一区段)
- (2)  $P' = \varphi$ ; // FPAA 向量
- (3) For(int  $i=0$ ;  $i < N$ ;  $i++$ ) begin
- (4) 计算  $P_i' = 1/\omega f(i) \sum_{j=n-\omega i+1}^{n-(i-1)\omega} p_j$
- (5) 将  $P_i'$  加到  $P'$  中
- (6) 返回  $P'$

FPAA 方法利用大小为  $\omega$  的滑动窗口沿时间轴滑动, 每滑动一次, 计算均值和函数影响因子, 从而计算出 FPAA 中该区段的索引量, 由此可得 FPAA 的时间复杂度为  $O(n)$ 。又因为 FPAA 需要额外的  $N$  维向量  $P'$  来保存中间的索引向量, 所以 FPAA 的空间复杂度为  $O(n)$ 。

### 3.2 距离计算

对于 FPAA 时间序列的距离计算仍可以采用平方后的加权欧氏距离, 此时:

$$D(P, Q) = 1/N \sum_{i=1}^N (P_i' - Q_i')^2 = \frac{1}{N} \sum_{i=1}^N [f(i) * \frac{1}{\omega} \sum_{j=n-\omega i+1}^{n-(i-1)\omega} (p_j - q_j)]^2 \quad (10)$$

引入  $f(x)$  后, 相邻两区段间的计算为  $f(x)/\sqrt{N}$ 。随着时间的向前推进, 函数的值越来越大, 当前区段对后面区段的影响越来越大, 符合时间序列的时间特性。

### 3.3 满足下界定理

Faloutsos 给出降维下界定理 (Lower Bounding), 即:

$$D_{\text{index}}(P, Q) \leq D(P, Q) \quad (11)$$

同样的 FPAA 方法也满足下界定理。

证明: 由于  $0 < f(x) < 1$ , 因此

$$D_{\text{FPAA}}(P, Q) = 1/N \sum_{i=1}^N [f(i) * \frac{1}{\omega} \sum_{j=n-\omega i+1}^{n-(i-1)\omega} (p_j - q_j)]^2$$

$$\begin{aligned} &< 1/N \sum_{i=1}^N [1/\omega \sum_{j=n-\omega i+1}^{n-(i-1)\omega} (p_j - q_j)]^2 \\ &= 1/N \sum_{i=1}^N [1/\omega \sum_{j=\omega(i-1)+1}^{\omega i} (p_j - q_j)]^2 \\ &= D_{\text{PAA}}(P, Q) \end{aligned} \quad (12)$$

因为  $D_{\text{PAA}}(P, Q)$  满足欧氏距离的下界定理<sup>[12]</sup>, 并从证明中得出  $D_{\text{FPAA}}(P, Q)$ , 针对  $D_{\text{PAA}}(P, Q)$  满足下界定理, 因此得出  $D_{\text{FPAA}}$  也满足下界定理。

## 4 实验

### 4.1 实验内容

本文使用 Java 3.6 分别实现 PAA, RPAA 以及 FPAA 算法。实验环境为 Windows XP Server, 120G 硬盘, 1G 内存。本实验使用 E. Keogh 提供的数据集 (<http://www.cs.ucr.edu/~eamonn/>), 选取了 OliveOil, Swedishlef, Beef 3 个数据集中的时间序列, 其中 Beef 含有 120 条时间序列, 时间序列长度为 470; Swedishlef 中含有 1125 条时间序列, 每条时间序列长度为 128; OlivOil 中含有 60 条时间序列, 每条时间序列长度为 570。以及 EEG Database 中的 egg\_full 数据集, 大小为 700M, 数据来自大量实验研究产生的脑电图数据, 每条时间序列包含 256 个数据点。

对原始时间序列进行查询, 可能出现的错误情况如下:

(1) 在原始数据中两点距离小于给定的阈值  $\epsilon$ , 但是在索引空间中该两点距离却大于  $\epsilon$ , 从而发生漏查现象。

(2) 在索引空间中两点距离小于给定的阈值  $\epsilon$ , 但是在原始数据中这两点距离却大于  $\epsilon$ , 从而在多索引空间的点查询的结果中出现错查现象。

由于已经证明 FPAA 表示方法满足下界定理, 保证不会出现漏查问题, 因此我们使用评价标准:

$$\text{查错率} = \text{查错记录数} / \text{总的查询记录} * 100\%$$

### 4.2 实验结果

在以上提到的标准数据集中的部分实验结果如表 1、表 2 所列。

表 1 3 种方法的查错率比较

方法	PAA <sup>[10]</sup>	RPAA <sup>[11]</sup>	FPAA
Beef ( $\omega=10$ )	28.5%	37.5%	21.4%
OliveOil ( $\omega=10$ )	21.4%	25%	17.8%
Swedishlef ( $\omega=8$ )	33.3%	33.3%	26.7%

表 2 3 种方法的查错率比较

方法	PAA <sup>[10]</sup>	RPAA <sup>[11]</sup>	FPAA
Co2a0000387 ( $\omega=8$ )	25.8%	22.2%	22.2%
Co2c0000373 ( $\omega=8$ )	29%	29%	25.8%
Co3a0000459 ( $\omega=8$ )	16.1%	12.6%	12.6%

### 4.3 实验分析

从表 1 中可以看出, FPAA 方法比 PAA 方法和 RPAA 方法的查错率都要低, 说明 FPAA 方法在这 3 组数据上是有有效的。从表 2 中可以看出 FPAA 方法的查错率要低于 PAA 方法, 而 FPAA 方法的查错率和 RPAA 方法的查错率在部分数据上是相同的, 但 FPAA 方法可以实时在线划分时间序列, 这对动态增长的数据环境是十分有用的。

**结束语** 在动态增长的数据流环境中实时准确地划分数

(下转第 160 页)

- [7] Mislove A, Marcon M, Krishna P, et al. Measurement and Analysis of Online Social Networks[C] // Proc. of IMC'07, New York: ACM, 2007; 29-41
- [8] Albert R, Jeong H, Barabasi A L. Error and attack tolerance of complex networks[J]. Nature, 2000, 406: 378-382
- [9] Hay M, Miklau G, Jensen D, et al. Anonymizing social networks[R], 07-19. University of Massachusetts Amherst, 2007
- [10] Zheleva E, Getoor L. Preserving the privacy of sensitive relationships in graph data[C] // Proc. of the 1st ACM SIGKDD Workshop on Privacy, Security, and Trust in KDD. Berlin: ACM, 2007; 153-171
- [11] Campan A, Truta T M. A clustering approach for data and structural anonymity in social networks[C] // Proc. of the 2nd ACM SIGKDD International Workshop on Privacy, Security, and Trust in KDD. Las Vegas: ACM, 2008; 93-104
- [12] Liu L, Wang J, Liu J, et al. Privacy preserving in social networks against sensitive edge disclosure[R]. CMIDA-HiPSCCS 006-08. Department of Computer Science, University of Kentucky, 2008
- [13] Liu K, Terzi E. Towards identity anonymization on graphs[C] // Proc. of the 2008 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2008; 93-106
- [14] Zhou B, Pei J. Preserving privacy in social networks against neighborhood attacks[C] // Proc. of the 24th IEEE International Conference on Data Engineering. Washington D C: IEEE, 2008: 506-515
- [15] Zou Lei, Chen Lei, Özsu M T. K-Automorphism: General Framework for Privacy Reserving Network Publication[J]. Proceedings of the VLDB Endowment, 2009, 2(1): 946-957
- [16] Cormode G, Srivastava D, Yu T, et al. Anonymizing bipartite graph data using safe groupings[J]. Proceedings of the VLDB Endowment, 2008, 1(1): 833-844
- [17] Panda G K, Mitra A, Prasad A, et al. Applying l-Diversity in anonymizing collaborative social network[J]. International Journal of Computer Science and Information Security, 2010, 8(2): 324-329
- [18] 谭建荣, 岳小莉, 陆国栋. 图形相似的基本原理、方法及其在结构模式识别中的应用[J]. 计算机学报, 2002, 25(9): 59-967
- [19] He P R, Zhang W J, Li Q. Some further development on the eigen system approach for graph isomorphism detection [J]. Journal of Franklin Institute, 2005, 342(6): 657- 673
- [20] Hay M, Miklau G, Jensen D, et al. Resisting Structural Reidentification in Anonymized Social Networks [C] // Proc. of the VLDB Endowment. New Zealand, Auckland: VLDB Endowment, 2008; 102-114

(上接第 155 页)

据序列,是当前一项重要且有现实意义的课题,它能够帮助用户发现数据中隐藏着的丰富知识,揭示数据变化规律及预测序列的发展趋势。本文提出了一种基于函数的时间序列分段线性表示方法,与 PAA 和 RPAA 方法相比,本文的方法不仅考虑到了时间序列的时间特性,而且能够实时划分时间序列,结果表明 FPAA 方法能够比较准确地表示原始时间序列的特性,进行有效的在线查询。但对于不同的数据,如何找到更加合适的函数影响因子来表现原始时间序列的时间特性,是一项有待研究的课题。

### 参 考 文 献

- [1] Fu T C, Chung F L, Ng V, et al. Pattern Discovery from Stock Time Series Using Self-Organizing Maps[C] // Proceedings of the 7th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Francisco: ACM Press, 2001: 23-27
- [2] Gavrilov M, Anguelov D, Indyk P, et al. Mining the Stock Market; Which Measure is Best? [C] // Proceedings of the 6th ACM International Conference on Knowledge Discovery and Data Mining. Boston: ACM Press, 2000; 487-496
- [3] Mantegna R N. Hierarchical Structure in Financial Markets[J]. European Physical Journal, 1999(11): 193-197
- [4] Bar-joseph Z, Gerber G, Gifford D, et al. A new Approach to Analyzing Gene Expression Time Series Data[C] // Proceedings of the 6th Annual International Conference on Research in Computational Molecular Biology. New York, USA: ACM Press, 2002; 39-48
- [5] Koski A, Juhola M, Meriste M. Syntactic Recognition of ECG Signals by Attributes Finite Automata[J]. Pattern Recognition, 1995, 28(12): 1927-1940
- [6] Harm S K, Reichenbach S, Goddard S E, et al. Data Mining in a Geospatial Decision Support System for Drought Risk Management[C] // Proceeding of the 1st National Conference in Digital Government. Los Angeles, CA, 2002; 9-16
- [7] Keogh E, Kasetty S. A Survey and Empirical Demonstration[C] // Proceeding of the 8th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Ganada; Edmonton, Alberta, 2002; 102-111
- [8] Keogh E, Chu S, Hart D, et al. An online algorithm for segmenting time series[C] // Proceedings of IEEE International Conference on Data Mining. Los Alamitos: IEEE Computer Society Press, 2001; 289-296
- [9] Keogh E, Chakrabarti K, Pazzani M J, et al. Dimensionality Reduction for Fast Similarity Search in Large Time Series Databases[J]. Knowledge and Information Systems, 2001, 3(3): 263-268
- [10] Yi B K, Faloutsos C. Fast Time Sequence Indexing for Arbitrary Lp Norms[C] // Proceeding of the 26th International Conference on Very Large Databases. San Francisco: Morgan Kaufmann Publishers Inc, 2000; 385-394
- [11] 王元珍, 李俊奎, 曹忠升. 一种基于时间特性的时间序列建模表示[J]. 计算机科学, 2007, 34(3): 83-86
- [12] Keogh E, Pazzani M. A Simple dimensionality reduction technique for fast similarity search in large time series databases[C] // Proceeding of the 4th Pacific-Asia Conference on Knowledge Discovery and Data Mining. Kyoto, Japan; Berlin, 2000; 122-133