控制与决策
Control and Decision

基于多元时间序列的自适应贪婪高斯分段算法

王玲, 李泽中

引用本文:
王玲,李泽中. 基于多元时间序列的自适应贪婪高斯分段算法[J]. 控制与决策, 2024, 39(2): 568−576.

在线阅读 View online: https://doi.org/10.13195/j.kzyjc.2022.0738

基于多元时间序列的自适应贪婪高斯分段算法

王 玲[1,2†], 李泽中[1,2]

(1. 北京科技大学 自动化学院，北京 100083；
2. 北京科技大学 工业过程知识自动化教育部重点实验室，北京 100083)

摘 要: 现有多元时间序列分段算法中分段点的选择以及分段个数的确定往往需要分别独立完成,大大增加了算法的计算复杂度.为解决上述问题,提出一种基于多元时间序列的自适应贪婪高斯分段算法. 该算法将多元时间序列各个分段所对应的数据解释为来自不同多元高斯分布的独立样本,进而将分段问题转化为协方差正则化的最大似然估计问题进行求解.为提高学习效率,采用贪婪搜寻方法使每个段的似然值最大化进而近似地找到最优分段点,并且在搜寻的过程中利用信息增益方法自适应地获取最优的分段个数,避免分段个数确定和分段点选择分别独立进行,从而减少计算的复杂度. 基于多种领域的真实数据集实验结果表明,所提出方法的分段精度以及运行效率均优于传统方法,并且能够有效完成多元时间序列的异常检测任务.

关键词: 多元时间序列；高斯分段模型；信息增益；自适应；贪婪搜索；异常检测

中图分类号: TP273 **文献标志码:** A

DOI: 10.13195/j.kzyjc.2022.0738

引用格式: 王玲,李泽中. 基于多元时间序列的自适应贪婪高斯分段算法[J]. 控制与决策, 2024, 39(2): 568-576.

Adaptive greedy Gaussian segmentation algorithm based on multivariate time series

WANG Ling[1,2†], *LI Ze-zhong*[1,2]

(1. School of Automation and Electrical Engineering，University of Science and Technology Beijing，Beijing 100083，China；2. Key Laboratory of Knowledge Automation of Industrial Processes of Ministry of Education，University of Science and Technology Beijing，Beijing 100083，China)

Abstract: For most multivariate time series segmentation algorithms, the selection of segmentation points and the determination of the number of segments often need to be completed independently, which greatly increase the computational complexity of the algorithm. In order to solve the above problem, an adaptive greedy Gaussian segmentation algorithm based on multivariate time series is proposed. The algorithm interprets the data points from the segmentations of multivariate time series as independent samples of different multivariate Gaussian distributions, and then transforms the segmentation problem into a covariance-regularized likelihood maximization problem to solve. In order to improve the learning efficiency, the greedy search method is adopted to maximize the likelihood value of each segment to find the optimal segment point approximately. During the search process, the information gain method is adopted to adaptively obtain the optimal number of segments, which avoids from realizing the determination of the number of segments and the selection of segment points independently to reduce the computational complexity. The experimental analysis is carried out on real datasets in many different fields. Compared with traditional methods, the proposed method can obtain higher accuracy and efficiency, and is able to detect outliers in multivariate time series effectively.

Keywords: multivariate time series; segmented Gaussian model; information gain; adaptive; greedy search; anomally detection

0 引 言

近年来,随着传感器技术的不断发展,许多应用领域(例如环境监测、便携式穿戴设备等)对于时间序列数据的收集量呈现出指数级的增长. 作为一种重要的时间序列预处理方法,时间序列分段技术能够根据时序数据的某些特性将其分割为一系列

收稿日期: 2022-05-02；**录用日期:** 2022-10-10.
基金项目: 国家自然科学基金项目(62076025, 61572073).
责任编委: 李少远.
†通讯作者. E-mail: lingwang@ustb.edu.cn.

离散的、非重叠的子序列,进而提供更加紧凑的表示来提取数据中感兴趣的模式,以实现进一步的知识挖掘,如分类、聚类以及规则挖掘等[1-4]. 早期的时间序列分段研究主要专注于对一元时间序列分段,采用的方法包括分段线性表示(piecewise linear representation, PLR)[5]、分支定界(AUG)[6]以及遗传算法(genetic algorithm, GA)[7-9]等. 这些分段方法对于一元时间序列能够取得良好的分段效果. 然而,在实际应用场景中,拥有更高数据维度的多元时间序列能够提供比一元时间序列更为丰富的信息量,并且多元时间序列所属多个变量的关联关系变化情况复杂,这将致使一元时间序列分段算法应用效果不理想. 因此,针对多元时间序列设计稳定且高效的分段算法具有重大意义.

目前,针对多元时间序列的分段算法大多包含以下3个部分[10]: 1) 成本函数; 2) 搜索(优化)方法; 3) 对于分段点个数的确定机制. 其中,文献[11]提出一种称为SMTS-DP的分段方法,该方法将Hubert分段成本[12]作为成本函数,采用动态规划作为优化方法进行分段位置搜索,并利用贝叶斯信息准则[13](BIC)确定最优分段点的个数. SMTS-DP在水文多元时间序列数据集上能够取得较好的分段效果. 文献[14]在文献[11]的基础上加以改进,提出了基于动态因子模型的分段算法(SMTS-FD). 该算法首先利用聚类方法将多元时间序列中相似的变量进行聚类,然后利用动态因子模型对变量降维,该步骤能够在一定程度上提高算法的运行效率. 文献[15]提出一种基于KPCA和G-G模糊聚类的分段算法FSTS-KPG. 该算法首先利用KPCA对多元时间序列降维,根据AP方法获取最大分段数目,再基于改进的G-G模糊聚类进行分段,并根据改进的DBI指数确定最佳分段数. FSTS-KPG方法在气象多元时间序列数据集上取得了良好的分段效果. 但上述方法对于分段点位置的确定以及最优分段数目的选择是两个独立进行的步骤,这增加了算法的复杂性. 为此,文献[16]提出一种称为IGTS的分段算法,它采用了基于熵的成本函数,同样利用动态规划对分段点位置进行寻优并同步确定最优分段数.

为了提高多元时间序列的分段效率,并且在分段点选择的同时获取最佳的分段数目,本文在文献[17]的基础上提出了一种基于多元时间序列的自适应贪婪高斯分段算法(adaptive greedy Gaussian segmentation of multivariate time series, AGGS-MTS). 该算法在贪婪搜寻分段点的过程中引入信息增益(information gain),自适应地获取最优的分段个数,从而避免了分段点位置确定和分段数目选择的孤立问题,较大程度上提升了算法的运行效率,实现计算复杂性和分段准确性的平衡.

1　问题定义

令原始的多元时间序列数据矩阵表示为

$$\boldsymbol{S}=\{x_1,\dots,x_t,\dots,x_n\}=\begin{bmatrix} x_{11} & x_{12} & \dots & x_{1t} & \dots & x_{1n}\\ x_{21} & x_{22} & \dots & x_{2t} & \dots & x_{2n}\\ \vdots & \vdots & \ddots & \vdots & \vdots & \vdots\\ x_{j1} & x_{j2} & \dots & x_{jt} & \dots & x_{jn}\\ \vdots & \vdots & \vdots & \vdots & \ddots & \vdots\\ x_{m1} & x_{m2} & \dots & x_{mt} & \dots & x_{mn}\end{bmatrix}. \tag{1}$$

其中: $x_t\in\boldsymbol{R}^m$, n代表时间序列的长度, $x_{jt}(j=1,2,\dots,m)$是第j个变量在时刻t的采样值.

假设多元时间序列$\boldsymbol{S}$能够划分为$k+1$个区间段$[b_0,b_1),\dots,[b_{r-1},b_r),\dots,[b_{k-1},b_k),[b_k,b_{k+1}](b_0=1,b_{k+1}=n)$,分段点集合为$\{b_0,\dots,b_r,\dots,b_k,b_{k+1}\}$. 本文的目标是从多元时间序列数据中找到最佳的分段数以及分段位置. 表1为文中涉及的符号及含义.

表1　符号及含义

符号	含义
$\boldsymbol{S}$	多元时间序列, $\boldsymbol{S}\in\boldsymbol{R}^{m\times n}$
x_t	多元时间序列在时刻t的采样值, $x_t\in\boldsymbol{R}^m$
x_{jt}	第$j(j=1,2,\dots,m)$个时间序列在时刻t的采样值
b	分段点集合$b=\{b_0,\dots,b_r,\dots,b_k,b_{k+1}\}$
k	分段点个数
$\boldsymbol{S}_r$	$\boldsymbol{S}$的第r个分段,表示为$[b_{r-1},b_r)$
$\mu^{(r)}$	第r个分段的均值向量
$\Sigma^{(r)}$	第r个分段的协方差矩阵
l_k	时间序列分成k段时信息增益损失函数值
ρ_k	时间序列分成k段时信息增益损失值变化率
$H(\boldsymbol{S})$	整个时间序列的信息增益
$H(\boldsymbol{S}_r)$	第r段的信息增益

2　基于多元时间序列的自适应贪婪高斯分段算法(AGGS-MTS)

2.1　高斯分段模型

令$\boldsymbol{S}_r$代表$\boldsymbol{S}=\{x_1,x_2,\dots,x_t,\dots,x_n\}$中第$r$个分段$[b_{r-1},b_r)$, $\boldsymbol{S}_r=\langle x_{b_{r-1}},\dots,x_{b_r-1}\rangle$. 第$r$个分段中每一个观测点$x_t$都服从多元高斯分布,其均值向量和协方差矩阵分别为

$$\mu^{(r)}=\frac{1}{b_r-b_{r-1}}\sum_{t=b_{r-1}}^{b_r-1}x_t, \tag{2}$$

$$\Sigma^{(r)}=\frac{1}{b_r-b_{r-1}}\sum_{t=b_{r-1}}^{b_r-1}(x_t-\mu^{(r)})(x_t-\mu^{(r)})^{\mathrm{T}}. \tag{3}$$

上述基于多元高斯分布的模型称为高斯分段模型(segmented Gaussian model, SGM),其可以将时间

序列分段问题转化为高斯分段模型的参数求解问题,即选择合适的参数使得整个时间序列数据可以用多个独立的高斯模型表示,如图1所示.

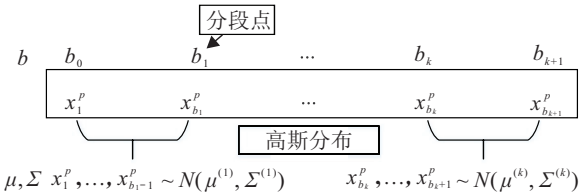


图1 高斯分段模型

为了学习高斯分段模型的相关参数,需要计算$\boldsymbol{S}=\{x_1,x_2,\dots,x_t,\dots,x_n\}$的对数似然函数以进行最大似然估计. $\boldsymbol{S}=\{x_1,x_2,\dots,x_t,\dots,x_n\}$的对数似然函数表示如下:

$$\begin{aligned}&L(b,\mu,\Sigma)=\\&\sum_{t=1}^{n}\Big(-\frac{1}{2}(\boldsymbol{x}_t-\mu_t)^{\mathrm{T}}\Sigma^{-1}(\boldsymbol{x}_t-\mu_t)-\\&\frac{1}{2}\log\det\Sigma-\frac{m}{2}\log(2\pi)\Big)=\\&\sum_{r=1}^{k+1}\sum_{t=b_{r-1}}^{b_r-1}\Big(-\frac{1}{2}(\boldsymbol{x}_t-\mu^{(r)})^{\mathrm{T}}(\Sigma^{(r)})^{-1}(\boldsymbol{x}_t-\mu^{(r)})-\\&\frac{1}{2}\log\det\Sigma^{(r)}-\frac{m}{2}\log(2\pi)\Big)=\\&\sum_{r=1}^{k+1}L^{(r)}(b_{r-1},b_r,\mu^{(r)},\Sigma^{(r)}).\end{aligned}\tag{4}$$

其中

$$\begin{aligned}&L^{(r)}(b_{r-1},b_r,\mu^{(r)},\Sigma^{(r)})=\\&\sum_{t=b_{r-1}}^{b_r-1}\Big(-\frac{1}{2}(\boldsymbol{x}_t-\mu^{(r)})^{\mathrm{T}}(\Sigma^{(r)})^{-1}(\boldsymbol{x}_t-\mu^{(r)})-\\&\frac{1}{2}\log\det\Sigma^{(r)}-\frac{m}{2}\log(2\pi)\Big)=\\&-\frac{1}{2}\sum_{t=b_{r-1}}^{b_r-1}(\boldsymbol{x}_t-\mu^{(r)})^{\mathrm{T}}(\Sigma^{(r)})^{-1}(\boldsymbol{x}_t-\mu^{(r)})-\\&\frac{b_r-b_{r-1}}{2}(\log\det\Sigma^{(r)}+m\log(2\pi)).\end{aligned}\tag{5}$$

其中:$b=\langle b_0,\dots,b_r,\dots,b_{k+1}\rangle$, $\mu=\langle\mu^{(1)},\dots,\mu^{(r)},\dots,\mu^{(k+1)}\rangle$, $\Sigma=\langle\Sigma^{(1)},\dots,\Sigma^{(r)},\dots,\Sigma^{(k+1)}\rangle$为高斯分段模型中的相关参数. 在式(4)和(5)中,如果Σ不是正定的,则$\log\det\Sigma$定义为$-\infty$, b_r-b_{r-1}代表第r段的长度.为避免多元时间序列的变量维数多于时间序列观测点个数时出现错误[18],在式(5)中添加一个协方差正则化项,即

$$\begin{aligned}&\varphi(b,\mu,\Sigma)=L(b,\mu,\Sigma)-\lambda\sum_{r=1}^{k+1}\mathrm{Tr}(\Sigma^{(r)})^{-1}=\\&\sum_{r=1}^{k+1}(L^{(r)}(b_{r-1},b_r,\mu^{(r)},\Sigma^{(r)})-\lambda\mathrm{Tr}(\Sigma^{(r)})^{-1}).\end{aligned}\tag{6}$$

其中:$\lambda\geqslant 0$是一个正则化参数,$\mathrm{Tr}(\Sigma^{(r)})$是分段$r$所对应协方差矩阵的迹. 一般认为协方差矩阵$\Sigma^{(r)}$是正定(可逆)的,但是在高维度协方差矩阵中往往不满足这一条件,因此在协方差矩阵中同样加入一个正则化项,则式(3)变为

$$\begin{aligned}\Sigma^{(r)}=&\frac{1}{b_r-b_{r-1}}\sum_{t=b_{r-1}}^{b_r-1}(\boldsymbol{x}_t-\mu^{(r)})(\boldsymbol{x}_t-\mu^{(r)})^{\mathrm{T}}+\\&\frac{\lambda}{b_r-b_{r-1}}I.\end{aligned}\tag{7}$$

为了书写方便,令

$$S^{(r)}=\frac{1}{b_r-b_{r-1}}\sum_{t=b_{r-1}}^{b_r-1}(\boldsymbol{x}_t-\mu^{(r)})(\boldsymbol{x}_t-\mu^{(r)})^{\mathrm{T}}.\tag{8}$$

将式(2)、(7)和(8)代入(6)中,经过协方差正则化的对数似然函数可以进一步表示为

$$\begin{aligned}&\varphi(b,\mu,\Sigma)=\\&\sum_{r=1}^{k+1}(L^{(r)}(b_{r-1},b_r,\mu^{(r)},\Sigma^{(r)})-\lambda\mathrm{Tr}(\Sigma^{(r)})^{-1})=\\&\sum_{r=1}^{k+1}\Big(-\frac{1}{2}\sum_{t=b_{r-1}}^{b_r-1}(x_t-\mu^{(r)})^{\mathrm{T}}(\Sigma^{(r)})^{-1}(x_t-\mu^{(r)})-\\&\frac{b_r-b_{r-1}}{2}\Big(\log\det\Big(S^{(r)}+\frac{\lambda}{b_r-b_{r-1}}I\Big)+\\&m\log(2\pi)\Big)-\lambda\mathrm{Tr}\Big(S^{(r)}+\frac{\lambda}{b_r-b_{r-1}}I\Big)^{-1}\Big)=\\&C+\sum_{r=1}^{k+1}\psi(b_{r-1},b_r).\end{aligned}\tag{9}$$

其中:$C=-(mn/2)(\log(2\pi)+1)$是一个常量,

$$\begin{aligned}&\psi(b_{r-1},b_r)=\\&-\frac{1}{2}\Big((b_r-b_{r-1})\log\det\Big(S^{(r)}+\frac{\lambda}{b_r-b_{r-1}}I\Big)+\\&\lambda\mathrm{Tr}\Big(S^{(r)}+\frac{\lambda}{b_r-b_{r-1}}I\Big)^{-1}\Big)\end{aligned}\tag{10}$$

为第r个分段的对数似然函数值. 最终高斯分段模型用于最大似然估计的对数似然函数如式(9)所示.

2.2 自适应贪婪高斯分段模型

为实现多元时间序列分段,最优分段数目的确定是较为重要的一环. 倘若确定的分段数过小,将致使分段算法仅仅捕捉到显著变化,无法充分揭示多元时间序列的演变过程,反之,过度关注微小变化趋势甚至噪声,将产生过拟合问题[10]. 文献[17]所提出的基于高斯分段模型的算法需要提前采取交叉验证确定分段数,而由此指定的分段数通常并不准确,并且交叉验证使得分段位置确定和分段数选择相互孤立. 为解决上述问题,本文提出基于多元时间序列的自适应贪婪高斯分段算法(AGGS-MTS). 该算法在采用贪

婪搜索分段点位置的过程中同时引入信息增益,可自适应地确定最优分段数目,避免分段位置确定和分段数选择相互独立,进而大大提升算法的分段效率.

AGGS-MTS计算高斯分段模型的协方差正则化对数似然函数(9),并采取贪婪搜索的方式,循环遍历每个时间点,找到使整体对数似然函数取最大值的点作为新的分段点,每次循环自上而下地增加一个分段点. 下面详细说明AGGS-MTS的具体运行过程.

在寻找第1个分段点的过程中($k=1$),算法首先初始化$b_0=0$, $b_2=n$,然后进行贪婪搜寻,遍历多元时间序列$\boldsymbol{S}$每一个时刻($t\in[b_0,b_2)$),并根据式(10)计算时刻t所对应的$\psi(b_0,t)+\psi(t,b_2)$值,即

$$\begin{aligned}&\psi(b_0,t)+\psi(t,b_2)=\\&-\frac{1}{2}\Big((t-b_0)\log\det\Big(S^{(1)}+\frac{\lambda}{t-b_0}I\Big)-\\&\lambda\mathrm{Tr}\Big(S^{(1)}+\frac{\lambda}{t-b_0}I\Big)^{-1}\Big)-\\&\frac{1}{2}\Big((b_2-t)\log\det\Big(S^{(2)}+\frac{\lambda}{b_2-t}I\Big)-\\&\lambda\mathrm{Tr}\Big(S^{(2)}+\frac{\lambda}{b_2-t}I\Big)^{-1}\Big).\end{aligned}\tag{11}$$

如果某一时刻t使得$\psi(b_0,t)+\psi(t,b_2)$在所有情况中取最大值,则认定时刻t为本次循环的最佳分段点,将该点加入分段点集合,并且更新该点的标号为b_1,这样便得到了将整个时间序列分为两段的段点集合$\{b_0,b_1,b_2\}$.

搜寻第2个分段点($k=2$)的过程与上述过程类似. 首先在第1个分段区间$[b_0,b_1)$中进行贪婪搜寻,找到某一时刻t使得$\psi(b_0,t)+\psi(t,b_1)$最大,即

$$\begin{aligned}&\psi(b_0,t)+\psi(t,b_1)=\\&-\frac{1}{2}\Big((t-b_0)\log\det\Big(S^{(1)}+\frac{\lambda}{t-b_0}I\Big)-\\&\lambda\mathrm{Tr}\Big(S^{(1)}+\frac{\lambda}{t-b_0}I\Big)^{-1}\Big)-\\&\frac{1}{2}\Big((b_1-t)\log\det\Big(S^{(2)}+\frac{\lambda}{b_1-t}I\Big)-\\&\lambda\mathrm{Tr}\Big(S^{(2)}+\frac{\lambda}{b_1-t}I\Big)^{-1}\Big).\end{aligned}\tag{12}$$

将该时刻t作为预选分段点保存. 然后,在第2个分段区间$[b_1,b_2)$进行贪婪搜寻,找到一个时刻t'使得$\psi(b_1,t^{'})+\psi(t^{'},b_2)$最大,即

$$\begin{aligned}&\psi(b_1,t')+\psi(t',b_2)=\\&-\frac{1}{2}\Big((t'-b_1)\log\det\Big(S^{(1)}+\frac{\lambda}{t'-b_1}I\Big)-\\&\lambda\mathrm{Tr}\Big(S^{(1)}+\frac{\lambda}{t'-b_1}I\Big)^{-1}\Big)-\\&\frac{1}{2}\Big((b_2-t')\log\det\Big(S^{(2)}+\frac{\lambda}{b_2-t'}I\Big)-\\&\lambda\mathrm{Tr}\Big(S^{(2)}+\frac{\lambda}{b_2-t'}I\Big)^{-1}\Big).\end{aligned}\tag{13}$$

将该时刻t'作为另一个预选分段点并保存.

当完成所有区间的贪婪搜寻后,比较$\psi(b_0,t)+\psi(t,b_1)+\psi(b_1,b_2)$和$\psi(b_0,b_1)+\psi(b_1,t')+\psi(t',b_2)$的大小,然后选择结果大的预选段点作为真正的分段点,将新的分段点加入分段点集合,按照时间次序更新所有分段点标号,进而得到整个时间序列的分段点集合$\{b_0,b_1,b_2,b_3\}$.

以此类推,在贪婪搜寻每个分段区间并添加候选分段点时,需要遵循使如下目标函数最大化的原则完成该步骤:

$$\begin{aligned}\max\ -\frac{1}{2}\sum_{r=1}^{k+1}\Big((b_r-b_{r-1})\log\det\Big(S^{(r)}+\\\frac{\lambda}{b_r-b_{r-1}}I\Big)-\lambda\mathrm{Tr}\Big(S^{(r)}+\frac{\lambda}{b_r-b_{r-1}}I\Big)^{-1}\Big).\end{aligned}\tag{14}$$

为实现在搜寻分段点位置的同时对分段数进行自适应确定,AGGS-MTS在每次确定增加一个分段点后,计算增加分段点前后信息增益损失值的变化. 例如,当分段点个数由1增加到2之后计算其分段前后的信息增益损失值变化率

$$\rho_1=\frac{l_1-l_0}{l_2-l_1}.\tag{15}$$

其中:l_0为时间序列未分段前的信息增益损失值,其值为0;l_1为获得第1个分段点后时间序列被分成两段时的信息增益损失值,有

$$l_1=H(\boldsymbol{S})-\sum_{r=1}^{2}\frac{|\boldsymbol{s}_r|}{|\boldsymbol{S}|}H(\boldsymbol{s}_r).\tag{16}$$

$|\boldsymbol{s}_r|=b_r-b_{r-1}$为第$r$段的长度,$H(\boldsymbol{S})$为整个时间序列未被分段的信息增益,有

$$H(\boldsymbol{S})=-\sum_{j=1}^{m}p_j\log p_j,\tag{17}$$

$$p_j=\sum_{t=1}^{n}x_{jt}\Big/\sum_{j=1}^{m}\sum_{t=1}^{n}x_{jt}.\tag{18}$$

其中$\sum_{t=1}^{n}x_{jt}$表示多元时间序列中第j个时间序列所有观测点的和,$\sum_{j=1}^{m}\sum_{t=1}^{n}x_{jt}$表示多元时间序列中所有时间序列所有观测点的和,$H(\boldsymbol{s}_r)$为第$r$段的信息增益,有

$$H(\boldsymbol{s}_r)=-\sum_{j=1}^{m}p_{rj}\log p_{rj},\tag{19}$$

$$p_{rj}=\sum_{t=b_{r-1}}^{b_r-1}x_{jt}\Big/\sum_{j=1}^{m}\sum_{t=b_{r-1}}^{b_r}x_{jt}.\tag{20}$$

l_2为获得第2个分段点后时间序列被分成3段时的信息增益损失值, 有

$$l_2 = H(\boldsymbol{S}) - \sum_{r=1}^{3} \frac{|\boldsymbol{s}_r|}{|\boldsymbol{S}|} H(\boldsymbol{s}_r). \tag{21}$$

同理, 当分段点个数为3时, 其分段前后的信息增益损失值变化率为

$$\rho_2 = \frac{l_2 - l_1}{l_3 - l_2}. \tag{22}$$

依次类推, 可以得到当分段点个数为k时的信息增益损失值l_k和信息增益损失值变化率ρ_k分别为

$$l_k = H(\mathrm{S}) - \sum_{r=1}^{k+1} \frac{|\boldsymbol{s}_r|}{|\boldsymbol{S}|} H(\boldsymbol{s}_r), \tag{23}$$

$$\rho_k = \frac{l_k - l_{k-1}}{l_{k+1} - l_k}. \tag{24}$$

为了自适应确定最优分段点的个数, AGGS-MTS算法使用信息增益损失值变化率作为评估指标. 当确定了k个分段点后便可以得到在分段过程中所有的信息增益损失值变化率集合$\{\rho_1, \rho_2, \ldots, \rho_k\}$, 最佳分段点的个数为拥有最大$\rho$所对应的$k$值, 因为当时间序列在该分段点个数时信息增益损失函数趋势的偏差急剧减小. 因此, 基于高斯分段模型的AGGS-MTS将贪婪搜寻与信息增益损失值变化率相结合, 可以快速地对多元时间序列进行分段, 同时获得最佳分段点的个数.

2.3 算法实现步骤

AGGS-MTS能够有效地处理高维的多元时间序列, 并且在整个分段过程中, 引入信息增益损失函数的变化率来自适应确定最佳的分段个数. 算法的整体流程如图2所示, 具体的实现过程如下:

输入: 多元时间序列$\boldsymbol{S} = \{x_1, x_2, \ldots, x_t, \ldots, x_n\}$, 最大分段点个数[17]$k_{\max} = (n/m)/3$;

输出: 分段点集合$b = \{b_0, b_1, \ldots, b_r, \ldots, b_k, b_{k+1}\}$, 最佳分段点个数$k$.

step 1: 初始化$b = \{b_0 = 0, b_2 = n\}$, $\rho_{\max} = 0$.

step 2: 令$k = 1$, 遍历初始时间序列$\boldsymbol{S}$, 找到时刻t使得$\psi(b_0, t) + \psi(t, b_2)$最大, 令该点作为第一个分段点, 并加入分段点集合.

step 3: 根据式(23)计算增加分段后的信息损失函数值l_k.

step 4: 根据式(24)计算增加分段前后的信息损失增益ρ_k.

step 5: 如果$\rho_k > \rho_{\max}$, 则$\rho_{\max} = \rho_k$.

step 6: 如果$k \geqslant k_{\max}$, 则转至step 8; 否则, 令$k = k + 1$转至step 7.

step 7: 遍历所有分段$\boldsymbol{s}_r (r = 1, 2, \ldots, k)$, 在每个$[b_{r-1}, b_r)$中找到一个时刻$t$使得$\psi(b_{r-1}, t) + \psi(t, b_r)$最大, 计算并以该点为新分段点时目标函数值; 将使目标函数最大的时刻t作为新的分段点, 并将新的分段点添加至分段点集合中, 更新分段点标号使得$b_0 < b_1 < \ldots < b_{k+1}$, 转至step 3.

step 8: 循环结束, 得到最佳分段点个数k以及分段点集合.

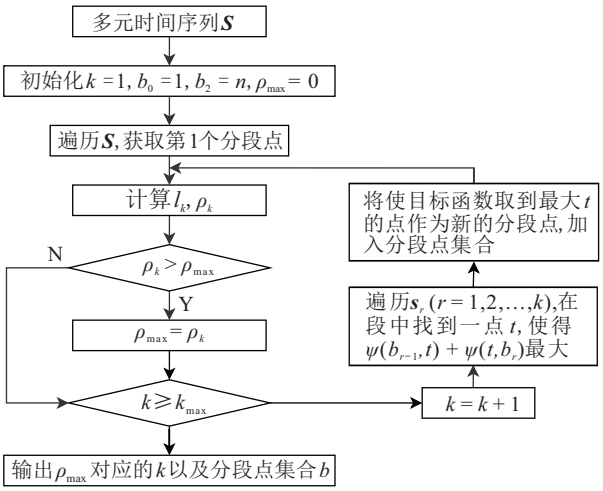


图 2 AGGS-MTS算法流程

2.4 计算复杂度分析

为便于评估算法的时间复杂度, 令多元时间序列的长度为n, 算法的最大分段数表示为$k_{\max}$. AGGS-MTS首先通过不断贪婪搜寻新的分段点, 假设平均每个段的长度为l, 在分段点$b_r \sim b_{r-1}$遍历新增加一个分段点的过程中, 计算在时刻t处添加分段点而产生的左右分段的经验均值和协方差, 其时间复杂度为$O(l^2)$. 求$\psi(b_{r-1}, t) + \psi(t, b_r)$的值, 其时间复杂度为$O((b_r - b_{r-1})l^3)$. 对于整个长度为$n$的多元时间序列而言, 采用贪婪搜寻依次添加$1, 2, \ldots, k_{\max}$个分段点, 总时间复杂度为$O(nk_{\max}(l^2 + l^3))$. 其次, AGGS-MTS利用信息增益确定最佳分段个数, 在分段的过程中计算每确定增加一个段点后的信息增益, 该过程的时间复杂度为$O(n)$. 综上所述, 算法的总时间复杂度为$O(k_{\max}n(l^2 + l^3 + 1))$, 与时间序列长度$n$呈线性关系.

3 实验结果

在实验验证部分, 本文首先对多元时间序列分段准确性的评价指标加以介绍, 其次采用真实的多元时间序列数据集来验证算法的分段性能, 最后将所提出算法用于真实多元时间序列数据集的异常检测. 实验将本文提出算法与SMTS_FD[11], SMTS_DP[14], FSTS_KPG[15], IGTS[16]四种分段算法进行对比. 所有实验均在Python 3.6、3.20 GHz处理器、16.00 GB内存

环境中执行.

3.1 评价指标

对于存在真实分段信息(ground truth)的数据集而言,为实现对于多元时间序列分段精度的评估,较为常用的指标为F_1分数[19],其公式如下所示:

$$F_1 = \frac{2 \times \mathrm{TP}}{2 \times \mathrm{TP} + \mathrm{FN} + \mathrm{FP}}. \tag{25}$$

假定数据集对应的真实分段点(ground truth)的位置集合表示为$\{\mathrm{GT}_i | i = 1, 2, \ldots, h\}$,待评估分段算法获得的实验分段点(experimental segmentation)的位置集合表示为$\{b_r | r = 1, 2, \ldots, k\}$,则式(25)中的真阳性TP代表$\{\mathrm{GT}_i | i = 1, 2, \ldots, h\}$与$\{b_r | r = 1, 2, \ldots, k\}$中元素相等的次数,假阳性FP代表$\{b_r | r = 1, 2, \ldots, k\}$中没有与之相等的真实分段点的元素个数. 假阴性FN代表$\{\mathrm{GT}_i | i = 1, 2, \ldots, h\}$中没有与之相等的实验分段点的元素个数. F_1分数越高,分段结果越精确.

文献[20]指出,直接将F_1分数作为评估指标将倾向于惩罚接近真实分段点的合理分割方案,为弥补这一缺陷,在数据集本身已经提供真实分段区间情形下,获得的实验分段点只要落在真实分段区间内, 则可标记为真阳性TP,对于仅给出真实分段点而未给出真实分段区间的数据集,本文依据文献[21],以每个真实分段点为中心,并以0.025 m为半径定义真实分段区间,其中m为整个多元时间序列的总长度,落入该区间的实验分段点可同样被标记为真阳性TP.

在特殊情况下,当两种待评估算法获得的实验分段点都同时落在真实分段区间时,F_1分数将无法继续评判两者的优劣. 为弥补上述缺陷,文献[22]提出一种用于评估分段算法准确性的指标MAE,该指标通过计算实验分段点位置与其最邻近真实分段点位置的绝对值距离来评判结果的准确性,即

$$\mathrm{MA}E = \sum_{i=1}^{h} |\mathrm{GT}_i - b_{\tilde{r}}| \Big/ m,$$

$$\tilde{r} = \arg\min_{r \in (1,k)} |\mathrm{GT}_i - b_r|. \tag{26}$$

MAE越低,分段结果越精确. 需要注意的是,该指标只适用于实验分段点总数与真实分段点总数相等的情况,而F_1分数可以适用于实验分段点和真实分段点不相等的情况,因此可与之形成互补.

3.2 真实数据集实验

该部分使用两个真实多元时间序列数据集进行实验,包括HY(hydrometeorological)水文气象数据集[23]和CMU人类动作捕捉数据集Mocap(motion capture)[24].

3.2.1 HY数据集的实验

HY数据集是一个真实多元时间序列数据集,是美国阿雷西沃地区的水文气象数据,其中包含windspeed, dir, gusts三个变量. 实验使用了2014-01-03～2014-01-05的数据,多元时间序列数据的总长度$T = 500$. 首先利用z-score对数据集进行标准化处理后,再将其作为算法的输入,并对其进行分段.

真实环境的数据集没有办法准确地得知其真正的分段点的位置,但可以通过时间序列的变化情况来判断其分段的好坏. 图3所示为windspeed, dir, gusts三个变量的时间序列变化曲线. 由图3(a)可以看出AGGS-MTS算法获得的分段的结果与时间序列的变化趋势相吻合,而其他算法在HY数集上的运行效果欠佳. 因此,对于真实水文气象数据集HY, AGGS-MTS算法能够较为准确地找到最佳的分段数.

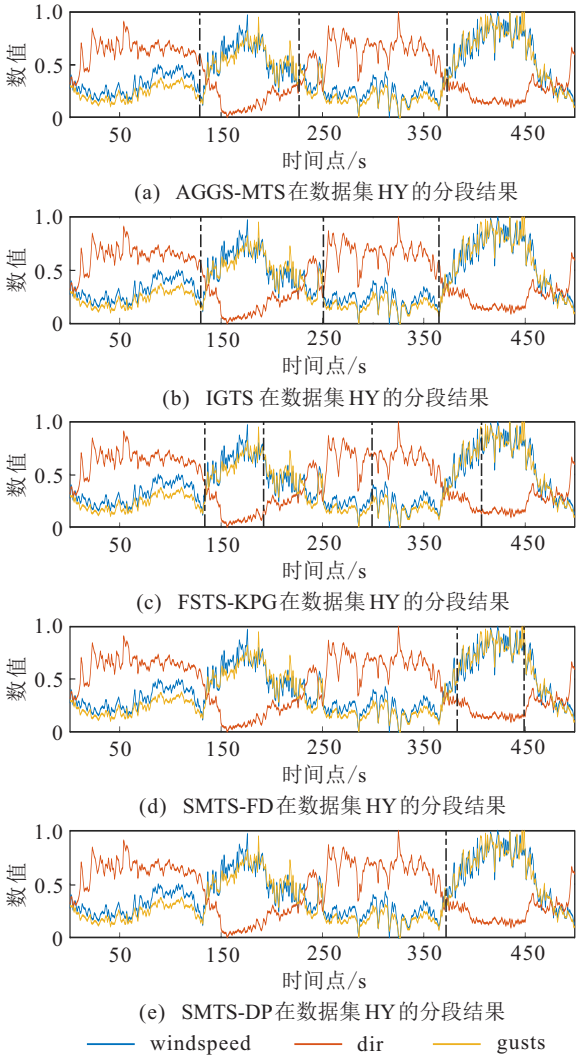


图3　不同算法在HY数据集上的分段结果

为了进一步验证所提出算法的分段性能,实验比较了不同算法在HY数据集上运行的效率与获得最佳分段数的个数,实验结果如表2所示. 由表2可以看出,本文提出的AGGS-MTS算法的运行时间在所有

算法中是最少的,特别是与SMTS_DP和SMTS_FD相比较,其运行效率是他们的将近1 000倍.

表 2 不同算法对HY数据集的分段结果

算法	运行时间/s	分段位置	分段数
AGGS-MTS	**0.34**	129, 227, 373	**4**
IGTS	0.72	130, 251, 365	**4**
FSTS_KPG	1.22	134, 192, 299, 407	5
SMTS_FD	317.35	383, 449	3
SMTS_DP	317.30	372	2

从最佳分段个数来看,AGGS-MTS和IGTS找到了相同的最佳分段个数,其他算法找到的最佳分段个数各不相同. 再结合图3(a)中时间序列的变化趋势,可以确定所提出算法能够找到更加合适的分段数,从而能够准确地对多元时间序列进行分段.

3.2.2 MoCap数据集的实验

MoCap数据集来自CMU动作捕捉数据库. 在这个数据集中,每个运动都被表示为数百帧的序列. 它由64维向量序列组成,实验中选择了其中4维(左右腿和左右臂)的数据. 它由“走”“蹲”“跑”等几个连续的动作组成,每个动作都可看作是由左臂、右臂、左腿、右腿4个变量所构成的多元时间序列,利用算法可以从这些运动序列中找到具体的运动(如“行走”和“奔跑”). 在本实验中选取了长度$T = 2\,000$的时间序列数据集. 其真实最优分段点(ground truth)以及真实分段区间由文献[25]提供. 实验首先通过比较不同算法在该数据集上的分段的结果与真实分段点进行比较来判断算法的分段准确程度;其次为了评价所提出算法运行性能的好坏,实验又比较了不同算法在数据集上的运行时间.

图4为不同算法在数据集上的分段结果,图4(a)中两条虚线表示真实分段点(ground truth)的位置,阴影部分表示真实分段区间. 由图4(b)~图4(f)可以看出,SMFTS_FD和SMFTS_DP算法获得的分段点的位置全部位于给定的真实分段点范围之外,两种算法的分段准确性较差. IGTS算法的第1个分段点的位置位于真实分段区间外,而AGGS-MTS和FSTS_KPG算法所获得的分段点的位置都位于真实分段区间的范围内,因此这两种算法的分段结果较为准确.

表3汇总了不同算法在MoCap数据集上的运行结果,其中包括算法的运行时间、分段位置、F_1分数以及MAE. 可以看出,SMFTS_FD和SMFTS_DP的运行效率较为低下,并且在分段的准确性上均表现欠佳. AGGS-MTS和FSTS_KPG都能够得到较为准确的分段结果,但FSTS_KPG的运行效率明显低

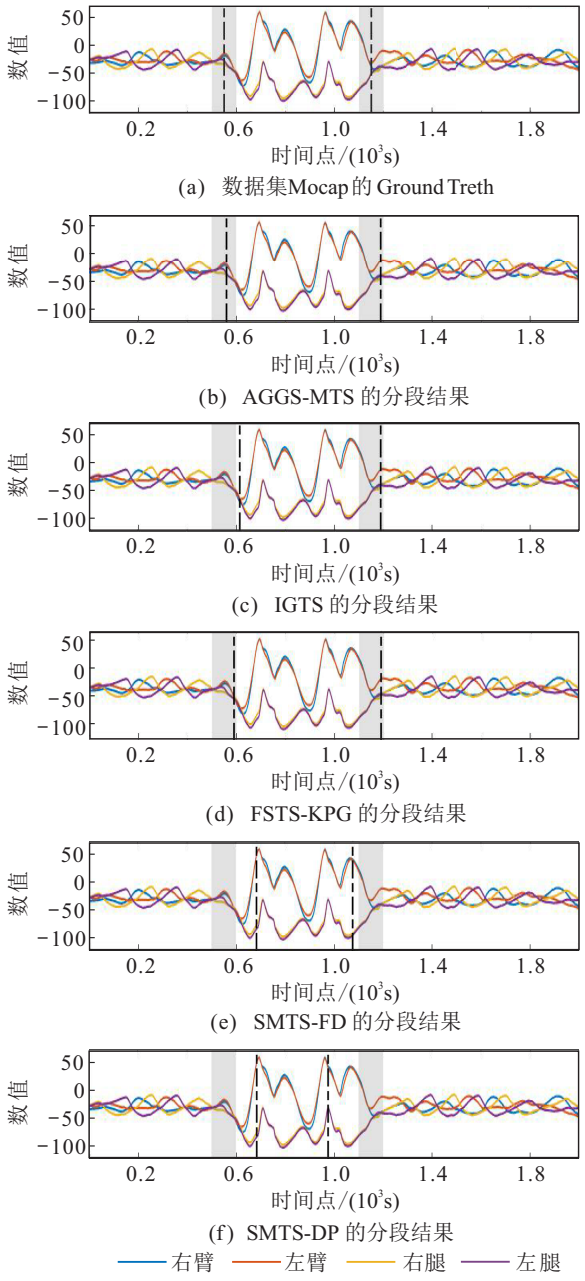


图 4 不同算法在MoCap数据集上的分段结果

于AGGS-MTS,IGTS算法的运行效率与AGGS-MTS较为接近,但该算法的分段准确率明显低于AGGS-MTS. 综上,对于真实动作捕捉数据集Mocap,本文所提出的AGGS-MTS算法能够获得最佳的分段结果.

表 3 不同算法对MoCap数据集的分段结果

算法	运行时间/s	分段位置	F_1	MAE
AGGS-MTS	**1.67**	**561, 1 189**	**1**	**0.025**
IGTS	1.85	615, 1 185	0.500	0.050
FSTS_KPG	19.31	590, 1 189	**1**	0.040
SMTS_FD	3 040.56	684, 1 076	0	0.104
SMTS_DP	4 933.49	685, 976	0	0.154

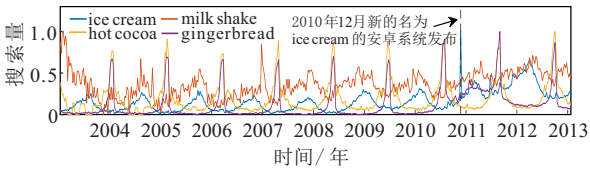
3.3 AGGS-MTS在异常检测任务中的应用

为了进一步验证AGGS-MTS算法的实际应用效果,本文将其用于多元时间序列的异常检测任

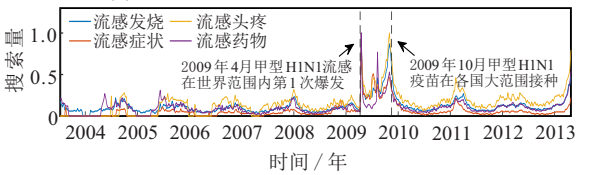
务. 所采用的实验数据包含信息检索,环境监测以及金融分析等领域的多元时间序列真实数据集,且均具有真实的异常点位置信息,具体信息见表4. 其中, sweets[19] 包含了用户使用Google搜索引擎在一段时间内查询与“甜点”话题相关的4种关键词的搜索量; flus[19] 包含了用户使用Google搜索引擎在一段时间内查询与“流感”话题相关的4种关键词的搜索量; occupancy[26] 记录了办公室内5种环境监测传感器在固定时间段内的数值; exchange[27] 则记录了1996至2016年7个国家每日汇率的变化情况. 所有数据集在进行异常检测实验之前均采用归一化方法完成预处理. 利用AGGS-MTS对上述数据集分别进行分段,所得到的分段位置点(如图5中黑色虚线所示)与上述数据集所有真实异常点的位置完全一致.

表 4　多元时间序列异常检测真实数据集

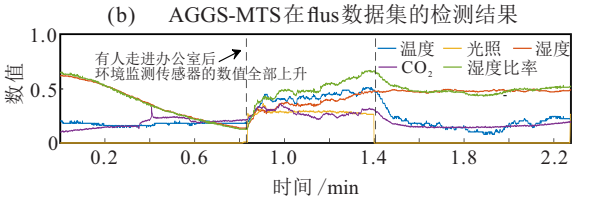
数据集	时间戳总数	变量维数	所属领域
sweets	484	4	信息检索
flus	472	4	信息检索
occupancy	2 275	5	环境监测
exchang	7 588	7	金融分析



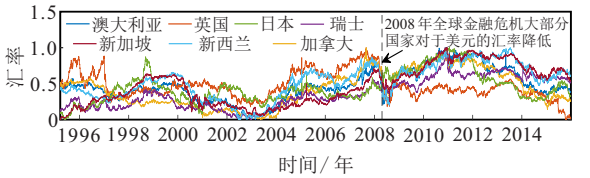
(a) AGGS-MTS在sweet数据集的检测结果



(b) AGGS-MTS在flus数据集的检测结果



(c) AGGS-MTS在occupancy数据集的检测结果



(d) AGGS-MTS在exchange数据集的检测结果

图 5　AGGS-MTS在真实数据集上的异常检测结果

图5(a)中的虚线显示了AGGS-MTS对于sweets数据集异常点位置的检测结果. 用户在Google搜索引擎搜索与“甜点”话题相关的4种关键词(“ice cream”“gingerbread”等)的搜索量在2010年以前具有明显的周期性,然而在2010年12月前后“ice cream”的搜索量曲线出现了尖峰,原因是由于安卓发布了名为“ice cream”的操作系统,引起了歧义. 图5(b)中的虚线显示了AGGS-MTS对于flus数据集异常点位置的检测结果. 用户在Google搜索引擎搜索与“流感”话题相关的4种关键词(“流感发烧”“流感症状”等),在2009年4月和2009年10月发生了两大历史事件: 甲型H1N1流感在世界范围内的首次爆发以及疫苗的大范围接种,致使上述4种关键词搜索量的趋势在上述两个时间点发生了明显的变化. 图5(c)中的虚线显示了AGGS-MTS对于occupancy数据集异常点位置的检测结果. 办公室内传感器持续监测室内的环境信息(温度,光照等),当有人走进办公室进行办公活动时,所有传感器的数值都会呈现上升趋势,当所有人离开办公室后,传感器数值则会呈现一定幅度的回落趋势. 图5(d)中的虚线显示了AGGS-MTS对于exchange数据集异常点位置的检测结果,当2008年全球金融危机爆发后,大部分国家对于美元的汇率降低. 综上所述,本文所提出的AGGS-MTS算法能够准确检测出上述多种领域真实的时序数据中存在的异常时间点.

4　结　论

为了解决现有分段算法不适用于多元时间序列的分段,且效率低下问题,本文提出了一种基于多元时间序列的自适应贪婪高斯分段算法AGGS-MTS. 该算法采用包含不同均值与协方差的多元高斯分布对多元时序数据的各个分段进行拟合,进而将分段问题转化为协方差正则化的最大似然估计问题进行求解. AGGS-MTS通过贪婪策略搜寻最优的分段点组合,并采用信息增益自适应地获取最优分段数,以避免分段点位置确定和分段数目选择的孤立问题,显著提高算法的运行效率. 实验部分分别以人工和真实数据集的分段任务以及真实多元时间序列的异常检测任务为例,验证所提出算法的有效性和优越性. 结果表明,AGGS-MTS在运行效率以及分段准确性方面均优于其他对比算法,并且在异常检测领域具有较好的运用价值.

本文所提出的算法也存在一定的局限性: 1) 当输入数据的分布情况与多元高斯分布差距较大时(譬如输入数据为重尾分布),AGGS-MTS输出的分段结果将有可能与真实分段信息存在较大偏差; 2) 输入数据中的噪声点可能对AGGS-MTS产生不良影响; 3) 当前的AGGS-MTS算法主要适用于离线分段场景. 在接下来的研究中,将着重于增强AGGS-MTS对复杂分布以及噪声数据的处理能力,并将其运用于流数据的分段场景.

参考文献(References)

[1] Mello C E, Carvalho A S T, Lyra A, et al. Time series classification via divergence measures between probability density functions[J]. Pattern Recognition Letters, 2019, 125: 42-48.

[2] Garcia-Vega S, León-Gómez E A, Castellanos-Dominguez G. A time-series prediction framework using sequential learning algorithms and dimensionality reduction within a sparsification approach[J]. Pattern Recognition Letters, 2020, 129: 287-292.

[3] Bokde N, Beck M W, Álvarez F M, et al. A novel imputation methodology for time series based on pattern sequence forecasting[J]. Pattern Recognition Letters, 2018, 116: 88-96.

[4] Gharghabi S, Yeh C C M, Ding Y F, et al. Domain agnostic online semantic segmentation for multi-dimensional time series[J]. Data Mining and Knowledge Discovery, 2019, 33(1): 96-130.

[5] Hu Y P, Guan P Y, Zhan P, et al. A novel segmentation and representation approach for streaming time series[J]. IEEE Access, 2018, 7: 184423-184437.

[6] Gedikli A, Aksoy H, Unal N E. Segmentation algorithm for long time series analysis[J]. Stochastic Environmental Research and Risk Assessment, 2008, 22(3): 291-302.

[7] Pérez-Ortiz M, Durán-Rosal A M, Gutiérrez P A, et al. On the use of evolutionary time series analysis for segmenting paleoclimate data[J]. Neurocomputing, 2019, 326/327: 3-14.

[8] Durán-Rosal A M, Gutiérrez-Peña P A, Martínez-Estudillo F J, et al. Time series representation by a novel hybrid segmentation algorithm[M]. Lecture Notes in Computer Science. Cham: Springer International Publishing, 2016: 163-173.

[9] Nikolaou A, Gutiérrez P A, Durán A, et al. Detection of early warning signals in paleoclimate data using a genetic time series segmentation algorithm[J]. Climate Dynamics, 2015, 44(7): 1919-1933.

[10] Truong C, Oudre L, Vayatis N. Selective review of offline change point detection methods[J]. Signal Processing, 2020, 167: 107299.

[11] Kehagias A, Nidelkou E, Petridis V. A dynamic programming segmentation procedure for hydrological and environmental time series[J]. Stochastic Environmental Research and Risk Assessment, 2006, 20(1): 77-94.

[12] Hubert P. The segmentation procedure as a tool for discrete modeling of hydrometeorological regimes[J]. Stochastic Environmental Research and Risk Assessment, 2000, 14(4): 297-304.

[13] Schwarz G. Estimating the dimension of a model[J]. The Annals of Statistics, 1978, 6(2): 461-464.

[14] Wang L, Xu P P, Peng K X. Segmentation of multivariate time series with factor model and dynamic programming[J]. Control and Decision, 2020, 35(1): 35-44.

[15] Wang L, Zhu H. Fuzzy segmentation of multivariate time series with KPCA and G-G clustering[J]. Control and Decision, 2021, 36(1): 115-124.

[16] Sadri A, Ren Y, Salim F D. Information gain-based metric for recognizing transitions in human activities[J]. Pervasive and Mobile Computing, 2017, 38: 92-109.

[17] Hallac D, Nystrup P, Boyd S. Greedy Gaussian segmentation of multivariate time series[J]. Advances in Data Analysis and Classification, 2019, 13(3): 727-751.

[18] Witten D M, Tibshirani R. Covariance-regularized regression and classification for high-dimensional problems[J]. Journal of the Royal Statistical Society Series B, Statistical Methodology, 2009, 71(3): 615-636.

[19] Matsubara Y, Sakurai Y, Faloutsos C. Autoplait: Automatic mining of co-evolving time sequences[C]. Proceedings of the 2014 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2014: 193-204.

[20] Lin J F S, Karg M, Kuli D N. Movement primitive segmentation for human motion modeling: A framework for analysis[J]. IEEE Transactions on Human-Machine Systems, 2016, 46(3): 325-339.

[21] Muralidhar N, Tabassum A, Chen L Z, et al. Cut-n-reveal: Time series segmentations with explanations [J]. ACM Transactions on Intelligent Systems and Technology, 2020, 11(5): 1-26.

[22] Gharghabi S, Ding Y F, Yeh C C M, et al. Matrix profile VIII: Domain agnostic online semantic segmentation at superhuman performance levels[C]. 2017 IEEE International Conference on Data Mining. New Orleans: IEEE, 2017: 117-126.

[23] Spinrad R W. NOAA/NOS/COŋOPS[DB/OL]. [2002-06-27]. http://coops.nos.noaa.gov.

[24] HODGINS J K. CMU graphics lab motion capture database[DB/OL]. [2020-06-27]. http://mocap.cs.cmu.edu/.

[25] Barbi J, Safonova A, Pan J Y, et al. Segmenting motion capture data into distinct behaviors[C]. Proceedings of Graphics Interface 2004. New York: ACM, 2004: 185-194.

[26] Candanedo L M, Feldheim V. Accurate occupancy detection of an office room from light, temperature, humidity and CO2 measurements using statistical learning models[J]. Energy and Buildings, 2016, 112: 28-39.

[27] Lai G K, Chang W C, Yang Y M, et al. Modeling long- and short-term temporal patterns with deep neural networks[C]. The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval. New York: ACM, 2018: 95-104.

作者简介

王玲(1974−), 女, 教授, 博士, 从事数据挖掘、模式识别等研究, E-mail: lingwang@ustb.edu.cn;

李泽中(1998−), 男, 硕士生, 从事数据挖掘、模式识别等研究, E-mail: g20208697@xs.ustb.edu.cn.