文章编号: 1001-0920 (2017) 01-0001-11 DOI: 10.13195/j.kzyjc.2016.0462

时间序列数据挖掘的相似性度量综述

陈海燕[a,b†], 刘晨晖[a], 孙 博[a]

(南京航空航天大学 a. 计算机科学与技术学院，b. 软件新技术与产业化协同创新中心，南京 210016)

摘 要: 在时间序列数据挖掘中, 时间序列相似性是一个重要的概念. 对于诸多算法而言, 能否与一种合适的相似性度量方法结合应用, 对其挖掘性能有着关键影响. 然而, 至今仍没有统一的度量相似性的方法. 对此, 首先综述了常用的相似性度量方法, 分析了各自的优点与不足; 其次, 讨论了近年来出现的时序相似性的新解释及其度量方法; 再次, 探讨了相似性度量在时序挖掘任务中的应用以及与挖掘精度的关系; 最后给出了关于时序相似性度量进一步的研究方向.

关键词: 时间序列数据挖掘；时间序列相似性；相似性度量；挖掘精度

中图分类号: TP273 **文献标志码:** A

Survey on similarity measurement of time series data mining

CHEN Hai-yan[a,b†], *LIU Chen-hui*[a], *SUN Bo*[a]

(a. School of Computer Science and Technology，b. Collaborative Innovation Center of Novel Software Technology and Industrialization，Nanjing University of Aeronautics and Astronautics，Nanjing 210016，China)

Abstract: Similarity measure is an important concept in time series data mining. For many data mining algorithms, whether it can be used in combination with a suitable time series similarity measure method has a key influence on mining performance. However, there is no uniform definition and measure of similarity. Therefore, we first introduce the most popular similarity measures, and analyze the advantages and disadvantages of each measure. Then, the new interpretations of the time series similarity and the corresponding measures are discussed. Furthermore, we analyze the applications of similarity measures in clustering, classification and regression of time series data, and the relationship between similarity measure and mining precision. Finally, several directions for the future research are given.

Keywords: time series data mining；time series similarity；similarity measure；mining accuracy

0 引 言

时间序列是按时间顺序获得的一系列观测值. 随着信息技术不断发展, 时间序列数据量与日俱增, 并存在于社会生活的各个方面, 如金融收益、气象研究、网络安全等[1]. 时间序列数据挖掘的主要任务是分析时间序列的历史数据, 预测将来一段时间的可能数据, 并分析未来发展趋势.

时间序列挖掘算法在训练过程中, 通常都需要计算输入样本之间的相似度. 对于诸多的机器学习算法而言, 选择一种合适的度量方式评估数据之间的相似性对学习算法的性能有着非常关键的影响. 许多研究工作也表明, 经过精心设计的相似性度量可以显著提高算法的性能[2]. 同时, 在对一批时间序列数据进行聚类、分类和回归等挖掘工作之前, 如果能先对样本点的时间序列相似性进行度量, 发现它们之间相似的规律或特征, 舍弃相似度偏低的序列数据, 着重研究相似度较高的数据, 则必能有效提高数据后续挖掘的精度和效率.

目前, 人们针对时序相似性问题从以下几个方面进行了研究: 1) 如何定义和度量时间序列的相似性; 2) 如何使用相似性度量指导时序数据挖掘; 3) 相似性度量与时间序列数据挖掘的关系. 下面将以这 3 个方面为线索, 对时间序列的相似性度量进行详细综述. 首先概述时间序列相似性度量的基本概念及其影响因素; 其次, 对常用的相似性度量方法进行特性分析; 再次, 从 3 个方面探讨时间序列相似性度量的研究进

收稿日期: 2016-04-18；**修回日期:** 2016-09-13.

基金项目: 国家自然科学基金项目(61501229)；中央高校基本科研业务费专项资金项目(NS2015091, NJ20160013).

作者简介: 陈海燕 (1979−), 女, 讲师, 博士, 从事机器学习、数据挖掘等研究；刘晨晖(1992−), 女, 硕士生, 从事数据挖掘、机器学习的研究.

†通信作者. E-mail: chenhaiyan@nuaa.edu.cn

展; 最后进行总结, 并给出进一步的研究方向.

1 基本概念及影响因素

1.1 时间序列的定义

时间序列是将研究数据的变化过程按时间先后顺序记录下来而形成的序列[3], 定义如下:

定义 1 假设大部分时间序列都是离散的, 记时间序列为

$$S = \langle (v_1, t_1), (v_2, t_2, \cdots, (v_n, t_n)) \rangle. \tag{1}$$

其中: 设 $s_i = (v_i, t_i)$ 表示 t_i 时刻、数据为 v_i 的元素, 且 $i < j \Leftrightarrow t_i < t_j$. 采样序列的间隔相同, 所以 S 被简记为 $S = \langle s_1, s_2, \cdots, s_n \rangle$. 数据点的数目 n 表示长度, 记为 $|S| = n$. 两个数据点 s_i, s_{i+t} 之间的部分时间序列称为时间序列的子序列, 记为 $S[s_i, s_{i+t}] = s_i, s_{i+1}, \cdots, s_{s+t}$.

1.2 时间序列的相似性

时间序列相似性的定义由 Agrawal 等[4]在 1993 年提出, 描述如下:

定义 2 假设给定两时间序列 S_1 和 S_2, 一个相似性度量函数 $\mathrm{Dist}(S_1, S_2)$. 如果序列 S_1 和 S_2 满足 $\mathrm{Dist}(S_1, S_2) \leqslant \varepsilon$, 则称时间序列 S_1 和 S_2 是相似的. 其中 ε 是时间序列相似的阈值. $\mathrm{Dist}(S_1, S_2)$ 表示 S_1 与 S_2 的某种距离, 通过这种距离来度量两者的相似性.

1.3 时间序列相似性的影响因素

两个序列是否相似主要取决于变化趋势是否一致[5]. 由于时间序列数据具有维度高、数据类型多等特点, 任意两个序列之间都可能存在某种差异, 影响这些差异的因素有: 噪声、振幅平移和伸缩、线性漂移、不连续性、时间轴伸缩等变形等[6]. 图 1 给出了一个原始序列在 6 种不同影响因素下产生的序列改变.

1) 噪声: 指时间序列中无法提取的波动, 存在于大多数时间序列数据中.

2) 振幅平移: 指时间序列绕着偏高或偏低的均值上下波动.

3) 振幅伸缩: 指时间序列的振幅变高或变低.

4) 线性漂移: 指时间序列呈现出线性递增或递减的趋势.

5) 不连续性: 指时间序列在个别时间点或时间段上出现间断的现象.

6) 时间轴伸缩: 指时间序列在时间轴上按比例伸缩.

然而, 实际收集到的时间序列数据受多重因素共同影响, 这给时间序列相似性研究增加了相当的难度. 因此, 如何针对不同的序列数据找到合适的相似性度量方法就显得尤为重要.

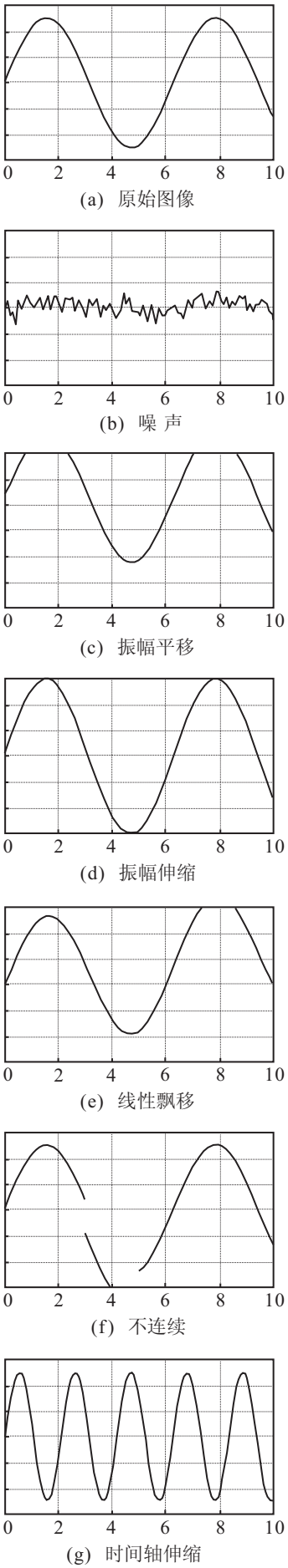


图 1 时间序列影响因素

2　经典的相似性度量方法

经典的时间序列相似性度量方法总体被分为两类: 锁步度量 (lock-step measures) 和弹性度量 (elastic measures)[7]. 锁步度量是时间序列进行 “一对一” 的比较; 弹性度量允许时间序列进行 “一对多” 的比较.

2.1　锁步度量

在两个时间序列之间点对点比较的度量方法称为锁步度量[8], 最常见的是欧氏距离 (ED)[9].

欧氏距离易于理解、计算简单且效率高, 对不同类型的数据都能适用, 是普遍首选的相似性度量方法[10]. 但欧氏距离要求被度量的两个序列长度相等, 且对于某些情况 (如两个变相的时间序列), 度量结果往往存在很大误差[11]. 例如, 序列 $X=\langle 1,1,1,10,2,3\rangle$ 与序列 $Y=\langle 1,1,1,2,10,3\rangle$ 的欧氏距离为 $D(X,Y)=128$, 距离较大, 说明 X 与 Y 具有较低的相似性, 但实际上这两个序列非常相似.

2.2　弹性度量

与锁步度量不同, 弹性度量可将两个时间序列进行 “一对多” 的比较或 “一对零” 的比较, 常见的有动态时间规整和基于编辑距离的度量.

2.2.1　动态时间规整

动态时间规整 (DTW)[12]通过将时间序列拉伸或收缩匹配来计算两个时间序列之间的相似性, 度量原理如图 2 所示.

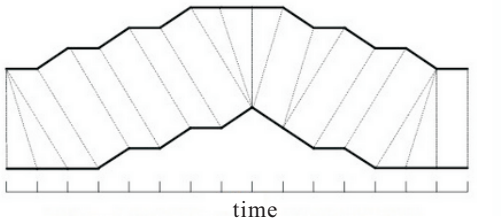


图 2　时间动态规整

图 2 中的实线表示两个时间序列, 虚线连接序列之间相似的点. DTW 使用所有相似点之间距离的综合来衡量两个时间序列之间的相似性, 计算公式如下:

$$D_{\mathrm{dtw}}(X,Y)=\begin{cases}0,\ i=j=0;\\ \infty,\ i=0\text{ or }j=0;\\ \mathrm{Com\_Dist}(x_1,y_1)+\\ \quad\min\begin{cases}D_{\mathrm{dtw}}(\mathrm{Rest}(X),\mathrm{Rest}(Y)),\\ D_{\mathrm{dtw}}(\mathrm{Rest}(X),Y),\\ D_{\mathrm{dtw}}(X,\mathrm{Rest}(Y)),\end{cases} & \text{otherwise.}\end{cases}\tag{2}$$

其中 $\mathrm{Rest}(X)$ 表示序列 X 除去第一个点之后的剩余序列.

当序列 X 与 Y 等长时, 可直接采用欧氏距离计算二者的距离; 当 X 和 Y 长度不同时, 采用 DTW 方法寻找两个时间序列中的最佳规整路径, 并计算相应的距离. 用 DTW 计算序列 $X=\langle 1,1,1,10,2,3\rangle$ 与 $Y=\langle 1,1,1,2,10,3\rangle$ 的距离为 $D_{\mathrm{dtw}}(X,Y)=18$. 可见, 相比于欧氏距离度量, DTW 的度量结果能更准确地反映两个序列之间的相似程度.

DTW 容许时间序列中的点经过自我复制之后再进行等长匹配, 克服了欧氏距离由于时间序列发生扭曲变形而无法匹配的问题[13]; 支持时间序列平移, 可灵活地处理多相位时间序列; 引入了动态窗口的时间弯曲 ε, 提高了计算效率和相似性度量精度[14]. 但 DTW 对噪声非常敏感, 若不对匹配过程进行优化, 则计算复杂度可能会达到 $O(n^2\times L)$[15], L 为序列长度. 但 DTW 不满足三角不等式, 不能成为度量函数.

2.2.2　基于编辑距离的度量方法

针对 DTW 时间复杂度高的问题, 学者们提出了基于编辑距离 (edit distance) 的时间序列度量方法. 编辑距离指两个字符串之间转换时所需要的最少编辑操作步数, 包括字符替换、插入和删除[16]. 例如两个序列 $A=\langle c,o,f,f,e,e\rangle$ 和 $B=\langle c,a,f,e\rangle$ 的编辑距离 $D(A,B)=3$, 即序列 A 转换为序列 B 所需最少编辑数为 3. 显然, 编辑距离的大小与序列相似性成反比. 基于编辑距离的相似性度量方法主要有以下 3 种.

1) 最长公共子序列.

如果一个序列 S 分别是两个或多个已知序列中符合条件的最长子序列, 则 S 就是已知序列的最长公共子序列 (LCSS)[17], 其长度就是两序列之间的距离. LCSS 用两字符串最大公共字串的长度与最长字符串长度的比值进行相似性度量[18], 计算公式为

$$D_{\mathrm{lcss}}(X,Y)=\begin{cases}0,\ i=j=0;\\ 1+D_{\mathrm{lcss}}(\mathrm{Rest}(X),\mathrm{Rest}(Y)),\ |X_{d,1}-Y_{d,1}|\leqslant\varepsilon;\\ \max\begin{cases}D_{\mathrm{lcss}}(\mathrm{Rest}(X),Y),\\ D_{\mathrm{lcss}}(X,\mathrm{Rest}(Y)),\end{cases} & \text{otherwise.}\end{cases}\tag{3}$$

当两个时间序列在大部分时间段都有相似形态, 仅在小范围内有扭曲突变或断点时, 欧氏距离和 DTW 将不再适用, 这时应采用 LCSS 距离度量. 例如, 两个时间序列 $X=\langle 1,1,1,10,2,3\rangle$ 和 $Y=\langle 1,1,1,2,10,3\rangle$, 欧氏距离和 DTW 距离分别为 $D_{\mathrm{ed}}(X,Y)=128$, $D_{\mathrm{dtw}}(X,Y)=18$, 而 $D_{\mathrm{lcss}}(X,Y)=5$, 可见, 此时 LCSS 距离有较高的准确度. 由于 LCSS 距离能克服

时间序列的短期突变或由断点造成的相似性变差的问题, 使它对噪声有较强的适应能力[19]. 但是, LCSS在处理振幅平移以及时间轴伸缩等形变问题时的效果不理想.

2) 实序列编辑距离.

实序列编辑距离 (EDR) 主要针对距离函数对时序数据中的噪声、位移及误差较为敏感等问题而提出[20], 被广泛用于生物信息学及语音识别领域中. 序列 X 与 Y 之间的EDR是指, 将 X 转换为 Y 需要的插入、删除或替换时所需要的操作数目. 序列 X 与 Y 之间的EDR距离定义为[21]

$$D_{\rm edr}(X,Y)=\begin{cases} i,\ j=0;\\ j,\ i=0;\\ \min\begin{cases}\text{subcost}(X,Y)+\\ D_{\rm edr}(\text{Rest}(X),\text{Rest}(Y)),\\ 1+D_{\rm edr}(\text{Rest}(X),Y),\\ 1+D_{\rm edr}(X,\text{Rest}(Y)),\end{cases} & \text{otherwise.}\end{cases}\tag{4}$$

其中subcost计算如下[22]:

$$\text{subcost}(X,Y)=\begin{cases}0,\ |X_i-Y_j\leqslant\varepsilon|;\\ 1,\ |X_i-Y_j>\varepsilon|.\end{cases}\tag{5}$$

EDR通过阈值 ε 将时间序列中的两个元素量化为0和1 (LCSS中也采用了相同手法), 有效减少了噪声的影响, 在处理异常数据时相比于ED和DTW有更好的鲁棒性; 同时, 寻找将一段序列转换为另一段序列所需要的最少编辑操作数, 使EDR能够处理本地时间转换问题[23]; EDR根据两个时间序列之间间隙长度定义了“处罚值”, 即subcost, 使它比LCSS有更高的精度. 例如, 4个时间序列 $Q=\langle1,2,3,4\rangle$, $A=\langle10,9,8,7\rangle$, $B=\langle1,100,2,3,4\rangle$ 和 $C=\langle1,100,101,3,4\rangle$, Q 为查询序列, 分别采用不同相似性度量方法对 A、B、C 序列与 Q 序列之间的相似性进行排序. 正确的相似排序序列为 B, C, A. 欧氏距离度量结果为 A, B, C; DTW与欧氏距离的结果相同; LCSS的度量结果为 B, C, A; 而EDR的度量结果为 B, C, A, 与正确的相似性排序序列一致. 但是, EDR不满足三角不等式, 不能成为度量函数.

3) 实补偿编辑距离.

实补偿编辑距离 (ERP)[24]是时间序列相似性度量方法中正处于发展阶段的一种方法. 在运用ERP距离进行时间序列度量时, 可在两个序列中添加一些符号 (即间隙) 把两条长度不同的序列排列成相同的长度[25], 从而方便一对一比较. 序列 X 与 Y 之间的ERP距离定义为

$$D_{\rm erp}(X,Y)=\begin{cases}\sum\limits_{k=1}^{j}|y_k-g|,\ i=0;\\ \sum\limits_{k=1}^{i}|y_k-g|,\ j=0;\\ \min\begin{cases}|x_i-y_j|+D_{\rm erp}(\text{Rest}(X),\text{Rest}(Y)),\\ |x_i-g|+D_{\rm erp}(\text{Rest}(X),Y),\\ |y_j-g|+D_{\rm erp}(X,\text{Rest}(Y)),\end{cases}\\ \quad\text{otherwise.}\end{cases}\tag{6}$$

其中: $|\cdot|$ 表示第1范氏距离, 间隙 g 是一个初始值为0的常数. ERP的目标是寻找弯曲路径中最小的路径; 间隙 g 将两个长度不同的时间序列对齐, 使ERP能有效处理本地时间转换问题[26]; ERP满足三角不等式, 所以它是一个度量函数[27].

2.3 经典相似性度量方法的对比分析

前两节对5种经典的相似性度量方法进行了较为全面的介绍, 由于提出的角度不同, 各种方法在计算复杂度、是否支持非等长序列、是否满足三角不等式、是否支持序列的变相和平移等方面表现出不同的特性. 表1对上述5种相似性度量方法从8个方面进行了综合对比.

文献[28]在一个分类问题上分析评估了上述5种经典的时间序列相似性度量方法, 得出以下结论:

1) 随着训练集规模的逐渐变大, 基于弹性度量的分类精度与基于欧氏距离的分类精度越来越接近, 而当训练集较小时, 基于弹性度量的分类精度远强于欧氏距离;

表 1 相似性度量方法特性对比

方法	时间复杂度	支持非等长	支持平移	支持噪声	支持扭曲断点	支持伸缩	本地时间转换	三角不等式
ED	$O(n)$							√
DTW	$O(n^2)$	√	√		√	√		
LCSS	$O(n^2)$	√	√	√	√			
EDR	$O(n^2)$	√	√	√	√	√	√	
ERP	$O(n^2)$	√	√	√	√	√	√	√

2) 编辑距离的精度与DTW很相近, 其中, 只有EDR的精度高于DTW;

3) 基于编辑距离的度量与DTW相比, 其计算复杂度低于DTW.

3　时间序列相似性度量研究进展

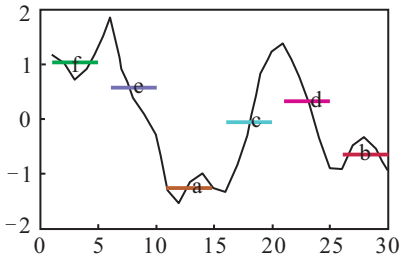
现阶段时间序列相似性度量的研究主要集中在3个方面: 一是提出新的时间序列相似性解释和度量方法; 二是利用相似性度量提高时间序列数据挖掘性能; 三是探讨时间序列相似性度量对时间序列数据挖掘精度的影响.

3.1　时间序列相似性新的解释和度量

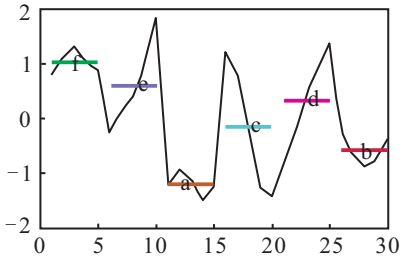
除了第1节给出的时间序列相似性定义外, 近几年, 学者们又从符号化、变化趋势、形状等角度给出了时间序列相似性的新解释, 并提出了相应的相似性度量方法.

3.1.1　时间序列的符号化相似性及其度量

时间序列的符号化(symbolic aggregate approximation)是指通过对时间序列 $X=\langle x_1,x_2,\cdots,x_n\rangle$ 进行规则化、主成分分析降维和离散化, 将数值形式时间序列转换为字符串形式 $S=\langle a_1,a_2,\cdots,a_n\rangle$. 其中: $a_k\in a,b,c,d,\cdots,\ k=1,2,\cdots,n$. 进而, 通过计算两个序列的SAX距离来度量其相似性. 图3显示了两个不相似的时间序列的SAX表示. 由于传统的SAX没有考虑时间片段的趋势(或方向), 导致不同时间片段的平均值可能映射到同一个符号.



(a) 序列 a



(b) 序列 b

图3　序列 a 和序列 b 的SAX表示

文献[29]提出在SAX计算过程中增加趋势变量来提高SAX的计算精度的方法(SAX_TD), 即把序列 X 和序列 Y 分别表示成

$$'_{0.2}f_{1.2}e_{-0.1}a_{-1.2}c_1d_{-0.2}b'_{-0.3}$$

和

$$'_{0.3}f_{-0.8}e_0a_{1.3}c_{-1.4}d_{0.4}b'_{0.3}.$$

这一相似性度量精度大大超过了SAX, 其相似性度量函数为

$$\mathrm{TDIST}(\hat{X},\hat{Y})=\sqrt{\frac{n}{w}}\sqrt{\sum_{i=1}^{w}(\mathrm{Com\_Dist}(\hat{x}_i,\hat{y}_i))^2+\frac{w}{n}(\mathrm{td}(x_i,y_i)^2)}. \quad (7)$$

其中: $\hat{X}$ 和 $\hat{Y}$ 分别是长度为 n 的时间序列 X 和 Y 的符号化表示; w 为符号的个数; $\mathrm{td}(x,y)$ 表示时间序列分段 x 与 y 之间趋势变量间的欧氏距离; $\mathrm{Com\_Dist}(\hat{x},\hat{y})$ 是 $\hat{X}$ 和 $\hat{Y}$ 中的每个字符之间的距离, 由下式计算:

$$\mathrm{Com\_Dist}(x,y)=\begin{cases}0, & |\hat{x}-\hat{y}|\leqslant 1;\\ \beta_{\max(\hat{x},\hat{y})-1}-\beta_{\min(\hat{x},\hat{y})}, & \text{otherwise.}\end{cases} \quad (8)$$

其中 β_i 为时间序列分段符号表示的 $N(0,1)$ 高斯分布值.

文献[30]提出一种基于复杂度的SAX相似性度量CM_SAX, 通过分段聚集近似地将时间序列平均细分成三元组 $s=(\mathrm{sax},m,w)$ 表示的子分段. 其中: sax为SAX符号化后的符号表示; m 为子分段模式; w 为子分段权值. 然后利用点距离、模式距离和复合距离公式计算时间序列间的相似性.

1) 点距离计算公式为

$$D_1(X_{\mathrm{sax}_j},Y_{\mathrm{sax}_j})=\left|\frac{\mathrm{SAX}(X_{\mathrm{sax}_j},Y_{\mathrm{sax}_j})}{\mathrm{SAX}(\mathrm{sym}_{\max},\mathrm{sym}_{\min})}\right|. \quad (9)$$

其中: $\mathrm{SAX}(X_{\mathrm{sax}_j},Y_{\mathrm{sax}_j})$ 表示序列 X 和 Y 中第 j 个分段之间的符号距离; $\mathrm{sym}_{\max}$ 表示对序列的符号化过程中出现的字母表中的最大字母; $\mathrm{sym}_{\min}$ 是出现的最小字母.

2) 模式距离计算公式为

$$D_2(X_{m_j},Y_{m_j})=\left|\frac{\sin X_{m_j}-\sin Y_{m_j}}{2}\right|, \quad (10)$$

其中 $\sin\theta$ 表示该子分段的变化趋势.

3) 复合距离计算公式为

$$\mathrm{KPDIST}(\hat{X},\hat{Y})=\sqrt{\sum_{i=1}^{W}(\mathrm{Com\_Dist}(\overline{X_i},\overline{Y_i})^2)}. \quad (11)$$

其中: W 为分段数; $\mathrm{Com\_Dist}(\overline{X_i},\overline{Y_i})$ 为第 i 个平均分段距离, 计算公式如下:

$$\mathrm{Com\_Dist}(\overline{X_i},\overline{Y_i})=\sum_{j=1}^{a}\omega_j\times\left[\frac{D_1(X_{\mathrm{sax}_j},Y_{\mathrm{sax}_j})+D_2(X_{m_j},Y_{m_j})}{2}\right]. \quad (12)$$

借助上述3种距离计算公式, CM_SAX准确地描

述了时间序列的重要特征信息, 有效度量了序列的相似性.

3.1.2 时间序列的趋势相似性及其度量

时间序列之间的趋势相似性可以定义为: 两个时间序列分别为 $X=\langle x_1,x_2,\cdots,x_m\rangle$, $Y=\langle y_1,y_2,\cdots,y_n\rangle$, 相应的趋势变化向量为 $X_v=\langle l_1,l_2,\cdots,l_m\rangle$, $Y_v=\langle s_1,s_2,\cdots,s_n\rangle$. 当 $1\leqslant i\leqslant n$ 时, $l_i=s_i$, 则 X 和 Y 被认为趋势相似, 即 X 与 Y 是相似的.

文献 [31] 提出一种基于趋势的时间序列相似性度量方法 (DT). 首先对时间序列进行区间划分并判断区间内趋势, 生成短趋势符号序列; 然后计算各趋势符号的一阶连接指数的塔尼莫特系数 $S_{X,Y}$ ($S_{X,Y}\in[0,1]$) 并进行相似性度量. $S_{X,Y}$ 的计算公式如下:

$$S_{X,Y}=\frac{\sum\limits_{m=1}^{5}\mathrm{Id}_X(\mathrm{ts}_m)\times\mathrm{Id}_Y(\mathrm{ts}_m)}{\sum\limits_{m=1}^{5}\mathrm{Id}_X^2(\mathrm{ts}_m)+\sum\limits_{m=1}^{5}\mathrm{Id}_Y^2(\mathrm{ts}_m)-\sum\limits_{m=1}^{5}\mathrm{Id}_X^2(\mathrm{ts}_m)\times\mathrm{Id}_Y(\mathrm{ts}_m)}. \tag{13}$$

其中: $\mathrm{ts}_m\in(\mathrm{ts}_{\mathrm{up}},\mathrm{ts}_{\mathrm{dw}},\mathrm{ts}_{\mathrm{pk}},\mathrm{ts}_{\mathrm{st}},\mathrm{ts}_{\mathrm{th}})$, 分别表示上升、下降、上凸、下凹和平稳的趋势符号序列. 若 $S_{X,Y}>\varepsilon$, 则 X 与 Y 相似, ε 为相似性阈值.

文献 [32] 提出一种通过趋势距离度量时间序列趋势相似性的方法 (SMVT), 采用分段聚合近似对时间序列进行降维, 进而将序列数据符号化并从变化趋势的角度对时间序列进行度量, 直观且有效地度量了序列趋势的相似性. 趋势序列 $X_1=\langle x_{11},x_{12},\cdots,x_{1i},\cdots,x_{1M}\rangle$ 和 $X_2=\langle x_{21},x_{22},\cdots,x_{2j},\cdots,x_{2N}\rangle$ 的趋势距离 $\mathrm{TD}(X_1,X_2)$ 为

$$\begin{cases}\mathrm{Dist}(0,0)=0,\\ \mathrm{Dist}(i,0)=\mathrm{Dist}(i-1,0)+\mu_d,\\ \mathrm{Dist}(0,j)=\mathrm{Dist}(0,j-1)+\mu_i,\\ \mathrm{Dist}(i,j)=\min\{\mathrm{Dist}(i-1,j)+\mu_d,\\ \qquad\qquad \mathrm{Dist}(i,j-1)+\mu_i,\\ \qquad\qquad \mathrm{Dist}(i-1,j-1)+\mu_r(i,j)\},\\ \sigma_{\max}=\max\{\mathrm{abs}(\max(X_1))-\min(X_2),\\ \qquad\qquad \mathrm{abs}(\min(X_1-\max(X_2)))\},\\ \mu_t(i,j)=\mathrm{abs}(x_{1i}-x_{2j})/\sigma_{\max},\\ \mathrm{TD}(X_1,X_2)=\mathrm{Dist}(M,N).\end{cases} \tag{14}$$

其中: μ_d、μ_i 和 $\mu_r(i,j)$ 分别表示插入、删除和替换操作的代价.

文献 [33] 提出一种基于二值变化趋势的时间序列相似性度量方法 (FTSC). 首先得到序列的变化向量 $X_V=\langle v_1,v_2,\cdots,v_n\rangle$ 和方向向量 $X_d=\langle b_1,b_2,\cdots,b_n\rangle$, 其中: $v_i=x_i-x_{i-1}$, $b_i=\begin{cases}0,\ v_i\geqslant 0\\ 1,\ v_i<0\end{cases}$; 然后由距离函数 $d_n(X,Y)$ 计算相似性, 即

$$d_n(X,Y)=\sum_{j=1}^{n}d_{\mathrm{binary}}(b_{X_j},b_{Y_j}), \tag{15}$$

其中

$$d_{\mathrm{binary}}(b_1,b_2)=\begin{cases}0,\ b_1=b_2;\\ 1,\ \mathrm{otherwise}.\end{cases} \tag{16}$$

即, 如果两序列在某点变化方向相同, 则二值距离为 0.

3.1.3 时间序列的形状相似性及其度量

时间序列的形状相似性主要是依据时间序列具体点值之间的距离来判断其相似性. 例如, 时间序列 $X=\langle x_1,x_2,\cdots,x_n\rangle$ 和 $Y=\langle y_1,y_2,\cdots,y_n\rangle$, 设置相似度阈值为 δ, 若两序列具体点间的距离 $\mathrm{Dist}(x_i,y_i)\leqslant\delta$, 则认为序列 X 与 Y 形状相似, 即序列相似.

文献 [34] 介绍了一种基于形状的时间序列数据相似性度量方法 (AMSS), 把时间序列看作一个向量序列数据集, 比较序列的形状和经过余弦相似度转换后的序列形态的相似性, 度量函数如下:

$$\begin{aligned}&\mathrm{AMSS}(X_n,Y_m)=\\ &\max\{\mathrm{AMSS}(X_{n-1},Y_{m-1})+2\mathrm{sim}(x_n,y_m),\\ &\mathrm{AMSS}(X_{n-2},Y_{m-1})+2\mathrm{sim}(x_{n-1},y_m)+\mathrm{sim}(x_n,y_m),\\ &\mathrm{AMSS}(X_{n-1},Y_{m-2})+2\mathrm{sim}(x_n,y_{m-1})+\mathrm{sim}(x_n,y_m)\},\end{aligned} \tag{17}$$

其中 $\mathrm{sim}(x_n,y_m)$ 为序列 X 的第 n 个向量与序列 Y 第 m 个向量之间的余弦相似度. 该方法基于向量的方向, 对时间转换、振幅平移和缩放具有较强的健壮性, 并且能以子序列的形状来正确计算时间序列之间的全局相似配对.

针对噪声和波动性导致的时间序列失真的情况, 文献 [35] 提出了一种新的相似性度量方法 SIMshape. 结合离散 Haar 小波变换, 对需要比较的序列进行多尺度形状特征提取并结合尺度权函数计算相似性, 计算公式为

$$\begin{aligned}&\mathrm{SIMshape}(X_t,Y_t)=\mathrm{sim}(S_{X_t},S_{Y_t})=\\ &\sum_{i=1}^{k-1}(\mathrm{xcom}(i)\times\mathrm{weight}(i)).\end{aligned} \tag{18}$$

其中: S_{X_t} 和 S_{Y_t} 表示利用多尺度形状提取模型获取到的 X_t 和 Y_t 的多尺度形状信息; $\mathrm{xcom}(i)$ 用于记录 i 尺度上的两个元素是否相等, 若相等, 则 $\mathrm{xcom}(i)=1$, 否则 $\mathrm{xcom}(i)=0$; $\mathrm{weight}(i)$ 为第 i 尺度的权重, $\mathrm{weight}(i)=5/6^{k-i+1}$. 该方法在保证 SIMshape 的效

率和精确性的前提下使用相对权重函数以提高SIMshape的健壮性.

3.1.4 时间序列的事件相似性及其度量

用户关注的问题即为一个事件序列, 记为$E(T)$, T表示事件的时间长度. 事件序列是指事件$E(T)$在时间T上发生程度的排序. 时间序列的事件相似性即为事件序列的相似性.

文献[36]首次提出了基于事件的时间序列相似性度量方法(SMBE). 将用户需求定义为事件序列, 构建SMBE模型, 进而根据事件相似性构建相似性矩阵M, $M(i,j)=\mathrm{Sim\_Event}(X_E[i],Y_E[i])$, 然后遍历整个矩阵, 计算成本最大路径, 即最优路径值, 作为序列之间的相似性度量. 事件相似性计算公式为

$$\mathrm{Sim\_Event}(X_E[i],Y_E[i])=\frac{(X_E[i],Y_E[i])\times\omega(X_E[i],Y_E[i])}{\gamma(i,j)}. \tag{19}$$

其中: $X_E[i]$和$Y_E[i]$分别表示时间序列X和Y中的事件; ω为权重函数; $\gamma(i,j)$为时间惩罚因子, 计算公式为

$$\gamma(i,j)=1+\frac{|i-j|}{T}. \tag{20}$$

文献[37]提出了一种基于模糊集的事件识别与处理机制(RPBE), 通过在两个时间序列中识别相同的事件来计算序列的相似性, 计算公式为

$$\mathrm{SIM}(X,Y)=\frac{\sum\limits_{i,j}\mathrm{length\_pair}((E_{X_i},E_{Y_j}))}{\sum\limits_{m}\mathrm{length}(E_m)}. \tag{21}$$

其中: E_{X_i}表示时间序列X中的第i个事件; E_m表示两个序列中的第m个事件; $\mathrm{length_pair}(\cdot)$表示事件序列对$(E_{X_i},E_{Y_i})$的持续时间长度; $\mathrm{length}(\cdot)$表示单个事件的持续时间长度, 计算公式为

$$\mathrm{length\_pair}((E_{A_i},E_{B_j}))=\mathrm{length}(E_{A_i})+\mathrm{length}(E_{B_j}), \tag{22}$$

$$\mathrm{length}(E_m)=|\mathrm{time\_stamp\_final}(E_m)-\mathrm{time\_stamp\_inicial}(E_m)|. \tag{23}$$

为确定时间序列X中的事件是否出现在时间序列Y中, 必须将事件当作属性, 并与Y序列的其他事件进行比较, 以得到一个更接近现实、更精确的相似性度量值. 在脑电图时间数据上的实验表明, 该方法可以大大提高时间序列相似性度量的精确性.

表2对上述4种时间序列相似性新的解释和度量方法进行了归纳.

表 2　时间序列相似性新的解释和度量

新解释	度量相似性的依据	代表方法
符号化相似性	符号间的欧氏距离结合变化趋势间的欧氏距离	SAX_TD(2014)
	符号间的点距离与漠视距离构成的符号距离	CM_SAX(2013)
趋势相似性	塔尼莫特系数	DT(2014)
	序列趋势属性间的编辑距离	SMVT(2014)
	由变化向量和方向向量定义的二值距离	FTSC(2007)
形状相似性	形状向量序列的余弦相似度	AMSS(2013)
	多尺度形状信息及尺度权重	SIMshape(2014)
事件相似性	事件序列的最优路径值	SMBE(2010)
	事件序列持续的时间长度	RPBE(2016)

3.2 相似性度量用于时间序列数据挖掘

相似性度量通常会作为前序步骤, 与时间序列数据挖掘算法结合起来完成不同的挖掘任务, 包括分类算法、聚类算法和预测算法等.

3.2.1 相似性度量与聚类方法结合

时间序列挖掘的聚类任务是将时间序列数据集划分成若干相似的组或类的过程, 同组内序列间的相似度较高, 而不同组间的序列不相似. 显然, 聚类分析本质上是一个相似性度量的过程, 因此, 准确的相似性度量一定能帮助聚类分析方法得到更好的聚类结果. 例如, K-means通常与欧氏距离结合进行时间序列聚类分析.

文献[38]提出了一种K-means与DTW距离相结合的时间序列数据聚类方法, 采用基于重心法的平均DTW方法DEA计算样本间的距离, 进而得到不同的聚类簇, 并在标准数据集上验证了这一方法的有效性.

文献[39]提出了一种基于加权序列模式相似性的序列聚类算法. 首先构造基于加权序列元素的新的序列间相似性度量函数, 将每个序列按照相似性分配给相似度最高的簇; 然后针对每个聚类簇, 计算各序列向量的加权和以获得聚类中心序列. 这种方法使更

新后的中心更贴合真正的聚类中心, 提高了聚类质量, 更加准确地诠释了聚类结果的含义.

文献 [40] 给出了一种基于引力场的概念相似性度量方法, 数据之间的行程时间越短, 它们的相似性就越高. 基于此, 提出了一种新的层次聚类方法 —— TTHC (travel-time based hierarchical clustering), 利用相似性度量结果计算每个数据的边缘加权数, 再进行簇划分. 实验表明, TTHC 具有更高的鲁棒性.

3.2.2 相似性度量与分类算法结合

时间序列数据挖掘的分类任务是对新的时间序列标记类标签的过程. 若时间序列数据不等长, 则一般分类算法不能直接应用, 即使序列长度相同, 不同序列相同位置的数值往往也不能直接比较[41]. 定义恰当的度量方法给相近的序列标记相同的分类标签是解决这类问题的常用方法.

文献 [42] 中提出了一种 DTW 与 SAX 结合的时间序列分类技术 DTW_SAX, 通过计算 DTW 距离重构时间序列的特征以适应机器学习算法. 实验表明, DTW_SAX 能有效提高时间序列分类的性能. 文献 [43] 将 DTW 与 Shapelets 结合, 提出了一种线性分类模型 —— LTSD (learning time series shapelets with DTW). Shapelets 是时间序列中最具有代表性的一类子序列, 该方法用 DTW 计算时间序列的每一个分段与 Shapelets 之间的距离作为分类特征, 然后使用线性分类模型 LTSD 对时间序列进行分类, 能够达到很好的效果.

文献 [44] 将一种新的 DTW 算法用于快速准确地对时间序列进行分类. 在此基础上, 文献 [45] 提出了一种平均时间序列 DTW 方法 (DBA), 采用平均 "扭曲" 时间序列结合 DBA 方法, 获得了最接近 "质心" 的分类结果.

文献 [46] 提出了一种相似性度量与神经网络相结合的分类查询方法 (SM_NN), 通过衡量 K-means 概念的查询的可能性, 使用相似性度量方法, 从不同属性的角度计算训练对象与查询对象的相似性, 在分类错误率方面与其他方法相比具有较好的效果.

3.2.3 相似性度量与时间序列回归

时间序列挖掘的回归任务是根据时间序列的历史数据建立回归模型来定量预测时间序列未来取值的过程.

文献 [47] 提出了一种基于相似性度量与 kNN 相结合的时间序列预测算法 (kNN-TSPI), 使用相似性度量找出时间序列中相似性较高的前 k 个序列, 并结合 kNN 算法, 对后续的序列进行回归预测. 对 55 组数据的对比实验表明, 所提出的方法能有效提高时间序列预测的性能.

文献 [48] 研究了一种网络链路的时间序列回归预测方法, 首先计算多个不同节点的相似度, 然后建立时间预测模型 ARIMA, 基于一种能够结合过去节点相似度以及其他外部原因的方法来对时间序列进行预测, 能够达到更好的性能.

表 3 对上述 6 种时间序列相似性度量在时序挖掘中的作用进行了归纳.

表 3 相似性度量在时序挖掘中的作用

时序挖掘任务	相似性度量方法的应用
聚类	度量时间序列样本间的相似性, 为簇划分提供依据
	度量目标时序样本与聚类簇的相似性, 完成簇分配
分类	用相似性距离构造分类特征
	计算目标时序样本与训练样本的相似性, 完成分类
回归	根据相似性为已知序列归类, 建立回归模型
	为目标时序样本找到相似的历史节点, 建立回归模型

3.3 相似性度量和时序挖掘精度

上一节从不同的时序数据挖掘任务的角度探讨了相似性度量在其中的影响. 大量研究表明, 正确度量时间序列相似性是提高挖掘效率和效果的关键步骤, 但在理论上相似性度量与时序数据挖掘精度的关系仍然没有明确的解释.

文献 [49] 设置了两个与弹性度量有关的假设检验: 1) 用最近邻分类器来检验不同相似性度量在时间序列数据挖掘精度上是否存在显著差异; 2) 检验将全局分类方案与弹性度量相结合的方法是否能取得更高的精度. 实验得出的结论是: 1) 不同弹性度量所达到的分类精度之间并没有显著差异; 2) 与弹性度量相结合分类方法的精度远高于单一分类方法.

文献 [50] 在来自不同科学领域中的 38 个数据集上对 9 种相似性度量方法的分类精度进行了广泛的比较, 所得主要结论之一是, 尽管最新的相似性度量在理论上很具有吸引力, 但是大多数情况下其有效性都低于已被广泛使用的度量方法. 具体而言, DTW 方法被认为是一贯优于其他方法的度量方法; 欧氏距离仍是一个相当准确、强大且简单有效的时间序列相似性度量方法. 文献 [51] 在来自不同科学领域的 45 种时间序列数据集上比较了 7 种不同的相似性度量方法对样品分类精度的影响, 分别从分类方案、交叉验证、显著性差异以及参数选择等方面研究了相似性度量方法对时间序列分类精度的影响. 实验结果表明, 各种相似性度量方法在分类精度上没有显著统计差异.

文献 [52] 提出了一个多标签分类框架, 能根据时间序列特性与不同相似性度量方法之间的关系自动

选择最合适的度量方法. 而且在合成的时序序列数据集上对5种常用的相似性度量方法进行了一套完整的实验来验证该分类框架的有效性, 同时认为, 相似性度量对时序数据挖掘精度的影响与序列数据的类型和特性密切相关. 同时, Tiago 等[53]也得出了同样的结论, 他们比较了9种不同的相似性度量方法对时间序列聚类结果的影响, 实验表明, 相似性度量是基于属性计算的, 相似度是两个时间序列之间所有属性的相似性的综合, 因此, 相似性度量对时序数据聚类结果的影响与属性的特性密切相关.

总之, 若能根据具体的时序挖掘任务和数据特性选择合适的相似性度量, 则一定能提高挖掘的效率和精度; 反之, 则不能发现明显的效果. 对于不同相似性度量方法在时间序列挖掘中的有用性、差异性及其与挖掘精度的关系还需作进一步的理论探讨.

4　总结与展望

时间序列是一种普遍存在的序列数据, 利用数据挖掘技术能够对时间序列进行快速且有效的信息发现和获取[54]. 本文首先对常用的时间序列相似性度量方法进行了综述, 探讨了各相似性度量方法的优缺点, 并对各种方法进行了一个综合比较; 然后讨论了时间序列相似性度量的研究进展, 即有关相似性的新的解释和度量; 最后研究了相似性度量在时间序列数据挖掘中的应用, 以及与时序挖掘精度的关系.

今后有待进一步研究的工作包括以下几点:

1) 将现有的时间序列相似性度量方法在多种时间序列数据中进行比较, 通过实验直观地分析它们在各个不同时间序列中的特性, 使研究者对各个度量方法有更加显式直观的了解, 对提出更多、更好的相似性方法起到启发作用.

2) 相对于传统的锁步度量方法而言, 弹性度量方法在时间相似性度量上具有较高的效率和精确性. 因此, 在以后的研究中, 可以考虑多将弹性度量方法应用到时间序列相似性度量中, 同时可针对不同的实际问题, 在弹性度量的基础上对其进行扩展.

3) 目前, 大多数度量方法都假设获得的训练集是准确无误差的. 但在实际中, 训练集往往包含着大量噪声以及其他多种结构的数据, 现有的相似性度量方法在处理噪声问题时效果往往不完全尽如人意. 因此, 在今后的研究中, 可以考虑将鲁棒统计学的相关技术引入相似性度量学习方法中, 使相似性度量能很好地克服噪声以及数据多种结构对其的影响, 从而找到一个更加有效的相似性度量方法.

参考文献(References)

[1] 李海林, 杨丽彬. 时间序列数据降维和特征表示方法[J]. 控制与决策, 2013, 28(11): 1718-1722.
(Li H L, Yang L B. Time series data reduction and feature representation method[J]. Control and Decision, 2013, 28(11): 1718-1722.)

[2] Frank J, Mannor S, Pineau J, et al. Time series analysis using geometric template matching[J]. IEEE Trans on Pattern Analysis and Machine Intelligence, 2013, 35(3): 740-754.

[3] Chen M Y. A high-order fuzzy time series forecasting model for internet stock trading[J]. Future Generation Computer Systems, 2014, 37(7): 461-467.

[4] Agrawal R K, Adhikari R. A combination of artificial neural network and random walk models for financial time series forecasting[J]. Neural Computing & Applications, 2014(6): 1-9.

[5] Hoang Nguyen. A new similarity measure for intuitionistic fuzzy sets[C]. Proc of the 8th Asian Intelligent Information and Database Systems. Vietnam: Springer, 2016: 574-584.

[6] 李正欣, 张凤鸣, 李克武. 多元时间序列模式匹配方法研究[J]. 控制与决策, 2011, 26(4): 565-570.
(Li Z X, Zhang F M, Li K W. Study on the method of multivariate time series pattern matching[J]. Control and Decision, 2011, 26(4): 565-570.)

[7] Michael R Berthold, Frank Hppner. On clustering time series using Euclidean distance and pearson correlation[J]. Expert Systems with Applications, 2016, 52(6): 26-38.

[8] Sariel Har-Peled, Benjamin Raichel. Net and prune: A linear time algorithm for Euclidean distance problems[C]. Proc of the 45 Annual ACM Symposium on Theory of Computing. New York: Eprint Arxiv, 2014: 605-614.

[9] Bai S, Qi H D, Xiu N. Constrained best Euclidean distance embedding on a sphere: A matrix optimization approach[J]. Siam J on Optimization, 2015, 25(1): 439-467.

[10] Isnanto R R, Zahra A A, Widianto E D. Palmprint recognition system based on principle-lines feature using Euclidean Distance and neural network[C]. Proc of Int Conf on Information Technology, Computer, and Electrical Engineering. Guangzhou: IEEE, 2015: 153-158.

[11] Berndt D J, Clifford J. Using dynamic time warping to find patterns in time series[C]. Proc of Working Notes of the Knowledge Discovery in Databases Workshop. Singapore: IEEE, 1994, 10(16): 359-370.

[12] Che-Jui Hsu, Kuo-Si Huang, Chang-Biau Yang, et al. Flexible dynamic time warping for time series classification[C]. Proc of the Int Conf on Computational Science, Computational Science at the Gates of Nature. Reykjavik: Elsevier, 2015: 2838-2842.

[13] 毛红保, 吴虎胜, 李正欣, 等. 多元时间序列相似性度量方法研究[J]. 控制与决策, 2011, 26(4): 565-570.
(Mao H B, Wu H S, Li Z X, et al. Research on similarity measurement methods for multivariate time series[J]. Control and Decision, 2011, 26(4): 565-570.)

[14] 李正欣, 张凤鸣, 李克武. 基于DTW的多元时间序列模式匹配方法[J]. 模式识别与人工智能, 2011, 24(3): 425-430.
(Li Z X, Zhang F M, Li K W. Multi time series pattern matching method based on DTW[J]. Pattern Recognition and Artificial Intelligence, 2011, 24(3): 425-430.)

[15] Ozsoy A, Chauhan A, Swany M. Fast longest common subsequence with general integer scoring support on gpus[C]. Proc of Programming Models and Applications on Multicores and Manycores. New Orleans: ACM, 2014: 92.

[16] Górecki T. Using derivatives in a longest common subsequence dissimilarity measure for time series classification[J]. Pattern Recognition Letters, 2014, 45(1): 99-105.

[17] Daniel Roggen, Luis Ponce Cuspinera, Guilherme Pombo, et al. Limited-Memory warping LCSS for real-time low-power pattern recognition in wireless nodes[C]. Proc of the 12th European Conf of Wileless Sensor Networks. Portugal: Springer, 2015: 151-167.

[18] Bukh B, Ma J. Longest common subsequences in sets of words[J]. Siam J on Discrete Mathematics, 2014, 28(4): 2042-2049.

[19] Wang H, Su H, Zheng K, et al. An effectiveness study on trajectory similarity measures[C]. Proc of the 24th Australasian Database Conf. Darlinghurst: Australia Computer Society, 2013: 13-22.

[20] Chairunnanda P, Gopalkrishnan V, Chen L. Enhancing edit distance on real sequences filters using histogram distance on fixed reference ordering[C]. Proc of Int Conf on Pattern Recognition. Washington DC: IEEE, 2006, 3: 582-585.

[21] Chen L, Zsu M T, Oria V. Robust and fast similarity search for moving object trajectories[C]. Proc of SIGMOD. Hong Kong: IEEE, 2005: 491-502.

[22] Conti J C, Farial F A, Almeida J, et al. Evaluation of time series distance functions in the task of detecting remote phenology patterns[C]. Proc of Int Conf on Pattern Recognition. Stockholm: IEEE, 2014: 3126-3131.

[23] Radosaw Karbarz, Jan Mulawka. Data quality system using reference dictionaries and edit distance algorithms[C]. Proc of XXXVI Symposium on Photonics Applications in Astronomy, Communications, Industry, and High-energy Physics Experiments. Wilga: Int Society for Optics and Photonics, 2015: 96623A-96623A-11.

[24] Chen L, Ng R. On the marriage of Lp-norms and edit distance[C]. Proc of the 30th Int Conf on Very Large Data Bases. Toronto: VLDB Endowment, 2004, 30: 792-803.

[25] Kurbalija V, Radovanovi M, Geler Z, et al. The influence of global constraints on similarity measures for time-series databases[J]. Knowledge-Based Systems, 2014, 56(3): 49-67.

[26] Conti J C, Farial F A, Almeida J, et al. Evaluation of time series distance functions in the task of detecting remote phenology patterns[C]. Proc of Int Conf on Pattern Recognition. Stockholm: IEEE, 2014: 3126-3131.

[27] Jia D, Zhang D, Li N. Pulse waveform classification using support vector machine with Gaussian time warp edit distance kernel[J]. Computational & Mathematical Methods in Medicine, 2014(2): 947254.

[28] Junejo I N, Junejo K N, Aghbari Z A. Silhouette-based human action recognition using SAX-shapes[J]. Visual Computer, 2014, 30(3): 259-269.

[29] Sun Y, Li J, Liu J, et al. An improvement of symbolic aggregate approximation distance measure for time series[J]. Neurocomputing, 2014, 138(11): 189-198.

[30] Guo Gongde, Liu Fen. A method of similarity of time series similarity based on symbolic aggregation[J]. Computer Application, 2013, 33(1): 192-198.

[31] Xiao R, Liu G H. Research on trend-based time series similarity measure and cluster[J]. Application Research of Computers, 2014, 31(9): 2600-2605.

[32] Zhang H T, Li Z H, Sun Y, et al. New similarity measure method on time series[J]. Computer Engineering & Design, 2014, 35(4): 1279-1284.

[33] Abughali I K A, Minz S. Binarizing change for fast trend similarity based clustering of time series data[M]. Pattern Recognition and Machine Intelligence. Shenzhen: Springer Int Publishing, 2015: 257-267.

[34] Nakamura T, Taki K, Nomiya H, et al. A shape-based similarity measure for time series data with ensemble learning[J]. Formal Pattern Analysis & Applications, 2013, 16(4): 535-548.

[35] He X, Shao C, Xiong Y. A new similarity measure based on shape information for invariant with multiple distortions[J]. Neurocomputing, 2014, 129(5): 556-569.

[36] Wu X Y, Huang D. Similarity measurement method for time series based on events[J]. Computer Application, 2010, 30(7): 1944-1946.

[37] Lizcano D, Ares J, Lara J A, et al. A soft computing framework for classifying time series based on fuzzy sets of events[J]. Information Sciences, 2016, 330(10): 125-144.

[38] Jeong Y S, Jayaraman R. Support vector-based algorithms with weighted dynamic time warping kernel function for time series classification[J]. Knowledge-Based Systems, 2014, 75(C): 184-191.

[39] Kotsifakos A, Athitsos V, Papapetrou P. Query-sensitive distance measure selection for time series nearest neighbor classification[J]. Intelligent Data Analysis, 2016, 20(1): 5-27.

[40] Anh D T, Thanh L H. An efficient implementation of k-means clustering for time series data with DTW distance[J]. Int J of Business Intelligence & Data Mining, 2015, 10(3): 213-232.

[41] Zhang Wei. Research on fault diagnosis of microcomputer monitoring system based on data mining[D]. College of Computer Science and Technology, Lanzhou Jiaotong University, 2010.

[42] Kate R J. Using dynamic time warping distances as features for improved time series classification[J]. Data Mining & Knowledge Discovery, 2016, 30(2): 1-30.

[43] Mit Shah, Josif Grabocka, Nicolas Schilling, et al. Learning DTW-shapelets for time-sereis classification[C]. Proc of the 3rd Conf on Data Science. Pune, 2016: 13-16.

[44] Grabocka J, Schilling N, Wistuba M, et al. Learning time-series shapelets[C]. Proc of the 20th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2014: 392-401.

[45] Petitjean F, Forestier G, Webb GI, et al. Dynamic time warping averaging of time series allows faster and more accurate classification[C]. Proc of the 2014 IEEE Int Conf on Data Mining. Shenzhen: IEEE, 2014: 470-479.

[46] Sadahiro Y, Kobayashi T. Exploratory analysis of time series data: Detection of partial similarities, clustering, and visualization[J]. Computers Environment & Urban Systems, 2014, 45(5): 24-33.

[47] Lu Y, Hou X, Chen X. A novel travel-time based similarity measure for hierarchical clustering[J]. Neurocomputing, 2015,173(1): 3-8.

[48] Parmezan A R S, Batista G E A P A. A study of the use of complexity measures in the similarity search process adopted by kNN algorithm for time series prediction[C]. Proc of the Int Conf on Machine Learning and Applications. California: IEEE, 2015: 45-51.

[49] Ismail Günes, Sule Gündüz-Oğüdücü, Cataltepe Z. Link prediction using time series of neighborhood-based node similarity scores[J]. Data Mining & Knowledge Discovery, 2016, 30(1): 1-34.

[50] Jason Lines, Anthony Bagnall. Ensembles of elastic distance measures for time series classification[C]. Proc of the 2014 Int Conf on Data Mining. Philadelphia, 2014: 524-532.

[51] Wang X, Mueen A, Ding H, et al. Experimental comparison of representation methods and distance measures for time series data[J]. Data Mining and Knowledge Discovery, 2013, 26: 275-309.

[52] Mori U, Mendiburu A, Lozano J A. Similarity measure selection for clustering time series databases[J]. IEEE Trans on Knowledge & Data Engineering, 2016, 28(1): 181-195.

[53] Tiago R L, Santos T R L D, Zárate L E. Categorical data clustering: What similarity measure to recommend[J]. Expert Systems with Applications, 2015, 42(3): 1247-1260.

[54] 张建业, 潘泉, 张鹏, 等. 基于斜率表示的时间序列相似性度量方法[J]. 模式识别与人工智能, 2007, 20(2): 271-274.
(Zhang J Y, Pan Q, Zhang P, et al. The similarity measure method of time series based on slope representation[J]. Pattern Recognition and Artificial Intelligence, 2007, 20(2): 271-274.)

(责任编辑：李君玲)