一种基于 Shapelets 的懒惰式时间序列分类算法

王志海 张 伟 原继东 刘海洋
(北京交通大学计算机与信息技术学院 北京 100044)

摘 要 近些年,时间序列分类问题研究受到了越来越多的关注.基于 shapelets 的时间序列分类技术是一种有效的方法.然而,其在提取最优 shapelet 的过程中要建立包含大量冗余元素的候选 shapelets 集合,一般所获得的 shapelets 只在平均意义上具有某种鉴别性;与此同时,普通模型往往忽略了待分类实例所具有的局部特征.为此,我们提出了一种依据待分类实例显著局部特征的懒惰式分类模型.这种模型为每个待分类实例构建各自的数据驱动的懒惰式 shapelets 分类模型,从而逐步缩小了与其分类相关的时间序列搜索空间,使得所获得的 shapelets 能够直接反映待分类实例的显著局部特征.实验结果表明该文提出的模型具有较高的准确率和更强的可解释性.

关键词 时间序列;懒惰式学习;分类;shapelets;可解释性
中图法分类号 TP311 **DOI 号** 10.11897/SP.J.1016.2019.00029

A Novel Lazy Time Series Classification Algorithm Based on the Shapelets

WANG Zhi-Hai ZHANG Wei YUAN Ji-Dong LIU Hai-Yang
(*School of Computer and Information Technology, Beijing Jiaotong University, Beijing* 100044)

Abstract In order to discover the characteristics of data and explain the prediction process of classification model, the study of interpretable model has become increasingly prevalent in recent years. In reality, we can get massive time series data in many fields, such as weather forecast, medical monitoring, and anomaly detection. Time series classification is an important research field of time series data mining. Time series is different from the traditional attribute vector data, and it has no explicit attributes. Even with the sophisticated feature selection techniques, the dimensionality of potential feature space is still beyond the acceptable range. This poses a challenge to learn an accurate classification model with strong interpretability. Since shapelet is a new primitive that can be used to construct interpretable model, time series classification based on shapelet has recently attracted considerable interest. Shapelet-based classification algorithm is a typical shape-based algorithm. Shapelet can help us give a high sight on the local discriminative features of time series. According to the usage of shapelet, the shapelet-based models can be divided into two categories. One type method establishes a much smaller yet more discriminative feature set through the top-k shapelets to transform the origin dataset. Furthermore, traditional classification algorithms can be applied on the converted low-dimensional dataset. The other employs selected shapelets to build the classification model directly. However, these global shapelet-based models have some obvious shortcomings. First, the global model always needs to create a candidate shapelet set which contains a large number of redundant elements in the process of extracting the

收稿日期:2017-08-02;在线出版日期:2018-07-19.本课题得到国家自然科学基金(61672086,61702030,61771058)、中国博士后科学基金(2018M631328)、中央高校基本科研业务费专项资金(2017YJS036)和北京市自然科学基金(4182052)资助.**王志海**,男,1963 年生,博士,教授,博士生导师,中国计算机学会(CCF)会员,主要研究领域为数据挖掘和机器学习.E-mail:zhhwang@bjtu.edu.cn.**张 伟**,男,1987 年生,博士研究生,主要研究方向为数据挖掘和机器学习.**原继东**,男,1989 年生,博士,讲师,主要研究方向为数据挖掘和模式识别.**刘海洋**,男,1987 年生,博士研究生,主要研究方向为多标记分类、时间序列分类.

best shapelet. Due to the impact of redundant instances and intra-class variation, the extracted shapelets are merely good for the training instances in the average sense. The established shapelet-based model may not be suitable and efficient for the test cases. Second, the shapelets obtained may be from different instances or approximate solutions, which cannot indicate the local characteristics of the test case exactly. Third, since the class value of the local features from the test case is unknown, the characteristics of test cases are always ignored. In order to solve the above problems, a data driven local model based on shapelets for each test case is proposed. In our model, instead of global similarity, local similarity is considered as the basis for classification. The local features of the test case are evaluated directly to find the most discriminative shapelet. And then the shapelet is used to reduce the searching space of class attribute value progressively. Since the shapelets are from the test example, they directly reflect the salient local features of the test case and can answer the question why the model assigns a certain class value to the instance. Meanwhile, in the shapelet evaluation progress, instances are selected to reduce the impact of redundant instances and intra-class variation. The lazy classification model presented in this paper is compared with two shapelet decision tree models, 1NN models based on different distance functions, and C4.5 models based on different top-k shapelets transformation algorithms. Experimental results show that the proposed model has higher accuracy and stronger interpretability.

Keywords time series; lazy learning; classification; shapelets; interpretability

1 引 言

近些年,时间序列分类问题受到极大关注.时间序列数据来源广泛,天气预测[1]、恶意软件检测[2]、电压稳定评估[3]、医疗监护[4]、网络异常检测[5]等许多问题领域产生海量时间序列数据.一般,时间序列是一组有序的实值数据,其通常是按照某一采样率对某一过程在有序等距的时间节点上观测获得的.时间序列数据的获得方式决定了时间序列不同于传统的属性向量数据,时间序列没有明确的属性,即使基于成熟的属性选择技术,潜在属性的维度都非常高,这给时间序列分类问题提出了挑战[6].

虽然不同学者提出了很多时间序列分类算法,但大量实验表明传统的 1NN 分类器仍然在许多问题领域具有相对较好的性能表现[7-8].但是 1NN 分类器存在明显的缺点,他需要存储和搜索整个数据集合,同时建立的分类器可解释性不足,1NN 模型并不能指出同类实例之间有哪些相似之处,不同类实例之间有哪些差异.实际中,除了分类准确率,不同类实例所具有的局部特征也是我们关注的重点,这些特征有助于我们深入理解数据本身和提高分类模型的可解释性.

基于 shapelets 的时间序列分类方法是一种典型的基于形状的分类算法,近些年受到极大关注[9-10].Shapelets 是由 Ye 和 Keogh 提出的一种新概念,它是时间序列中可用来决定类属性归属的特殊子序列[11].基于 shapelets 的时间序列分类算法大致可以分为两类.一类方法基于提取出的 shapelets 对数据集进行转换,然后在转换后的数据集上建立分类模型[12-17].这类方法的优点是可利用多种分类模型对转换后的数据集进行处理.Lines 等人最早利用 top-k shapelets 对时间序列数据集进行转换,并在提取 shapelets 过程中使用 F-statistic 代替信息增益度量 shapelets 的鉴别性[12].Hills 等人进一步对秩和检验、F-statistic 方差分析和 Mood 中位数三种不同的 shapelets 鉴别性评价方式进行了分析,并通过对 shapelets 进行聚类加强了转换后数据的可解释性[13].针对转换算法中提取出的 top-k shapelets 存在较大相似性的问题,为了提高用于时间序列数据转换的 shapelets 的质量,Yuan 等人将 shapelets 剪枝技术和 shapelets 覆盖方法应用到时间序列 shapelets 提取过程[14-15].针对转换过程忽略 shapelets 之间逻辑关系的问题,Yuan 等人提出了逻辑 shapelets 转换方法[16].上述模型的缺点是算法的可解释性较差.

另一类方法直接利用提取出的 shapelets 建立分类模型[11,18-21].例如,Ye 等人提出在训练集合上

递归发现当前最优的 shapelet 建立决策树模型[11]. 针对单个 shapelet 可解释性不足的问题，Mueen 等人提出了具有更强可解释性的逻辑 shapelets 的概念[18]. 针对提取最优 shapelet 过程中需要建立庞大的候选 shapelets 集合和计算量大的问题，Rakthanmanon 等人提出一种基于符号聚合近似(Symbolic Aggregate approximation，SAX)离散化表示的快速 shapelets 发现算法[19]；Grabocka 等人提出了一种启发式的梯度下降 shapelets 搜索算法，并用该方法分类提取 shapelets 建立预测模型[20]，上述两种方法降低了 shapelets 发现过程的时间复杂度，但得到的 shapelets 都是近似解，和 Brute-Force 算法找到的 shapelets 可能不同.

虽然上述两类方法都能在不同程度上提高基于 shapelets 的时间序列分类算法中 shapelets 的发现效率和分类效果，但是这些方法一般都是在数据集合上建立全局模型，这些全局模型存在如下缺点：

(1) 全局模型都是基于整个训练集合对 shapelets 进行评价和选择，往往忽略待分类实例所蕴含的信息，对每个待分类实例所具有的特征的研究没有得到应有的重视.

(2) 全局模型获得的 shapelets 来自不同类实例，并不能直接用来解释每个待分类实例是由于具有哪些局部特征而被归属于某一类.

(3) 由于受到冗余实例、实例类内变异等的影响，提取出的 shapelets 对于数据集合中的实例只在平均意义上是最优的，并不能够准确反映待分类实例所具有的局部特征，用它们建立的 shapelets 模型对于待分类实例并非是合适高效的.

针对建立在整个训练集合上的全局 shapelets 模型存在的问题，本文将懒惰式学习策略和 shapelets 特征提取相结合，为每个待分类实例建立一种新的数据驱动的 shapelets 分类模型. 不同于建立在整个集合上的全局 shapelets 模型，在对待分类实例分类过程中，我们不再遍历训练集合寻找最有鉴别性特征，而是每次只从待分类实例的所有的局部特征中搜索一个最优的特征来逐步降低其类属性归属的不确定性，从而渐进式的消除不确定. 图 1 中给出了来自 UCR 时间序列知识库①的二分类数据集 ItalyPowerDemand 中的三条时间序列及其 shapelets 的示意图，其中 T_j 表示第 j 个实例，T_1 和 T_2 是不同类实例，T_3 和 T_2 是同类实例，S_i^j 表示第 j 个实例的第 i 个 shapelet.

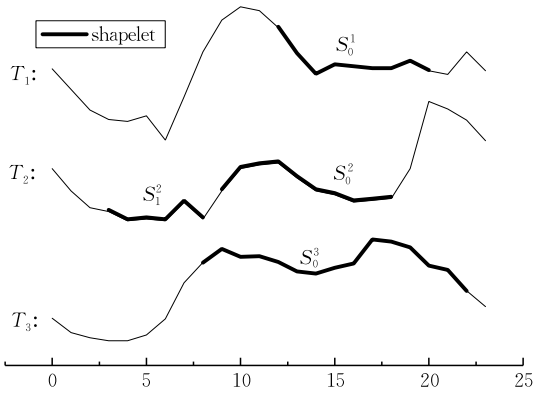


图 1　三条时间序列及其 shapelets

从图 1 可看到不仅不同类实例间有差异，同类实例间的局部特征也有明显不同. 此外，从图 1 中可看出不同实例类属性的预测过程需要的局部特征特点和个数都可能是不同的. T_1 需要一个特征就可以确定类属性归属，即我们可以利用实例 T_1 的一个局部特征 S_0^1 将它的近邻集合中的不同类实例排除，使得近邻集合中只剩下包含该局部特征的一类实例，我们将该类属性作为 T_1 的预测值. 这是我们模型的基本思想，不断利用待分类实例的局部特征逐步降低其类属性的不确定性. T_2 通过两个局部特征才将其类属性的不确定性降为 0，而和 T_2 同类的实例 T_3 预测类属性只需要一个局部特征.

和全局模型相比，本文提出的模型具有更强的针对性和可解释性. 本文的主要贡献概括如下：

(1) 不同于基于全局相似性的 1NN 模型，本文基于局部相似性进行分类预测. 值得注意的是，本文模型对每个待分类实例的预测结果不一定和该实例最近邻的实例的类属性相同，这正是在一些数据集上本文模型的性能显著优于 1NN 模型的主要原因.

(2) 本文使用训练集中的部分不同类实例对候选 shapelets 的鉴别性进行评价，较小的训练集合不仅可以排除同类差异较大实例对鉴别性评价的干扰，还可以充分减少模型的计算量和提高模型的分类准确率.

(3) 针对 Brute-Force 算法建立候选 shapelets 集合过程中产生大量冗余 shapelets 的问题，我们提出只从当前待分类实例上提取候选 shapelets 的策略. 这种策略可以保证提取出的 shapelets 能够准确

① Chen Y P, Keogh E, et al. The UCR time series classification archive. http://www.cs.ucr.edu/~eamonn/time_series_data/, 2015, 10, 8

地反映待分类实例的局部特征.

(4)我们将本文提出的懒惰式分类模型和两种 shapelets 决策树模型、基于不同距离函数的 1NN 模型以及基于不同 top-k shapelets 转换算法的 C4.5 模型进行了比较,本文的算法具有较高的准确率和可解释性.

本文第 2 节介绍相关概念和基本理论;第 3 节介绍本文的模型和算法设计;第 4 节为本文所提出的模型进行了实验设计,并和相关算法进行了比较,最后给出了算法的详细分析;第 5 节给出本文的结论.

2 相关概念与基本理论

Ye 和 Keogh 最初将 shapelet 的提取过程嵌入到了模型的建立过程中[11]. Lines 等人将二者分离,并利用提取出的 shapelets 对数据集进行转换[12]. 由于基于 shapelets 可以建立具有可解释的分类模型,时间序列 shapelet 概念一经提出,就受到了广泛关注. 下面我们介绍基于 shapelets 的时间序列分类模型中的一些重要概念.

2.1 时间序列

时间序列的分类问题和传统的分类问题相同,都是希望找到一个函数将时间序列实例映射到一个类属性值. 下面我们首先介绍文中和时间序列分类相关的一些定义.

定义 1. 时间序列.

时间序列是由 m 个有序的实际观测值 $t_1,t_2,\cdots,t_m$ 组成的实值序列 $T=\{t_1,t_2,\cdots,t_m\},t_i\in R$.

定义 2. 时间序列的子序列.

设 $T=\{t_1,t_2,\cdots,t_m\}$ 为一条完整的时间序列,由序列 T 中连续的 l 个值组成的序列 $S=\{t_i,t_{i+1},\cdots,t_{i+l-1}\}$ 称作时间序列 T 的子序列,其中 $1\leq i\leq m-l+1$.

任意长度为 m 的时间序列包括 $m-l+1$ 个长度为 l 的子序列. 时间序列 T_i 所有长度为 l 的子序列集合为 $W_{i,l}$,时间序列数据集合 D 的所有长度为 l 的子序列集合为 $W_l=W_{1,l}\cup W_{2,l}\cup\cdots\cup W_{n,l}$.

定义 3. 时间序列 shapelet.

一条 shapelet 是由一个子序列 S 和一个阈值 δ 组成的元组(S,δ).

数据集合 D 的所有候选 shapelets 的集合可以表示为 $W=W_{min}\cup W_{min+1}\cup\cdots\cup W_{max}$,其中 min 表示候选 shapelets 子序列的最小长度,max 表示候选 shapelets 的最大长度. 长度为 l 的候选 shapelets 集合 W_l 中有 $O(m^2)$ 个候选 shapelets,整个数据集合上的候选 shapelets 集合 W 中有 $O(nm^2)$ 个候选 shapelets[11].

基于 shapelets 的分类模型的本质是通过局部特征学习来区分不同类属性. 我们将 shapelet 区分不同类属性的性质称作 shapelet 的鉴别性. 例如,图 1 中的实例 T_1 的 shapelet 可以将他对应的近邻集合中的不包含该局部特征的实例排除.

基于 shapelets 的分类模型中都需要根据 shapelets 的鉴别性强弱对 shapelets 进行选择. 这就需要对 shapelets 的鉴别性进行评价,常用的 shapelets 评价方法有信息增益、秩和检验、F-statistic 方差分析以及 Mood 中位数[13]. 由于我们的模型需要在每个节点根据一个分裂点阈值对实例进行选择,而以上四种 shapelets 的鉴别性评价方式只有信息增益在计算过程中能获得一个分裂距离,所以本文通过信息增益来计算每个局部特征对数据集中不同实例的区分度.

shapelet 有序距离线(Orderline)是计算信息增益过程中的重要概念[11],下面进行介绍.

定义 4. shapelet 有序距离线.

shapelet 有序距离线是记录了 shapelet 和训练集中每个实例的距离的一维数组,数组中的元素按距离递增排序,且每个元素对应一个类属性值.

图 2 中展示了一个 shapelet 的有序距离线. 我们依次计算训练集中每个实例和给定 shapelet 间的距离,并根据距离由小到大将训练实例的类属性映射到给定的数轴上,这样就得到图 2 中的有序距离线.

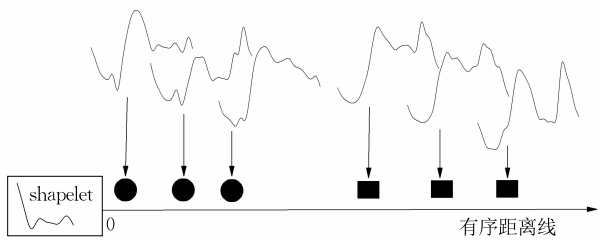


图 2 shapelet 有序距离线

定义 5. 信息熵.

数据集 D 上的信息熵 $H(D)$ 为

$$H(D)=-\sum_{i=1}^{C}\frac{n_i}{N}\log\frac{n_i}{N} \tag{1}$$

其中,N 表示数据集 D 中的实例个数,C 表示数据集 D 中的类属性个数,n_i 表示数据集中类属性 c_i 对

应的实例个数，$1\leq i\leq C$.

定义 6.　分裂点的信息增益.

给定一个 shapelet 对应于数据集 D 的有序距离线，根据有序距离线上的分裂点距离 s 可将数据集 D 划分为两个独立的子集合：$D_{\text{left}}=\{T_i\mid d(T_i,\text{shapelet})\leq s\}$ 和 $D_{\text{right}}=\{T_i\mid d(T_i,\text{shapelet})>s\}$，则 shapelet 在分裂点 s 处的信息增益为

$$I(\text{shapelet},s)=H(D)-\frac{|D_{\text{left}}|}{|D|}H(D_{\text{left}})-\frac{|D_{\text{right}}|}{|D|}H(D_{\text{right}}) \quad (2)$$

其中 $H(D)$ 表示集合 D 的信息熵，$|*|$ 表示集合中的实例数.

下面我们介绍如何基于 shapelet 的有序距离线计算 shapelet 的信息增益. 给定一个 shapelet 的有序距离线，我们依次取 shapelet 有序距离线上相邻两个距离点的中点作为分裂点来计算信息增益. 最后，我们将计算得到的最大信息增益作为候选 shapelet 的信息增益. 即

$$IG(\text{shapelet})=\arg\max_s I(\text{shapelet},s) \quad (3)$$

上述公式计算得到的 shapelet 的信息增益体现了 shapelet 对数据集合 D 中类属性归属不确定性的减少程度. 假定一个二分类数据集中包含 10 个实例. 图 3 中给出了这 10 个实例到最优 shapelet 的距离组成的有序距离线，其中不同符号代表不同类属性. 这个数据集的信息熵 $H(D)=-(1/2\times\log(1/2)+1/2\times\log(1/2))=0.3010$. 根据每个分裂点 s_i 可计算得到一个分裂点信息增益，例如，第一个分裂点的信息增益 $I(\text{shapelet},s_1)=H(D)-1/10\times H(D_{\text{left}})-9/10\times H(D_{\text{right}})=0.3010-1/10\times 0-9/10\times 0.2983=0.0325$. 同理可求得 $I(\text{shapelet},s_2)=0.0712,\cdots,I(\text{shapelet},s_6)=0.1836$ 等. 最后得到最大的信息增益为 $I(\text{shapelet},s_6)$，于是我们将该 shapelet 的信息增益取为 0.1836，对应的分裂阈值为 s_6.

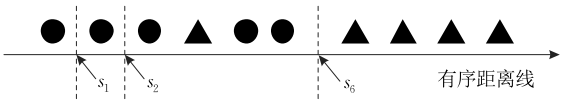


图 3　shapelet 信息增益计算

2.2　相似性度量

几乎各种时间序列分类模型都依赖于序列的相似性度量. 相似性度量方式通常是基于一个距离函数，它以两条时间序列作为输入，返回二者的距离作为两条时间序列的相似程度. 通常在计算两条时间序列的距离之前，需要对两条时间序列进行规范化处理，这样做可以消除度量单位的不同或采样过程中的偏移对相似性造成的影响. 下面介绍本文使用的 z-规范化方法[22]，我们以欧式距离(Euclidean Distance，ED)函数为例进行说明，长度为 m 的时间序列 T 的均值和标准差分别为

$$\bar{t}=\frac{1}{m}\sum_{i=1}^{m}t_i \quad (4)$$

$$\sigma_T=\sqrt{\frac{1}{m}\sum_{i=1}^{m}(t_i-\bar{t})^2} \quad (5)$$

时间序列 T 的 z-规范化公式：

$$t_{\text{norm}}=\frac{t-\bar{t}}{\sigma_T} \quad (6)$$

接下来，我们讨论的时间序列都是指经过规范化的时间序列.

设函数 $d(T_i,T_j)$ 为一种时间序列距离度量函数，给定两条等长时间序列 T_i 和 T_j，$d(T_i,T_j)$ 返回一个非负值 d，这个值反映了 T_i 和 T_j 间的相似程度. 一般情况下，两条时间序列间的距离越小意味着越相似. 距离度量函数也可以用来度量两个不等长序列间的相似程度. 下面我们用符号 $|X|$ 表示时间序列 X 的长度.

给定两条不等长的时间序列 T,S，其中 $|S|<|T|$. $W_{|S|}$ 表示时间序列 T 的长度为 $|S|$ 的子序列集合，则两条不等长时间序列间的相似程度为

$$d(T,S)=\min(d(S_i,S)),S_i\in W_{|S|} \quad (7)$$

Ding 等人的研究表明并没有一种距离度量方式比其他所有度量方式更好[7]. ED 和其他 L_p 形式的距离度量函数被广泛用于时间序列间的距离度量[8]，然而这些距离度量方法在相似性的度量上有时表现较差，不具有鲁棒性[7,9]. ED 在度量两条时间序列相似性的过程中不能避免两条时间序列间由于时间延迟或数据噪声导致的误差，而动态时间规整可以有效解决这一问题.

动态时间规整方法(Dynamic Time Warping，DTW)是处理时间轴局部扭曲问题中常用的方法[22-26]. 这种度量方法能够动态调整时间轴来对应一条时间序列中的各个片段，最后形成一条规整路径. 一条规整路径是一组连续的矩阵坐标，它表示的是两条时间序列之间的一种映射关系. 动态规整的目的就是寻找一条最短的规整路径，并将该路径计算出的距离作为两条时间序列间的距离. DTW 的时间复杂度为 $O(m^2)$[24]. 一般，通过引入下界度量方法可

以加速 DTW 的计算过程，例如，Keogh 等人利用上下边界序列计算下界距离[25]，Rakthanmanon 等人提出用两条时间序列的起点(终点)间的欧式距离作为下界[22]等. 本文分别尝试使用 ED 和 DTW 来计算时间序列间的相似性.

3 待分类实例 shapelets 提取技术

这部分介绍本文提出的懒惰式 shapelet 分类模型和模型中用到的一些重要算法. 计算每个候选 shapelet 和每条时间序列的距离是寻找最优 shapelet 过程中最耗时的部分. 为了减少寻找最优 shapelet 过程中的计算量，我们从两个方面进行改进：其一，建立待分类实例的候选 shapelets 集合；其二，尝试使用部分训练实例对候选 shapelets 进行评价. 下面首先介绍本文提出的建立近邻集合和候选 shapelets 集合的方法.

3.1 待分类实例 shapelets 鉴别性评价实例集合

对规模巨大的数据集，若不进行实例选择，模型的运行时间难以承受. 而基于传统的近邻搜索算法建立的近邻实例集合存在分类路径的初始节点对应的近邻实例集合中的实例属于同一类的情况，此时模型退化为单节点分类模型，特别是当近邻实例个数设为 1 时，模型退化为 1NN 分类模型，不能有效提取鉴别性特征. 此外，用于评价 shapelets 鉴别性强弱的数据集合的类属性分布会直接影响获得的 shapelets 的质量. 从信息熵的角度分析，若用于评价的实例集合中只有一类实例，则所有 shapelets 的鉴别性都为 0，理想情况是用于评价的实例集合具有类平衡的特点，此时实例集合中类属性的不确定性最大. 实际上，用于对 shapelets 鉴别性进行评价的集合只要具有一定的不确定性，就可以用来对 shapelets 的鉴别性进行评价.

针对以上问题，我们提出为待分类实例建立包含不同类属性实例的近邻实例集合，这样就保证了我们可以对待分类实例的局部特征的鉴别性进行评价. 特别是对于二分类数据集，本文的方法保证每个待分类实例在初始节点对应的数据集合具有最大的信息熵，即类属性归属具有最大的不确定性. 此外，通过实例选择还可以消除同类差异较大训练实例对待分类实例局部特征鉴别性评价的影响.

下面给出本文提出的建立待分类实例的相似实例集合的算法.

算法 1. *BuildingDiffClassValuesDataset*(T, D, k).

输入：待分类实例 T，包含 n 个训练实例的集合 $D=\{T_1, T_2, \cdots, T_n\}$，同类和异类实例个数 k

输出：待分类实例的相似实例集合 $D_k(T)$

1. FOR $i=1$ to n
2. $d(T_i, T) \leftarrow$ compute the distance between T_i and T
3. END FOR
4. sort the instances in the dataset D according to the distance $d(T_i, T)$ in ascending order
5. double $c \leftarrow$ the class value of T's nearest instance
6. $D_{same}(T) \leftarrow$ select the top k instances with the class value c from the set D
7. $D_{diff}(T) \leftarrow$ select the top k instances with the class values different from the given value c in the set D
8. $D_k(T) \leftarrow D_{same}(T) \cup D_{diff}(T)$
9. RETURN $D_k(T)$

算法 1 中第 6 步和第 7 步分别挑选 k 个与待分类实例最近的训练实例类属性相同和不同的实例，然后将这两个子集合合并组成待分类实例的近邻实例集合，这意味着实际中待分类实例的近邻集合包含 $2\times k$ 个实例.

3.2 求解待分类实例的候选 shapelets 集合

在 Ye 等人提出的建立 shapelets 决策树方法中[11]，生成候选 shapelets 集合过程中需要遍历训练实例集合中每条时间序列中指定长度范围内的所有的子序列，这会导致产生大量的冗余、相似候选 shapelets. 为了减少模型提取 shapelets 过程中的计算量. 同时，为了使得提取出的 shapelets 能够更好的反映待分类实例所具有的特征，我们提出了一种数据驱动的 shapelets 搜索算法寻找最优的 shapelet，我们只从待分类实例的子序列中寻找最优的 shapelet，这样提取出的 shapelets 准确地反映了每个待分类实例的局部特征.

下面给出本文建立待分类实例的候选 shapelets 集合的算法.

算法 2. *GenerateAllCandidates*(T, *min*, *max*).

输入：待分类实例 T，shapelets 的最小长度 *min*，shapelets 的最大长度 *max*

输出：时间序列 T 生成的所有候选 shapelets 集合：*Candidates_Set*

1. *Candidates_Set* $\leftarrow \varnothing$
2. FOR $i=0$ to $|T|$
3. FOR $j=min$ to max
4. $S \leftarrow Candidate(T, i, j)$ 为 T 上起始位置为 i 长度为 j 的子序列

5.　*Candidates_Set.add*(*S*)
6.　END FOR
7. END FOR
8. RETURN *Candidates_Set*

Ye 等人建立 shapelets 决策树过程中每个节点对应的候选 shapelet 集合中有 $O(nm^2)$ 个候选 shapelets，其中 n 表示时间序列的个数，m 表示时间序列的长度. 而本文提出的懒惰式 shapelets 分类路径中每个节点对应的候选 shapelet 集合中只有 $O(m^2)$ 个候选 shapelets，本文将每个节点需要评估的候选 shapelets 集合的规模降了一个量级.

我们用信息增益对候选 shapelet 的鉴别性进行评价，首先，需要计算候选 shapelet 和实例集合中每个实例的距离，这一过程要计算 $O(n)$ 个不等长序列间的距离；其次，计算一个候选 shapelet 和一条完整时间序列之间的距离的时间复杂度为 $O(m^2)$；最后，我们可以根据计算得到的这组距离和对应的类属性值计算每个候选 shapelet 的最大信息增益，这部分计算量可忽略不计. 综上所述，shapelets 决策树中使用 Brute-Force 算法寻找最优的 shapelet 需要执行 $O(n^2m^4)$ 次计算，而本文为每个待分类实例寻找最优的 shapelet 只需执行 $O(lm^4)$ 次计算，其中 l 为每个待分类实例的近邻集合中的实例数. 考虑到 l 比 n 小，本文的懒惰式 shapelets 分类模型在节点寻找最优 shapelet 过程中的计算量显著更少. Ye 等人的决策树模型对于规模较大的训练集，模型的训练时间难以承受，而由于本文将针对单个实例寻找最优 shapelet 算法的复杂度降了一个量级，使得我们可以在规模较大的训练集上为待分类实例建立本文提出的模型. 下面介绍本文用于寻找最优 shapelet 的算法思想.

3.3　寻找待分类实例的最优 shapelet

寻找最优 shapelet 的目的是希望利用局部特征最大程度上减少待分类实例类属性归属的不确定性，这种不确定性体现在每次利用 shapelet 划分后的子集合中实例类属性的分布情况. 例如，对于二分类问题，理想的情况是根据某一 shapelet 及其分裂阈值可将包括两种类属性实例的实例集合分为两个子集，每个子集都只有一类实例. 实际中，单个 shapelet 的鉴别性可能不足，无法将不同类的实例清楚的区分开. 为此，我们在子集合上递归提取最优的 shapelet，并利用提取出的 shapelet 对数据集进行进一步划分，直到划分后的子集只包含同类实例，这样获得的一组可以有效将训练集合的信息熵降为 0 的 shapelet 就可以用来判定待分类实例属于某一类，这是本文模型的基本思想. 本文我们在寻找待分类实例最优的 shapelet 过程中，首先，选定一个包含不同类属性的近邻集合，然后在这个集合上对待分类实例的候选 shapelets 进行评价，最后将信息增益最大的 shapelet 作为最优 shapelet.

在寻找最优 shapelet 的过程中，本文使用了两种常用的提高 shapelet 搜索效率的方法[11,18]：

(1) 早期放弃机制. 这种机制用于减少计算 shapelet 和完整时间序列间距离过程中的计算量；

(2) shapelet 熵剪枝. 该方法用来减少寻找最优的 shapelet 过程中的计算量.

3.4　懒惰式 shapelets 分类模型

本文提出为每个待分类实例建立基于 shapelets 的分类路径. 本文模型直接对待分类实例的每个局部特征进行评价，模型中提取的 shapelets 都来自待分类实例，这意味着所有局部特征和待分类实例的距离都为 0. 建立分类路径过程中我们将节点对应的数据集中和 shapelet 的距离大于分裂阈值的实例排除，距离小于等于分裂阈值的训练实例组成新的子集合. 若子集合中实例属于同一类，则将该类属性作为预测值，否则继续在子集合上对候选 shapelets 进行评价提取最优的 shapelet，并对节点集合进一步划分，直到子集中只包含同类实例. 此时，我们将该类实例的类属性作为预测值. 下面我们首先给出本文提出的在每个节点划分数据集的算法.

算法 3.　*GeneratingNodeDataset* (*shapelet*, *D*, *T*).

输入：用于划分数据集的 shapelet，时间序列数据集合 *D*，待分类实例 *T*
输出：*D* 的子集合：*subD*
1. $d \leftarrow ComputeDistance(T,$ shapelet$)$
2. $\delta \leftarrow$ shapelet 对应的分裂阈值
3. $n \leftarrow$ 数据集 *D* 中的实例个数
4. FOR $i=1$ to n
5.　$d_i \leftarrow ComputeDistance(T_i,$ shapelet$)$
6.　if $d_i \leq \delta$, then add the instance T_i into *subD*
7. END FOR
8. RETURN *subD*

算法 3 介绍了如何利用 shapelet 对模型中节点对应的数据集中的实例进行选择. 我们只保留那些和 shapelet 的距离不大于 shapelet 分裂阈值 δ 的训练实例. 我们利用图 3 对算法 3 进行说明. 算法 3 通过计算数据集中每个实例和 shapelet 的距离可以得到图 3 中有序距离线上的一组有序距离. 然后，我们选择位于 shapelet 的分裂阈值左边的实例组成新的训练子集，分裂阈值右边的实例删除.

下面给出本文提出的懒惰式 shapelet 分类算法.

算法 4. $LSCR(D,T,k)$.

输入:数据集合 D,待分类实例 T,与待分类实例最近的实例同类和异类的实例个数 k

输出:待分类实例 T 的分类路径:CRouteForT

1. $D_k(T) \leftarrow BuildingDiffClassValuesDataset(T,D)$;
2. 若节点对应集合 $D_k(T)$ 中所有的实例具有相同的类属性值 C_k,则搜索结束,返回分类路径 CRouteForT,并将 C_k 作为类属性的预测值;否则:
3. $best\_shapelet \leftarrow FindingShapletBF(T,D_k(T),min,max)$
4. $D' \leftarrow GeneratingNodeDataset(best\_shapelet, D_k(T), T)$
5. 构建子节点,在 D' 上重复 2~5 步,直至结束
6. RETURN 待分类实例 T 的分类路径:CRouteForT

算法 4 中第 1 步为待分类实例建立包含不同类属性实例的近邻实例集合;第 2 步基于节点对应的数据集判断是否满足终止条件,本文的分类模型不会在初始节点满足终止条件,即不会退化为单节点路径;第 3 步提取节点数据集对应的最优的 shapelet;第 4 步用提取出的 shapelet 对节点数据集进行划分,生成子节点对应的数据集;第 5 步建立子节点,并重复 2~5 步,直到满足终止条件,返回待分类实例的基于 shapelets 的分类路径.

4 实验分析

我们在 18 个来自 UCR 时间序列知识库的数据集上对本文提出的算法进行实验分析,表 1 中给出了数据集的介绍,其中 *MinNum* 表示训练集中分布最少的类属性对应的实例数.

实验运行环境的 CPU 为 3.00 GHz,内存为 4 GB,操作系统是 Windows 7. 本文所有的算法都是使用 Java 在 Weka 框架中实现的,我们将实验中用到的数据集都分为训练部分和测试部分,在训练部分上构建对应的分类模型,在测试部分上计算模型的分类准确率. 下面介绍和实验相关的一些预处理步骤:

(1) shapelets 的长度参数. 长度从 3 到时间序列的完整长度,每次递增 1.

(2) 同类和异类近邻实例数 k. 本文我们建立的近邻实例集合由两部分实例组成,即 k 个和待分类实例最近的实例类属性相同的实例以及 k 个和待分类最近实例类属性不同的实例.

接下来,我们首先对本文模型邻域大小的参数设置进行实验分析.

表 1 实验数据集介绍

Dataset	Instances (train/test)	Length	Classes	*MinNum*
BeetleFly	20/20	512	2	10
Coffee	28/28	286	2	14
ECGFiveDays	23/861	136	2	9
Gun_Point	50/150	150	2	24
ItalyPowerDemand	67/1029	24	2	33
MoteStrain	20/1252	84	2	10
SonyAIBORobotSurface	20/601	70	2	6
SonyAIBORobotSurfaceII	27/953	66	2	11
Wafer	1000/6164	152	2	97
CBF	30/900	128	3	8
ProximalPhalanxOLAG	400/205	80	3	72
OliveOil	30/30	570	4	4
Beef	30/30	470	5	6
DistalPhalanxTW	400/139	80	6	18
Synthetic-control	300/300	60	6	50
Lighting7	70/73	319	7	5
MedicalImages	381/760	99	10	6
FacesUCR	200/2050	131	14	4

4.1 参数分析及选择

通过实例选择不仅可以排除异类差异较大的实例,也可以排除同类间差异较大的实例,这样有利于提高 shapelet 的搜索效率和分类效果. 本节研究了两种确定本文模型中实例选择参数 k 值的方法,并对不进行实例选择的情况进行了实验分析. 第一种方法通过观察指定范围内各数据集上模型的准确率和时间随 k 值变化的趋势来确定最优的 k 值,第二种方法通过交叉验证在指定的范围内寻找最优的 k 值. 下面我们对第一种方法的实验结果进行分析.

图 4 中给出了基于 DTW 的 LSCR 模型(DTWLSCR)在 9 个二分类数据集上的准确率随 k 值的变化规律. 从图中我们可以看出多数数据集的准确率在 $k=5$ 时达到一个较高值后都呈现出明显的下降趋势. 因此,接下来的实验中我们将二分类数据集上用于对比的 DTWLSCR 中使用的 k 值设为 5.

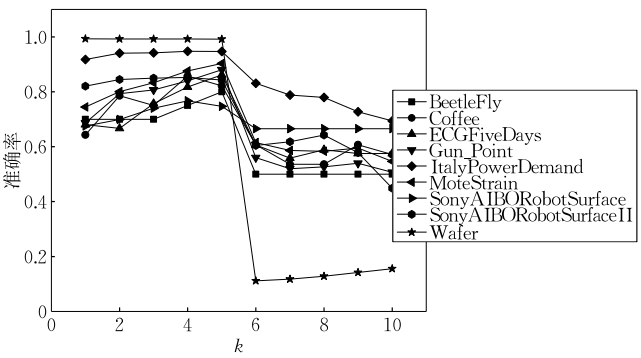


图 4 二分类数据集上准确率随 k 值的变化趋势

图 5 中给出 DTWLSCR 在 9 个多类数据集上的准确率随 k 值的变化趋势. 从图中我们可以看出,

除了 Beef 数据集准确率有波动外，随着 k 值增大其他 8 个数据集的准确率都呈现出递增趋势，当 $k\geq 6$ 后各数据集上的准确率逐渐趋于稳定，因此，在接下来的对比实验中，我们将 DTWLSCR 在多类数据集合上的 k 值设为 6.

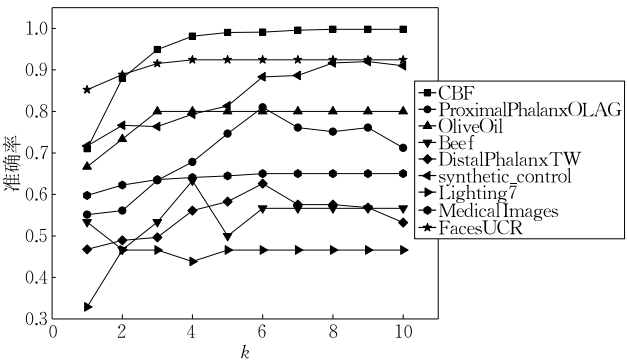


图 5　多类数据集上算法准确率随 k 值的变化趋势

表 2 给出了分别使用遍历指定范围和交叉验证两种确定 k 值方式的 DTWLSCR 模型的准确率，其中我们给出了上界分别设为 10 和 50 时的交叉验证的准确率，实验结果显示两个不同上界对应的模型准确率几乎完全一样，这是由于本文使用的基于 kNN 模型的交叉验证方法获得的最优参数 k 倾向于选择较小值的结果. 从表 2 中我们可以看出第一种方法在 18 个数据集中的 16 个上获得了更高的准确率，18 个数据集上的平均准确率比第二种方法获得准确率高出 10%. 因此，接下来的对比实验中我们使用第一种方法来设置 k 值.

表 2　基于不同 k 值设定方法的 DTWLSCR 模型的准确率

Dataset	DTWLSCR (k=5/6)/%	DTWLSCR (CV10)/%	DTWLSCR (CV50)/%
BeetleFly	**80.00**	70.00	70.00
Coffee	**82.14**	64.29	64.29
ECGFiveDays	**86.18**	68.06	68.06
Gun_Point	**88.00**	68.67	68.67
ItalyPowerDemand	**94.66**	94.17	94.17
MoteStrain	**90.34**	74.44	74.44
SonyAIBORobotSurface	**74.71**	67.55	67.55
SonyAIBORobotSurfaceII	84.26	**84.99**	**84.99**
Wafer	99.16	**99.32**	**99.32**
CBF	**99.11**	71.00	71.00
ProximalPhalanxOLAG	**80.98**	76.10	66.34
OliveOil	**80.00**	66.67	66.67
Beef	**56.67**	53.33	53.33
DistalPhalanxTW	**62.59**	57.55	57.55
synthetic_control	**88.33**	76.33	76.33
Lighting7	**46.58**	32.88	32.88
MedicalImages	**65.00**	59.74	59.74
FacesUCR	**92.44**	85.22	85.22
Average	**80.62**	70.57	70.03

图 6 中给出了 18 个数据集上 DTWLSCR 模型的运行时间随 k 值的变化趋势.

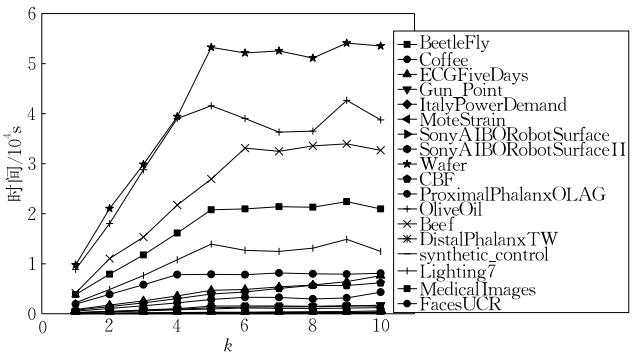
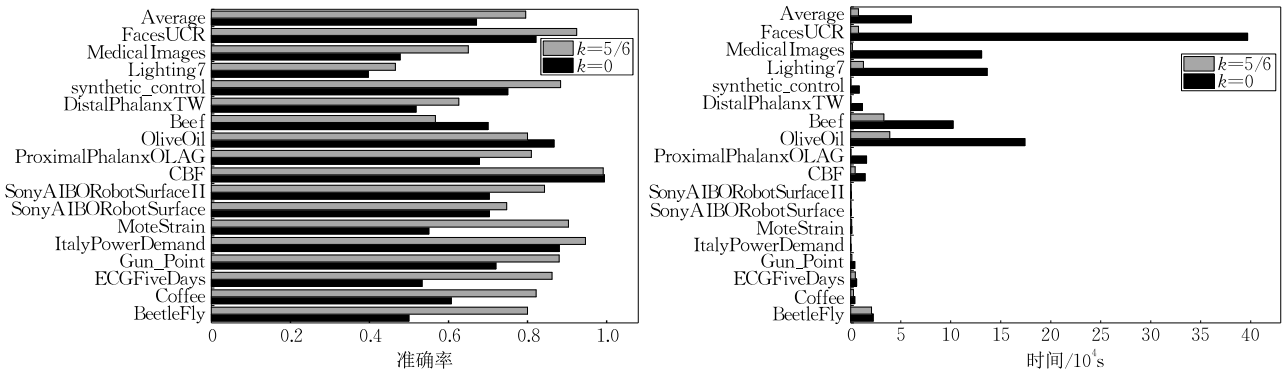


图 6　18 个数据集上模型运行时间随参数 k 的变化趋势

从图 6 中我们可以看出随着 k 的增大，模型的运行时间显著增大. 由于本文模型中的参数满足 $1\leq k\leq MinNum$，即近邻集合中各类实例的个数都不能超过数据集合中分布最少类属性的实例个数，近邻集合的大小的上限为 $2\times MinNum$. 当近邻集合的大小达到上限后，模型的计算量趋于稳定，正如图 6 中所示，模型的运行时间存在拐点，达到拐点后模型的运行时间趋于稳定，例如，数据集 OliveOil，Beef 和 Lighting7 上模型的运行时间在 k 值达到 $MinNum$ 后趋于稳定.

接下来，我们在 17 个数据集上对初始节点不进行实例选择的情况下的准确率和时间进行对比分析. 由于 Wafer 数据集上不进行实例选择条件下 DTWLSCR 模型的运行时间难以承受，这里我们没有考虑. 图 7 是 17 个数据集上基于不同参数 k 设置的 LSCR 模型的准确率和时间的柱状对比图. 图 7 中每个数据集对应两条柱，其中上面一条表示进行实例选择的情况，即 $k=5/6$；下面一条表示不进行实例选择的情况，即 $k=0$. 图 7 中 Average 代表 17 个数据集上的平均值.

从图 7(a)中我们可以看出在初始节点不进行实例选择条件下，DTWLSCR 准确率在 17 个数据集中的 14 个上低于进行实例选择的情况，且平均准确率低了 12.46%. 从图 7(b)可以看出除了在规模较小或时间序列长度较小的数据集 BeetleFly，Coffee，ECGFiveDays，MoteStrain，SonyAIBORobotSurface 上时间相近外，随着数据集规模或时间序列长度的增大，不进行实例选择的模型运行时间变得难以承受. 例如，在规模较大的数据集 synthetic_control 上初始节点不进行实例选择的模型运行时间是进行实例选择模型的 88 倍，而准确率低了 13.33%. 基于图 7 的实验分析结果，我们可以得到结论：为每个待



(a) 准确率对比 (b) 时间对比

图 7 17 个数据集上基于不同参数的 DTWLSCR 模型的准确率和时间的对比图

分类实例进行训练实例选择是必要的,这既可以提高模型的准确率,也可以加快模型的运行时间.

4.2 LSCR 模型准确率分析

本节我们对本文提出的模型的准确率进行分析,表 3 给出了 DTWLSCR 模型和其他 7 个模型的准确率及排名的对比结果.这 7 个模型分别为:基于欧式距离的本文模型(EDLSCR),Ye 等人提出的 shapelets 决策树模型(Shapelets Decision Tree, SDT)[11],Rakthanmanon 等人提出的快速 shapelets 决策树模型(Fast Shapelets Decision Tree,FS)[19],基于 DTW 和欧式距离的 1NN 模型(分别记作 DTW1NN 和 ED1NN)以及分别基于 Yuan 等人[14]和 Lines 等人[12]提出的 shapelets 转换算法的 C4.5 模型(分别记作 SSC4.5 和 STC4.5).其中,DTWLSCR 在二分类数据集上将 k 设为 5,在多分类数据集上的 k 值设为 6;EDLSCR 在所有数据集上的参数 k 都设为 5;SDT 的候选 shapelets 的长度每次递增 1,使用和本文模型相同的加速方式;FS 的准确率我们采用 Rakthanmanon 等人提供的值;基于 shapelets 转换算法的 C4.5 模型采用对应论文中的参数设置.表 3 中倒数第二行给出了各个模型在 18 个数据集上的平均准确率,表 3 中最后一行给出了 DTWLSCR 和用于对比的模型的准确率大小的比较统计结果,其中 w 表示本文模型准确率更高的数据集个数,t 表示相同的数据集个数,l 表示差的数据集个数.表 3 中的实验结果说明 EDLSCR 模型在 5 个数据集上比 DTWLSCR 模型更好,在其余 13 个数据集上的实验效果都较差.这再次验证了欧式距离不能很好处理数据采集过程中由于时间延迟或噪声导致的误差,而基于 DTW 的 LSCR 模型具有更强的鲁棒性.因此,我们在表 3 中将基于 DTW 的 LSCR 模型和其他模型进行对比.

表 3 算法的准确率和排名

Dataset	DTWLSCR	EDLSCR	SDT	FS	DTW1NN	ED1NN	SSC4.5	STC4.5
BeetleFly	80.00%(2)	**85.00%**(1)	75.00%(4.5)	79.55%(3)	70.00%(6.5)	75.00%(4.5)	70.00%(6.5)	60.00%(8)
Coffee	82.14%(6.5)	89.29%(3)	89.29%(3)	**93.21%**(1)	82.14%(6.5)	71.43%(8)	89.29%(3)	85.71%(5)
ECGFiveDays	86.18%(6)	89.78%(5)	99.42%(2)	**99.59%**(1)	76.77%(8)	81.42%(7)	98.95%(3)	95.82%(4)
Gun_Point	88.00%(7)	87.33%(8)	**94.67%**(1)	93.93%(2)	90.67%(4.5)	90.67%(4.5)	91.33%(3)	89.33%(6)
ItalyPowerDemand	94.66%(6)	95.24%(3.5)	**95.82%**(1)	90.51%(8)	95.04%(5)	95.24%(3.5)	94.17%(7)	95.43%(2)
MoteStrain	**90.34%**(1)	88.98%(2)	82.27%(5)	79.28%(7)	83.55%(4)	84.50%(3)	81.79%(6)	75.96%(8)
SonyAIBORobotSurface	74.71%(3)	72.55%(5.5)	73.54%(4)	68.55%(7)	72.55%(5.5)	68.22%(8)	84.86%(2)	**87.69%(1)**
SonyAIBORobotSurfaceII	84.26%(3)	84.58%(2)	65.90%(8)	78.52%(6)	83.11%(4)	**85.31%**(1)	79.54%(5)	75.66%(7)
Wafer	99.16%(6)	99.04%(7)	99.72%(2)	99.64%(3)	97.99%(8)	99.43%(4)	99.33%(5)	**100.00%(1)**
CBF	99.11%(2)	98.44%(3)	93.89%(6)	94.71%(5)	**99.67%**(1)	83.56%(8)	97.89%(4)	88.11%(7)
ProximalPhalanxOLAG	80.98%(2)	79.51%(5)	78.05%(7)	79.72%(4)	80.49%(3)	78.54%(6)	**84.39%(1)**	77.07%(8)
OliveOil	80.00%(2.5)	70.00%(6.5)	66.67%(8)	78.67%(4)	**86.67%**(1)	76.67%(5)	70.00%(6.5)	80.00%(2.5)
Beef	56.67%(3)	53.33%(4.5)	50.00%(6.5)	55.33%(4.5)	50.00%(6.5)	**60.00%(1.5)**	**60.00%(1.5)**	30.00%(8)
DistalPhalanxTW	62.59%(5)	53.96%(8)	**64.75%(1.5)**	62.33%(6)	58.99%(7)	64.03%(3)	63.31%(4)	**64.75%(1.5)**
synthetic_control	88.33%(6.5)	69.67%(8)	94.00%(2)	91.90%(4)	**99.33%**(1)	88.33%(6.5)	92.33%(3)	90.00%(5)
Lighting7	46.58%(3.5)	43.84%(5.5)	36.99%(8)	**59.73%**(1)	49.32%(2)	43.84%(5.5)	38.36%(7)	46.58%(3.5)
MedicalImages	65.00%(2)	61.58%(5)	51.58%(7)	56.70%(6)	**68.03%**(1)	63.42%(4)	63.82%(3)	48.68%(8)
FacesUCR	**92.44%(1)**	81.12%(3)	64.54%(7)	70.07%(5)	90.49%(2)	75.02%(4)	63.12%(8)	66.20%(6)
Average	**80.62%**	77.96%	76.45%	79.55%	79.71%	76.92%	79.03%	75.39%
$w/t/l$		13/0/5	11/0/7	12/0/6	10/1/7	11/1/6	9/0/9	8/2/8

我们采用 Demšar 等人提出的方法对表 3 中的多个分类器在 18 个数据集上的性能表现进行显著性检验[27]. 从图 8 中我们可以发现 18 个数据集上 8 个模型不存在显著性差异，但 DTWLSCR 分类结果明显更好一些.

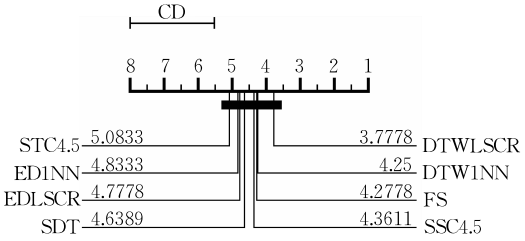


图 8　8 个分类模型在 18 个数据集上的临界差异图

从表 3 中最后一行给出的统计结果我们可以看出 DTWLSCR 模型比其他模型在更多的数据集上更准确. 在 MoteStrain，FacesUCR，OliveOil，SonyAIBORobotSurfaceII 等数据集上本文模型的准确率和建立在整个训练集合上的 shapelets 决策树模型相比显著提高，在 FacesUCR 上准确率提高最为明显，比 SDT 和 FS 模型的准确率高了 20%以上.

和两种基于不同 shapelets 转换方法的 C4.5 模型相比，本文提出的模型将 BeetleFly、MoteStrain 和 FacesUCR 数据集上的准确率提高了 8%以上，其中 FacesUCR 数据集上的准确率提高了 25%以上.

DTWLSCR 模型在 BeetleFly，ECGFiveDays 数据集上的准确率比 DTW1NN 模型的准确率高了接近 10%，在 MoteStrain，Beef 两个数据集上的准确率比 DTW1NN 模型高了 5%以上. 和 ED1NN 模型相比也有类似结果. 表 4 给出了 DTWLSCR 模型比 DTW1NN 模型准确率更高的 5 个数据集上的不同预测情况统计. 表 4 中的符号 IstNum 表示两种模型预测结果不同的实例数，DTW1NN_R 和 DTWLSCR_R 分别表示在上述预测结果不同的实例中对应模型预测正确的实例数，其中括号中的值表示占测试集的百分比.

表 4　两个模型在 5 个数据集上的不同预测结果统计

	BeetleFly	ECGFiveDays	MoteStrain	Beef	FacesUCR
IstNum	6(30%)	203(23.58%)	249(19.89%)	17(56.67%)	198(9.67%)
DTWLSCR_R	4(20%)	142(16.49%)	167(13.34%)	7(23.33%)	95(4.63%)
DTW1NN_R	2(10%)	61(7.08%)	82(6.55%)	5(16.67%)	55(2.68%)

从表 4 中可以看出 DTWLSCR 模型在不同数据集上的预测结果和 DTW1NN 的预测情况存在明显不同. 这说明本文模型虽然根据最近的近邻实例的类属性对训练实例进行选择，但本文模型对每个待分类实例的预测结果并不一定和距离他最近的训练实例的类属性相同. 显然，这决定了本文模型在表 4 中 5 个数据集上获得了更好的准确率.

以上实验结果表明基于局部相似性的本文模型与不同 shapelet 决策树模型、基于全局相似性的 1NN 模型以及基于 top-k shapelets 转换算法的 C4.5 模型相比具有一定的竞争力.

4.3　探索性数据分析

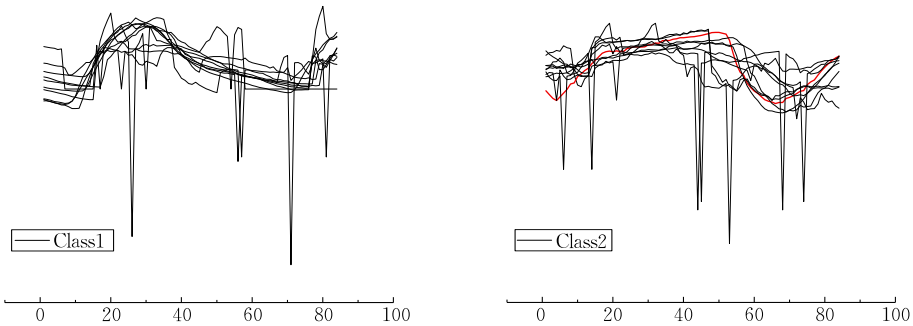
这部分我们分别在 UCR 时间序列知识库中的二分类和多分类数据集上对本文模型的可解释性进行深入分析.

4.3.1　MoteStrain 数据集

MoteStrain 中的传感数据最初用来在线检测传感器网络中的潜在变量. Eamonn Keogh 将这些数据进行规范化处理后作为研究时间序列分类的标准数据集. 这个数据集上的分类任务是通过时间序列来区分传感器是用于湿度测量还是用于温度测量. 本文提出的模型在该数据集上的分类结果比用于对比的模型的效果显著更好，和目前该数据集上最优的分类结果相近.

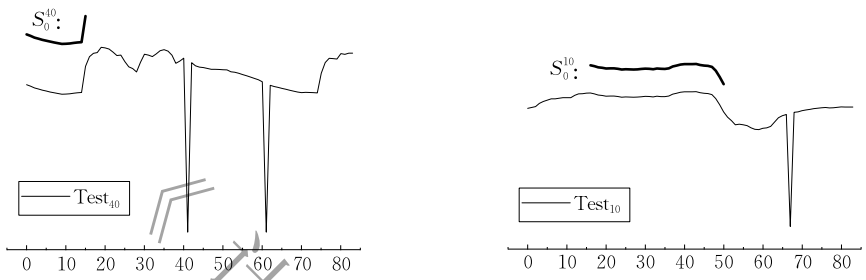
首先，图 9 给出了 MoteStrain 数据集中两类时间序列的图示. 从图 9(a)和(b)可以看出两类时间序列间局部特征并不显著，同类实例间的共同特征并不明显，而且同类实例间的一些局部特征差异也比较大，这加大了数据集上提取 shapelets 的复杂度. 接下来，我们在 MoteStrain 数据集上对本文的模型进行分析.

图 10 中给出了本文模型预测正确而 shapelets 决策树预测错误的两个待分类实例通过本文模型提取出的 shapelets 的图示，这两个实例属于不同类. 图中 S_i^j 表示第 j 个待分类实例的分类路径上节点 i 对应的 shapelet，Test_j 表示 MoteStrain 测试集中第 j 个实例.



(a) 第一类 (b) 第二类

图 9 MoteStrain 数据训练集中的两类实例



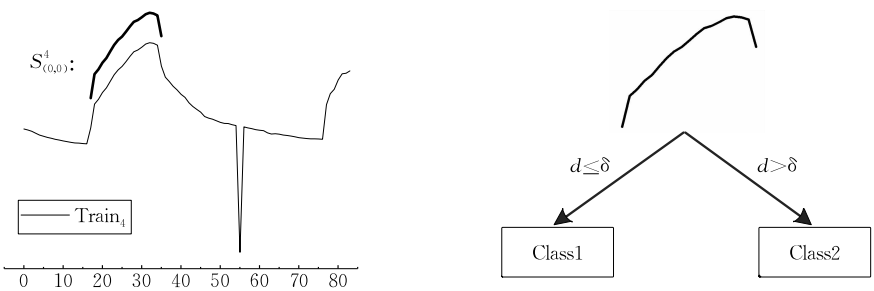
(a) 第40个待分类实例 (b) 第10个待分类实例

图 10 MoteStrain 数据集中两个待分类实例及其 shapelets

从图 10 中可以看出两个待分类实例都只提取出了一个 shapelet 就可以将待分类实例类属性的不确定性降为 0. 本文模型提取出的每个 shapelet 都来自待分类实例本身,这些 shapelets 可以很好的解释每个待分类实例由于具有哪些特征而被分到某一类. 如图 10(a)所示第 40 个待分类实例归属于类属性 1 的原因是他初始阶段所具有的局部特征 S_0^{40},而图 10(b)所示第 10 个待分类实例归属于类属性 2 的原因是他的中间部分所具有的局部特征 S_0^{10}.

为了进一步对本文的模型进行对比分析,我们在图 11 中给出了 SDT 模型中提取出的 shapelets 以及 shapelets 决策树模型,图 11(a)中 $S_{(0,0)}^4$ 表示 shapelets 决策树的根节点对应的来自训练集中第 4 个实例的 shapelet,Train$_4$ 表示根节点对应的 shapelet 来自的训练实例.

SDT 模型在 MoteStrain 数据集上建立的的决策树如图 11(b)所示,该 shapelets 决策树中只有一个 shapelet,其中的 d 表示待分类实例和 shapelet 的距离,δ 表示 shapelet 对应的分裂阈值. 基于图 11(b)中的 shapelets 决策树对待分类实例进行预测时,当待分类实例和根节点对应的 shapelet 的距离不大于分裂阈值时,则该待分类实例的类属性预测值为 Class1,否则,为 Class2. 我们不难发现图 11 中 shapelets 决策树模型中对应的 shapelet 和图 10 中给出的两个待分类实例分类路径上的 shapelets 差异较大,结合图 9 给出的训练集实例图示,我们不难发现 shapelets 决策树遍历整个训练集获得的最优 shapelet 从分类效果和实际形状特征来看对待分类实例并不是最优的,这验证了从整个数据集合上提取出的 shapelets 的鉴别性只在平均意义下对每个训练集中的实例是最优的这一结论,而这也是 shapelets 决策树在 MoteStrain 数据集上性能较差



(a) 第4个训练实例 (b) shapelets决策树

图 11 shapelets 决策树及 shapelet 所属训练实例

的原因，而本文模型提取出的 shapelet 很好的反映了待分类实例的局部特征，基于这样的局部特征我们可以有效逐步确定待分类实例的类属性.

4.3.2　FacesUCR 数据集

这节我们基于多类数据集 FacesUCR 对本文模型进行分析. 这个数据集中的每条时间序列都是由人脸的面部轮廓转化而来，整个数据集中共有 14 个人的面部轮廓，即该数据集的类属性有 14 种可能取值. 图 12 给出了 FacesUCR 数据集中 14 类时间序列的图示.

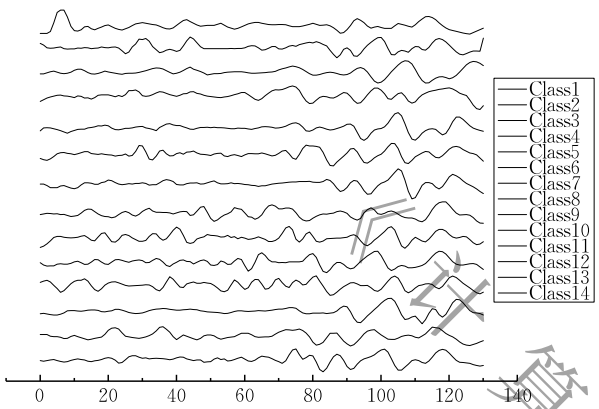
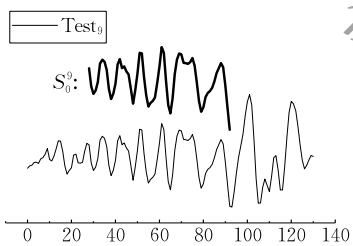


图 12　FacesUCR 数据集中的 14 类时间序列

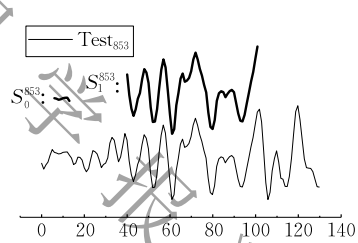
从图 12 我们可以发现多类数据集中各类实例间的局部特征显得错综复杂，表 3 中的实验结果显示建立在整个数据集上的 SDT 模型和 FS 模型的准确率都较差，而 DTWLSCR 在 FacesUCR 上的准确率比 SDT 和 FS 的分别高出了 27.90%，22.36%. 下面我们对两种模型进行分析.

SDT 模型在 FacesUCR 数据集上建立的 shapelets 决策树模型的高度为 7，树中从根节点到叶节点的最短路径的长度为 3，这意味着要确定一个待分类实例的类属性归属至少要和不同的 shapelets 进行三次距离计算. 而本文提出的模型在 FacesUCR 上对待分类实例进行预测的分类路径长度的最大值为 3，分类过程最多只需三步就可以确定待分类实例的类属性，对单个实例本文模型的预测时间相对更短.

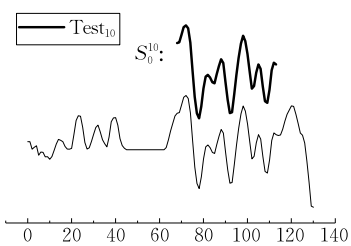
下面我们给出 4 个本文模型可以正确预测而 shapelets 决策树预测错误的实例及本文模型提取出的 shapelets 图示，图 13 中给出了 FacesUCR 的测试集中第 9、10、401 和 853 个待分类实例的分类路径及其 shapelets，这四个待分类实例的类属性分别为 Class7、Class11、Class11 和 Class4.



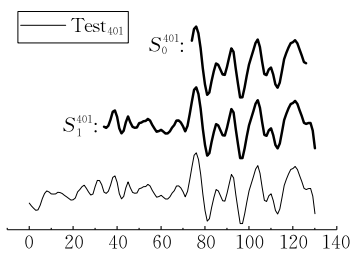
(a) 第9个待分类实例



(b) 第853个待分类实例



(c) 第10个待分类实例



(d) 第401个待分类实例

图 13　FacesUCR 数据集合中 4 个待分类实例和它们的 shapelets

本文提出的模型不仅可以有效说明待分类实例类属性归属的原因，还可以反映出同类待分类实例间差异，从图 13(a)和图 13(b)中我们可以看出在 FacesUCR 数据集上本文模型为不同类实例建立的路径长度可能不同，提取出的 shapelets 也不同，与此同时，从图 13(c)和图 13(d)可以看出本文模型可以反映同类实例间的差异，它们的分类路径长度和 shapelets 都可能不同，但它们的 shapelets 之间有相似之处，这是建立在整个数据集合上全局模型做不到的，全局模型完全忽略待分类实例间的差异. 此外，我们可以从图 13(d)看出模型不同节点提取的 shapelets 之间存在较大相似性，这是由于不同类实例间的差异性导致的. 单一固定长度的局部特征不足以将某一类和其他所有类实例甚至某一类的所有

实例区分开,这使得具有较大鉴别性的不同长度的相似局部特征被重复利用.

本文提出的模型可以用来逐步减少待分类实例类属性的不确定性.通过提取 shapelet 将和待分类实例存在局部差异的不同类实例逐步排除,直至最后节点对应的数据集合中只剩下同一类实例.最后,将剩下实例的类属性值作为待分类实例的预测值.图 13 中每个待分类实例的分类路径节点对应的 shapelets 很好的实现了这样一个目的,而表 3 中的实验结果验证了本文模型的有效性.

5 结 论

针对建立在整个训练集合上的 shapelets 分类模型存在的问题,本文为每个待分类实例构建一种数据驱动的 shapelets 分类模型.本文模型使用训练集中的部分实例对候选 shapelets 的鉴别性进行评价.为了避免模型在初始节点退化为单节点路径,我们不是简单的挑选 k 个距离待分类实例最近的实例,而是向近邻集合中加入不同类实例.较小的训练集合不仅减少了模型的运算时间,还提高了 shapelets 的质量.一般,基于 shapelets 的决策树模型是根据通过检测待分类实例包含或不包含某些特征来决定其类属性归属,而本文模型是基于待分类实例建立的,因此,本文模型提取出的 shapelets 一定来自待分类实例,这些局部特征可直接用来回答待分类实例由于具有哪些局部特征而被归属于某一类.本文模型对待分类实例的类属性归属有更强的可解释性,该模型可用于研究每个待分类实例所蕴含的局部特征信息,这是全局 shapelets 分类模型做不到的,具有较大的实际应用价值,例如,在医疗领域,获得每个患者的具体病因对于制定针对性的治疗方案很重要.

致 谢 在此感谢审稿人对本文提出的宝贵意见!

参 考 文 献

[1] McGovern A, Rosendahl D H, Brown R A, et al. Identifying predictive multi-dimensional time series motifs: An application to severe weather prediction. Data Mining and Knowledge Discovery, 2011, 22(1-2): 232-258

[2] Patri O, Wojnowicz M, Wolff M. Discovering malware with time series shapelets//Proceedings of the 50th Hawaii International Conference on System Sciences. Hawaii, USA, 2017: 6079-6088

[3] Zhu L, Lu C, Sun Y. Time series shapelet classification based online short-term voltage stability assessment. IEEE Transactions on Power Systems, 2016, 31(2): 1430-1439

[4] Burkom H S, Murphy S P, Shmueli G. Automated time series forecasting for biosurveillance. Statistics in Medicine, 2007, 26(22): 4202-4218

[5] Zhong S, Khoshgoftaar T M, Seliya N. Clustering-based network intrusion detection. International Journal of Reliability, Quality and Safety Engineering, 2007, 14(2): 169-187

[6] Xing Z, Pei J, Keogh E. A brief survey on sequence classification. ACM SIGKDD Explorations Newsletter, 2010, 12(1): 40-48

[7] Ding H, Trajcevski G, Scheuermann P, et al. Querying and mining of time series data: Experimental comparison of representations and distance measures. Proceedings of the VLDB Endowment, 2008, 1(2): 1542-1552

[8] Keogh E, Kasetty S. On the need for time series data mining benchmarks: A survey and empirical demonstration. Data Mining and Knowledge Discovery, 2003, 7(4): 349-371

[9] Antunes C M, Oliveira A L. Temporal data mining: An overview//Proceedings of the KDD Workshop on Temporal Data Mining. San Francisco, USA, 2001: 1-13

[10] Bagnall A, Lines J, Bostrom A, et al. The great time series classification bake off: A review and experimental evaluation of recent algorithmic advances. Data Mining and Knowledge Discovery, 2017, 31(3): 606-660

[11] Ye L, Keogh E. Time series shapelets: A new primitive for data mining//Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Paris, France, 2009: 947-956

[12] Lines J, Davis L M, Hills J, et al. A shapelet transform for time series classification//Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Beijing, China, 2012: 289-297

[13] Hills J, Lines J, Baranauskas E, et al. Classification of time series by shapelet transformation. Data Mining and Knowledge Discovery, 2014, 28(4): 851-881

[14] Yuan J D, Wang Z H, Han M. A discriminative shapelets transformation for time series classification. International Journal of Pattern Recognition and Artificial Intelligence, 2014, 28(6): 1-28

[15] Yuan Ji-Dong, Wang Zhi-Hai, Han Meng. Shapelet pruning and shapelet coverage for time series classification. Journal of Software, 2015, 26(9): 2311-2325(in Chinese)
(原继东,王志海,韩萌. 基于 Shapelet 剪枝和覆盖的时间序列分类算法. 软件学报, 2015, 26(9): 2311-2325)

[16] Yuan Ji-Dong, Wang Zhi-Hai, Han Meng, et al. A logical Shapelets transformation for time series classification. Chinese Journal of Computers, 2015, 38(7): 1448-1459(in Chinese)
(原继东,王志海,韩萌等. 基于逻辑 Shapelets 转换的时间序列分类算法. 计算机学报, 2015, 38(7): 1448-1459)

[17] Zalewski W, Silva F, Maletzke A G, et al. Exploring shapelet transformation for time series classification in decision trees. Knowledge-Based Systems, 2016, 112: 80-91

[18] Mueen A, Keogh E, Young N. Logical-shapelets: An expressive primitive for time series classification//Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. San Diego, USA, 2011: 1154-1162

[19] Rakthanmanon T, Keogh E. Fast shapelets: A scalable algorithm for discovering time series shapelets//Proceedings of the 13th SIAM International Conference on Data Mining. Austin, USA, 2013: 668-676

[20] Grabocka J, Schilling N, Wistuba M, et al. Learning time-series shapelets//Proceedings of the 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. New York, USA, 2014: 392-401

[21] Karlsson I, Papapetrou P, Boström H. Generalized random shapelet forests. Data Mining and Knowledge Discovery, 2016, 30(5): 1053-1085

[22] Rakthanmanon T, Campana B, Mueen A, et al. Searching and mining trillions of time series subsequences under dynamic time warping//Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. Beijing, China, 2012: 262-270

[23] Berndt D J, Clifford J. Using dynamic time warping to find patterns in time series//Proceedings of the KDD Workshop. Seattle, USA, 1994: 359-370

[24] Ratanamahatana C A, Keogh E. Everything you know about dynamic time warping is wrong//Proceedings of the 3rd Workshop on Mining Temporal and Sequential Data. Seattle, USA, 2004: 1-11

[25] Keogh E, Ratanamahatana C A. Exact indexing of dynamic time warping. Knowledge and Information Systems, 2005, 7(3): 358-386

[26] Cuturi M, Blondel M. Soft-DTW: A differentiable loss function for time-series//Proceedings of the 34th International Conference on Machine Learning. Sydney, Australia, 2017: 894-903

[27] Demšar J. Statistical comparisons of classifiers over multiple data sets. Journal of Machine Learning Research, 2006, 7(1): 1-30

WANG Zhi-Hai, born in 1963, Ph. D., professor, Ph. D. supervisor. His research interests include data mining and machine learning.

ZHANG Wei, born in 1987, Ph. D. candidate. His research interests include data mining and machine learning.

YUAN Ji-Dong, born in 1989, Ph. D., lecturer. His research interests include data mining and pattern recognition.

LIU Hai-Yang, born in 1987, Ph. D. candidate. His research interests include data mining and pattern recognition.

Background

Time series classification has received great attention in recent years. We can get massive time series data in many fields, such as weather forecast, malware detection, voltage stability assessment, medical monitoring, and anomaly detection. Time series is usually composed of a set of ordered real data, which is usually obtained by observing a certain process at a certain time interval. Time series is different from the traditional attribute vector data, and it has no explicit attributes. Even with the sophisticated feature selection techniques, the dimensionality of potential features is difficult to be reduced to a reasonable range. This poses a challenge to time series classification.

The goal of time series classification is to find a function to predict the class value of the time series. In the study of time series classification problem, except the accuracy, we think highly of the interpretability of model. Since the time series have no explicit features, it is difficult to conduct an interpretable time series classification model. Different scholars have studied this issue. A new concept of time series shapelet has been put forward. The shapelet is a subsequence of a times series that can be used to determine the class value. The shapelet-based model can give us an insight to the data. So, the time series classification model based on shapelets has received much attention. In this paper, we emphasize importance to the local characteristics of the instances to be classified. The lazy learning strategy is combined with the local feature extraction technique. On this basis, a data driven model based on the shapelets for each test case is proposed, which can improve the classification accuracy whilst the shapelets obtained in the model can directly reflect the salient local features of the test case.

This work is supported by the National Natural Science Foundation of China (Nos. 61672086, 61702030, 61771058), the China Postdoctoral Science Foundation (2018M631328), the Fundamental Research Funds for the Central Universities (2017YJS036) and the Beijing Natural Science Foundation (4182052). These projects are devoted to the study of time series classification algorithms closely related to practical applications. Until now, our team has published more than 40 research papers. The research directions involved in these papers include time series classification, data stream mining and Bayesian networks learning.