基于FastICA和G-G聚类的多元时序自适应分段

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

摘 要: 现有多元时间序列的分段方法主要通过检测时序数据统计特性或形状的变化情况,并以此为依据对分段点的位置进行"硬划分". 然而,这些分段方法无法对两个分段之间的过渡区间长度进行准确估计,且普遍需要人为预先设置参数,在高维且噪声较强的情况下分段效果较差. 本文针对现有分段方法存在的诸多不足,提出一种基于FastICA(Fast Independent Component Analysis)和G-G(Gath-Geva)模糊聚类的多元时序自适应分段方法. 该方法利用FastICA进行特征提取,采用DW(Durbin-Watson)指数自动选取高信噪比的主成分,并根据最小描述长度(Minimum Description Length,MDL)设计基于G-G模糊聚类的自适应分段模型,实现对于多元时间序列的"软划分". 基于多种领域的真实数据集实验结果表明:与现有主流的分段方法相比,本文方法在上述数据集上的平均F_1和MAE(Mean Absolute Error)可分别提升8.4%~16.8%和3.06%~6.56%.

关键词: 多元时间序列;自适应分段;快速独立主成分分析;Gath-Geva聚类;最小描述长度
基金项目: 国家自然科学基金(No.62076025,No.61572073)
中图分类号: TP273 **文献标识码:** A **文章编号:** 0372-2112(2023)05-1235-10
电子学报URL:http://www.ejournal.org.cn **DOI:**10.12263/DZXB.20220649

Adaptive Segmentation of Multivariate Time Series with FastICA and G-G Clustering

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, School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing* 100083, *China*)

Abstract: The existing segmentation methods detect the statistical or shape changes of multivariate time series, and perform crisp segmentation on the location of change points. However, these methods fail to estimate the length of the transition interval between two segments, cannot accurately segment multivariate time series with high dimension, strong noise, and need to set parameters in advance. To address such matters, an adaptive multivariate time series segmentation method based on FastICA (Fast Independent Component Analysis) and G-G (Gath-Geva) clustering is proposed. In this method, the key features of multivariate time series are extracted via FastICA, and DW (Durbin-Watson) criterion is used to automatically select main components with high signal-to-noise ratio. According to the minimum description length (MDL), an adaptive multivariate time series segmentation model based on G-G clustering is designed, which is able to perform soft segmentation of multivariate time series. The experimental analysis is carried out on real datasets in many different fields. Compared with state-of-art benchmarks, the average F_1 and MAE (Mean Absolute Error) of the proposed method on the above-mentioned datasets improve 8.4%~16.8% and 3.06%~6.56%, respectively.

Key words: multivariate time series; adaptive segmentation; fast independent component analysis; Gath-Geva clustering; minimum description length
Foundation Item(s): National Natural Science Foundation of China (No.62076025, No.61572073)

1 引言

时间序列是对某个物理量按时间的先后顺序进行定量观测的有序集合,已被广泛应用于经济学[1],气象学[2],生命科学[3]等各个研究领域. 作为一种重要的时间序列预处理技术,时间序列分段的目标是将原始的时序数据分割为若干个离散且同质的数据片段以反映原始

收稿日期:2022-06-07;修回日期:2022-08-22;责任编辑:覃怀银

数据的底层模式,从而使得海量时序数据的分析任务变得易于处理. 早期的时间序列分段研究主要针对一元时间序列,采用的方法包括基于分段线性表示(piecewise linear representation)[4]、基于进化计算(evolutionary computing)[5]、基于变化点检测(change point detection)[6]等. 然而,在许多应用场景下仅凭一元时间序列所提供的信息量将不足以生成合理的分割结果,譬如在动作捕捉领域中,仅凭手部佩戴的传感器数据将很难区分受试者步态模式的改变. 因此,针对拥有更高数据维度的多元时间序列设计行之有效的分段方法具有重大意义.

目前,针对多元时间序列的分段方法主要可分为三大类:基于统计模型的方法[7~12],基于形状变化的方法[13~15]以及基于聚类的方法[16~19]. 上述分段方法已经在分段准确性方面取得重要进展,但仍存在以下几点不足:(1)这些分段方法普遍属于"硬划分"的分段机制. 然而,在实际情况中,多元时间序列的变化趋势是缓慢而模糊的. 两组有明确语义的片段之间很少会发生瞬态转换,更多情况下其间会夹杂一段无明确语义的,高熵的过渡区间. 譬如在动作捕捉领域中,受试者在"踢腿"和"跑步"的动作转换过程中会做出一系列过渡动作进行衔接. "硬划分"的分段机制通常并不能准确识别完整的过渡区间,而是倾向于将其割裂开,或将其简单划归至"踢腿"或"跑步"的任意一段. 更为合理的方法则是将完整的过渡区间作为独立的段进行分割,以实现对多元时间序列的"软划分". (2)上述分段方法大都需要人为给定至少一种超参数,诸如时间序列的分段总数,滑动窗口长度或聚类个数等,调试超参数的冗长过程很大程度上增加了计算复杂性. (3)高维数据中的冗余部分以及混迭噪声的影响会使得上述分段方法对于真实分段点的识别产生偏差. (4)当前大多数方法对于分段结果的准确性评估仍旧以视觉为基准[13],当数据维度或分段总数增多的情况下,评估将会变得困难,并且缺乏客观性.

针对当前多元时间序列分段方法中存在的诸多不足,本文提出一种基于快速独立主成分分析(Fast Independent Component Analysis, FastICA)以及G-G(Gath-Geva)模糊聚类的自适应分段方法(Adaptive Multivariate Time Series segmentation model based on FastICA and G-G clustering, AMTS-FG). 该方法具有以下优点:(1)不同于"硬划分"的分段机制,所提出方法通过G-G模糊聚类获得任意时间点属于各个分段的模糊隶属度,并依据模糊隶属度实现对过渡区间长度的准确估计;(2)基于最小描述长度(Minimum Description Length, MDL)[20]自动选取最优分段总数,能够实现对于多元时间序列的自适应分段,避免人为设置参数;(3)采用FastICA实现数据的特征提取,并引入DW(Durbin-Watson)指数自适应确定信噪比较高的主成分,能够提高对高维且噪声较强的多元时间序列数据类型的分段准确性;(4)采用多种评价指标全面评估分段算法的有效性,而不是仅以视觉为基准,增强实验的客观性.

2 相关理论

2.1 独立主成分分析

在现实生活中,由于数据采集设备所处的真实环境较为复杂,并且不同设备之间存在相互作用,致使收集到的多元时间序列数据易与噪声发生混叠现象. 因此,研究有效的多元时间序列降噪方法,提取原始数据信息的主要成分,对于后续分段算法准确性的提升具有重大意义.

作为实现信号降噪分离最为流行的方法之一,独立主成分分析(Independent Component Analysis, ICA)[21]已被广泛应用于故障诊断和模式识别等领域. 给定原始数据矩阵$\boldsymbol{S}\in\mathbb{R}^{q\times N}$,其中$q$代表变量维数,$N$代表样本个数,则在ICA算法中,$\boldsymbol{S}$与潜在的独立主成分$\boldsymbol{X}\in\mathbb{R}^{d\times N}$存在以下关系:

$$\boldsymbol{S}=\boldsymbol{A}\boldsymbol{X}+\boldsymbol{E} \tag{1}$$

式(1)中,$\boldsymbol{A}\in\mathbb{R}^{q\times d}$代表混合矩阵,$\boldsymbol{E}\in\mathbb{R}^{q\times N}$代表残差矩阵. 在仅已知原始数据矩阵$\boldsymbol{S}$的情况下,ICA通过估计解混矩阵$\boldsymbol{W}=\boldsymbol{A}^{-1}$,最终得到潜在的独立主成分:

$$\boldsymbol{X}=\boldsymbol{W}\boldsymbol{S} \tag{2}$$

目前求解ICA模型的方法可以分为很多种. 在众多方法中,本文选择FastICA算法[22]实现对原始数据的分离降噪. FastICA采用固定点迭代的优化方法,在优化计算$\boldsymbol{W}$的过程中无需选择步长参数,具有易用性强,可靠性高的优点.

2.2 基于G-G聚类的多元时间序列分段

在现实生活中,许多情况下多元时间序列的变化趋势通常是缓慢而模糊的,对这种存在"过渡效应"的数据进行准确分段具有较大的挑战性. 最近,Abonyi等人[23]基于G-G模糊聚类,提出了一种新型多元时间序列分段算法. 该方法将多元时间序列的不同分段表示为具有非固定边界的模糊集,而不是将其分割为不相交的区域. 具体而言,给定多元时间序列数据矩阵$\boldsymbol{X}=\left[\boldsymbol{x}_1\ \cdots\ \boldsymbol{x}_i\ \cdots\ \boldsymbol{x}_N\right]$及其对应的时间戳$\boldsymbol{T}=\left(t_1,\cdots,t_i,\cdots,t_N\right)$,其中$\boldsymbol{x}_i=\left(x_{i1},x_{i2},\cdots,x_{id}\right)^{\mathrm{T}}$代表时刻$t_i$的$d$维采样值. 在已知分段数$C$的情况下,基于G-G聚类的分段算法通过最小化如下目标函数以获得最优的模糊分段:

$$J_{(\boldsymbol{X}|\mathrm{GG})}=\sum_{i=1}^{N}\sum_{k=1}^{C}\left(u_{i,k}\right)^{m}\mathrm{dis}^2\left(\boldsymbol{z}_i,\eta_k\right) \tag{3}$$

其中,$m\in(1,\infty)$为决定各分段模糊性的加权指数,通常取2. $\boldsymbol{z}_i=\left[\boldsymbol{x}_i\ \ t_i\right]$为同时包含采样值$\boldsymbol{x}_i$及其对应时间戳$t_i$的数据点,其中$i=1,\cdots,N$. $\eta_k=\left\{\boldsymbol{\eta}_k^{x},\boldsymbol{\eta}_k^{t}|k=1,\cdots,C\right\}$代表第$k$个分段的原型,$\mathrm{dis}^2\left(\boldsymbol{z}_i,\eta_k\right)$代表数据点$\boldsymbol{z}_i$与$\eta_k$的距离度

量,$u_{i,k}$表示$\boldsymbol{z}_i$隶属于第k个分段的程度,如式(4)所示:

$$u_{i,k}=\frac{1}{\sum_{j=1}^{C}\left(\mathrm{dis}^2\left(\boldsymbol{z}_i,\eta_k\right)/\mathrm{dis}^2\left(\boldsymbol{z}_i,\eta_j\right)\right)^{\frac{2}{m-1}}},\tag{4}$$

$$\text{s.t.}\forall i,k,u_{i,k}\in[0,1];\ \forall k,0<\sum_{i=1}^{N}u_{i,k}<N;\ \forall i,\sum_{k=1}^{C}u_{i,k}=1$$

不断迭代优化$\eta_k=\left\{\boldsymbol{\eta}_k^x,\boldsymbol{\eta}_k^t|k=1,\cdots,C\right\}$以及$u_{i,k}$,可以得到模糊分段结果.

3　基于FastICA和G-G聚类的多元时间序列自适应分段方法

本文提出基于FastICA以及G-G模糊聚类的多元时间序列自适应分段算法AMTS-FG,该方法的整体架构如图1所示,主要分为分离降噪模块与模糊分段模块.分离降噪模块负责对含有高维冗余信息以及强噪声的多元时间序列数据进行预处理,采用FastICA技术实现特征提取,并利用DW指数自适应确定最优主成分数,提取出高信噪比的主成分输入下一模块;模糊分段模块则采用G-G模糊聚类对输入数据中每个时间点属于各个分段的模糊隶属度进行求解,并采用最小描述长度自适应确定最优分段数,最终根据输出的模糊隶属度进行过渡区间长度的估计,实现对于多元时间序列的"软划分".以下各节将对AMTS-FG方法架构中的各个步骤进行详细介绍.

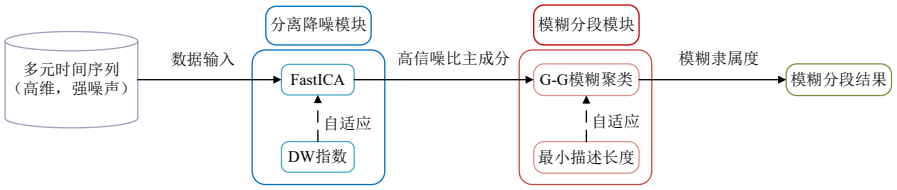


图1　AMTS-FG方法的整体架构图

3.1　基于FastICA的多元时间序列分离降噪

假定$\boldsymbol{S}\in\mathbb{R}^{q\times N}$为含有噪声的原始多元时间序列数据矩阵,其中$q$代表变量维数,$N$代表时间长度,$\boldsymbol{T}=\left(t_1,\cdots,t_i,\cdots,t_N\right)$为其对应的时间戳.采用FastICA能够在仅已知$\boldsymbol{S}$的情况下对其进行分离降噪,最终得到主成分矩阵$\boldsymbol{X}\in\mathbb{R}^{d\times N}$.然而该方法需要在运行前人为指定主成分数$d$.倘若指定的$d$值小于$\boldsymbol{S}$固有的最佳主成分数$d_{\mathrm{opt}}$,则会导致反映原始数据的重要信息在主成分矩阵$\boldsymbol{X}$中发生混叠,倘若指定的$d$值大于$d_{\mathrm{opt}}$,则将致使得到的主成分矩阵$\boldsymbol{X}$引入多余的噪声.

为解决上述问题,本文引入DW指数矩阵[24]实现最优主成分数d_{opt}的自适应选择.DW指数矩阵已被广泛用于解决结构化信号的信噪比估计问题.当信号中没有噪声时,DW指数矩阵所有元素的值将接近于0,并随着噪声量的增加而增加;当信号中(几乎)只含有噪声时,DW指数矩阵中将出现接近于2的元素值.

具体而言,首先在[1,q]范围内遍历主成分数d并分别执行FastICA算法,根据式(1)得到主成分数d对应的残差矩阵$\boldsymbol{E}^{(d)}$满足:

$$\boldsymbol{E}^{(d)}=\begin{bmatrix}e_{11}^{(d)} & \cdots & e_{1N}^{(d)}\\ \vdots & e_{ji}^{(d)} & \vdots\\ e_{q1}^{(d)} & \cdots & e_{qN}^{(d)}\end{bmatrix}\in\mathbb{R}^{q\times N},\tag{5}$$

$$1\le j\le q,1\le i\le N,1\le d\le q$$

其中,$e_{ji}^{(d)}$代表主成分数为d时变量j在时刻i所对应的残差值.其次,根据式(6)计算残差矩阵集合$\{\boldsymbol{E}^{(1)},\cdots,\boldsymbol{E}^{(d)},\cdots,\boldsymbol{E}^{(q)}\}$所对应的DW指数矩阵$\boldsymbol{W}$:

$$\boldsymbol{W}=\begin{bmatrix}w_1^{(1)} & \cdots & w_q^{(1)}\\ \vdots & w_j^{(d)} & \vdots\\ w_1^{(q)} & \cdots & w_q^{(q)}\end{bmatrix}\in\mathbb{R}^{q\times q},\tag{6}$$

$$1\le j\le q,1\le d\le q$$

其中,$w_j^{(d)}$表示主成分数为d时,残差矩阵$\boldsymbol{E}^{(d)}$的第j行所对应的DW指数,其值由式(7)进行计算:

$$w_j^{(d)}=\frac{\sum_{i=2}^{N}\left(e_{ji}^{(d)}-e_{j(i-1)}^{(d)}\right)^2}{\sum_{i=2}^{N}\left(e_{ji}^{(d)}\right)^2},\tag{7}$$

$$1\le j\le q,1\le i\le N,1\le d\le q$$

最终,在$\boldsymbol{W}$中自上而下逐行进行搜索,当搜索至第d行时,如果该行中存在至少有一个元素值接近于2,则证明以当前行数d作为主成分数的FastICA提取的主成分中引入了过多噪声,此时停止遍历,输出最优主成分数$d_{\mathrm{opt}}=d-1$,并将经由FastICA处理后得到的主成分$\boldsymbol{X}\in\mathbb{R}^{d_{\mathrm{opt}}\times N}$作为模糊分段模块的输入数据.否则执行$d=d+1$,继续遍历搜索,直到$d>q$结束.

3.2 基于G-G聚类的多元时间序列模糊分段

给定经由分离降噪模块处理后得到的主成分矩阵 $\boldsymbol{X}=\left[\boldsymbol{x}_1\ \cdots\ \boldsymbol{x}_i\ \cdots\ \boldsymbol{x}_N\right]$ 及分段数 C，其中 $\boldsymbol{T}=\left(t_1,\cdots,t_i,\cdots,t_N\right)$ 为主成分矩阵 $\boldsymbol{X}$ 对应的时间戳，$\boldsymbol{x}_i=\left(x_{i1},x_{i2},\cdots,x_{id_{\text{opt}}}\right)^{\mathrm{T}}$ 代表时刻 t_i 的 d_{opt} 维采样值. 在式(3)所给出的基于G-G聚类的分段算法的目标函数中，由于 $\boldsymbol{x}_i$ 和 t_i 是相互独立的，于是目标函数中距离度量 $\text{dis}^2\left(\boldsymbol{z}_i,\eta_k\right)$ 的定义如式(8)所示：

$$\begin{aligned}\text{dis}^2\left(\boldsymbol{z}_i,\eta_k\right)&=\frac{1}{\alpha_k p\left(\boldsymbol{z}_i,\eta_k\right)}\\&=\frac{1}{\alpha_k p\left(\boldsymbol{x}_i,\boldsymbol{\eta}_k^x\right)p\left(t_i,\boldsymbol{\eta}_k^t\right)}\end{aligned}\tag{8}$$

式(8)中，α_k 表示第 k 个分段对应的混合系数，如式(9)所示：

$$\alpha_k=\frac{\sum_{i=1}^{N}u_{i,k}}{\sum_{j=1}^{C}\sum_{i=1}^{N}u_{i,k}},\ \text{s.t.}\sum_{k=1}^{C}\alpha_k=1,\ \alpha_k\geqslant 0,\ k=1,2,\cdots,C\tag{9}$$

式(8)中的概率密度函数 $p\left(\boldsymbol{x}_i,\boldsymbol{\eta}_k^x\right)$ 由高斯概率密度函数 $G\left(\boldsymbol{x}_i;\ \boldsymbol{v}_k^x;\ \boldsymbol{\Sigma}_k^x\right)$ 给出，表示数据点 $\boldsymbol{x}_i$ 属于第 k 个分段的概率，满足：

$$\begin{aligned}p\left(\boldsymbol{x}_i,\boldsymbol{\eta}_k^x\right)=G\left(\boldsymbol{x}_i;\ \boldsymbol{v}_k^x;\ \boldsymbol{\Sigma}_k^x\right)&=\frac{1}{(2\pi)^{\frac{d_{\text{opt}}}{2}}\sqrt{\det\left(\boldsymbol{\Sigma}_k^x\right)}}\\&\cdot\exp\left(-\frac{1}{2}\left(\boldsymbol{x}_i-\boldsymbol{v}_k^x\right)^{\mathrm{T}}\left(\boldsymbol{\Sigma}_k^x\right)^{-1}\left(\boldsymbol{x}_i-\boldsymbol{v}_k^x\right)\right)\end{aligned}\tag{10}$$

其中，$\boldsymbol{v}_k^x$ 与 $\boldsymbol{\Sigma}_k^x$ 分别表示 $G\left(\boldsymbol{x}_i;\ \boldsymbol{v}_k^x;\ \boldsymbol{\Sigma}_k^x\right)$ 中对应的均值向量与协方差矩阵，表示为

$$\begin{aligned}\boldsymbol{v}_k^x&=\frac{\sum_{i=1}^{N}\left(u_{i,k}\right)^m\boldsymbol{x}_i}{\sum_{i=1}^{N}\left(u_{i,k}\right)^m},\\\boldsymbol{\Sigma}_k^x&=\frac{\sum_{i=1}^{N}\left(u_{i,k}\right)^m\left(\boldsymbol{x}_i-\boldsymbol{v}_k^x\right)\left(\boldsymbol{x}_i-\boldsymbol{v}_k^x\right)^{\mathrm{T}}}{\sum_{i=1}^{N}\left(u_{i,k}\right)^m}\end{aligned}\tag{11}$$

式(8)中的概率密度函数 $p\left(t_i,\boldsymbol{\eta}_k^t\right)$ 由高斯概率密度函数 $G\left(t_i;\ v_k^t;(\sigma_k^t)^2\right)$ 给出，表示时间戳 t_i 属于第 k 个分段的概率，满足：

$$\begin{aligned}p\left(t_i,\boldsymbol{\eta}_k^t\right)&=G\left(t_i;\ v_k^t;(\sigma_k^t)^2\right)\\&=\frac{1}{\sqrt{2\pi(\sigma_k^t)^2}}\cdot\exp\left(-\frac{1}{2}\frac{\left(t_i-v_k^t\right)^2}{(\sigma_k^t)^2}\right)\end{aligned}\tag{12}$$

其中，v_k^t 与 $(\sigma_k^t)^2$ 分别表示 $G\left(t_i;\ v_k^t;(\sigma_k^t)^2\right)$ 中的均值与方差，表示为

$$\begin{aligned}v_k^t&=\frac{\sum_{i=1}^{N}\left(u_{i,k}\right)^m t_i}{\sum_{i=1}^{N}\left(u_{i,k}\right)^m},\\(\sigma_k^t)^2&=\frac{\sum_{i=1}^{N}\left(u_{i,k}\right)^m\left(t_i-v_k^t\right)^2}{\sum_{i=1}^{N}\left(u_{i,k}\right)^m}\end{aligned}\tag{13}$$

将式(8)带入式(3)，可得到完整的目标函数，如式(14)所示：

$$\begin{aligned}J_{(\boldsymbol{X}|\text{GG})}&=\sum_{i=1}^{N}\sum_{k=1}^{C}\frac{\left(u_{i,k}\right)^m}{\alpha_k p\left(\boldsymbol{x}_i,\boldsymbol{\eta}_k^x\right)p\left(t_i,\boldsymbol{\eta}_k^t\right)}\\&=\sum_{i=1}^{N}\sum_{k=1}^{C}\frac{\left(u_{i,k}\right)^m}{\alpha_k G\left(\boldsymbol{x}_i;\ \boldsymbol{v}_k^x;\ \boldsymbol{\Sigma}_k^x\right)G\left(t_i;\ v_k^t;\ \left(\sigma_k^2\right)\right)}\end{aligned}\tag{14}$$

不断迭代优化 $\eta=\left\{\boldsymbol{\eta}_k^x,\boldsymbol{\eta}_k^t|1\leqslant k\leqslant C\right\}=\left\{\boldsymbol{v}_k^x,\boldsymbol{\Sigma}_k^x,v_k^t,\left(\sigma_k^t\right)^2|1\leqslant k\leqslant C\right\}$ 以及 $a=\{\alpha_k|1\leqslant k\leqslant C\}$，使目标函数 $J_{(\boldsymbol{X}|\text{GG})}$ 最小化，可视为分段数 C 对应的G-G聚类完成.

3.3 基于MDL自适应确定最优分段数

基于G-G聚类的分段算法需要人为预先指定分段数 C. 在完成多元时间序列分段任务的过程中，最优分段数 C_{opt} 的确定是较为重要的一环. 若指定的分段数 C 小于最优分段数 C_{opt}，将使分段算法不能充分揭示多元时间序列的趋势变化，反之将产生过拟合问题，增加计算复杂度. 为解决上述问题，本文将使用最小描述长度[20]作为有效性指标以自适应确定最优分段数.

最小描述长度是一种基于无损压缩原则的模型选择方法，其遵循这样的假设：模型可以压缩的数据越多，该模型就能够越准确的揭示数据本身所蕴含的底层模式，因此分配最小位数以进行无损数据压缩的模型被视为最佳模型. 给定经由FastICA处理过后的主成分 $\boldsymbol{X}\in\mathbb{R}^{d_{\text{opt}}\times N}$，根据最小描述长度原理，构建目标函数如式(15)所示：

$$\text{Cost}_{\text{MDL}}=\text{Cost}\left(\boldsymbol{X}|\text{GG}\right)+\text{Cost}(\text{GG})\tag{15}$$

式(15)中，$\text{Cost}\left(\boldsymbol{X}|\text{GG}\right)$ 表示在给定基于G-G聚类的分段模型的条件下，描述 $\boldsymbol{X}$ 所花费的成本，$\text{Cost}(\text{GG})$ 表示描述基于G-G聚类的分段模型本身所花费的成本. 对于式(15)中的第一项 $\text{Cost}\left(\boldsymbol{X}|\text{GG}\right)$，使用哈夫曼编码(Huffman coding)[10]进行位数分配：

$$\mathrm{Cost}(\boldsymbol{X}|\mathrm{GG})=-\sum_{i=1}^{C}\sum_{k=1}^{N}\left(u_{i,k}\right)^{m}\log_2\left(\alpha_k\cdot G\left(\boldsymbol{x}_i;\ \boldsymbol{v}_k^x;\ \boldsymbol{\Sigma}_k^x\right)G\left(t_i;\ v_k^t;\ \left(\sigma_k^2\right)\right)\right)\tag{16}$$

式(15)中的第二项Cost(GG)等同于对主成分长度N,主成分维数d_{opt},分段数C,分段原型$\eta=\left\{\boldsymbol{\eta}_k^x,\boldsymbol{\eta}_k^t|1\le k\le C\right\}=\left\{\boldsymbol{v}_k^x,\boldsymbol{\Sigma}_k^x,v_k^t,\left(\sigma_k^t\right)^2|1\le k\le C\right\}$以及混合系数$a=\{\alpha_k|1\le k\le C\}$进行位数分配,如式(17)所示:

$$\mathrm{Cost}(\mathrm{GG})=\log^*(N)+\log^*\left(d_{\mathrm{opt}}\right)+\log^*(C)+\mathrm{FB}\times C\times\left(3+d+\frac{d\times(d+1)}{2}\right)\tag{17}$$

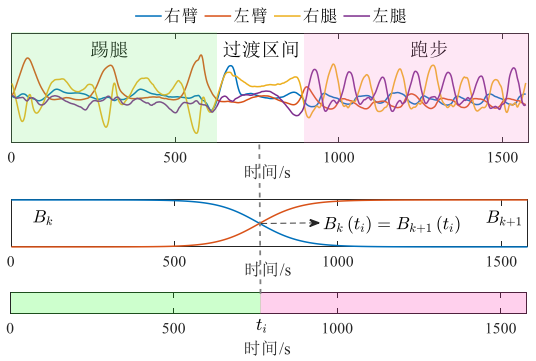
其中,主成分长度N,维数d_{opt}以及分段数C都为整数,因此我们采用通用整数编码为其进行位数分配,总共需要$\log^*(N)+\log^*(d)+\log^*(C)$位. 其中$\log_2^*(x)$表示通用整数码长,该编码方式定义为:$\log_2^*(x)\approx\log_2(x)+\log_2\log_2(x)+\log_2\log_2\log_2(x)+\cdots$,其中只有正项包含在总和之中[25]. 分段原型$\eta=\left\{\boldsymbol{\eta}_k^x,\boldsymbol{\eta}_k^t|1\le k\le C\right\}=\left\{\boldsymbol{v}_k^x,\boldsymbol{\Sigma}_k^x,v_k^t,\left(\sigma_k^t\right)^2|1\le k\le C\right\}$以及混合系数$a=\{\alpha_k|1\le k\le C\}$中的每一个元素都采用小数表示,因此需要分配$\mathrm{FB}\times C\times\left(3+d_{\mathrm{opt}}+\frac{d_{\mathrm{opt}}\times\left(d_{\mathrm{opt}}+1\right)}{2}\right)$位,其中FB为浮点数位数,满足FB=4×8. 最终,能够使目标函数式(15)取得最小值的分段数C被视为最优分段数C_{opt}:

$$C_{\mathrm{opt}}=\underset{C}{\mathrm{argmin}}\left(\mathrm{Cost}(\boldsymbol{X}|\mathrm{GG})+\mathrm{Cost}(\mathrm{GG})\right)\tag{18}$$

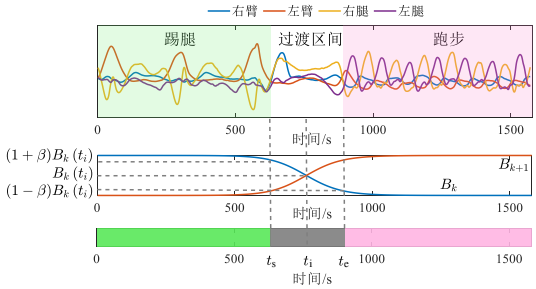
在确定最优分段数目C_{opt}之后,可以回溯得到相应的簇原型$\eta=\left\{\boldsymbol{\eta}_k^x,\boldsymbol{\eta}_k^t|1\le k\le C_{\mathrm{opt}}\right\}$,依据式(19)得到所有时间点$\{t_i|1\le i\le N\}$隶属于各个分段$\{k|1\le k\le C_{\mathrm{opt}}\}$的最优模糊隶属度集合$\{B_k(t_i)|1\le k\le C_{\mathrm{opt}}\}$.

$$B_k(t_i)=\frac{\exp\left(-\frac{1}{2}\frac{\left(t_i-v_k^t\right)^2}{\sigma_{k,t}^2}\right)}{\sum_{j=1}^{C_{\mathrm{opt}}}\exp\left(-\frac{1}{2}\frac{\left(t_i-v_k^t\right)^2}{\sigma_{k,t}^2}\right)},1\le i\le N,1\le k\le C_{\mathrm{opt}}\tag{19}$$

得到隶属度集合后,计算使得相邻隶属度取值相等的点的集合$\{t_i|B_k(t_i)=B_{k+1}(t_i),1\le i\le N,1\le k\le C_{\mathrm{opt}}-1\}$作为"硬划分"的分段点集合. 如图2(a)所示,相邻的隶属度函数B_k(蓝线)和B_{k+1}(橙线)在t_i处取值相等,因此选择将t_i作为"硬划分"的分段点. 在求得t_i的位置后,为实现对多元时间序列的"软划分",AMTS-FG以t_i作为预估过渡区间的中心点,进一步计算模糊隶属度B_k取值分别为$(1+\beta)B_k(t_i)$和$(1-\beta)B_k(t_i)$所对应的时间戳作为预估区间的左右边界点,如图2(b)中的t_s(左边界点)以及t_e(右边界点). 经过多次实验测试,β=0.4时过渡区间的预估结果较为合理.



(a) "硬划分"的分段结果



(b) "软划分"的分段结果

图2　AMTS-FG实现"硬划分"和"软划分"的原理示意图

3.4　AMTS-FG的实现步骤

上文对AMTS-FG各个模块的组成及原理进行了详细阐述,本小节则主要介绍AMTS-FG的具体实现细节,如算法1.

4　实验验证

在实验验证部分,本文对多元时间序列分段准确性的评价指标加以介绍,并采用多种真实数据集对所提出算法的分段准确性与可解释性进行评估. 文中的所有实验均在Python3.8、1.80 GHz处理器、16.0 GB内存环境中执行.

4.1　多元时间序列分段准确性评估指标

在多元时间序列分段结果的准确性评估过程中,选择合适的评价指标尤为关键. 目前采用较为广泛的评价指标为F_1[10,16],如式(20)所示:

$$F_1=\frac{2\times\mathrm{TP}}{2\times\mathrm{TP}+\mathrm{FN}+\mathrm{FP}}\tag{20}$$

假定待评估算法生成的实验分段点(experimental segmentation)的位置集合表示为$\{\mathrm{ES}_j|j=1,\cdots,n\}$,数据集对应的真实分段点(ground truth)的位置集合表示为$\{\mathrm{GT}_i|i=1,\cdots,m\}$,则式(20)中的真阳性TP代表$\{\mathrm{ES}_j|j=$

算法1 AMTS-FG

输入：原始多元时间序列数据矩阵$\boldsymbol{S}\in\mathbb{R}^{q\times N}$，G-G聚类的终止阈值$\varepsilon=10^{-4}$；

输出：预估过渡区间集合；

Step1：初始化主成分数$d=1$，执行FastICA算法，并根据式(1)计算得到残差矩阵$\boldsymbol{E}^{(d)}$；

Step2：令$d=d+1$，返回Step1，直至$d>q$；

Step3：根据式(6)和式(7)计算残差矩阵集合$\{\boldsymbol{E}^{(1)},\cdots,\boldsymbol{E}^{(d)},\cdots,\boldsymbol{E}^{(q)}\}$对应的DW指数矩阵$\boldsymbol{W}$；

Step4：令$d=1$；

Step5：在$\boldsymbol{W}$中进行自上而下的遍历搜索，如果$\boldsymbol{W}$的第d行中至少存在一个元素值接近于2，停止遍历，输出最优主成分数$d_{\text{opt}}=d-1$以及最优主成分$\boldsymbol{X}\in\mathbb{R}^{d_{\text{opt}}\times N}$，转至Step7；

Step6：令$d=d+1$，返回Step5，直至$d>q$；

Step7：初始化分段数$C=1$，$l=1$，$\text{Cost}_{\text{MDL}}^{(C)}=\infty$；

Step8：令$C=C+1$；

Step9：对$u_{i,k}(l)$进行初始化：

$$u_{i,k}(l)=\begin{cases}1,\ 1\le i\le\left\lfloor\dfrac{N}{C}\right\rfloor\\0,\ \left\lfloor\dfrac{N}{C}\right\rfloor+1\le i\le N\end{cases},\ 1\le i\le N,\ 1\le k\le C;$$

Step10：根据式(9)、式(11)、式(13)更新参数$a(l)=\{\alpha_k(l)|1\le k\le C\}$以及$\eta(l)=\left\{\boldsymbol{v}_k^x(l),\boldsymbol{\Sigma}_k^x(l),v_k^t(l),\left(\sigma_k^t(l)\right)^2|1\le k\le C\right\}$；

Step11：令$l=l+1$，根据式(4)更新$u_{i,k}(l)$，若$\|\boldsymbol{U}(l)-\boldsymbol{U}(l-1)\|<\varepsilon$，转至Step12，否则返回Step10；

Step12：根据式(15)~(17)计算$\text{Cost}_{\text{MDL}}^{(C)}$，如果$\text{Cost}_{\text{MDL}}^{(C)}\le\text{Cost}_{\text{MDL}}^{(C-1)}$，则令$\text{Cost}_{\text{MDL}}^{(C)}=\text{Cost}_{\text{MDL}}^{(C-1)}$并返回Step8，否则转至Step13；

Step13：设定最优分段数$C_{\text{opt}}=C-1$，根据式(19)得到最优模糊隶属度集合$\{B_k(t_i)|1\le k\le C_{\text{opt}}\}$；

Step14：依据最优模糊隶属度集合计算“硬划分”的分段点集合$\{t_i|B_k(t_i)=B_{k+1}(t_i),1\le i\le N,1\le k\le C_{\text{opt}}-1\}$；

Step15：以分段点集合中的每个t_i为中心点，计算使隶属度B_k取值分别为$(1+\beta)B_k(t_i)$和$(1-\beta)B_k(t_i)$的时间戳t_{s}和t_{e}，将其作为预估过渡区间的左右边界点，最终将所有预估过渡区间以集合形式输出.

$1,\cdots,n\}$与$\{\text{GT}_i|i=1,\cdots,m\}$中元素相等的次数，假阳性FP代表$\{\text{ES}_j|j=1,\cdots,n\}$中没有与之相等的真实分段点的元素个数，假阴性FN代表$\{\text{GT}_i|i=1,\cdots,m\}$中没有与之相等的实验分段点的元素个数. F_1越高，分段结果越精确.

然而，最近有研究者[26]指出，直接将F_1作为评价指标将倾向于惩罚接近真实分段点的合理分割方案. 例如对于真实分段点位置为10 000的多元时间序列，F_1只奖励能够获得实验分段点位置为10 000的算法，而惩罚获得实验分段点位置为10 001或9 999的算法，尽管它们的分割方案同样是合理的. 因此，为弥补上述缺陷，在数据集本身未提供真实过渡区间信息的情况下，本文以每个真实分段点为中心，并以$0.025N$为半径定义真实过渡区间，其中N代表整个多元时间序列的总长度. 只要实验分段点落入该区间中，则可被标记为真阳性TP.

在一些特殊情况下，譬如两种待评估算法获得的实验分段点同时落在真实过渡区间中，F_1将无法继续评判两者的优劣. 为解决上述问题，Gharghabi等人[14]提出了一种用于多元时间序列分段的准确性评价指标MAE(Mean Absolute Error)，该指标通过计算实验分段点位置与其对应最邻近的真实分段点位置的绝对值距离来评判分段结果的准确性，如式(21)所示：

$$\text{MAE}=\frac{\sum_{i=1}^{m}|\text{GT}_i-\text{ES}_{\tilde{j}}|}{N},\ \tilde{j}=\underset{j\in(1,n)}{\text{argmin}}|\text{GT}_i-\text{ES}_j| \quad (21)$$

MAE越低，分段结果越精确. 需要注意的是，MAE只适用于实验分段点总数等于真实分段点总数的情况，而F_1适用于实验分段数与真实分段数不相等的情况，因此可与之形成互补.

4.2 多元时间序列分段真实数据集实验

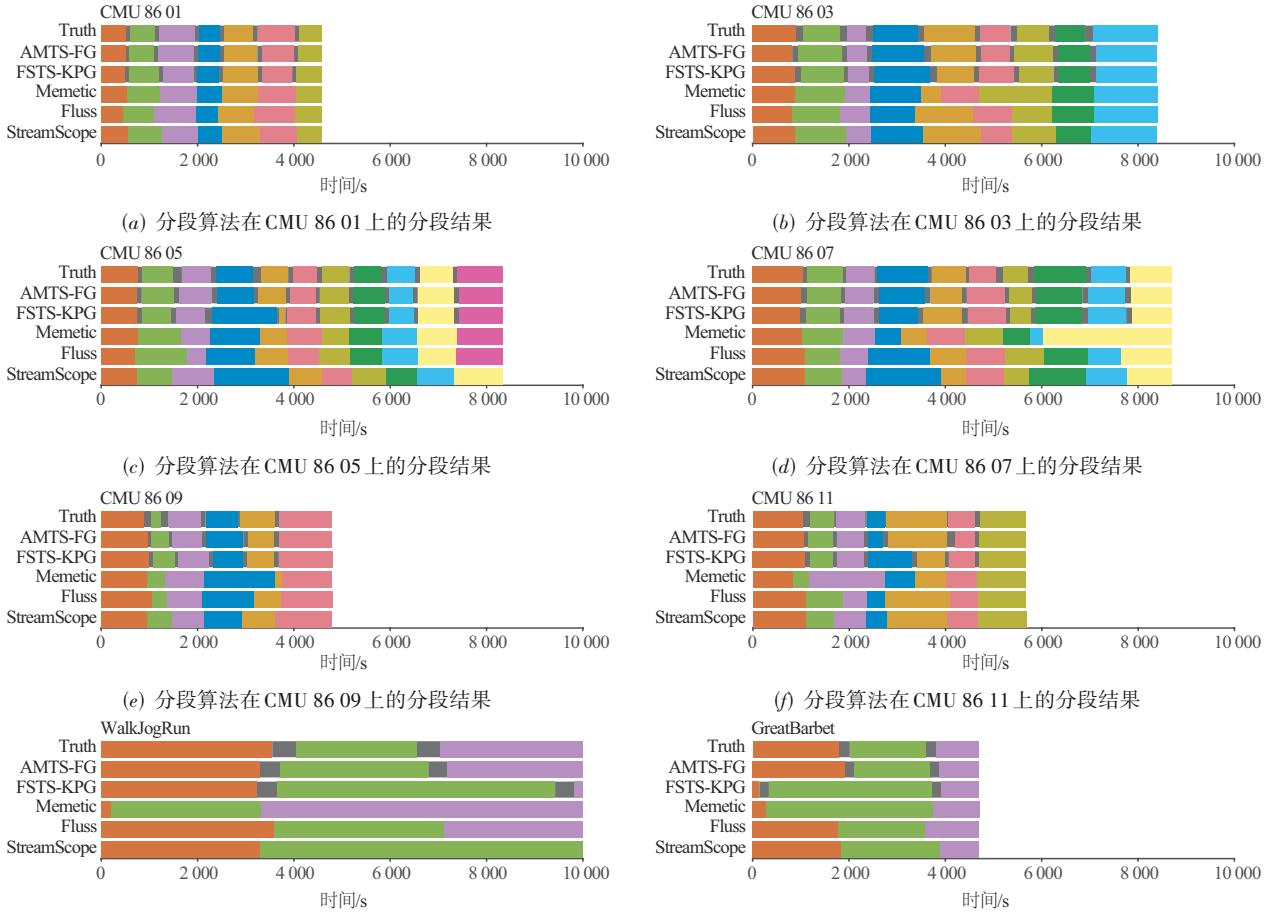
本节将从分段准确性以及可解释性两个方面来验证所提出的AMTS-FG算法的性能. 实验数据采用包含动作捕捉，信息检索，语音发音以及能源电力等各行业的多元时间序列真实数据集. 其中，除NILM和flus之外，所有数据集均具有真实分段信息. 实验数据的具体信息见表1.

表1 多元时间序列真实数据集

数据集	时间戳总数	变量维数	真实分段数	真实过渡区间
CMU 86 01	4 579	42	7	√
CMU 86 03	8 401	42	9	√
CMU 86 05	8 340	42	11	√
CMU 86 07	8 702	42	10	√
CMU 86 09	4 794	42	6	√
CMU 86 11	5 674	42	7	√
WalkJogRun	10 000	2	3	—
GreatBarbet	4 700	2	3	—
NILM	720	5	—	—
flus	472	4	—	—

表1中，CMU 86[27]中包含的6个数据集分别记录了具有不同时间戳总数的42维动作捕捉序列，不同动作之间转换的真实过渡区间及真实分段点(区间中心)由文献[28]提供；WalkJogRun[13]收集了受试者在走路，单脚跳以及跑步时左下臂以及左腿的传感器数值，真实分段点为动作转换时间；GreatBarbet[13]收集了三种不同鸟的叫声，不同种类的鸟切换叫声的时间代表其真实分段点；Nilm[29]收集了5种家用电器在一天24小时内的功率变化情况；flus[10]包含了用户使用Google搜索引擎在一段时间内查询与流感相关的4种关键词的搜索量. 所有数据集在实验时均经过z-score标准化处理.

为验证AMTS-FG的有效性,本文使用StreamScope[12],Memetic[11],Fluss[13]以及FSTS-KPG[17]作为对比算法进行实验. StreamScope利用隐马尔可夫模型以及MDL实现自适应分段,无需预设任何参数;Memetic将多元时间序列的分段问题转化为协方差正则化的最大似然估计问题并进行优化求解,其需要用户预设种群大小ps,生成种群大小gs,种群代数gn,交叉概率pc,变异概率pm,局部细化概率pl以及分段数k;Fluss将多元时间序列对应的矩阵轮廓结构的局部最小值作为候选分割点,其需要预设子序列长度L及分段数k;FSTS-KPG通过KPCA对多元时序数据进行降维,并采用G-G聚类和MDBI指标实现最优分割,其需要预设KPCA的核函数带宽σ及分段数k. 上述对比算法所采用的所有超参数均基于最好情况进行设置. 此外,本实验设定FSTS-KPG中G-G聚类的区间估计参数β与AMTS-FG相同,以便对比AMTS-FG与FSTS-KPG对于过渡区间的估计效果. 在完成对比算法的参数设置后,本文在所有具有真实分段信息的数据集上对各种方法的分段效果进行检验,结果如图3所示.



(a) 分段算法在CMU 86 01上的分段结果　(b) 分段算法在CMU 86 03上的分段结果

(c) 分段算法在CMU 86 05上的分段结果　(d) 分段算法在CMU 86 07上的分段结果

(e) 分段算法在CMU 86 09上的分段结果　(f) 分段算法在CMU 86 11上的分段结果

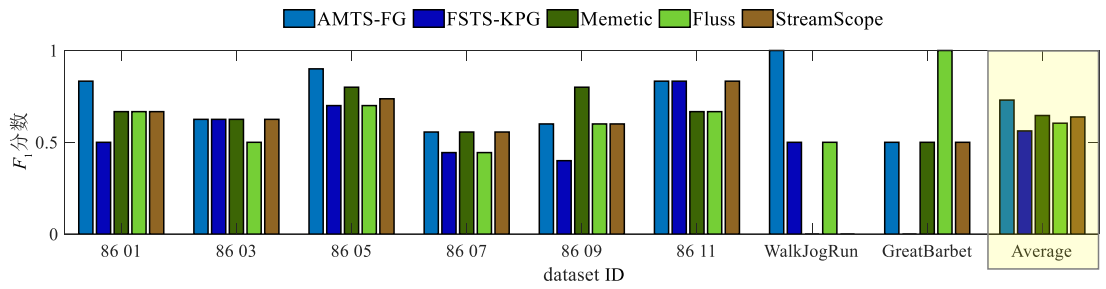
(g) 分段算法在WalkJogRun上的分段结果　(h) 分段算法在GreatBarbet上的分段结果

图3　分段算法在不同数据集上的分段结果

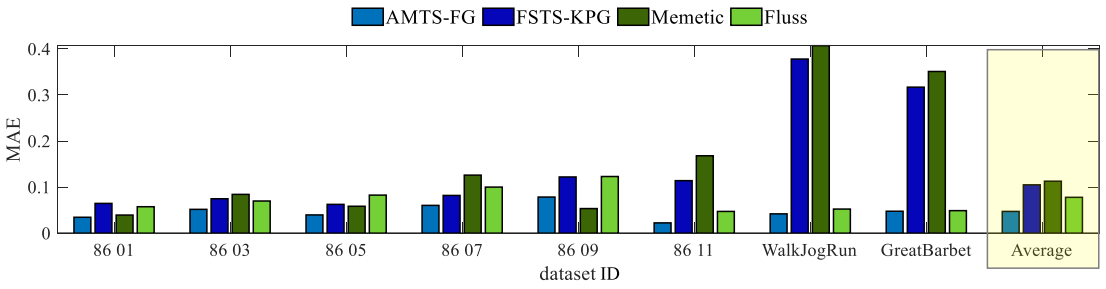
图3(a)~(h)中,Truth以及各个分段算法的条形图中的彩色色块分别代表真实分段区域和实验分段区域,Truth,AMTS-FG以及FSTS-KPG的条形图中的灰色阴影分别代表真实过渡区间和实验预估区间. 显然,无论是分段点的"硬划分"还是过渡区间的估计,AMTS-FG都是最接近Truth的;FSTS-KPG和Memetic对于变量维数较小的数据集(诸如GreatBarbet)容易产生较差的分段结果,Fluss则容易受到冗余变量的干扰(诸如CMU 86数据集)而导致分段结果不理想,并且这三种方法都需要人为设置超参数;StreamScope虽然无需预设超参数,但其对于均值和方差等统计特性变化不大的数据集(诸如WalkJogRun)通常会给出分段数小于真实情况的结果. 表2以及图4显示了分段算法在不同真实数据集上的准确性指标评估情况,由于StreamScope在一些数据集上(诸如CMU 86 05及WalkJogRun)无法给出与真实分段数相等的分段结果,因此该算法在图4(b)中不予显示MAE. 显然,AMTS-FG的平均F_1和MAE显著优于对比算法.

表2 分段方法在不同真实数据集上的F_1和MAE 单位:%

方法 数据集	AMTS-FG		FSTS-KPG[17]		Memetic[11]		Fluss[13]		StreamScope[12]	
	F_1	MAE	F_1	MAE	F_1	MAE	F_1	MAE	F_1	MAE
CMU 86 01	**83.3**	**3.45**	50.0	6.46	66.7	3.93	66.7	5.74	66.7	6.18
CMU 86 03	**62.5**	**5.17**	**62.5**	7.47	**62.5**	8.42	50.0	6.98	**62.5**	5.67
CMU 86 05	**90.0**	**3.96**	70.0	6.25	80.0	5.84	70.0	8.26	73.7	—
CMU 86 07	**55.6**	**6.03**	44.4	8.19	**55.6**	12.6	44.4	10.0	**55.6**	8.29
CMU 86 09	60.0	**7.84**	40.0	12.2	**80.0**	5.34	60.0	12.3	60.0	8.47
CMU 86 11	**83.3**	2.23	**83.3**	11.4	66.7	16.8	66.7	4.72	**83.3**	**1.18**
GreatBarbet	50.0	**4.76**	0	31.7	50.0	35.1	**100.0**	4.89	50.0	5.85
WalkJogRun	**100.0**	**4.19**	50.0	37.8	0	40.7	50.0	5.23	0	—
average	**73.0**	**4.72**	56.2	10.49	64.6	11.28	60.4	7.78	63.8	—



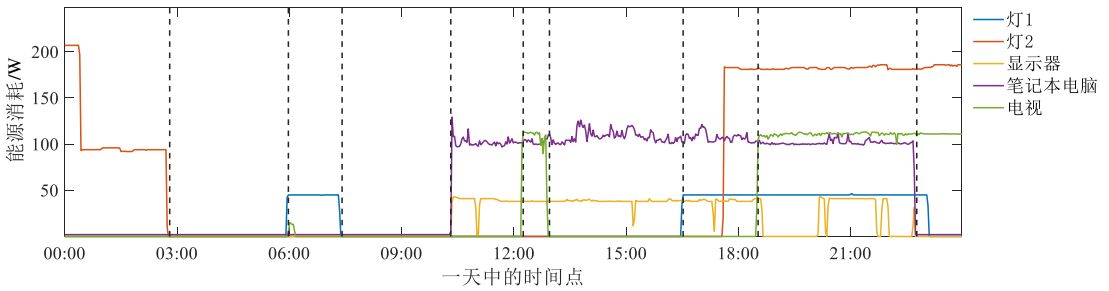
(*a*) F_1的计算情况



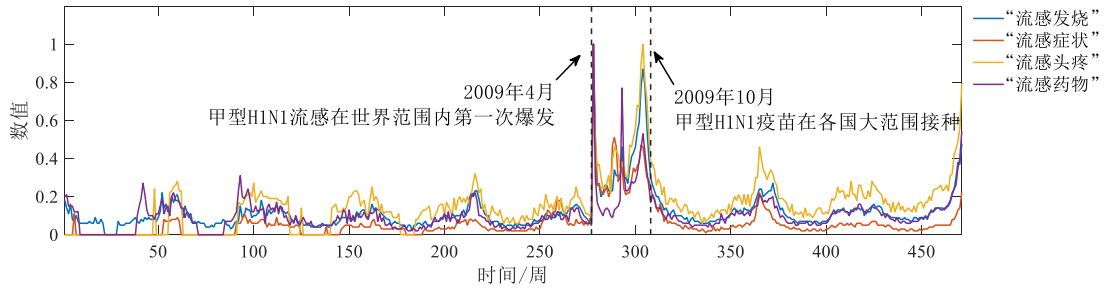
(*b*) MAE的计算情况

图4 各个评价指标的计算情况

在完成准确性评估后,本文利用AMTS-FG对NILM和flus数据集所获得的分段结果进行可解释性评估,可视化结果如图5所示.由图5(*a*)可知,AMTS-FG的分段结果与NILM中不同家用电器的开关状态一致,进而可总结出该家庭的能源消耗模式.如图所示,一共形成9个分段,第1分段表示夜晚期间灯2调暗后逐渐关闭的过程;第2分段表示深夜所有家用电器处于关闭状态;第5分段代表前半段工作日的耗电量情况;第6分段表示午休期间打开电视的事件;第7分段代表后半段工作日的耗电量情况;第9段与第10段可被视为晚上的娱乐时间.图5(*b*)则显示了AMTS-FG对于flus数据集的分段结果,获得的两个分段点刚好对应了“2009年4月甲型H1N1流感在世界范围内的第一次爆发高峰”,以及“2009年10月甲型H1N1流感疫苗在各国大范围接种”两大历史事件.上述两个实验体现出AMTS-FG的分段结果具有良好的可解释性,能够较好地应用于不同领域的多元时间序列数据的模式识别.



(*a*) AMTS-FG对于NILM的分段结果



(*b*) AMTS-FG对于flus的分段结果

图5　AMTS-FG对NILM以及flus的分段结果

5　结论

本文提出一种基于FastICA以及G-G聚类的多元时间序列自适应分段算法AMTS-FG. 首先通过FastICA对原始多元时间序列进行特征提取,并采用DW指数自适应确定高信噪比的主成分,其次基于MDL设计基于G-G聚类的自适应分段方法,实现对于多元时间序列的“软划分”. 在真实数据集上的实验结果表明,AMTS-FG能够有效对高维且噪声较强的多元时间序列进行较为精准的分段,并且获得的分段结果显示出较强的可解释性. 未来,我们考虑将AMTS-FG扩展至在线场景,实现对多元时间数据流的在线分段.

参考文献

[1] MAYA S, YAMAGUCHI A, NISHINO K, et al. Lag-aware multivariate time-series segmentation[C]//Proceedings of the 2020 SIAM International Conference on Data Mining. Philadelphia: SIAM, 2020: 622-630.

[2] XU J, ZHOU J, TAN P N, et al. Spatio-temporal multi-task learning via tensor decomposition[J]. IEEE Transactions on Knowledge and Data Engineering, 2019, 33(6): 2764-2775.

[3] CHE Z, PURUSHOTHAM S, CHO K, et al. Recurrent neural networks for multivariate time series with missing values[J]. Scientific Reports, 2018, 8(1): 1-12.

[4] KEOGH E, CHU S, HRT D, et al. An online algorithm for segmenting time series[C]//Proceedings 2001 IEEE International Conference on Data Mining. Piscataway: IEEE, 2001: 289-296.

[5] CHUNG F L, FU T C, NG V, et al. An evolutionary approach to pattern-based time series segmentation[J]. IEEE Transactions on Evolutionary Computation, 2004, 8(5): 471-489.

[6] LIU S, YAMADA M, COLLIER N, et al. Change-point detection in time-series data by relative density-ratio estimation[J]. Neural Networks, 2013, 43: 72-83.

[7] WANG P, WANG H, WANG W. Finding semantics in time series[C]//Proceedings of the 2011 ACM SIGMOD International Conference on Management of Data. New York: ACM, 2011: 385-396.

[8] LI L, MCCANN J, POLLARD N S, et al. Dynammo: mining and summarization of coevolving sequences with missing values[C]//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2009: 507-516.

[9] HALLAC D, NYSTRUP P, BOYD S. Greedy gaussian segmentation of multivariate time series[J]. Advances in Data Analysis and Classification, 2018, 13(3): 727-751.

[10] 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.

[11] LIM H, CHOI H, CHOI Y, et al. Memetic algorithm for multivariate time-series segmentation[J]. Pattern Recognition Letters, 2020, (138): 60-67.
[12] KAWABATA Y, MATSUBARA Y, SAKURAI Y. Automatic sequential pattern mining in data streams[C]//Proceedings of the 28th ACM International Conference on Information and Knowledge Management. New York: ACM, 2019: 1733-1742.
[13] GHARGHABI S, YEH C C M, Ding Y, et al. Domain agnostic online semantic segmentation for multi-dimensional time series[J]. Data Mining and Knowledge Discovery, 2019, 33(1): 96-130.
[14] GHARGHABI S, DING Y, 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. Piscataway: IEEE, 2017: 117-126.
[15] DELDARI S, SMITH D V, SADRI A, et al. Espresso: Entropy and shape aware time-series segmentation for processing heterogeneous sensor data[J]. Proceedings of the ACM on Interactive Mobile Wearable and Ubiquitous Technologies, 2020, 4(3): 1-24.
[16] HALLAC D, VARE S, BOYD S, et al. Toeplitz inverse covariance-based clustering of multivariate time series data[C]//Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York: ACM, 2017: 215-223.
[17] 王玲, 朱慧. 基于KPCA和G-G聚类的多元时间序列模糊分段[J]. 控制与决策, 2021, 36(1): 115-124.
WANG Ling, ZHU Hui. Fuzzy segmentation of multivariate time series with KPCA and G-G clustering[J]. Control and Decision, 2021, 36(1): 115-124. (in Chinese)
[18] ZHOU F, DE LA TORRE F, HODGINS J K. Hierarchical aligned cluster analysis for temporal clustering of human motion[J]. IEEE Transactions on Pattern Analysis and Machine Intelligence, 2013, 35(3): 582-596.
[19] HOAI M, DE LA TORRE F. Maximum margin temporal clustering[C]//Proceedings of the Fifteenth International Conference on Artificial Intelligence and Statistics (AISTAT). Cambridge: JMLR, 2012: 520-528.
[20] GRUNWALD P D. The Minimum Description Length Principle[M]. Cambridge: MIT Press, 2007: 132.
[21] HYVARINEN A, OJA E. Independent component analysis: algorithms and applications[J]. Neural Networks, 2000, 13(4): 411-430.
[22] HYVARINEN A. Fast and robust fixed-point algorithms for independent component analysis[J]. IEEE Transactions on Neural Networks, 1999, 3(10): 626-634.
[23] ABONYI J, FEIL B, NEMETH S, et al. Modified gath geva clustering for fuzzy segmentation of multivariate time-series[J]. Fuzzy Sets and Systems, 2005, 149(1): 39-56.
[24] BOUVERESSE D J R, RUTLEDGE D N. Independent Components Analysis: Theory and Applications[M]//Data Handling in Science and Technology Resolving Spectral Mixtures with Applications from Ultrafast Time-Resolved Spectroscopy to Super-Resolution Imaging. 1st ed. New York: Elsevier, 2016: 225-277.
[25] FENG W, LIU S, FALOUTSOS C, et al. Eaglemine: vision-guided micro-clusters recognition and collective anomaly detection[J]. Future Generation Computer Systems, 2021, (115): 236-250.
[26] LIN J F S, KARG M, KULIC D. Movement primitive segmentation for human motion modeling: A framework for analysis[J]. IEEE Transactions on Human-Machine Systems, 2016, 46(3): 325-339.
[27] HODGINS J K. CMU graphics lab motion capture database[DB/OL]. [2020-06-27]. http://mocap.cs.cmu.edu/.
[28] BARBIC J, SAFONOVA A, PAN J Y, et al. Segmenting motion capture data into distinct behaviors[C]//Proceedings of Graphics Interface 2004. Waterloo: Canadian Human-Computer Communications Society, 2004: 185-194.
[29] REINHARDT A, BAUMANN P, BURGSTAHLER D, et al. On the accuracy of appliance identification based on distributed load metering data[C]//2012 Sustainable Internet and ICT for Sustainability (SustainIT). Piscataway: IEEE, 2012: 1-9.

作者简介

王 玲 女,1974年出生于北京市. 现为北京科技大学自动化学院教授. 研究方向为数据挖掘、模式识别.
E-mail: lingwang@ustb.edu.cn

李泽中 男,1998年出生于山西省. 现为北京科技大学自动化学院硕士生. 研究方向为数据挖掘、模式识别.
E-mail: g20208697@xs.ustb.edu.cn