# shapelet

# some thoughts

kaikaifeng*

BUAA

2017-07-29

*E-mail: kaikaifengqh@163.com，Student ID: xxxxxxxx

# 1 shapelet——时间序列分类

## 1.1 简单综述

时间序列分类问题（Time-series classification or TSC）的困难之处在于如何度量序列之间的相似性。传统的分类问题中，属性的顺序通常是不重要的；然而对于时间序列，数据的排序对于找到好的分类特征至关重要。对TSC的研究（有一部分）集中在寻找最近邻（Nearest Neighbor or NN）分类器使用的距离度量。实际上，在小数据集上，NN（虽然看上去naive）效果“很好”：”There is a plethora of classification algorithms that can be applied to time series; however, all of the current empirical evidence suggests that simple nearest neighbor classification is very difficult to beat” [Batista et al.(2011)]。

最近邻分类器适用于那种通常的时域曲线1.1。曲线的（潜在的）基本形状的变动被认为是观察时引入的噪声导致的。由噪声导致的相位变化较小。

可以不太准确地说，最近邻以及其他关注完整曲线的方法是“全局方法”。

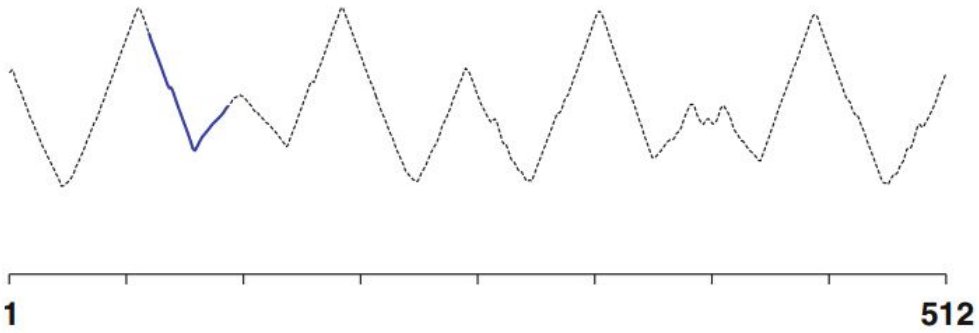


图 1: 全局匹配 [Hills et al.(2013)]

## 1.2 shapelet——关注局部

shapelet方法不直接关注时间序列的全局特征，而试图寻找序列间局部的相似性。一个shapelet的简介定义是：”A shapelet is a time-series subsequence that can be used as a primitive for TSC based on **local**, phase-independent similarity in shape” [Hills et al.(2013)]

shapelet的基本思想也并不复杂，如图1.2所示，一个shapelet应能够和训练集中的一部分时间序列的某些连续子序列匹配地很好，然而在剩余的时间序列中却不能找到匹配良好的连续子序列。

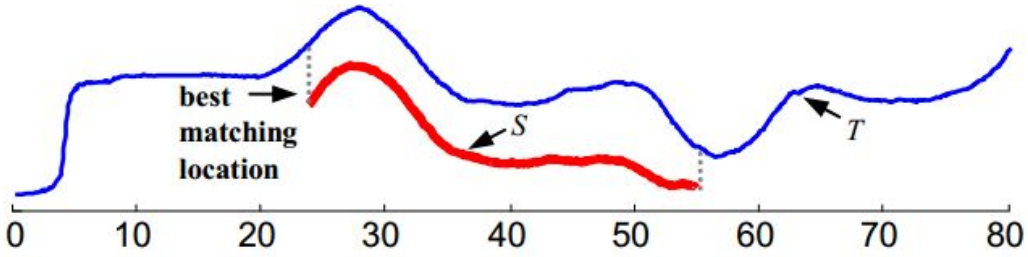


**Figure 5: Illustration of best matching location in time series *T* for subsequence *S***

图 2: 局部匹配 [Lexiang Ye & Eamonn Keogh(2009)]

下表1说明了将要使用的符号。

表 1: 符号表

| 符号 | 说明 |
|---|---|
| $T, R$ | 时间序列 |
| $S$ | 连续子序列 |
| $m, \|T\|$ | 时间序列长度 |
| $l, \|S\|$ | 连续子序列长度 |
| $\mathbf{D}$ | 时间序列数据集 |
| $A, B$ | 类别标签 |
| $S_k$ | $S$中的第$k$个数据点 |
| $\mathbf{S}_T^{\|S\|}$ | $T$的长度为$\|S\|$的连续子序列集合 |
| $\mathcal{D}$ | 距离序列 |

**定义 1 ：** 时间序列和连续子序列的距离。以$Dist(T, R)$表示两个等长的时间序列$T, R$的距离，时间序列和连续子序列的距离$SubsequenceDist(T, S)$ 定义为：

$$SubsequenceDist(T, S) = min(Dist(S, S^{'})), S^{'} \in \mathbf{S}_T^{|S|}$$

由此可见，单纯的*shapelet*关注的是完全与相位无关的“形状”特征，这还不是我们想要的。

### 1.3 shapelet——分类器

*shapelet*的提出最早是用于分类的。其基本算法如下*1*:

算法首先生成一些候选序列，然后使用信息增益（*Information Gain*）衡量候选序列的分类效果，并找到信息增益最大的候选序列。对集合的划分利用时间序列和连续子

**Algorithm 1** FindingShapeletBF(dataset **D**, *MAXLEN*, *MINLEN*)

```
candidates ← GenerateCandidates(D, MAXLEN,MINLEN)
bsf_gain ← 0
For each S in candidates
    gain ← CheckCandidate(D, S)
    If gain > bsf_gain
        bsf_gain ← gain
        bsf_shapelet ← S
    Endif
Endfor
Return bsf_shaplet
```

序列的距离$SubsequenceDist(T, S)$，如图所示*1.3*，*CheckCandidiate()*函数中，首先计算$\boldsymbol{D}$中所有序列与候选*shapelet*之间的距离，然后找到最佳的划分点，并得到对应的信息增益。

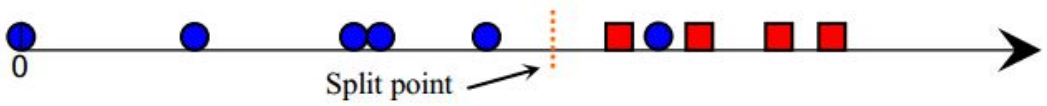


图 3: 划分点

划分点的选取需要遍历得到的距离序列中相邻点对的中点。

由基本算法*1*可以构造决策树，用于时间序列的分类。这是最初提出的*shapelet*用法。

# 2 shapelet——一种embedding方法

## 2.1 什么是embedding?

这个问题看似与*shapelet*无关，然而，所有构造集合间映射的方法都可以成为*embedding*方法；在数学上*embedding*有如下的定义 *[Serge Lang(2002)]*:

定义 **2** :"*A homomorphism* $f : G \longrightarrow G'$ *which establishes an isomorphism between* $G$ *and its image* $G'$ *will also be called an* ***embedding***"

### 2.2 shapelet assessment

上一节中提到的方法使用信息增益衡量*shapelet*的分类能力。使用信息增益导致的主要困难是确定划分点所需的运算量非常大。使用非参数检验替代信息增益可以有效提高计算效率。

#### 2.2.1 Kruskal-Wallis

*KW*检验是一个关于三组或更多数据的非参数性检验，实质是两独立样本的曼-惠特尼*U*检验在多个样本下的推广，也用于检验多个总体的分布是否存在显著差异。给定一个按照升序排序的距离序列$\mathcal{D}$，它依据某种分类被划分为一系列集合$D_1, \ldots, D_C$，再给定$D_i$中元素在$\mathcal{D}$中排序数的集合$R_i$，*KW*统计量定义为：

$$K = \frac{12}{n(n+1)} \sum_{i=1}^{C} |R_i|(\bar{R}_i - \bar{R})^2$$

其中$\bar{R}_i$是类别$i$的平均排序（平均秩），$\bar{R} = \frac{\sum_{i=1}^{n} i}{n}$，上式的化简为：

$$K = \frac{12}{n(n+1)} \sum_{i=1}^{C} \frac{\sum_{r_j \in R_i} r_j^2}{|R_i|} - 3(n+1)$$

#### 2.2.2 Analysis of variance F-statistic

一个好的分类函数（将一定形式的输入转化为数值），应该使得不同类输入对应数值尽可能不同，同时使得同类输入对应数值尽可能相同。也就是，使得组内方差尽可能小，同时使得组间方差尽可能大，据此，*F-statistic*定义如下：

$$F = \frac{\sum_i (\bar{\mathcal{D}}_i - \bar{\mathcal{D}})^2/(C-1)}{\sum_{i=1}^{C} \sum_{d_j \in \mathcal{D}_i} (d_j - \bar{\mathcal{D}}_i)^2/(n-C)}$$

其中$C$为类别数量，$n$为序列数量，$\bar{\mathcal{D}}_i$为类别$i$组内平均，$\bar{\mathcal{D}}$为$\mathcal{D}$的均值

#### 2.2.3 Mood's median

## 3 shapelet——加速方法

### 3.1 Early abandon

由于$SubsequenceDist(T, S)$是最小值，所以当当前距离已经大于当前最小值时即可放弃。

3.2 online normalisation and reordered

# 4 无监督扩展

## 4.1 监督学习方法

上述的方法设计为监督学习方法。对于数据集中的时间序列有分类的标注。但是我们可能不能对于股票数据给出类似的标注。首先，以现有的分类标准标注没有意义；第二，我们人为给出标注就牵扯到解释分类标准的问题。就同涨同跌这个问题而言，*shapelet* 可能更适合直接用作特征提取的方法。在提取特征方面，相比构造同涨同跌矩阵相比，*shapelet*又能够保留单只股票的信息，而不是仅仅记录股票之间的关系。

## 4.2 无监督的shapelet

*shapelet*方法的核心在于找到有好的分类效果的序列。对于这种序列的寻找并不一定依赖标签。对于没有标签的*shapelet*，也有相关的研究 *[Jesin Zakaria et al.(2012)]*中提出了使用无监督方式得到*shapelet*的方法。

*[Jesin Zakaria et al.(2012)]*中对无监督*shapelet(u-shapelet)*给出了如下的定义（翻译成中文）：

**定义 *3*** ：一个无监督*shapelet* $\acute{S}$是时间序列$T$的一个连续子序列。在数据集$\mathbf{D}$中，$\acute{S}$和一个子集$D_A$的$SubsequenceDist$远小于$\acute{S}$和剩余时间序列$D_B$的$SubsequenceDist$，即：

$$SubsequenceDist(\acute{S}, D_A) \ll SubsequenceDist(\acute{S}, D_B)$$

*[Jesin Zakaria et al.(2012)]*通过优化（最大化）如下的统计量得到具有好的分类效果的*shapelet*:

$$gap = \mu_B - \sigma_B - (\mu_A + \sigma_A)$$

其中，$\mu_A, \mu_B$分别是$mean(SubsequenceDist(\acute{S}, D_A))$ 和$mean(SubsequenceDist(\acute{S}, D_B))$，$\sigma_A, \sigma_B$ 分别为$std(SubsequenceDist(\acute{S}, D_A))$和$std(SubsequenceDist(\acute{S}, D_B))$，即均值和标准差

*gap*表示了$D_A$右端和$D_B$左端之间的距离，如*3*。

## 4.3 无监督shapelet求解

*[Jesin Zakaria et al.(2012)]*中使用了一种贪心搜索算法最大化*gap*。对于我们的问题，首先我们不一定要使用*gap*，举例而言，使用$K-Means$确定$D_A, D_B$，然后使用$KW$检验。

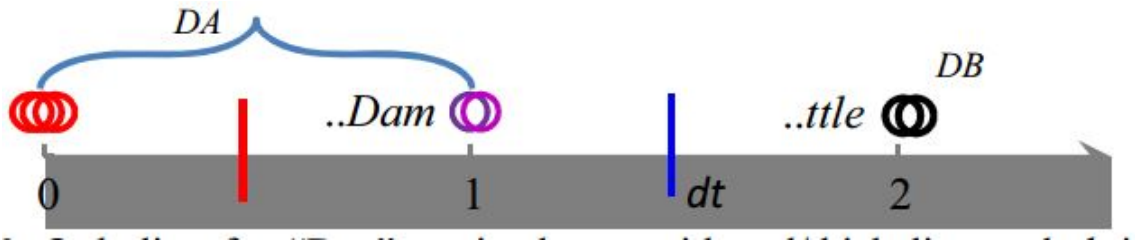


图 4: gap

# 5 后续

## 5.1 股价

*shapelet*特征的提取丢失了相位信息；对于股价时间序列而言，在*2015*年*1*月和*2014*年*1* 月类似的涨幅含义不同，所以在股价上使用*shapelet*时，至少需要记录特征值被提取的位置，对此，一支股票至少得到*2*个向量：特征值向量$E$，对应位置向量$P$。

最简单的后续手法是使用$E, P$聚类，得到完全从数据中取得的股票间相似性的信息。

## 5.2 张量

直觉上，$E, P$应当是共同作用的，所以可以看成是一个张量，这方面只是一个观察，没有完全想清楚（*1%*）。

## 参考文献

*[Batista et al.(2011)] Batista G, Wang X, Keogh E (2011) A complexity-invariant distance measure for time series. Proceedings of the eleventh SIAM conference on data mining (SDM)*

*[Lexiang Ye & Eamonn Keogh(2009)] Lexiang Ye, Eamonn Keogh (2009) Time Series Shapelets: A New Primitive for Data Mining. ACM Knowledge Discovery and Data Mining (KDD)*

*[Hills et al.(2013)] Jon Hills · Jason Lines · Edgaras Baranauskas James Mapp · Anthony Bagnall (2013) Classification of time series by shapelet transformation: Data Mining and Knowledge Discovery (DMKD)*

*[Jesin Zakaria et al.(2012)] Jesin Zakaria, Abdullah Mueen, Eamonn Keogh (2012) Clustering Time Series using Unsupervised-Shapelets: IEEE 12th International Conference on Data Mining (ICDM)*

*[Serge Lang(2002)] Serge Lang (2002) Algebra Revised Third Edition: Springer*