文章编号: 1001-0920(2018)11-1950-09 DOI: 10.13195/j.kzyjc.2017.0877

基于中心度的标签传播时间序列聚类方法

李海林†, 梁 叶

(华侨大学 工商管理学院，福建 泉州 362021)

摘 要: 为了实现时间序列自动聚类,以及更为细致地描述时间序列之间的结构关系,引入社区发现方法来研究时间序列聚类. 针对标签传播方法在标签传播过程中具有较强不确定性,以及算法对网络结构较为敏感等问题,提出一种基于中心度的标签传播时间序列聚类方法;通过构建时间序列网络空间结构,将每条时间序列看作一个节点,根据每个节点的中心度来得到标签更新顺序;计算节点对于每个簇的归属度,再利用节点的归属度和标签的传播实现节点的划分,从而实现时间序列聚类. 所提方法通过分析时间序列之间的连接关系来发现其在欧氏空间的结构特征,进而实现空间结构的有效划分. 实验结果表明,所提方法无需确定初始簇中心,能够有效划分人工数据网络和真实社会网络,在时间序列数据聚类中取得了良好的聚类效果.

关键词: 标签传播；时间序列；聚类分析；社区发现；中心度；数据挖掘

中图分类号: TP391 **文献标志码:** A

Time series clustering method with label propagation based on centrality

LI Hai-lin†, LIANG Ye

(School of Business Administration，Huaqiao University，Quanzhou 362021，China)

Abstract: In order to cluster time series automatically and describe be the structural relations between of time series in more detail, this paper introduces the community discovery method to study time series clustering. According to the ability of the label propagation method which has limitation of uncertainty in the process and the sensitivity of the algorithm to the network structure, a clustering method for time series with label propagation based on centrality is proposed. Time series network structure is built, each time series is treat as a node in the network, and an updating order of labels is obtained according to each node's centrality. The membership degree of each node belonging to each community is computed, and the community is divided using belonging degree and label propagation, so as to achieve time series clustering. The proposed method analyzes the connection relationships among time series to find the structure features in the Euclidean space, thereby achieing the valid division of space structure. The experimental results demonstrate that the proposed clustering method does not need to determine the initial cluster center objects. It not only can detect simulated data network and real social network, but also obtains better results in time series clustering.

Keywords: label propagation；time series；clustering analysis；community detection；centrality degree；data mining

0 引 言

在时间序列数据挖掘领域中,时间序列可以通过相似性来衡量两者的关系[1-2]. 然而,针对聚类、分类、预测等挖掘任务,仅利用时间序列的相似性未能够充分挖掘时间序列之间的结构关系. 复杂网络是一种强大的挖掘工具[3-5],能够描述任意两对或者两组数据样本之间的关系. 然而,利用复杂网络的分析方法来研究时间序列数据挖掘的相关工作仍然不多[6]. 在相关研究文献[7-8]中,大部分工作是将某条时间序列的所有数据点构建一个复杂网络,利用复杂网络分析来研究该时间序列的特性. 因此,将一条时间序列作为一个数据对象,利用网络的形式表征所有对象之间的关系,可为相关研究提供一种新的时间序列数据挖掘视角.

社区结构是复杂网络的一个重要特征[9],而时间序列聚类所形成的簇与“社区结构”有着许多相同之处,即“同簇节点连接紧密(同簇时间序列形态特征相似),异簇节点相互连接稀疏(异簇时间序列形态特征

收稿日期: 2017-07-04；**修回日期:** 2017-11-02.

基金项目: 国家自然科学基金项目(71771094, 61300139)；福建省社会科学规划基金项目(FJ2017B065)；福建省高等学校新世纪优秀人才支持计划项目(Z1625112).

作者简介: 李海林(1982−), 男, 副教授, 博士, 从事数据挖掘与机器学习等研究；梁叶(1992−), 女, 硕士生，从事数据挖掘与金融数据分析的研究.

†通讯作者. E-mail: hailin@hqu.edu.cn

差异大)". 大多文献中使用的聚类方法为K-means、K-medois、层次聚类等,但这些方法只能够根据指定的相似性度量方法识别特定形态的簇. 在欧氏空间中,通过基于网络的聚类可以根据数据任意的连接模式捕获任意形态的簇[6],并且当网络节点较为稀疏时,很多社区发现方法的时间复杂度几乎是线性的. 为此,利用社区发现技术来实现时间序列数据挖掘是一种新的研究思路. 与此同时,标签传播方法是一种简单高效的社区发现方法[10],能够利用节点之间的连接关系自动扩散信息,无需任何先验知识及假设. 将该机制应用到时间序列聚类分析,能够找到影响力大的时间序列,再利用时间序列之间的信息传播,自动达到"物以类聚"的效果.

为了克服传统时间序列聚类方法需要初始化簇中心,以及未能充分反映时间序列之间结构关系等问题,同时减少标签传播过程中存在的不确定性,本文提出一种基于中心度的标签传播时间序列聚类方法. 首先介绍经典标签传播的基本原理,以及标签传播和时间序列聚类的相关研究背景,然后提出基于中心度的标签传播时间序列聚类方法,并通过实验分析新的时间序列聚类方法的性能. 实验结果表明,新方法能够有效划分人工数据网络和真实社会网络,在时间序列聚类任务中取得了良好的效果.

1　相关理论基础

标签传播算法(Label propagation algorithm, LPA)的基本思想是假设一个网络具有一定的社区结构,任意一个节点所在的社区由它大多数的邻接点共同决定[3]. 标签传播算法无需社区大小和个数、优化目标函数等先验知识,思想简单且操作易于实现,同时具有接近线性的时间复杂度,能够高效地应用到大规模网络中. 给定一个无向无权网络$G(V,E)$,V表示节点集,E表示边集,通过算法得到节点的标签集L,以c_x表示节点x的标签. 根据经典LPA方法,其算法过程的伪代码如下.

算法1　标签传播算法.

输入:网络$G(V,E)$;

输出:节点的标签集L.

Step 1: $t=0$,初始化网络中所有节点的标签,针对节点x,有$L_x(0)=x$,$L_x(t)$表示节点x在第t次迭代过程中的标签.

Step 2: 设置$t=1$.

Step 3: 随机排序网络节点,得到排序后的节点集V'.

Step 4: 针对每个节点$x\in V'$,令$L_x(t)=f(L_{\mathrm{Nb}(x)}(t))$,其中$\mathrm{Nb}(x)$表示节点$x$的邻接点集,$f(\cdot)$函数表示众数,即返回$L_{\mathrm{Nb}(x)}(t)$中出现频率最多的标签.

Step 5: 若$L(t)$与$L(t-1)$相等($L(t)$表示第t次迭代得到的标签集),算法停止,反之$t=t+1$,返回Step 3,算法继续.

在LPA算法中,由于在双分或近似双分网络结构中,同步更新标签会导致循环振荡[1,11],因此采用异步更新. 当节点的邻接点具有多个数量最多的标签时,该节点随机从这些标签中选择一个作为自己的新标签,这种随机性使得社区结构不稳定,甚至有些情况会将网络中所有节点归为一个社区,具有较差的鲁棒性. 因此,为减小LPA局限性所带来的影响,国内外相关研究者进行了很多探索. Sun等[12]以邻接节点对当前节点的影响大小作为更新顺序,缓解了传统方法在随机更新节点标签和收敛困难对结果的影响;张鑫等[13]对标签初始化、随机队列和标签传播过程进行了改进,提高了社区发现结果的稳定性;Chen等[14]在节点标签更新的过程中,借助信息熵来衡量节点之间的关系,以此影响节点归属标签的权重,取得了较高的社区划分效果. 尽管上述方法在一定程度上提高了社区发现的性能和稳定性,但是都没有同时考虑节点之间的相似程度和节点对社区的归属程度. 若利用节点之间相似度这个特征,将更为相似的节点聚集起来,同时结合节点对社区的归属度来加强节点被划分到某社区的可信度,则能达到划分的目的. 因此,提出一种基于中心度的标签传播社区发现方法. 该方法通过计算每个节点的中心度,利用中心度来计算每个节点属于某个社区的归属度,按归属度的大小将节点划分到应该所在的社区.

时间序列聚类是时间序列数据挖掘中的重要任务之一. 由于时间序列数据的特殊结构,传统聚类算法不能直接应用于时间序列数据,因此聚类算法成为时间序列数据挖掘中的研究热点. 黄令贺等[15]利用K-means聚类方法对网络百科用户贡献行为时间序列进行聚类,研究用户的兴趣动态变化;Beńitez等[16]设计一种时间序列动态聚类方法对电量模式的动态演化进行研究,可以快速识别用户一天里的电量消费模式和演化模型;Aghabozorgi等[17]针对股价股票时间序列的高维特性,提出利用三阶段聚类模型对股票进行聚类,不仅可以发现公司之间的联动关系,而且能够进行增量式聚类. 上述时间序列聚类方法在聚类分析及应用中得到了不错的效果,但有些方法需要在聚类之前确定初始簇中心以及聚类数目,也没有反映时间序列之间的关系. 社区发现方法将连接紧密的节点聚集成一个簇,与聚类存在着异曲同工

之处. 此外,利用社区发现方法研究时间序列聚类的相关工作较少,将社区发现方法应用到时间序列聚类中,可为时间序列聚类的研究提供一种新的视角.

2 标签传播时间序列聚类

标签传播通过利用节点的连接关系自动传播信息,最终得到社区的划分结果. 时间序列聚类是根据某种聚类规则将时间序列划分为若干个簇,社区发现与时间序列聚类有着相似之处,因此本文利用标签传播自动识别社区结构的优势将其应用到时间序列聚类中. 首先介绍相关概念及定义,在此基础上提出新的社区发现方法,并将新方法应用到时间序列聚类中,同时通过数值实验验证新时间序列聚类方法的有效性和优越性.

2.1 相关概念及定义

本节将给出一些涉及到的定义以及计算公式,其中包括Liu等[18]提出的相关信心度、信心方差的相关概念和计算公式,并采用Sun等[19]提出的部分相关定义以及中心度计算方法来衡量节点的中心度,给定无向带权图$G(V,E,w)$来表示网络. 其中:V表示节点集,E表示边集,w表示边的权重,权重w取1时表示每条边权重一致.

定义1 节点相似度. 给定无向网络$G(V,E,w)$,节点u与节点z的Jaccard相似度为

$$\mathrm{sim}(u,z)=\frac{|\Gamma(u)\bigcap\Gamma(z)|}{|\Gamma(u)\bigcup\Gamma(z)|}. \tag{1}$$

其中$\Gamma(u)=\mathrm{Nb}(u)\bigcup u$,$\mathrm{Nb}(u)$表示节点$u$的邻接点集合.

定义2 度. 节点的度表示与该节点相连的节点个数,在无向图中也可以理解为节点邻接点的个数,描述了该节点在网络中的影响程度,表示如下:

$$k(u)=\sum_{v\in\mathrm{Nb}(u)}w(u,v). \tag{2}$$

其中$w(u,v)$表示节点u与节点v边的权重,在本文中取值为1,此时$k(u)$也可以表示为$k(u)=|\mathrm{Nb}(u)|$.

定义3 局部密度. 针对节点$u\in V$,u的局部密度为

$$\rho(u)=\frac{k(u)}{N-1}, \tag{3}$$

其中N表示网络G中共有N个节点.

定义4 与具有更高局部密度邻接点的相似度. 在邻接点$\mathrm{Nb}(u)$中,局部密度比$\rho(u)$还要大的节点与节点u的相似度为

$$S(u)=\max_{v\in\mathrm{Nb}(u)\wedge\rho(v)>\rho(u)}(\mathrm{sim}(u,v)). \tag{4}$$

当节点u与邻接点相比,局部密度$\rho(u)$为最大时,取$S(u)=\max\limits_{v\in\mathrm{Nb}(u)}(\mathrm{sim}(u,v))$.

定义5 中心度. 该值表明当前节点u对邻接节点的影响,给定一个无向带权图$G(V,E,w)$,针对节点$u\in V$的中心度$C(u)$的计算方法为

$$C(u)=\frac{\rho(u)}{S(u)}. \tag{5}$$

定义6 信心度. 每个邻接点对当前节点有着不同程度的影响,针对节点u和邻接点$v\in\mathrm{Nb}(u)$,信心度$\delta_u(v)$表示节点v对节点u的影响程度,计算方式为

$$\delta(u)=\frac{\mathrm{sim}(u,v)}{\sum\limits_{s\in\mathrm{Nb}(u)}\mathrm{sim}(u,s)}. \tag{6}$$

定义7 信心方差. 为了解决LPA节点标签更新顺序的随机性,根据信心方差从大到小的顺序更新节点. 文献[16]引入信心方差,其计算方法为

$$\sigma(u)=\sqrt{\frac{\sum\limits_{v\in\mathrm{Nb}(u)}\left[\delta_u(v)-\dfrac{1}{|\mathrm{Nb}(u)|}\sum\limits_{v\in\mathrm{Nb}(u)}\delta_u(v)\right]^2}{|\mathrm{Nb}(u)|}}. \tag{7}$$

2.2 基于中心度的标签传播社区发现

LPA随机为每个节点分配唯一的标签,按照初始标签的先后顺序进行迭代更新节点标签. 若邻接节点存在多个数量最多的标签,则随机选择其中一个标签作为更新标签,这种多重随机性导致标签循环震荡,或者最终将所有节点归为一个社区. 为此,提出一种新的归属度计算方法来解决节点随机选择多个数量最多的标签问题. 当更新节点标签时,节点u属于社区c的归属系数为

$$b(u,c)=\sqrt{\frac{\sum\limits_{v\in\mathrm{Nb}(u)}\theta(v,c)C(v)\delta_u(v)}{\sum\limits_{v\in\mathrm{Nb}(u)}C(v)\delta_u(v)}}. \tag{8}$$

其中

$$\theta(v,c)=\begin{cases}1,\ \mathrm{label}(v)=c;\\ 0,\ \mathrm{otherwise},\end{cases}$$

$\mathrm{label}(v)$表示节点v的标签. 由此看出,节点u属于社区c的归属度受到邻接点的信心度以及中心度的影响. 为了解决LPA迭代更新标签过程随机性带来的不稳定,提出基于中心度的标签传播社区发现算法(Label propagation algorithm based on centrality, LPC),简述如下.

算法2 基于中心度的标签传播社区发现算法.

输入:网络$G(V,E,w)$,迭代次数T;

输出:网络中每个节点的标签.

Step 1:计算节点$u(u\in V)$的$C(u)$、$\delta_u(v)(v\in\mathrm{Nb}(u))$和$\sigma(u)$;

Step 2: 将$\delta_u(v)$进行降序排列;

Step 3: 合并节点u与邻接节点$v \in \mathrm{Nb}(u)$的标签,得到节点u的待定标签集合L;

Step 4: 针对待定标签集合L中的每个标签c,利用式(8)计算u的归属度$b(u,c)$;

Step 5: 选择$c_u = \arg\max_{c\in L}(b,(u,c))$作为节点$u$的新标签;

Step 6: 重复T次Step 3 ~ Step5,迭代结束.

当节点u本身具有越高的中心度时,越有可能向它的邻接点传播自身的标签,从而在迭代更新过程中,标签得以来回传播,最终使得标签统一. 在Step 5中,当节点u存在若干个最大归属度时,若随机选择一个标签作为节点u的更新标签,则可以加快社区的划分速度. 由于$C(u)$和$\delta_u(v)$的值相对固定,节点u的待定标签集合L影响着θ的取值,这种划分方式能够较快得到划分的结果. 然而,真实社会网络通常存在重叠社区,不利于数据划分.

若归属度大于某个阈值,则保留对应的标签. 在一定程度上,LPC可以发现重叠社区. 然而,在时间序列聚类过程中,时间序列划分到某个簇中需要得到具体的标签,在一定程度上属于硬划分. 因此,算法LPC只选择使得节点u归属度最大的社区,当出现多个归属度相同的社区时,节点随机选择一个社区的标签作为更新标签.

2.3　基于中心度的标签传播时间序列聚类

基于网络的聚类可以发现任意形态的簇,为了将这种优越性推广到时间序列聚类,将LPC推广为时间序列聚类方法. 由于LPC是通过标签传播将数据进行划分,不仅可以在欧氏空间探索时间序列的内部空间结构,还可以发现个别时间序列的重要特征. 综上所述,提出LPC时间序列聚类方法(Time series clustering based on LPC, TLPC). 将数据集中的时间序列当作网络中的节点,利用给定的相似性度量方法度量两条时间序列之间的距离,符合某种距离要求的两条时间序列之间建立边连接,最终利用LPC进行聚类.

TLPC共分为4个步骤: 1) 标准化数据集. 由于时间序列存在量纲,会导致相似性度量结果的不准确,从而影响时间序列聚类效果. 因此,在度量之前需消除量纲的影响,对时间序列进行标准化操作,使其数据均值为0,方差为1. 2) 时间序列相似性度量. 在构建网络之前,为了确定哪两条时间序列之间有连接关系,需要对数据集中每两条时间序列进行相似性度量,构建距离矩阵. 3) 构建时间序列网络. 将距离矩阵转化成网络,每一个节点代表一条时间序列,每一条边代表两条时间序列的连接关系,说明这两条时间序列满足一定的距离要求. 构建网络的方法对时间序列聚类结果有着非常大的影响,常用的网络构建方法有k-NN和ε-NN. k-NN是指每条时间序列选择与之距离最小的前K条时间序列建立连接关系,ε-NN是指给定一个距离(或相似性)阈值,距离(相似性)小于(大于)ε的时间序列之间建立连接关系. Ferreira等[6]指出,ε-NN构建的时间序列网络的聚类效果优于k-NN的聚类效果,文章采用ε-NN方式构建时间序列网络,且ε指定为距离阈值,并确立一种阈值指定方式. 4) LPC时间序列聚类. 时间序列网络构建完成之后,利用LPC对时间序列数据集进行聚类. 具体方法步骤如下.

算法3　LPC时间序列聚类方法.

输入: 时间序列数据集,阈值ε,迭代次数T;

输出: 预测类标签L.

Step 1: 标准化时间序列数据集;

Step 2: 构建距离矩阵D;

Step 3: 针对每条时间序列,找到与之距离小于ε的时间序列作为邻接点,构建时间序列网络G;

Step 4: 根据基于中心度的标签传播社区发现算法,即LPC(G,T),返回时间序列数据的预测标签L,算法结束.

2.4　时间复杂度

TLPC算法的时间复杂度取决于每一步骤的时间复杂度: 针对n条长度为l的时间序列数据集,标准化数据集的时间复杂度为线性的,即$O(nl)$. 构建距离矩阵受到相似性度量公式的影响,例如,若采用欧氏距离作为度量公式,则该步骤时间复杂度为$O(n^2l)$;若采用动态时间弯曲(Dynamic time warping, DTW)作为度量公式,则为$O(n^2l^2)$. 因为每一条时间序列都需要经过比对其他时间序列才能找到合适的邻接点,所以构建时间序列网络G的过程时间复杂度为$O(n^2l)$. 另外,LPA时间接近线性,即$O(n)$,而LPC也仅在标签传播过程中多了计算归属度步骤,因此利用LPC对网络进行聚类也能近似为线性的,即$O(n)$. 综上所述,TLPC的时间复杂度为$O(n^2l^2)$.

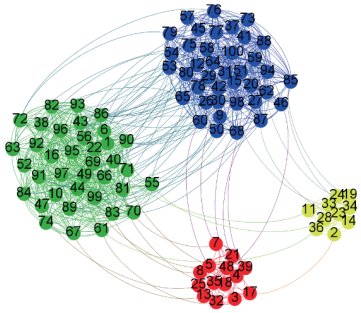
3　数值实验

为了体现LPC在社区发现的社区划分能力以及TPLC在时间序列聚类任务中的性能,将进行两部分的数值实验. 第1部分数值实验利用人工网络数据和真实社会网络数据来检验LPC的社区划分能力;第2部分数值实验利用时间序列数据集来测试TLPC构建网络能力以及时间序列聚类的效果,并与相关的聚类方法进行比较,验证TLPC的聚类性能.

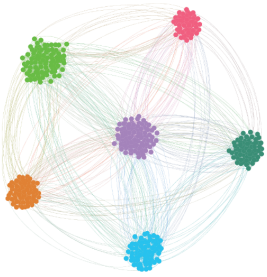
3.1 基于中心度的标签传播数值实验

3.1.1 人工网络数据

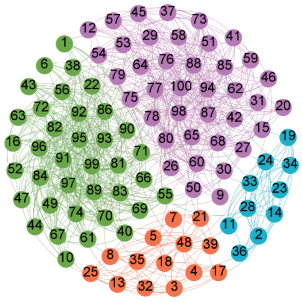
采用Lancichinetti等[20]提出的Lancichinetti-Fortunato-Radicchi基准网络,简称LFR基准网络. n为网络的节点数;k为节点平均度,$\max k$为度的最大值;μ为混合参数,表示节点与外部社区连接的概率,μ越大,社区发现的难度也越大;$\min c$为社区最小规模的节点数,$\max c$为社区最大规模的节点数. 实验中采用了两个人工数据集,分别取$n = 100, k = 15, \max k = 40, \mu = 0.1, \min c = 10, \max c = 40$; $n = 500, k = 15, \max k = 40, \mu = 0.1, \min c = 50, \max c = 200$. 其原始数据和LPC社区划分结果如图1所示.



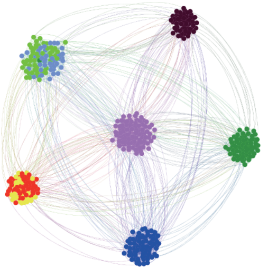
(a) 人工数据集1



(b) 人工数据集2



(c) 数据集1的LPC发现结果



(d) 数据集2的LPC发现结果

图1 LFR基准网络测试数据

图1(a)和图1(b)分别是节点数为100和500的人工网络数据集,图1(c)和图1(d)分别是这两组数据集LPC的社区发现结果,相同颜色的节点为同一个社区. 图1(a)的数据集1共有4个社区,图1(c)中也显示了LPC能够准确地发现对应的4个社区. 图1(b)中原始数据存在6个社区,LPC算法可将原始数据划分为8个社区,其中有两个社区被细分,这种细分可以在一定程度上有利于帮助挖掘社区的内部结构. 例如在股票市场中,相同行业的股票一般会呈现相似的波动形态,但由于某些公司可能存在着相同股东,或者有一些业务往来,这些公司的股票呈现的形态则更为相似.

3.1.2 真实网络数据

真实网络数据使用Karate[21]数据集,该数据集描述了美国一家空手道俱乐部成员之间的交往关系. 该网络由34个节点、78条边组成,节点和边分别代表该俱乐部的成员及其之间在俱乐部之外的个人交往关系. 同时,该俱乐部的经理与教练之间的分歧,导致该俱乐部划分为两个团体,分别表示支持经理和支持教练.

图2给出了Karate数据集原始网络以及LPC算法对该网络的划分情况. 相应地,节点1表示教练,节点34表示经理. 阴影区域所包含的节点表示LPC算法划分的社区,节点的颜色表示原始数据的社区. 容易看出,LPC能够正确地将这两个社区成功划分. 另外,节点可以通过邻接点的中心度来计算社区的归属度,邻接点的中心度越大,当前点越有可能接受邻接点的标签. 例如节点1和节点34的中心度很大,则越能够将其标签传播给与之相连的节点.

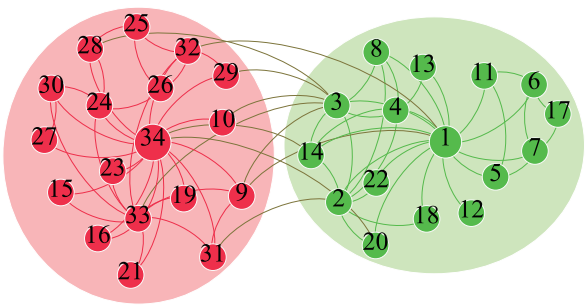


图2 Karate原始数据及LPC划分情况

3.2 基于中心度标签传播时间聚类数值实验

为了检验TLPC的有效性,随机选取Keogh教授提供的16个时间序列数据集[22]进行聚类分析. TLPC采用两种度量方式,即欧氏距离(Euclidean

distance, ED) 和 DTW[23], 简称 TLPC-ED 和 TLPC-DTW, 并与 K-means、谱聚类(Spectral clustering, SC)、改进的层次聚类(HC-DD_{DTW})[24] 以及 LPA 进行对比实验. 实验采用 Rand Index 和 Precise[25] 两种指标进行检验, 进一步验证 TLPC 的优越性. 数据集信息如表 1 所示.

表 1　时间序列数据集信息

ID	名称	类别数	训练集数	长度
1	Arrow Head	3	36	176
2	Beef	5	30	470
3	BeetleFly	2	20	512
4	CBF	3	30	128
5	Coffee	2	28	286
6	ECG200	2	100	96
7	ECG Five Days	2	23	136
8	Gun Point	2	20	150
9	Italy Power Demand	2	24	24
10	Mote Strain	2	20	84
11	Olive Oil	4	30	570
12	Sony AIBO Robot Surface	2	27	70
13	Sony AIBO Robot Surface II	2	20	65
14	ToeSegmentation1	2	40	277
15	ToeSegmentation2	2	36	343
16	Two Lead ECG	2	23	82

Rand Index (RI):

$$\mathrm{RI} = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{FPTN} + \mathrm{FN}}. \tag{9}$$

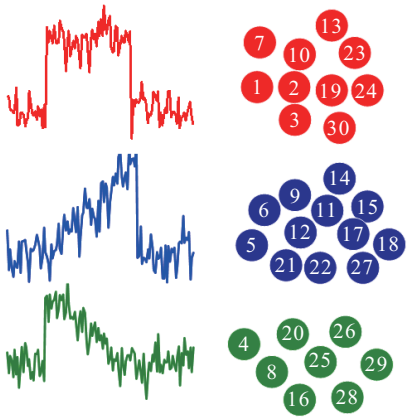
TP 指的是真正例, 表示相同类别的时间序列被正确分到同一个簇中的个数; TN 指的是真假例, 表示不同类别的时间序列被正确分到不同簇中的个数; FP 指的是假正例, 表示相同类别的时间序列被错误地划分到不同簇中的个数; FN 指的是假负例, 表示不同类别的时间序列被错误地划分到同一个簇中的个数.

Precise (P):

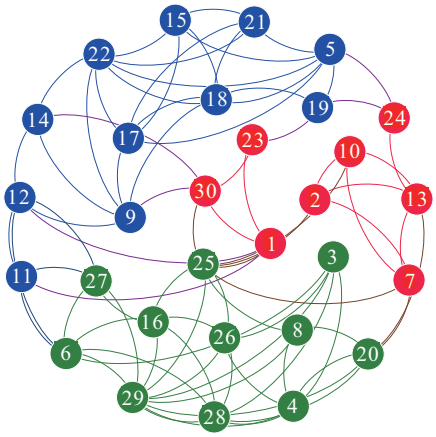
$$P = \sum_{j-1}^{c} \frac{|C_j|}{n} \times \max_{i=1,2,\cdots,k} p(H_i, C_j). \tag{10}$$

其中: $p(H_i, C_j) = \dfrac{|H_i \bigcap C_j|}{|C_j|}$, c 表示簇的个数, k 表示原始类标签, n 表示时间序列的总个数, C_j 表示第 j 个簇, H_i 表示簇中原始标签为 i 的时间序列集, $|\cdot|$ 表示统计时间序列的数量.

文献 [4] 指出, ε 的大小影响着网络的构建, 最终影响聚类效果. 若 ε 太大, 则最终将所有时间序列划分为一个簇中; 反之, 将时间序列分散到很多个簇中. 为了得到合理的聚类效果, 本文针对时间序列 S, 选取距离矩阵中对应的前 K 个最小距离值的均值作为 S 的 ε, 与 S 距离小于 ε 的时间序列将作为 S 的邻接点. 由于构建网络的影响会使得 TLPC 的聚类结果不一致, 为了方便比较聚类结果, 在 TLPC-ED 和 TLPC-DTW 的实验中, 指定 K 取值范围为 $[3, n-1]$. 若给定 K 产生的簇与原始类标签数目一致, 则将聚类结果与其他方法进行比较. 值得说明的是, 对比方法 HC-DD_{DTW} 需要对相似性度量方法给定参数, 实验中参数设定为 0.1. 图 3 所示的是 CBF 原始数据集详情以及利用 TLPC-DTW 对 CBF 数据集进行聚类的效果.



(a) CBF 数据集



(b) CBF 的 TLPC-DTW 划分结果

图 3　CBF 数据集 TLPC-DTW 聚类效果

从图 3(a) 可以看出, 左方为 CBF 数据集 3 种形态的时间序列, 对应右方的节点集为该类别的时间序列子集, 每条时间序列用其在数据集中的 ID 表示. 从图 3(b) 可以发现, 算法可将时间序列划分为 3 个簇, 且具有较好的聚类效果, 相同颜色的节点被划分为同一个簇. 具体分析图 3(b) 的聚类结果, 节点 3、6 和 27 被划分到了错误的簇中, 这 3 个节点与该簇中的节点连接较为紧密, 如节点 3 与中心度较大的节点 4、26、28、29 连接, 因此该簇中心度较大的节点能够将自身标签传播给这 3 个点, 使之聚为一类. 从效果上看, TLPC-DTW 不仅有较好的时间序列聚类效果, 且能够通过网络形式反映时间序列之间的空间网络布局情况. 例如某些节点只与同一个簇内的节点连接, 有些节点也会与其他簇的节点相连, 这样将导致其有

可能接受其他簇节点的标签,但最终标签取决于邻接点的中心度.

另外,考虑使用不同距离度量方法,将TPLC-ED和TLPC-DTW的聚类结果与其他4种聚类方法的结果进行比较. 其中,利用LPA作为聚类方法时,根据欧氏距离来构建网络,构建网络的方式与TLPC-ED和TLPC-DTW的相同. 同时,TLPC和LPA方法下不同K值将产生不同聚类结果,故取所有聚类结果下的RI和P的均值作为TLPC方法下的聚类结果,具体数值结果见表2和表3.

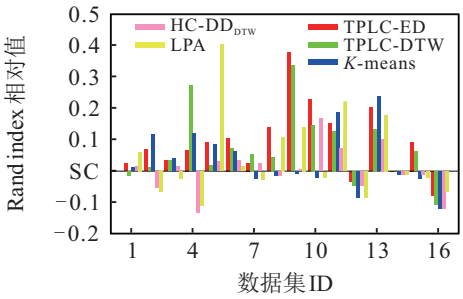
表2 不同聚类方法的Rand Index结果

ID	数据集	TLPC-ED	TLPC-DTW	K-means	SC	HC-DD$_{\mathrm{DTW}}$	LPA
1	Arrow Head	0.665 2	0.626 0	0.652 4	0.639 7	0.654 0	0.701 6
2	Beef	0.726 1	0.667 9	0.774 7	0.657 5	0.604 6	0.565 5
3	BeetleFly	0.513 2	0.513 8	0.521 1	0.478 9	0.494 7	0.473 7
4	CBF	0.582 3	0.788 0	0.634 5	0.514 9	0.381 6	0.503 4
5	Coffee	0.584 4	0.509 9	0.576 7	0.492 1	0.523 8	0.928 6
6	ECG200	0.674 0	0.640 2	0.631 5	0.567 9	0.601 8	0.528 7
7	ECG Five Days	0.528 1	0.556 5	0.478 3	0.502 0	0.525 7	0.478 3
8	Gun Point	0.650 2	0.555 0	0.497 1	0.510 2	0.497 1	0.622 0
9	Italy Power Demand	0.884 6	0.842 3	0.498 0	0.506 1	0.511 5	0.687 0
10	Mote Strain	0.723 1	0.642 1	0.473 7	0.494 7	0.663 2	0.473 7
11	Olive Oil	0.650 4	0.627 4	0.685 1	0.498 9	0.570 1	0.719 5
12	Sony AIBO Robot Surface	0.571 1	0.557 9	0.521 1	0.605 3	0.557 9	0.521 1
13	Sony AIBO Robot Surface II	0.702 8	0.633 0	0.737 9	0.498 6	0.601 1	0.686 6
14	ToeSegmentation1	0.496 9	0.500 3	0.487 2	0.498 7	0.488 5	0.487 2
15	ToeSegmentation2	0.604 3	0.574 1	0.487 3	0.511 1	0.500 0	0.485 7
16	Two Lead ECG	0.517 0	0.490 2	0.478 3	0.596 8	0.478 3	0.525 7
	MEAN	**0.629 6**	**0.607 8**	0.570 9	0.535 8	0.540 9	0.586 7

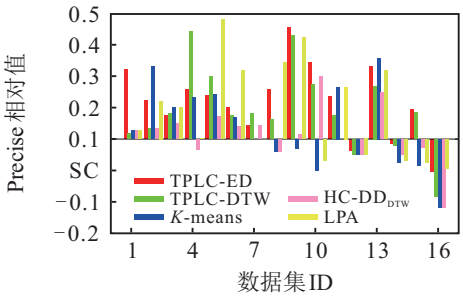
表3 不同聚类方法的Precise结果

ID	数据集	TLPC-ED	TLPC-DTW	K-means	SC	HC-DD$_{\mathrm{DTW}}$	LPA
1	Arrow Head	0.834 8	0.627 8	0.638 9	0.611 1	0.654 0	0.639 1
2	Beef	0.523 1	0.433 3	0.633 3	0.400 0	0.433 3	0.508 3
3	BeetleFly	0.625 0	0.631 2	0.650 0	0.550 0	0.600 0	0.649 8
4	CBF	0.625 6	0.810 3	0.600 0	0.466 7	0.433 3	0.466 7
5	Coffee	0.711 3	0.770 8	0.714 3	0.571 4	0.642 9	0.964 3
6	ECG200	0.790 6	0.764 6	0.760 0	0.690 0	0.730 0	0.810 0
7	ECG Five Days	0.652 2	0.691 3	0.608 7	0.608 7	0.652 2	0.608 7
8	Gun Point	0.759 1	0.663 5	0.560 0	0.600 0	0.560 0	0.820 0
9	Italy Power Demand	0.937 9	0.913 2	0.552 2	0.582 1	0.597 0	0.910 4
10	Mote Strain	0.843 8	0.775 0	0.500 0	0.600 0	0.800 0	0.512 2
11	Olive Oil	0.635 9	0.574 4	0.666 7	0.500 0	0.500 0	0.666 7
12	Sony AIBO Robot Surface	0.712 5	0.700 0	0.700 0	0.750 0	0.700 0	0.700 0
13	Sony AIBO Robot Surface II	0.824 1	0.762 3	0.851 9	0.592 6	0.740 7	0.814 8
14	ToeSegmentation1	0.561 1	0.555 6	0.500 0	0.575 0	0.525 0	0.501 8
15	ToeSegmentation2	0.706 6	0.696 2	0.527 8	0.611 1	0.583 3	0.532 7
16	Two Lead ECG	0.634 8	0.556 5	0.521 7	0.739 1	0.521 7	0.652 2
	MEAN	**0.716 9**	**0.686 3**	0.622 4	0.593 2	0.603 9	0.672 3

从表2和表3可以看出,TLPC的聚类结果总体上在RI指标和P指标上要好于其他3种比较方法. 然而,基于DTW的TLPC整体聚类效果稍逊于基于ED的TLPC聚类效果,其受到构建网络过程中数据集大小和参数ε计算方法等影响. 为了便于比较方法的优越性,以SC的聚类结果为基准线,优于SC的结果在y轴上方以正值呈现;反之,则在y轴下方以负值呈现,如图4所示.



(a) 数据集在不同方法下的RI值比较



(b) 数据集在不同方法下的P相对值比较

图 4　不同方法RI与P相对值比较

从图4可以看出,本文提出的两种方法的聚类比较结果大多分布均在y轴上方,表示新方法要优于谱聚类分析方法. 针对多个数据集条形图的值,本文方法的效果也相对优于其他方法,充分说明本文方法的优越性. 然而,新方法在ID为12、14、16的数据集出现的聚类精度略低于其他方法,其原因是由于相似性度量方法在此类数据中未能表现出较好的度量质量. 对于Two Lead ECG数据集两个类别的时间序列形态特征,从数据集的特点来看,这两种不同类别时间序列之间的数据或形态差异并不十分明显,不管是欧氏距离在时间点上数据匹配或者DTW从形态上的匹配,可能均较难从数值或者形态上区分这两种类别,使得相似性度量效果不佳,最终影响聚类效果. 因此,相似性度量质量的提升能够提高新方法的聚类质量.

3.3　时间效率

时间序列聚类的性能不仅体现在聚类效果上,而且也包含了算法的整体时间消耗. 为了进一步体现TPLC的性能,将比较上述所有方法的时间消耗. 以TLPC-DTW时间消耗为基准,时间消耗少于TLPC-DTW的结果在y轴上方以正值显示,反之在y轴下方以负值显示. 相对值的计算方法利用下式得到:

$$T_{\text{Relative}} = \frac{T_{\text{method}} - T_{\text{TLPC-DTW}}}{1\,000}. \tag{11}$$

其中: T_{method}为对比方法的时间消耗, $T_{\text{TLPC-DTW}}$为TLPC-DTW的时间消耗, T_{Relative}为时间消耗相对值. 整体的对比效果如图5所示.

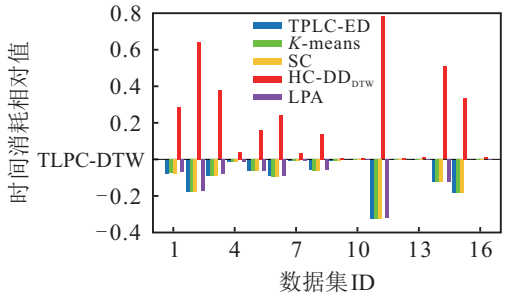


图 5　时间消耗比较分析

由图5可知,TLPC-ED、K-means、SC和LPA时间消耗的条形图均在y轴下方,这是由于TLPC-ED、K-means和SC在计算时间序列相似性时使用的是计算效率高的欧氏距离,LPA所消耗的时间是由欧氏距离构建网络的时间与标签传播的时间组成,而TPLC-DTW和HC-DD$_{\text{DTW}}$均利用DTW度量每两条时间序列之间的相似性,会导致整体时间消耗随着时间序列长度的增长而增加,因此方法TLPC-ED、K-means、SC和LPA整体时间效率也优于TLPC-DTW和HC-DD$_{\text{DTW}}$. HC-DD$_{\text{DTW}}$的条形图在所有数据集上都是呈现正值,表明所有数据集的时间消耗均大于TPLC-DTW. 由于HC-DD$_{\text{DTW}}$不仅需要在度量时间序列相似性之前得到时间序列的一阶导数序列,还要计算原始时间序列的DTW值和一阶导数时间序列的DTW值,再附加在聚类过程对时间序列进行两两比较,因此HC-DD$_{\text{DTW}}$的时间消耗会比TLPC-DTW的高,成为了这几种方法中时间消耗最多的方法. 综上所述,与传统方法相比,TLPC-ED和TLPC-DTW的聚类质量具有较高的优势,并且TLPC-ED的时间效率也较高,充分说明了新方法的有效性和优越性.

4　结　论

针对传统时间序列聚类方法需要确定初始簇中心的局限性,本文以社区发现方法为新的研究视角,提出一种基于中心度的标签传播时间序列聚类方法. 通过实验验证,与传统时间序列聚类方法相比,新聚类方法具有以下优势: 1) 新聚类方法可以构建时间序列网络,通过研究时间序列之间的连接关系发现时间序列在欧氏空间的结构特征; 2) 新聚类方法可以根据不同的距离阈值,将时间序列的空间结构进行合理地划分,找到适合于簇划分的形式; 3) 与传统聚类方法相比,新聚类方法能够得到更好的聚类效果,具有一定的优越性. 然而,由于相似性度量距离、构建网络的阈值以及时间序列数据的特征均可能影响聚类效果,如何提高相似性度量的效果、自动选择最优参数阈值构建时间序列网络和减小时间序列数据特征的影响,都是今后工作中值得关注的研究方向.

参考文献(References)

[1] Esling P, Agon C. Time-series data mining[J]. ACM Computing Surveys, 2012, 45(1): 1-12.

[2] 李海林, 梁叶. 基于数值符号和形态特征的时间序列相似性度量方法[J]. 控制与决策, 2017, 32(3): 451-458.
(Li H L, Liang Y. Similarity measure based on numerical symbolic and shape feature for time series[J]. Control and Decision, 2017, 32(3): 451-458.)

[3] 刘建国, 任卓明, 郭强, 等. 复杂网络中节点重要性排序的研究进展[J]. 物理学报, 2013, 62(17): 9-18.
(Liu J G, Ren Z M, Guo Q, et al. Node importance ranking of complex networks[J]. Acta Physica Sinica, 2013, 62(17): 9-18.)

[4] Pagani G A, Aiello M. The power grid as a complex network: A survey[J]. Physica A: Statistical Mechanics and Its Applications, 2013, 392(11): 2688-2700.

[5] Hao X, An H, Qi H, et al. Evolution of the exergy flow network embodied in the global fossil energy trade: Based on complex network[J]. Applied Energy, 2016, 162: 1515-1522.

[6] Ferreira L N, Zhao L. Time series clustering via community detection in networks[J]. Information Sciences, 2016, 326: 227-242.

[7] 周婷婷, 金宁德, 高忠科, 等. 基于有限穿越可视图的时间序列网络模型[J]. 物理学报, 2012, 61(3): 86-96.
(Zhou T T, Jin N D, Gao Z K, et al. Limited penetrable visibility graph for establishing complex network from time series[J]. Acta Physica Sinica, 2012, 61(3): 86-96.)

[8] Wang M, Tian L. From time series to complex networks: The phase space coarse graining[J]. Physica A: Statistical Mechanics and Its Applications, 2016, 461: 456-468.

[9] 朱牧, 孟凡荣, 周勇. 基于链接密度聚类的重叠社区发现算法[J]. 计算机研究与发展, 2013, 50(12): 2520-2530.
(Zhu M, Meng F R, Zhou Y. Density-based link clustering algorithm for overlapping community detection[J]. J of Computer Research and Development, 2013, 50(12): 2520-2530.)

[10] Raghavan U N, Albert R, Kumara S. Near linear time algorithm to detect community structures in large-scale networks[J]. Physical Review E, 2007, 76(3): 036106.

[11] 赵卓翔, 王轶彤, 田家堂, 等. 社会网络中基于标签传播的社区发现新算法[J]. 计算机研究与发展, 2011 48(S3): 8-15.
(Zhao Z X, Wang Y T, Tian J T, et al. A novel algorithm for community discovery in social networks based on label propagation[J]. J of Computer Research and Development, 2011, 48(S3): 8-15.)

[12] Sun H, Huang J, Zhong X, et al. Label propagation with α-degree neighborhood impact for network community detection[J]. Computational Intelligence and Neuroscience, 2014: 130689(9).

[13] 张鑫, 刘秉权, 王晓龙. 稳定标签传播的社区发现方法[J]. 哈尔滨工业大学学报, 2016, 48(11): 47-52.
(Zhang X, Liu B Q, Wang X L. Community discovery method based on stable label propagation[J]. J of Harbin Institute of Technology, 2016, 48(11): 47-52.)

[14] Chen N, Liu Y, Chen H, et al. Detecting communities in social networks using label propagation with information entropy[J]. Physica A: Statistical Mechanics and Its Applications, 2017, 471: 788-798.

[15] 黄令贺, 朱庆华, 沈超. 差异与稳定: 网络百科用户兴趣动态变化研究[J]. 图书情报知识, 2016(2): 101-113.
(Huang L H, Zhu Q H, Shen C. Variance and stability:dynamic change of user's interests in online encyclopedia[J]. Documentation, Information and Knowledge, 2016(2): 101-113.)

[16] Benítez I, Díez J L, Quijano A, et al. Dynamic clustering of residential electricity consumption time series data based on Hausdorff distance[J]. Electric Power Systems Research, 2016, 140: 517-526.

[17] Aghabozorgi S, Ying W T. Stock market co-movement assessment using a three-phase clustering method[J]. Expert Systems with Applications, 2014, 41(4): 1301-1314.

[18] Liu K, Huang J, Sun H, et al. Label propagation based evolutionary clustering for detecting overlapping and non-overlapping communities in dynamic networks[J]. Knowledge-Based Systems, 2015, 89: 487-496.

[19] Sun H, Liu J, Huang J, et al. CenLP: A centrality-based label propagation algorithm for community detection in networks[J]. Physica A: Statistical Mechanics and Its Applications, 2015, 436: 767-780.

[20] Lancichinetti A, Fortunato S, Radicchi F. Benchmark graphs for testing community detection algorithms[J]. Physical Review E, 2008, 78(4): 046110.

[21] Zachary W W. An information flow model for conflict and fission in small groups[J]. J of Anthropological Research, 1977, 33(4): 452-473.

[22] Chen Y P, Keogh E, Hu B, et al. The UCR time series classification archive[EB/OL]. (2015-07-01)[2015-12-01]. http://www.cs.ucr.edu/∼eamonn_series_data/.

[23] Sakoe H, Chiba S. Dynamic programming algorithm optimization for spoken word recognition[J]. IEEE Trans on Acoustics, Speech, and Signal Processing, 1978, 26(1): 43-49.

[24] Luczak M. Hierarchical clustering of time series data with parametric derivative dynamic time warping[J]. Expert Systems with Applications, 2016, 62: 116-130.

[25] Izakian H, Pedrycz W, Jamal I. Fuzzy clustering of time series data using dynamic time warping distance[J]. Engineering Applications of Artificial Intelligence, 2015, 39: 235-244.

(责任编辑：齐 霁)