Applied Soft Computing xxx (2015) xxx–xxx

Contents lists available at ScienceDirect

Applied Soft Computing

journal homepage: www.elsevier.com/locate/asoc

Effect of segmentation on financial time series pattern matching

Yuqing Wan, Xueyuan Gong, Yain-Whar Si*

Department of Computer and Information Science, University of Macau, Macao

ARTICLE INFO

Article history:
Received 16 January 2015
Received in revised form
25 September 2015
Accepted 6 October 2015
Available online xxx

Keywords:
Segmentation
Pattern matching
Financial time series
Technical pattern
Head-and-shoulders

ABSTRACT

In financial time series pattern matching, segmentation is often performed as a pre-processing step to reduce the data points from the input sequence. The segmentation process extracts important data points and produces a time series with reduced data points. In this paper, we evaluate the effectiveness and accuracy of four approaches to financial time series pattern matching when used with four segmentation methods, the perceptually important points, piecewise aggregate approximation, piecewise linear approximation and turning points methods. The pattern matching approaches analysed in this paper include the template-based, rule-based, hybrid, decision tree, and Symbolic Aggregate approXimation (SAX) approaches. The analysis is performed twice, on a real data set (of Hang Seng Index prices from the Hong Kong stock market) and on a synthetic data set containing positive and negative cases of a technical pattern known as head-and-shoulders.

1. Introduction

In financial trading, two types of analysis are usually used to predict future price movements. The first type, called “technical analysis”, involves forecasting a trend in the financial market based on historical data such as the daily prices and volumes traded. The historical data are commonly represented in the form of a time series. The second type is called “fundamental analysis”, which is the prediction of price movement based on the financial strength (health) of the company or on developments in the social, political and economic situation. In technical analysis, one of the crucial steps performed by traders is locating interesting patterns in the time series that can help to forecast future price trends. These patterns are commonly referred to as technical patterns (or chart patterns) in the financial domain. Some of the well-known technical patterns include head-and-shoulders (H&S), double top, triple top, cup-with-handle and range breakout [1].

Pattern matching is one of the most important tasks for the analysis of financial time series and many novel pattern matching approaches have appeared in recent years. Zapranis and Tsinaslanidis [2] proposed a new approach based on neural networks to identify a technical pattern known as H&S. Zhou et al. [3] proposed a geometrical similarity measure approach that is invariant to shifting and scaling, and which calculates the angle between two vectors after shift-eliminated transformation. In [4], a kernel regression estimator of a given time series is constructed. The extrema on the original time series is identified based on the local minimum or maximum in the regression line. The extremas in the original time series are then used to determine whether or not a pattern has occurred. Rao and Principe [5] proposed a generalized eigendecomposition algorithm using two step Principal Component Analysis (PCA) process for segmenting speech signals. In [6], Ge et al. proposed an approach to model time series with Hidden semi-Markov Model (HSMM) to detect specific waveform patterns. Symbolic Aggregate approXimation (SAX) [7] allows a time series of arbitrary length to be reduced to a string of arbitrary length. Barnaghi et al. [8] proposed an enhanced SAX which uses K-means clustering method to determine the zones of the symbols. To calculate the similarity measure, Damerau–Levenshtein distance [9,10] is also used to find the edit distance between two strings. Kullback–Leibler divergence [11], Jensen–Shannon divergence [12] or Bhattacharyya distance [13] are also used to compare the difference between two probability distributions.

A number of comprehensive surveys on time series pattern matching are also reported in literature. Fu [14] gives a review on time series data mining and categorizes the time series data mining research into representation, indexing, similarity measure, segmentation and visualization. Xing et al. [15] survey sequence classification methods in terms of methodologies and application domains. A comprehensive survey of control-chart pattern-recognition methods is also reported in [16].

Dynamic time warping (DTW) is one of the most popular similarity measure based approaches for time series pattern matching [17]. Some researchers have applied an extended version of dynamic time warping (DTW) for pattern matching. For instance, Li et al. [18] proposed a novel similarity measure approach based on piecewise linear approximation and derivative DTW. Junkui and Yuanzhen [19] accelerated the DTW process by terminating the calculation earlier when the values of the neighbour cells in the cumulative distance matrix exceeded the tolerance level. Chen et al. [20] proposed a new distance calculation method called DTW-D that combines DTW and Euclidean distance (ED) for time series semi-supervised learning algorithms. In [21], string kernels are used to measure similarity of strings in linear time by using annotated suffix trees. However, computing string kernels using suffix trees does not scale well to problems with large data size [22]. To alleviate this problem, Teo et al. [22] compute string kernels by designing a space efficient and scalable algorithm using Enhanced Suffix Arrays [23]. None of these pattern-matching approaches (DTW, ED, string kernels) require the size of the query pattern to be the same as the size of the sub-sequence.

A number of pattern-matching approaches require segmentation as a pre-processing step. These methods include the template-based (TB) [24], rule-based (RB) [24], hybrid (HY) [25] and decision tree (DT) approaches, which are all discussed in Section 2.4. All of these pattern matching approaches require the size of the query pattern to be the same as the size of the sub-sequence.

To reduce the number of data points in the original time series, segmentation methods have been commonly used as a pre-processing step in time series

* Corresponding author. Tel.: +853 83974454.
E-mail addresses: mb25466@umac.mo (Y. Wan), amoonfana@qq.com (X. Gong), fstasp@umac.mo (Y.-W. Si).

http://dx.doi.org/10.1016/j.asoc.2015.10.012
1568-4946/© 2015 Published by Elsevier B.V.

analyses. These segmentation methods include the perceptually important points (PIP) method [26], the piecewise aggregate approximation (PAA) method [27], the piecewise linear approximation (PLA) method [28] and the turning points (TP) method [29]. Three variations of the PIP method were compared by Fu et al. [24]. They considered the vertical distance PIP (PIP-VD) to be the best choice.

In using these segmentation methods, analysts must consider that each method involves a different process for the selection of data points from the input time series. Therefore, the resulting time series can be significantly different, depending on the segmentation method used. Such differences in segmented time series can have a profound effect on the pattern matching results. A number of studies have been conducted on the effects of segmentation on pattern matching methods. Chen et al. [30] compared the PIP-based evolutionary pattern discovery approach with the discrete wavelet transformation (DWT) approach (which is based on the pattern discovery approach). Their proposed approach solved the problems of information loss, distortion of segments and generation of meaningless patterns that had been associated with the DWT-based approach. Fu et al. [24] compared efficiency and effectiveness among several pattern matching approaches, namely the TB approach (which uses the PIP-VD segmentation method), the RB approach (which also uses the PIP-VD) and the PAA-based approach proposed by Keogh et al. [28]. The two approaches based on PIP performed better than the PAA-based approach. The TB approach provided an effective method, but the RB approach showed a better ability to describe query patterns. Zhang et al. [25] compared the processing time and accuracy of the proposed HY approach with that of two other pattern matching approaches the ED-based method and the slope-based method. The experimental results showed that the HY approach was more effective and efficient than the ED-based method or the slope-based approaches.

Comparisons of the PLA, PAA and PIP segmentation methods in terms of on-line use, representation interval and complexity were discussed by Si and Yin [29]. They reported that the PAA approach could be directly used for on-line representation, but the PIP and PLA approaches were unsuitable. PAA is based on segments with identical length calculations, but PIP and PLA are based on the degree of fluctuation in the time series. The complexity rating of PAA is $O(n)$, and the ratings of PIP and PLA are both $O(n^2)$. PAA is similar to the operation of removing redundant information from triangle meshes [31].

Si and Yin [29] also compared a segmentation method based on TPs with two common segmentation methods, PLA and PIP, in terms of capacity for reconstructing error and ability to keep trends. The PLA approach produced the least amount of errors and the fewest trends, and the proposed TP approach preserved more trends than the PLA or the PIP approaches. All of these studies, however, investigated only one or two pattern matching approaches each. To the best of our knowledge, no comparative analysis has been ever performed on all of the well-known methods of data segmentation methods and pattern matching.

In this paper, we evaluate the effectiveness and accuracy of four well-known approaches to pattern matching (the TB, RB, HB and the DT approaches) when used with four segmentation methods as the pre-processing step. These four segmentation methods are the PIP method [26], the PAA method [27], the PLA method [28] and the TP method [29]. We use the technical pattern known as H&S as a query pattern to better understand how these segmentation methods affect the pattern matching approaches.

The remainder of the paper is organised into four sections. We briefly review the algorithms used for segmentation and pattern matching in financial time series in Section 2. In Section 3, we report the experimental results obtained from evaluation of segmentation and pattern matching algorithms applied to price data from the Hong Kong stock market. In Section 4, we summarise our findings and discuss directions for future research.

2. Segmentation methods and pattern matching approaches

2.1. Terminology and notation

The term “time series” is defined as an ordered list $T = [(t_1, x_1), (t_2, x_2), \ldots, (t_n, x_n)]$. $Len(T)$ represents the number of points in T. As the value t_i is sequential (e.g., 1, 2,..., n), T can be simplified to $T = [x_1, x_2, \ldots, x_n]$ and T_i is often used to denote the element x_i. Accordingly, the sub-sequence S of T is $T_{i,j} = [x_i, x_{i+1}, \ldots, x_j]$, where i and j are the start and end points of the sub-sequence.

2.2. Segmentation methods

The aforementioned four pattern matching approaches (the TB, RB, HY and DT approaches) all require a pre-processing of the input sequence to reduce the number of data points until the length of the input sequence is the same as the query pattern. The well-known segmentation method, PIP, was first introduced by Chung et al. [26]. The variants of PIP are PIP-ED, PIP-VD and perpendicular distance PIP. Fu et al. [24] considered the PIP-VD method to be the best choice among these three variants in terms of efficiency and effectiveness. Therefore, we choose the PIP-VD method for our experiment. The generic algorithm of PIP is described in Algorithm 1 [24]. With the time series T, the first and the last data point in the time series are the first two PIPs. The third PIP is the point in T with maximum vertical distance to the line joining the first two PIPs. The fourth PIP is the point in T with maximum vertical distance to the line joining its two adjacent PIPs, either between the first and second PIPs or between the second and the last PIPs. This process continues until the length of the segmentation sequence SP is equal to the input sequence Q. An illustration showing the selection of five PIPs from a time series is shown in Fig. 1.

Algorithm 1. Pseudocode of the PIP identification [24]

```
  Function: PIP Identification (T, Q)
Input: sequence T of Len(T) = m, template Q of Len(Q) = n
Output: pattern SP of Len(SP) = n
  Set $SP_1 = P_1$, $SP_n = P_m$
  repeat
    Select point $T_j$ with maximum distance to the adjacent points in SP
  ($SP_1$ and $SP_n$ initially)
    Add $T_j$ to SP
  until all SP are all filled
  return SP
```

The PAA method was proposed by Keogh and Pazzani [27]. In PAA, a time series T of length n is represented by the compressed time series T' of length N. That is, $T = (x_1, \ldots, x_n)$ is represented by $T' = (x'_1, \ldots, x'_N)$. The time series T is divided into N equal-sized parts and each part is represented by the mean value of the data points in that part. The ith element of T' can be calculated by using Eq. (1).

$$x'_i = \frac{N}{n}\sum_{j=s_i}^{e_i} x_j \qquad (1)$$

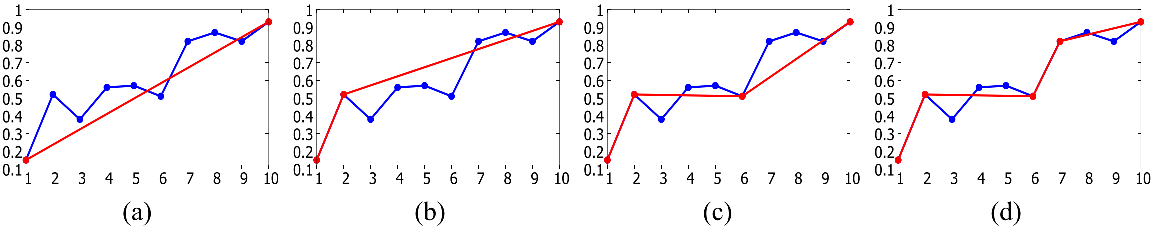
where s_i and e_i denote the start point and end point of the ith part, respectively. An illustration showing the selection of five points with PAA is shown in Fig. 2.

The PLA method uses several straight lines to segment a time series T. PLA can be obtained through the sliding window, top-down or bottom-up methods [28]. In our experiment, we choose the bottom-up method for obtaining PLA. The generic bottom-up algorithm [28] for PLA is described in Algorithm 2. In this bottom-up method, the time series is represented by a number of segments in the first FOR loop. The costs of merging the neighbour segments are calculated in the next FOR loop. In the WHILE loop, the two neighbouring segments with the lowest merge-costs are combined until the minimum merge cost is less than the threshold. In this experiment, the merging process continues until the number of data points in the sequence is equal those in the query pattern. Fig. 3 gives an illustration showing the selection of five points with PLA.

Algorithm 2. The generic algorithm of the PLA-bottom up [28]

```
Function: Seg_TS = Bottom_UP(T, max_error)
for i = 1: 2: Len(T) {Create initial fine approximation.} do
  Seg_TS = concat (Seg_TS, create_segment ($T_{i,i+1}$));
end for
for i = 1: Len(Seg_TS)-1 { Find the cost of merging each pair of segments.}
do
  merge_cost (i) = calculate_error ([merge (Seg_TS(i), Seg_TS(i +1))]);
end for
while min (merge_cost)< max_error {While not finished.} do
  i = min (merge_cost); { Find the cheapest pair to merge.}
  Seg_TS(i) = merge (Seg_TS (i), Seg_TS(i + 1))); {Merge them.}
  delete (Seg_TS(i + 1)); {Update records.}
  merge_cost (i) = calculate_error (merge (Seg_TS(i), Seg_TS(i + 1));
  merge_cost (i-1) = calculate_error (merge (Seg_TS(i-1), Seg_TS(i));
end while
```

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012



Q5

Fig. 1. An illustration of PIP. Five PIPs (shown as red points) are selected from a time series (shown as a blue line) based on Algorithm 1. The selection order is from (a) to (d). The segmentation result is the red line in (d). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

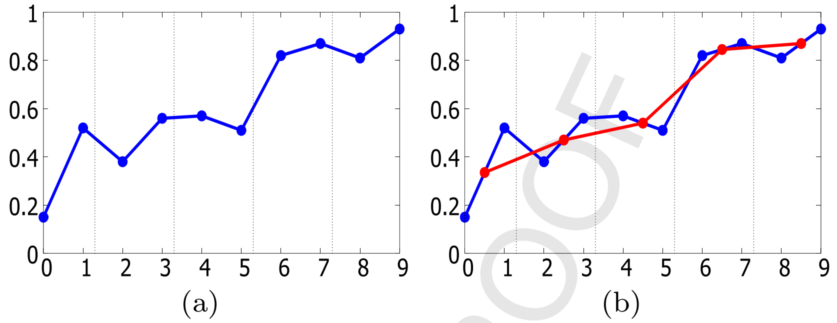


Fig. 2. An illustration of PAA. A time series (shown as the blue line in (a)) is divided into five parts. Each part consists of two points. Each part is then represented by a red point in (b), which is calculated by the mean of the blue points in that part. The segmentation result is the red line in (b). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

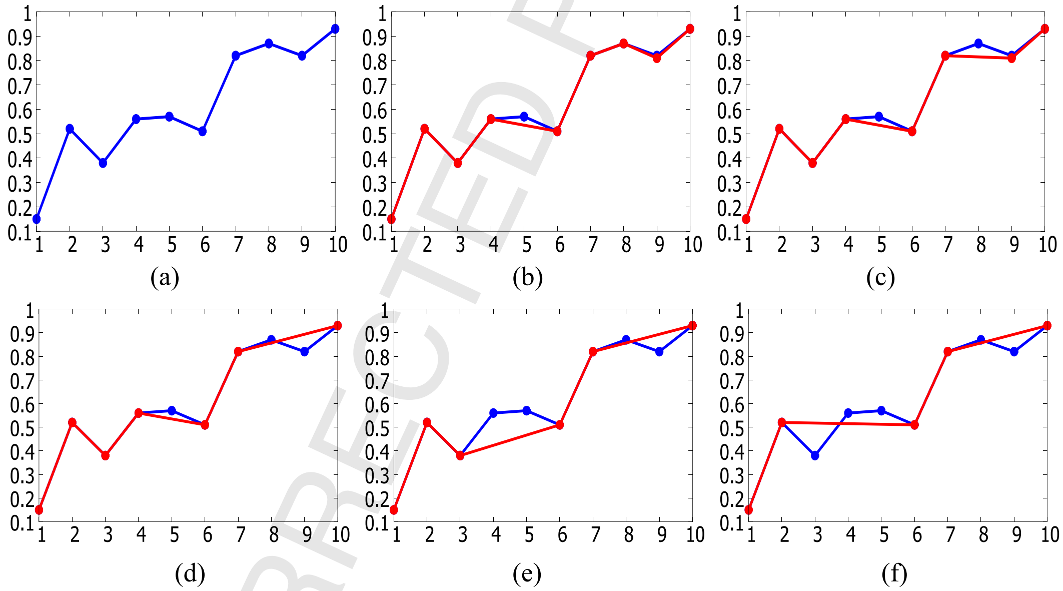


Fig. 3. An illustration of the PLA-bottom up method. Five points are chosen from the original time series (shown as the blue line in (a)). To select five points, the points in the original time series are removed point by point in the order from (b) to (f). In this process the blue points are removed and the red points remain. The segmentation result is depicted as a red line in (f). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

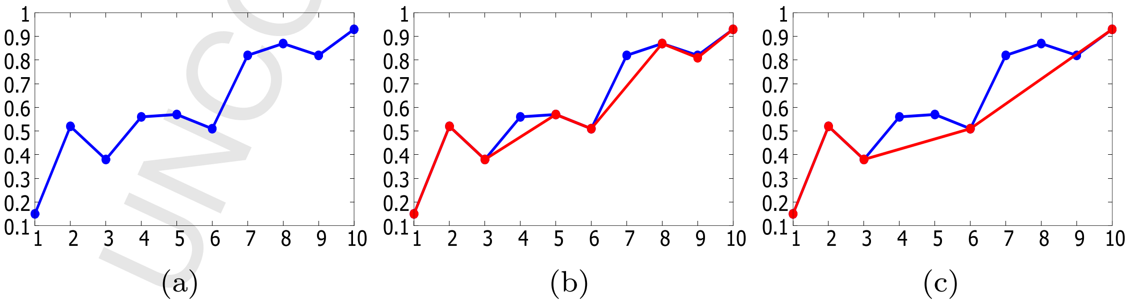


Fig. 4. An illustration of Algorithm 3. The blue line in (a) is the input time series. The red points in (b) are the TPs chosen from the time series. The importance is then assigned to each TP. The five most important points pop out from the stack to form the segmentation result. The red line in (c) is the segmentation result of TP. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

Table 1
The look-up table of breaking points for dividing the Gaussian curve into regions of equal area.

	$\alpha=3$	$\alpha=4$	$\alpha=5$	$\alpha=6$	$\alpha=7$
β_1	−0.43	−0.67	−0.84	−0.97	−1.07
β_2	0.43	0	−0.25	−0.43	−0.57
β_3		0.67	0.25	0	−0.18
β_4			0.84	0.43	0.18
β_5				0.97	0.57

The TP method [29] comprises an identification phase and an evaluation phase. The pseudocode for identifying and evaluating TPs is described in Algorithm 3. First, all of the local maximum and minimum points are marked as the TPs from the time series. Next, the importance of all of these points is evaluated. For retrieval, all of the TPs are then stored in a stack according to their importance. In our experiment, a certain number of TPs are pop out from the stack to represent a sequence. An illustration showing the selection of five TPs from a time series is shown in Fig. 4.

Algorithm 3. Pseudocode for identifying and evaluating TPs [29]

```
  Function: TPs_identification_phase
Input: T[1..n]
Output: TP[1..m]
  for each point i in T[1..n] do
    if T[i-1] < T[i] and T[i] > T[i+1] then
      put T[i] into TP
    end if
    if T[i-1] > T[i] and T[i] < T[i+1]
      put T[i] into TP
    end if
  end for
  Function: TPs_evaluation_phase
Input: Sequence TP[1..m]
Output: TPstack TPS[1..m]
  Calculate each TPs importance value.
  repeat
    Push the TP with lowest importance value into the stack TPS.
    if a point that is no longer minimum or maximum is found in TP[1..m]
    then
      Push it into TPS
      Update the neighbour TPs importance
  end if
  until all point in TPs are pushed into TPS
  return TPS
```

2.3. Symbolic aggregate approximation

Symbolic Aggregate approXimation (SAX) [7] transforms a time series into a sequence of symbols. A time series is firstly segmented using PAA into several parts and then each part is represented by a symbol. The total number of symbols (α) to represent a time series is determined in advance. Breakpoints are a sorted list of numbers $B=\beta_1, \ldots, \beta_{\alpha-1}$ such that the area under a standard Gaussian curve from β_i to $\beta_{i+1}=1/\alpha$ where β_0 and β_α are defined as $-\infty$ and ∞, respectively.

After the PAA step, each part segmented by PAA in the time series is represented by a mean value. These mean values are mapped into the regions in the look-up table and represented by the symbol in the corresponding region. The look-up table of breaking points which divide the Gaussian curve into regions of equal area is shown in Table 1. For example, to represent a time series with 3 characters (i.e. when $\alpha=3$), we divide the Gaussian curve into 3 regions of equal area. As shown in the Table 1, two break points β_1 and β_2 are needed and each region is represented by a symbol. The mean values within the three intervals $(-\infty, -0.43]$, $(-0.43, 0.43]$ and $(0.43, \infty)$ can now be represented as symbol A, B and C, respectively.

After the PAA step, two time series $P=p_1, \ldots, p_n$ and $Q=q_1, \ldots, q_n$, are transformed into symbolic sequences $P\prime = p\prime_1, \ldots, p\prime_w$ and $Q\prime = q\prime_1, \ldots, q\prime_w$ where n and w are the lengths of the time series and the symbolic sequence. The minimum distance between two symbolic sequences is calculated by [32]:

$$MINDIST(P\prime, Q\prime) = \sqrt{\frac{n}{w}}\sqrt{\sum_{i=1}^{w}(dist(p_i, q_i))^2} \qquad (2)$$

where dist(·) is a function for calculating the distance between two symbols and it can be defined as:

$$dist(c_i, c_j) = \begin{cases} 0 & |i-j| \le 1 \\ \beta_{j-1} - \beta_i & i < j-1 \\ \beta_{i-1} - \beta_j & i > j+1 \end{cases} \qquad (3)$$

2.4. Pattern matching approaches

Most pattern matching approaches (such as the TB, RB, HY and DT approaches) can perform their operations after the time series are segmented to reduce the data points.

The TB approach [24] measures the similarity between the defined pattern templates and the sequences by calculating their point-to-point amplitude distance (AD) and temporal distance (TD). These distances are conditional expected deviations. The pattern template and the segmented sequence may have different amplitudes. Therefore, the data points need to be rescaled so that the comparison between sequences in different amplitudes can be performed. Rescaling can be performed by normalizing all the sequence values to a given range. The amplitude distance (AD) is defined as:

$$AD(SP, Q) = \sqrt{\frac{1}{n}\sum_{k=1}^{n}(sp_k - q_k)^2} \qquad (4)$$

To address the horizontal distortion of the segmented sequence against the pattern templates, rescaling of the time dimension should be performed. The temporal distance (TD) is defined as:

$$TD(SP, Q) = \sqrt{\frac{1}{n-1}\sum_{k=2}^{n}(sp_k^t - q_k^t)^2} \qquad (5)$$

The similarity measure is

$$D(SP, Q) = w1 \times AD(SP, Q) + (1 - w1) \times TD(SP, Q) \qquad (6)$$

where Q denotes the pattern template and SP denotes the segmented sequence. Variable $w1 \in [0, 1]$ is the weight for compensating between vertical and horizontal variations among different kind of patterns. The variables sp_k and q_k denote the points in SP and the pattern template, respectively. The variables sp_k^t and q_k^t denote the time coordinates of the points sp_k and q_k. As with the weights configuration described by Fu et al. [24], we set $w1 = 0.5$ for our experiment.

The RB approach [24] uses predefined rules to identify patterns. A sequence is recognised as a matching pattern if its segmented sequence complies with the rules of a given pattern. The rules for the H&S pattern are defined as follows:

$sp_4 > sp_2$ and sp_6
$sp_2 > sp_1$ and sp_3
$sp_6 > sp_5$ and sp_7
$sp_3 > sp_1$
$sp_5 > sp_7$
$\text{diff}(sp_2, sp_6) < 15\%$
$\text{diff}(sp_3, sp_5) < 15\%$

sp_k (k =1, 2, 3, 4, 5, 6, 7) denotes the seven points of H&S.

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

Table 2
Rules for the DT approach.

	P1	P2	P3	P4	P5	P6	P7
Rule 1	6	4	2	1	3	5	7
Rule 2	7	4	2	1	3	5	6
Rule 3	6	5	2	1	3	4	7
Rule 4	7	5	2	1	3	4	6
Rule 5	6	4	3	1	2	5	7
Rule 6	7	4	3	1	2	5	6
Rule 7	6	5	3	1	2	4	7
Rule 8	7	5	3	1	2	4	6

The HY approach [25] first calculates the Spearman's correlation coefficient of the sequence. The Spearman's correlation coefficient is calculated by

$$\rho = 1 - \frac{6 \times \sum_{i=1}^{n} d_i^2}{n(n^2 - 1)} \tag{7}$$

The variable d_i is the difference between the predefined rank of a data point on the template pattern and the rank of a corresponding data point on the segmented sequence. The Spearman's correlation coefficient of the sequence is compared with the threshold of eight technical patterns (H&S, double top, triple top, spike top, H&S reversed, double top reversed, triple top reversed and spike top reversed) to get the post patterns. The post patterns that pass the rules of the given patterns are identified as the matching patterns. In our experiment, we test the similarity between the sequence and the technical pattern H&S (H&S). The rules of this technical pattern are defined as follows:

Rule 1: $\left|SP2\text{–}SP6\right| < 15\%$,
Rule 2: $\left|SP3\text{–}SP5\right| < 15\%$,
Rule 3: the ranking of SP4 is first,
Rule 4: the ranking of SP2 and SP6 must be 2 and 3,
Rule 5: the ranking of SP1 and SP7 must be 5 or 6 or 7.

SPk (k = 1, 2, 3, 4, 5, 6, 7) denotes the ranks of the seven points in H&S.

As in the RB and HY approaches, the relative positions of the points on the pattern can be used to define the rules. Based on these rules, a DT approach is proposed in this paper. From our testing of the previously proposed pattern matching approaches, we find that the relative position of the points in each pattern can be adopted to define the rules. For example, there are seven points in H&S. The fourth point must be the highest point among the six. Therefore, the relative position of the fourth point must be 1. By analysing the possibilities for the relative positions of each point in H&S, we can define the rules as shown in Table 2.

In Table 2, each rule represents one possible combination of seven points, and $P1, P2, \ldots, P7$ denote the first point, second point, . . ., seventh point and so on. The number in each entry in the table represents the relative position of that point. For example, 1 represents the highest point, and 7 represents the lowest point.

After defining these eight positive cases, we define the negative cases for training a DT. For generating negative cases, we simply generate all permutations of 1, 2, . . ., 7. Except for the eight rules shown in Table 2, all other generated rules are defined as negative training examples. After these positive and negative cases are created, a DT can be trained.

There are other pattern matching methods which do not require segmentation as a pre-processing step. These methods include the Euclidean distance (ED) and the dynamic time warping (DTW) approaches. Both of these approaches measure the similarity between the original time series and the query sequence directly, without segmentation. In the following paragraphs, we briefly review these two approaches. A pattern matching approaches for symbolic strings and a similarity measure between two probability distributions are also reviewed.

The ED approach measures the similarity of time and query sequences by calculating the point-to-point ED between the two sequences. This distance between the two sequences $X(x_1, \ldots, x_n)$ and $Y(y_1, \ldots, y_n)$ can be represented by the following equation [33]:

$$ED(X, Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2} \tag{8}$$

The two sequences are said to be similar if the ED is less than the threshold.

Berndt and Clifford [17] proposed DTW to measure the similarity between time series with different lengths and to match the similar sequences that are out of phase. Given two sequences $X(x_1, \ldots, x_n)$ and $Y(y_1, \ldots, y_m)$, an n-by-m matrix M is constructed. The elements $d(x_i, y_j)$ in the matrix represent the ED between the points x_i and y_j. The warping path $W = w_1, w_2, \ldots, w_k$ $(max(m, n) \leq K < m + n - 1)$ is a neighbouring set of elements in M. The warping path follows three constraints, that is, the boundary conditions, continuity and monotonicity. The boundary conditions are $w_1 = (x_1, y_1)$ and $w_K = (x_n, y_m)$. Continuity means that given $w_k = (a, b)$, then $w_{k-1} = (a', b')$, where $a - a' \leq 1$ and $b - b' \leq 1$. Monotonicity means that $a - a' \geq 0$ and $b - b' \geq 0$. The optimal warping path $DTW(x, y)$ is defined as

$$DTW(x, y) = \min \sqrt{\sum_{k=1}^{k=K} w_k} \tag{9}$$

The optimal warping path $DTW(x, y)$ that minimises the warping cost is calculated by dynamic programming. The cumulative distance γ(i, j) is defined as the distance $d(x_i, y_j)$, which is found in the current cell, and the minimum of the cumulative distances from the adjacent elements.

$$\gamma(i, j) = d(x_i, y_j) + \min\{\gamma(i-1, j-1), \gamma(i-1, j), \gamma(i, j-1)\} \tag{10}$$

Damerau–Levenshtein distance [9,10] can be used to calculate the edit distance between two strings (i.e. symbolic sequences). The edit distance between two strings A and B is defined as the minimum number of edit operations needed in converting strings A into B or vice versa. Edit operations are defined as insertion, deletion, substitution and transposition in Damerau-Levenshtein distance. Given two strings A and B of length m and n $(m \leq n)$, the edit distance between A and B is computed by filling an $(m+1) \times (n+1)$ matrix D. Each cell $D(i, j)$ in matrix D is filled by following recurrences [34]:

$$D[i, 0] = i \; D[0, j] = j$$

$$D[i, j] = \begin{cases} D[i-1, j-1], \quad \text{if} \quad A[i] = B[j]. \\ D[i-1, j-1], \quad \text{if} \quad A[i-1..i] = B^R[j-1..j] \quad \text{and} \quad D[i-1, j-1] > D[i-2, j-2]. \\ 1 + \min(D[i-1, j-1], D[i-1, j], D[i, j-1]), \quad \text{otherwise.} \end{cases} \tag{11}$$

where $A[i]$ denotes the ith character in A, $A[i..j]$ denotes the substring of A and the superscript R denotes the reverse string. The cell $D[m+1, n+1]$ is the edit distance between A and B. A financial time series is represented by real numbers. Therefore, we need to transform a financial time series to a symbolic sequence if we apply the Damerau–Levenshtein distance to calculate the similarity.

Kullback–Leibler divergence [11], Jensen–Shannon divergence [12] and Bhattacharyya distance [13] are commonly used to compare the difference between two probability distributions in information theory and statistics. In [35], the similarity between two time series is calculated by the Kullback–Leibler distance between the transition matrixes of the two time series. To obtain the transition matrix of a time series, the time series should be transformed into a Markov chain. Given a time series $x = (x_0, \ldots, x_i, \ldots)$, where each x_i belongs to one of the m states of a variable X, the generating process of the sequence is a Markov chain if the conditional probability that the variable x_t visits state i at time t is independent of $(x_0, \ldots, x_{t-2})$, that is, $p(x_t = i|(x_0, \ldots, x_{t-1})) = p(x_t = i|x_{t-1})$. A Markov chain is represented by the $m \times m$ transition matrix P and each element in P is $p_{ij} = p(x_t = i|x_{t-1} = j)$. The Bayesian estimate of p_{ij} can be defined as follows [35]:

$$\widehat{p_{ij}} = \frac{\alpha_{ij} + n_{ij}}{\alpha_i + n_i} \tag{12}$$

$$n_i = \sum_j n_{ij} \tag{13}$$

$$\alpha_i = \sum_j \alpha_{ij} \tag{14}$$

where n_{ij} denotes the transition frequency from state i to state j and $\alpha_{ij} = \frac{1}{m}$ is a hyper-parameter. In [36], Kullback–Leibler distance is used to calculate the distance between two categorical sequence and each time series can be represented as a sequence of states (i.e. a Markov chain). For two transition probability matrix $P1$ and $P2$ for two Markov chains, let $P1_{ij}$ and $P2_{ij}$ denotes the probabilities of the transition from state i to state j in $P1$ and $P2$. The asymmetric Kullback–Leibler distance of $P1$ and $P2$ is computed by [35]:

$$d(P1_i, P2_i) = \sum_{j=1}^{m} P1_{ij} \log \frac{P1_{ij}}{P2_{ij}} \tag{15}$$

The symmetric version of Kullback–Leibler distance is defined as:

$$D(P1_i, P2_i) = \frac{d(P1_i, P2_i) + d(P2_i, P1_i)}{2} \tag{16}$$

The average distance between P1 and P2 is:

$$D(P1, P2) = \sum_i \frac{D(P1_i, P2_i)}{m} \tag{17}$$

3. Experiments

We evaluate the performance of the four segmentation methods with a synthetic data set and a real data set. For the synthetic data set, positive and negative cases of the H&S pattern are generated. Based on the synthetic positive and negative cases of H&S, we test the performance of the four approaches to pattern matching when each of them is paired with the four segmentation methods.

The above-described experiment is then repeated with a real data set. For this experiment, a sliding window is used that is shifted one data point at a time to get a sub-sequence. Next, four segmentation methods are applied to segment the sub-sequence. The segmentation results of the same time series can be varied, depending on the segmentation method used. The resulting four segmented sub-sequences are then tested with each of the pattern matching approaches. As the H&S pattern has seven data points, the length of the whole sequence in the synthetic data set and the sub-sequences in the real data set must be reduced to seven data points after segmentation. In addition, for the PAA segmentation method, the input time series should be divided into seven equal-sized parts. Therefore, we set the length of the sequence in the synthetic data set and the window size of the sliding windows in terms of integer values that are multiples of seven.

3.1. Synthetic data experiment

To generate synthetic data, we adopt the methods proposed by Fu et al. [24] and Zhang et al. [25]. There are three steps in the generation process, time scaling, time warping and noise adding. Time scaling alters the length of the pattern to generate different sub-sequences with different lengths. Time warping changes the positions of important points and therefore the sub-sequence looks warped compared to the pattern. Noise adding alters the value of each point to generate sub-sequences with smaller fluctuations. The pseudocode is given in Algorithm 4.

Algorithm 4. The algorithm for generating synthetic data with H&S

```
Time Scaling
Input a pattern $P$ (the length of $P$ is $n$) and a number $m$ ($m$ = 49, 91 and 133
  in our experiments)
for each two adjacent $P_i$ and $P_{i+1}$ do
  $X = (m - n) \div (n - 1)$
end for
Time Warping
Input a pattern $P$
for each critical point $P_i$ do
  Change the position of $P_i$ between $P_{i-1}$ and $P_{i+1}$ randomly
end for
Noise Adding
Input a pattern $P$
for each point $P_i$ do
  Generate a probability $R$ randomly
  if $R$ < threshold (0.5 is used in this paper) then
    Generate a random value $A$ ($A \in [0, 0.3]$)
    $P_i = P_i + A\,(P_{i+1} - P_i)$
  end if
end for
```

We extend the methods proposed by Fu et al. [24] and Zhang et al. [25], as we consider that negative cases are also important for testing accuracy. Note that although it is possible to generate sufficient positive cases, it is impossible to generate all possible negative cases, as their numbers could be infinite. Therefore, several negative cases in various forms are generated randomly. To generate negative cases, we revise the time warping and noise adding codes. Specifically, the position of P_i is adjusted based on P_{i-1} and P_{i+1} and the range of parameter A is changed from [0, 0.3] to [0, 3].

Firstly, we generate a H&S template with seven data points. For illustration purpose, we name these seven points as original data points. Fig. 5(a) is a template pattern H&S of seven original data points. The coordinate of the seven original data points are (0, 0), (1, 0.7), (2, 0.3), (3, 1), (4, 0.3), (5, 0.7), (6, 0). Secondly, time warping is performed on the template pattern H&S. We randomly change the positions of the seven original data points based on the time warping step from Algorithm 4. Fig. 5(b) shows that after Time Warping, the coordinates of the seven original data points become (0, 0), (15, 0.7), (19, 0.3), (30, 1), (35, 0.3), (47, 0.7), (48, 0). For example, after time warping step, the second original data point changes its position from (1, 0.7) to (15, 0.7). Thirdly, time scaling is performed on the resulted template. In order to extend the length of template from 7 to 49, we insert 42 data points into the pattern. We name these 42 points as added data points. Fig. 5(c) shows how additional points are inserted between original data points to generate a time series of length 49. We insert 14 points between (0, 0) and (15, 0.7). The Y values of 14 added points are calculated by the

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

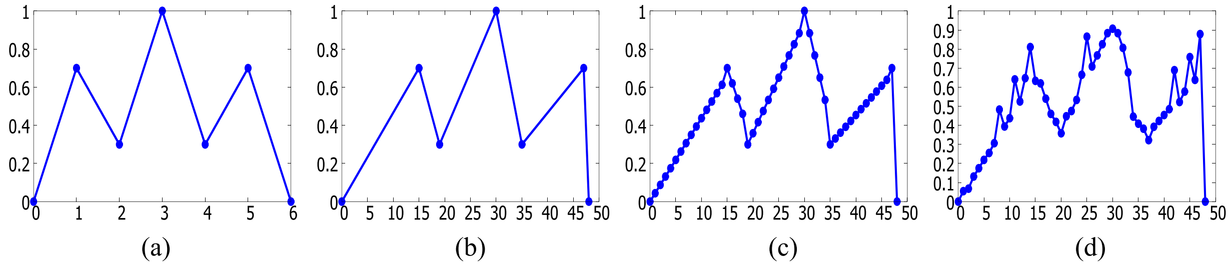


Fig. 5. Steps for generating a synthetic positive test case: (a) The template pattern H&S of 7 original data points. (b) After time warping step. (c) After time scaling step. (d) After noise adding step.

linear function which is formed by the two data points (0, 0) and (15, 0.7). Finally, noise is added to the template. *Y* axis values of all data points in the template are randomly modified and *X* axis values are kept unchanged. The final result is shown in Fig. 5(d).

In our experiment, we create 240 synthetic time series with lengths of 49, 91 and 133. These 240 synthetic time series consist of 120 positive cases (synthetic H&S patterns) and 120 negative cases. The accuracy, precision and sensitivity/recall of these cases are defined in Eqs. (18)–(20).

$$\text{Accuracy} = \frac{Number\ \ of\ \ True\ \ Positive\ \ and\ \ True\ \ Negative}{Number\ \ of\ \ Positive\ \ cases\ \ and\ \ Negative\ \ cases} \tag{18}$$

$$\text{Precision} = \frac{Number\ \ of\ \ True\ \ Positive}{Number\ \ of\ \ Test\ \ Positive\ \ cases} \tag{19}$$

$$\text{Sensitivity/Recall} = \frac{Number\ \ of\ \ True\ \ Positive}{Number\ \ of\ \ Positive\ \ cases} \tag{20}$$

The accuracy, precision and sensitivity of the different segmentation methods are shown in Fig. 6. All four segmentation methods have similarly high precision with each of the pattern matching approaches (see Fig. 6(b)). The figure also illustrates that each pattern matching approach can recognise most of the negative cases correctly, regardless of which segmentation method is used. The correct classification of the positive cases (based on Eq. (20)) is crucial for comparison of the performance of each segmentation method.

In Fig. 6(a) and (c), we can also observe that PIP and PLA have similarly high accuracy/sensitivity with each of the pattern matching approaches. In addition, PIP and PLA have higher accuracy/sensitivity than the other segmentation methods with all four of the pattern matching approaches. PAA has higher accuracy/sensitivity with TB than with the other three pattern matching approaches. TP has low accuracy/sensitivity with all of the pattern matching approaches except for DT.

In Fig. 7, we compare the accuracy of the different segmentation methods for time series of different lengths. PIP and PLA have higher and more stable accuracy than the other segmentation methods for all of the tested lengths in the four pattern matching approaches. PAA has high and stable accuracy with the TB approach. With the other three pattern matching approaches, the accuracy of PAA is stable, but lower than that of PIP or PLA. The accuracy of TP tends to decrease with increases in the length of the time series with each of the pattern matching approaches.

In the following sections, we analyse the performance of each segmentation method. For the sake of illustration, the original results (or the normalised results) of the segmentation methods are depicted with red lines. The original synthetic time series are depicted with blue lines and the normalised H&S pattern is shown with green lines.

3.1.1. Analysis of the PAA method

The PAA method has higher accuracy and sensitivity when used with the TB approach than with the other three pattern matching methods. The reason for this higher performance is that PAA divides the sequence into seven equal-sized parts and every part is represented by a data point that is calculated by the mean value of all of the data points in that part. For the TB approach, TD is one of the contributing factors to the similarity measure. The time coordinates of most of the represented data points in the results of PAA are consistent with the data points in the pattern template. This result leads to the higher accuracy observed when PAA is used with the TB approach for pattern matching.

Note that the TB approach measures the similarity between the template and the input time series according to the absolute positions of the data points in the segmentation results. However, the PAA method cannot accurately retain the up and down trends of the original sequence due to its smoothing effect. Therefore, the accuracy is low when PAA is used with pattern matching approaches that rely on the evaluation of relative fluctuations in the data points for their results in segmenting the time series. The approaches that rely on relative fluctuations of data points include the RB, HY and DT approaches. As the PAA method cannot preserve relative fluctuations, the synthetic time series of H&S (shown in blue in Fig. 8(a)) is identified as a positive case only with the TB approach. The other pattern matching approaches cannot recognise the results of PAA. However, in Fig. 8(b) we can see that the time coordinates of the normalised segmentation result of PAA (shown in red) are consistent with the pattern template (shown in green).

3.1.2. Analysis of the TP method

Unlike the PIP or PLA methods, which take all of the data points in the original sequence into account, the TP method selects seven of the most important TPs to represent a time series. In Fig. 9, we compare the segmentation results of the PIP, PLA and TP methods. The fourth data points of the segmentation results from the PIP (Fig. 9(a)) and PLA (Fig. 9(b)) are representative points for indicating the shape of the synthetic H&S pattern. As the fourth representative point is not a TP, it is ignored by the TP algorithm. Therefore, the results of the TP method cannot embody the shape of the synthetic H&S pattern. For each pattern matching approach, the synthetic positive case (shown in blue in Fig. 9) is identified as a positive case when using the PIP or PLA methods to segment the time series, but it is identified as a negative pattern when using the TP method.

The PIP, PLA and TP methods all select points from the original time series. In many cases the points chosen by PIP and PLA are similar or even identical. The segmentation results of TP, however, are different from the results of PIP or PLA. For example, in Fig. 10, the data points in the results of PIP and PLA are evenly distributed, but the data points in the result from TP are not.

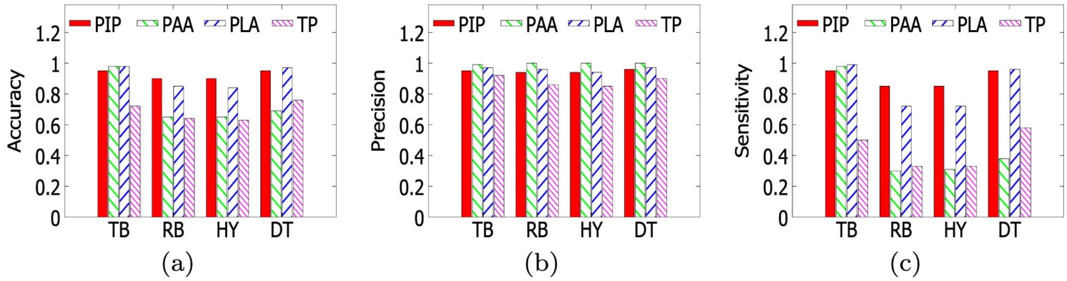


Fig. 6. (a) Accuracy, (b) precision and (c) sensitivity for the four pattern matching methods TB, RB, HY, and DT when they are paired with four segmentation methods PIP, PAA, PLA, and TP.

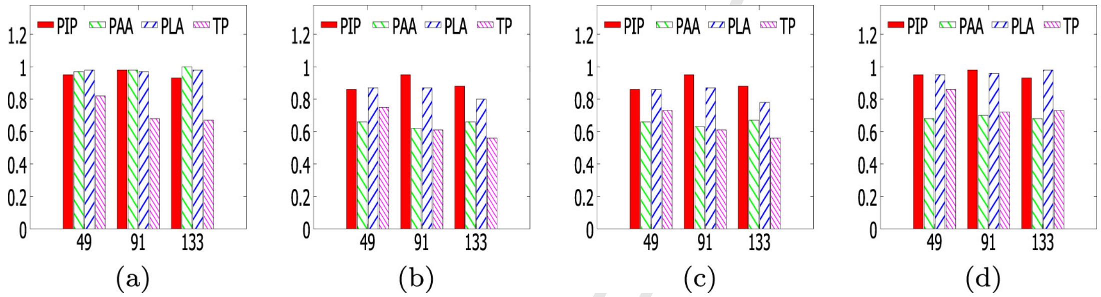


Fig. 7. Accuracy of the four segmentation methods PIP, PAA, PLA, TP with increasing lengths of the time series, using the (a) TB, (b) RB, (c) HY and (d) DT pattern matching approaches.

3.1.3. *Analysis of the PIP and PLA methods*

The PIP and PLA methods both select points from among all of the points in the original time series. Our experimental results show that PIP and PLA have relatively high accuracy and sensitivity with all four of the pattern matching approaches. The PIP and PLA methods are both able to preserve the overall shape of the original time series relatively well. When used with the TB and DT approaches, PLA and PIP have similar accuracy and sensitivity. With the RB and HY approaches, however, PIP has a slightly higher accuracy and sensitivity than PLA. PLA works better than the other segmentation methods with the TB and DT approaches. PIP works well with all of the pattern matching approaches and it works better than any other segmentation method with the RB and HY approaches.

The segmentation results of PIP and PLA and the normalised results of PLA are show in Fig. 11. For the RB and HY approaches, the synthetic positive H&S time series (shown in blue) is identified

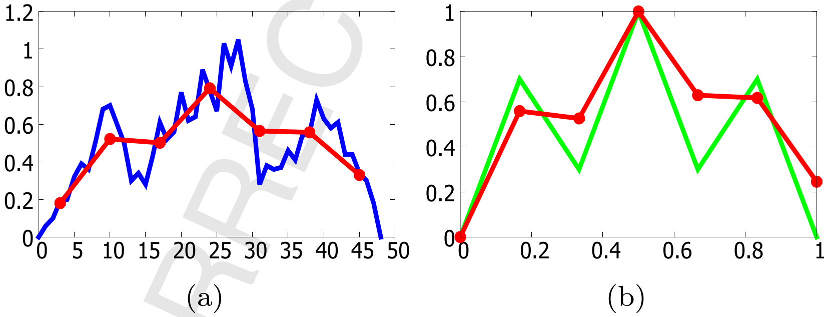


Fig. 8. (a) The result of PAA (shown in red) on the synthetic H&S time series (shown in blue). (b) The normalised result of PAA (in red) on the H&S pattern template (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

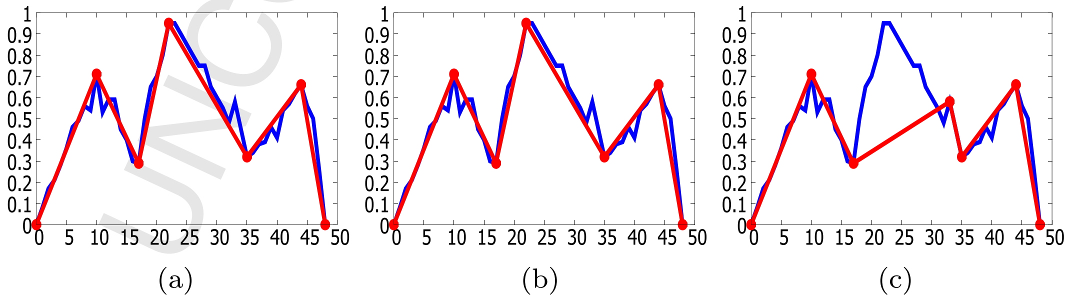


Fig. 9. The results of (a) PIP, (b) PLA and (c) TP (all shown in red) on the synthetic H&S time series (shown in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

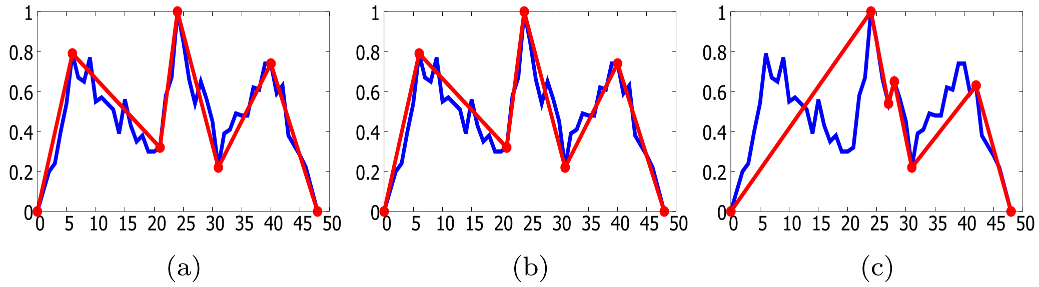


Fig. 10. The results of (a) PIP, (b) PLA and (c) TP (all shown in red) on the synthetic H&S time series (shown in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

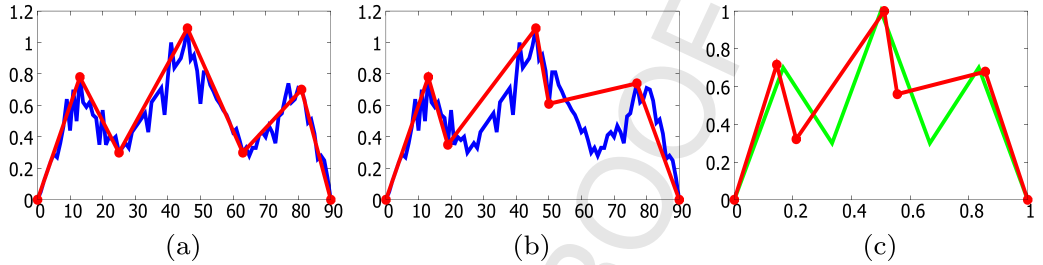


Fig. 11. The results of (a) PIP and (b) PLA (both shown in red) on the synthetic H&S time series (shown in blue) (c) The normalised result of PLA (in red) on the H&S pattern template (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

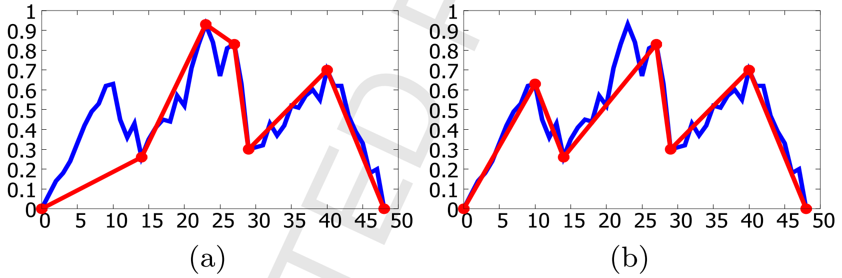


Fig. 12. The results of (a) PIP and (b) PLA (both shown in red) on the synthetic H&S time series (shown in blue). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

as a positive case when using PIP to segment the time series, but this time series is returned as a negative pattern when using PLA. In the TB approach, the synthetic positive H&S time series is identified as a positive case when PIP and PLA are used for segmentation. In Fig. 11(c), the data points in the results of normalised PLA are distributed evenly and most of the data points absolute positions are consistent with the pattern template in TB. However, the relative positions of the data points fail to satisfy the rules in RB and HY. In our experiment, we find that the PIP method almost always works well, except for one or two cases. One such exception is shown in Fig. 12. When PLA is used for segmentation, the synthetic positive time series is considered as a positive case with all of the pattern matching approaches. However, this time series is returned as a negative pattern when PIP is used for segmentation.

3.2. Experiment with SAX on synthetic data

In our experiment, we create 240 synthetic time series with lengths of 49, 91 and 133. Each length has 80 time series with 40 positive cases and 40 negative cases. Altogether, there are 240 synthetic time series including 120 positive cases (synthetic H&S patterns) and 120 negative cases. Recall that H&S template pattern has 7 data points. For our experiment, we assign the first and last data points as symbol A, the second and the sixth data points as symbol C, the third and the fifth data points as symbol B. The fourth data point is represented by symbol D. Therefore, the template pattern H&S can be represented by ACBDBCA. In the synthetic data experiment, each time series in the dataset is transform into sequences of symbols of length 7 with 4 characters. If the minimum distance (MINDIST) is equal to zero, then the time series is recognized as a positive example of H&S. In the experiment of testing the accuracy in recognizing H&S pattern with SAX approach, we use the MATLAB code available at [37]. Note that the minimum distance (MINDIST) calculated between symbolic sequence ABBA and ABBB is zero because the algorithm cannot distinguish between two strings that differ only in the fourth place. Actually, most of the positive H&S symbol sequences are not exactly the same as the symbolic sequence of the defined template pattern H&S ACBDBCA. Usually, they may have one or two symbols which are different from the template. We conduct experiments on these two situations. In Case 1, a time series is recognized as a positive H&S when the minimum distance (MINDIST) of the two symbolic sequences of the subsequence and the template pattern H&S is zero. In Case 2, a time series is recognized as a positive H&S when its symbolic sequence is identical to the template pattern H&S ACBDBCA.

As shown in Fig. 13, the experiment on Case 1 has higher accuracy and sensitivity than Case 2. In Case 1, 221 true positive and true negatives time series are identified out of 240 synthetic time series. In Case 2, 134 true positive and true negatives time series

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

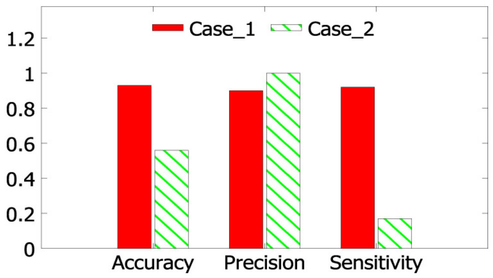


Fig. 13. The accuracy, precision, sensitivity of two cases.

are identified out of 240 synthetic time series. For precision, all the negative synthetic time series are accurately recognized in Case 2, while 109 out of 120 negative synthetic time series in Case 1. For sensitivity, in Case 1, SAX approach can recognize 112 out of 120 positive synthetic time series, while in Case 2 only 14 out of 120 positive synthetic time series are accurately recognized.

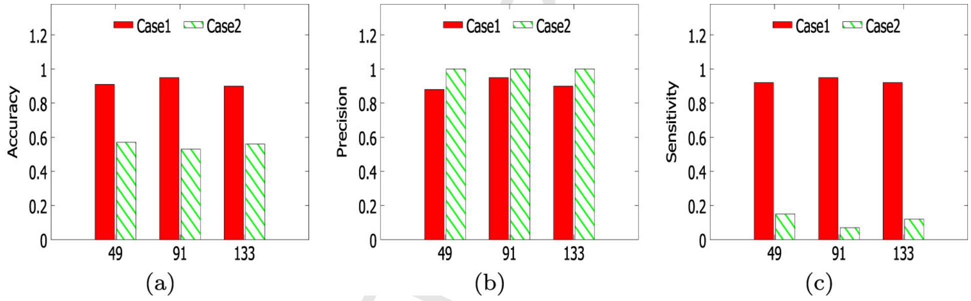
Figure 14 shows the accuracy, precision and sensitivity of two cases with increasing window size. From the experiment results, we can observe that the window size has no significant effect on the accuracy of both cases. We also find that the precision and sensitivity are not affected by the window size.

Table 3
The number of patterns found with different window sizes. Q6

	PIP	PLA	TP	*PAA*		PIP	PLA	TP	PAA
Window size: 35					Window size: 49				
TB	9	6	10	8	TB	8	5	4	1
RB	3	3	6	1	RB	2	1	5	0
HY	4	3	6	1	HY	3	1	6	0
DT	5	4	10	2	DT	5	2	7	0
Total	21	16	32	12	Total	18	9	22	1
Window size: 63					Window size: 91				
TB	5	4	3	2	TB	4	4	3	3
RB	3	3	4	0	RB	1	2	2	0
HY	3	3	5	0	HY	1	2	2	0
DT	3	3	6	0	DT	3	3	6	0
Total	14	13	18	2	Total	9	11	13	3

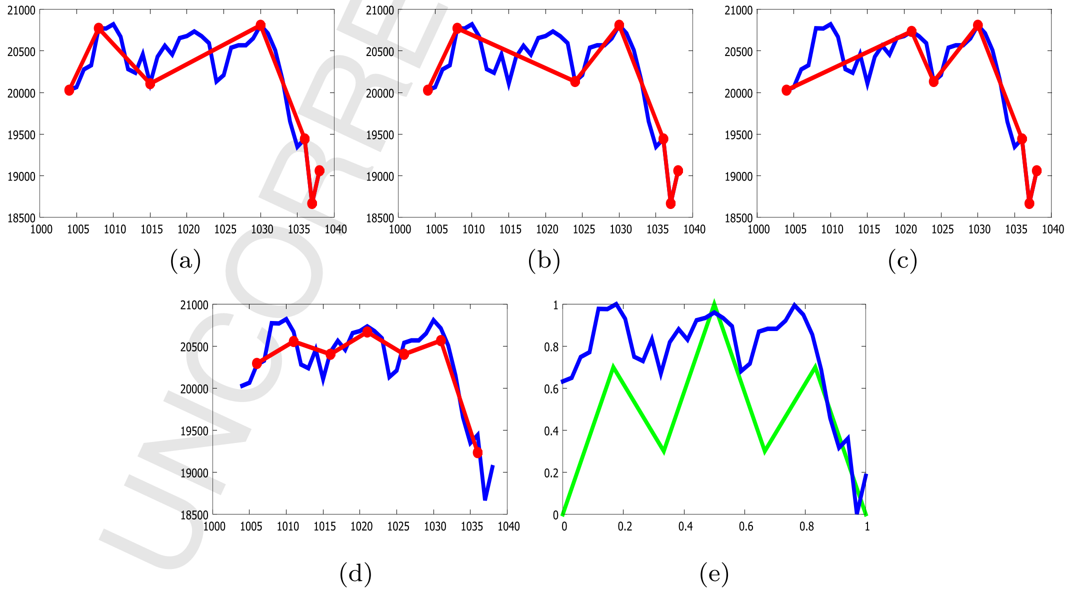
3.3. Real data experiment

For the experiment on a real data set, we use the historical prices of the Hang Seng Index from 1 January 2003 to 31 December 2012, which include 2506 points. The window sizes used are 35, 49, 63 and 91. For this experiment, redundant patterns that are within a +1 or −1 difference in position are eliminated. Table 3 shows the number of patterns found by the TB, RB, HY and DT approaches when



(a) (b) (c)

Fig. 14. (a) The accuracy of two cases with increasing window size. (b) The precision of two cases with increasing window size. (c) The sensitivity of two cases with increasing window size.



(a) (b) (c)

(d) (e)

Fig. 15. The segmentation results of (a) PIP, (b) PLA, (c) TP and (d) PAA (all shown in red) on the sub-sequence from 16 January 2007 to 6 March 2007 (shown in blue) (e) The normalised sub-sequence from 16 January 2007 to 6 March 2007 (in blue) on H&S (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

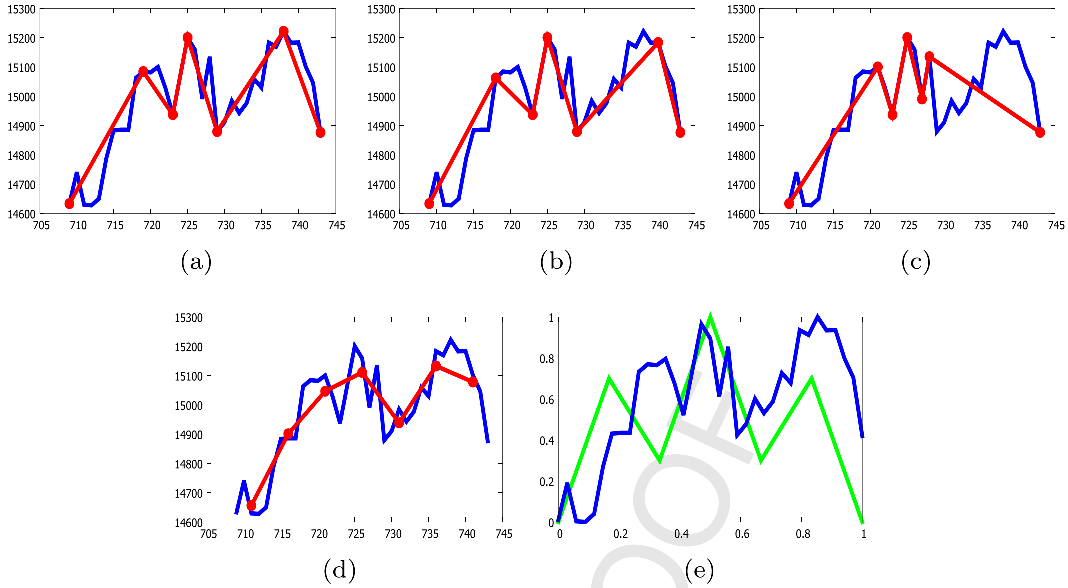


Fig. 16. The segmentation results with (a) PIP (b) PLA, (c) TP and (d) PAA (all shown in red) on the sub-sequence from 10 November 2005 to 30 December 2005 (shown in blue) (e) The normalised sub-sequence from 10 November 2005 to 30 December 2005 (in blue) on H&S (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

their results are paired with each of the four different segmentation methods. We choose the results of window size 35 to further analyse the performance of the four segmentation methods when used with the four pattern matching approaches. Table 4 in the appendix shows the exact position of the patterns when window size 35 is used. The following discussions are based on the results of window size 35.

In the following illustrations, the original results of the segmentation methods are depicted in red. The original sub-sequences from the time series and their normalised versions are shown in blue, and the normalised H& S pattern is depicted in green.

With each of the pattern matching approaches, the lowest number of patterns is found with PAA. In addition, in most cases the patterns found with PAA are different from those found with the other three segmentation methods. For example, with the PAA segmentation method, a pattern that the RB, HY and DT approaches find from 16 January 2007 to 6 March 2007 (shown in blue) is depicted in Fig. 15(d) and (e). However, this sub-sequence from 16 January 2007 to 6 March 2007 is not recognised as an H&S pattern when PIP, PLA or TP are used with the RB, HY and DT approaches. The segmentation results from PIP, PLA and TP methods are depicted in Fig. 15(a)–(c). In Fig. 15(d), we can notice that the shape of the segmentation result of PAA (shown in red) is different from results of the other segmentation methods shown in (a), (b) and (c). This difference in shape is caused by the smoothing effect of PAA on the input time series.

The highest number of patterns is found with TP in each of the pattern matching approaches. However, some of the patterns found

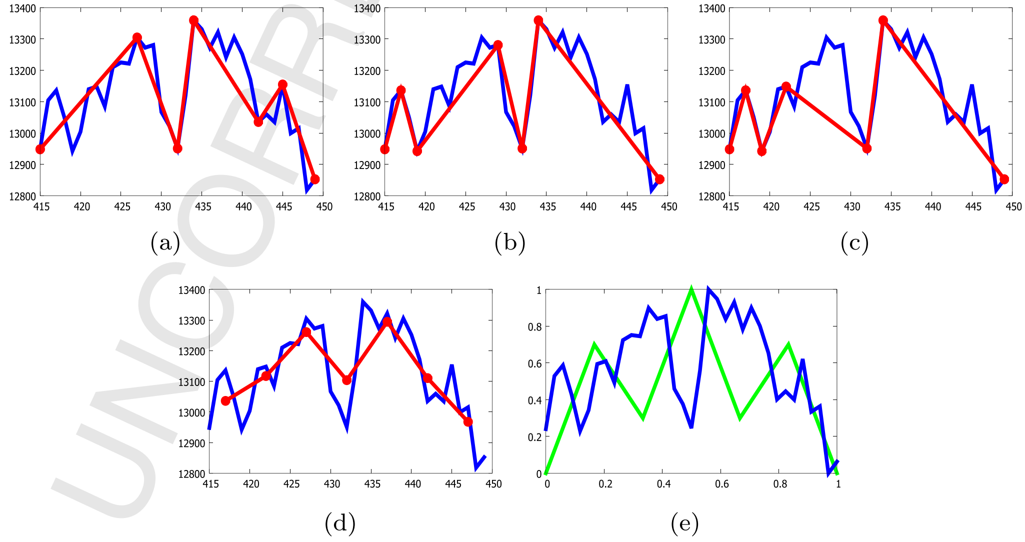
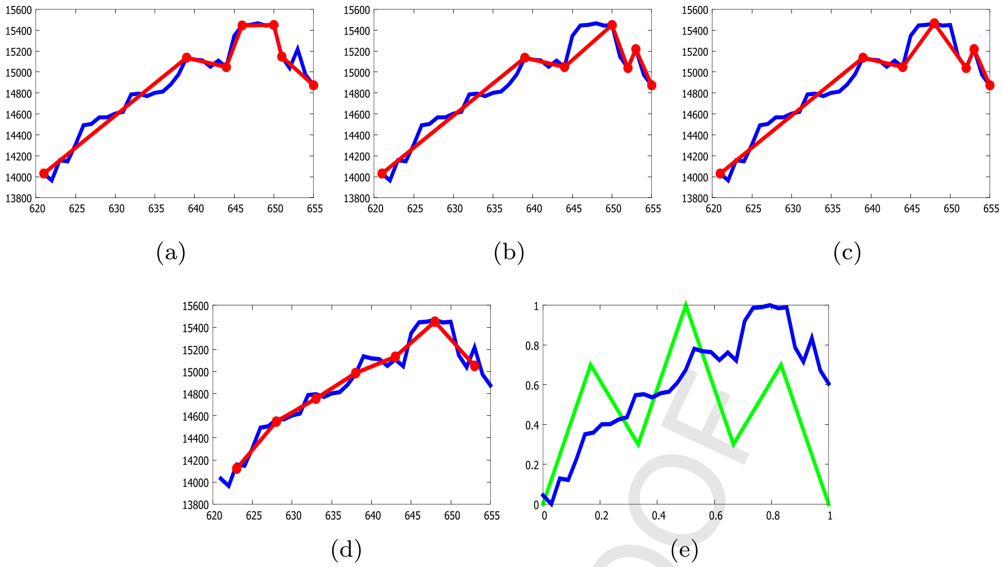


Fig. 17. The segmentation result of (a) PIP, (b) PLA, (c) TP and (d) PAA (all shown in red) on the sub-sequence from 3 September 2004 to 26 October 2004 (shown in blue) (e) The normalised sub-sequence from 3 September 2004 to 26 October 2004 (in blue) on H&S (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012



(a) (b) (c)

(d) (e)

Fig. 18. The segmentation results of (a) PIP, (b) PLA, (c) TP and (d) PAA (all shown in red) on the sub-sequence from 7 July 2005 to 24 August 2005 (shown in blue) and (e) the normalised sub-sequence from 7 July 2005 to 24 August 2005 (in blue) on H&S (in green). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

with TP are different from those found by the other three segmentation methods. For instance, a sub-sequence from 10 November 2005 to 30 December 2005 (shown in Fig. 16(e)) is recognised as an H&S pattern by the RB and HY approaches when TP is used as the segmentation method. However, with the other three segmentation methods, this sub-sequence is not recognised as a pattern by RB or the HY approaches. We can also observe that the shape of the segmentation result from TP (shown in red in Fig. 16(c)) is similar to that of PIP and PLA in Fig. 16(a) and (b).

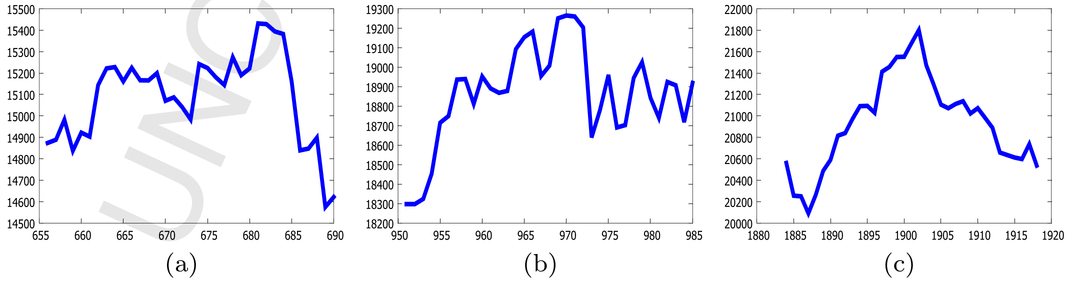
From our experimental results, we find that the patterns found with PIP are also found in at least one of the other three segmentation methods, except for one particular case. This case is further analysed in Fig. 17. A sub-sequence found with PIP from 3 September 2004 to 26 October 2004 (shown in blue in Fig. 17(e)) is recognised as an H&S pattern by the TB and DT approaches. However, this sub-sequence is not recognised as a pattern when the other three segmentation methods are used with the same pattern matching approaches.

Fig. 18(e) shows a sub-sequence from 7 July 2005 to 24 August 2005 (shown in blue). This sub-sequence is recognised as a pattern by the RB, HY and DT approaches when the PLA and TP methods of segmentation methods are used. However, this sub-sequence is not considered a pattern when the PIP and PAA methods are used.

3.4. *Experiment with SAX on real data*

We detect H&S pattern on real data set of HSI with SAX approach. The window sizes used are 35, 49, 63 and 91. For this experiment, redundant patterns that are within a +1 or -1 difference in position are eliminated. For the H&S pattern, each time series in the window is transformed into a sequence of symbols of length 7 using 4 characters. Next, we calculate the distance between the template and subsequence. If the minimum distance (MINDIST) is zero, the subsequence is recognized as a pattern. The experiment on real data set shows that SAX approach can find 14, 6, 6, and 6 patterns in the window sizes of 35, 49, 63 and 91, respectively.

For illustration, we select three patterns found by SAX approach for window size 35 in Fig. 19. From our experiment, we find that 10 out of 14 patterns found by SAX approach are the same as the patterns found by TB approach, 5 out of 14 patterns found by SAX approach are the same as the pattern found by RB and HY. 7 out of 14 patterns found by SAX approach are the same as the pattern found by DT approach. However, the subsequences found by SAX approach from Fig. 19(a)–(c) are not recognized by any other four pattern matching approaches (TB, RB, HY, DT) as patterns. In addition, we find that the symbolic sequences of these three cases (ACCCCDA), (ACCDCBB), (ABCDCBA) are different from the template H&S pattern (ACBDBCA) although the minimum distances among them are zero.



(a) (b) (c)

Fig. 19. Three subsequences found by SAX approach from HSI data. (a) A subsequence from 14 October 2005 to 25 August 2005. (b) A subsequence from 30 October 2005 to 15 December 2006. (c) A subsequence from 15 July 2010 to 1 November 2010.

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

4. Conclusions

The TB pattern matching approach measures the similarity of the absolute positions of the data points between the pattern template and the segmentation result of the sequence. The RB, HY and DT approaches measure similarity according to the relative fluctuations of the data points in the segmentation result of the sequence. From our experiments, we find that the PIP segmentation method can perform well to preserve the overall shape of the sequence. As a result, the PIP method achieves better performance than the other segmentation methods and is especially superior when used with the RB and HY pattern matching approaches. The goal of the PLA method is to minimise reconstruction errors. We find that PLA works well with all four pattern matching approaches, especially with the TB and DT approaches. Although the TP method can preserve more trends, it is inferior to the other segmentation methods when the length of the pattern is known or predefined by the user. The PAA method is designed to smooth out the input sub-sequence. As a result, PAA cannot preserve the up and down trends of the original sequence. The PAA method performs poorly with the RB, HY and DT approaches.

SAX transforms a real value time series into a symbolic sequence and MINDIST function is used to calculate the distance between two symbolic sequences. The synthetic data experiment shows that Case 1 has higher accuracy than Case 2 while Case 2 has higher precision than Case 1. When it is compared to other approaches, TB with PIP has higher accuracy than SAX in Case 1. The real data experiment shows that SAX is just as good as other pattern matching approaches.

When SAX is used to represent a query time series, it smoothes the time series since PAA is used as a preprocessing step. Therefore, the outcome of PAA can affect the overall process of SAX approach. When SAX is used to represent a template pattern, analysts need to decide the number of symbols that are used to represent the template and how to assign the symbols to the real value points from the template. Therefore, any bias in decision making can significantly affect the results. Although SAX approach is selected for evaluation in our experiments, other approaches can be used to measure the similarity between two symbolic sequences (e.g. edit distance of two strings). We can also model each symbol as a state and then use a Markov model to calculate the probability of the time series.

For future work in this area, we plan to develop an adaptive pattern matching system for streaming on-line data with appropriate pattern matching algorithms and segmentation methods.

Acknowledgement

Q7 This research was funded by the University of Macau under grants MYRG041(Y1-L1)-FST13-SYW and MYRG2015-00054-FST.

Appendix A.

Table 4 shows the results of window size 35 for the experiment with a real data set.

Table 4
The results of window size 35 for the experiment with a real data set.

Start date	End date	PIP	PLA	TP	PAA
TB approach					
2003/10/9	2003/11/26	0	1	1	1
2004/5/24	2004/7/14	1	1	1	0
2004/9/3	2004/10/26	1	0	0	0
2005/2/2	2005/3/29	1	0	1	0
2006/4/4	2006/5/26	0	0	1	0
2006/7/25	2006/9/11	0	0	1	0
2007/1/12	2007/3/2	1	1	0	1
2007/4/12	2007/5/30	1	0	0	1
2007/10/4	2007/11/21	1	1	1	1
2009/5/22	2009/7/13	0	0	1	0
2009/7/20	2009/9/4	0	0	0	1
2010/3/15	2010/5/3	1	1	0	0
2010/12/24	2011/2/14	0	0	0	1
2011/10/7	2011/11/24	0	0	1	0
2012/2/7	2012/3/26	1	0	1	1
2012/6/7	2012/7/25	1	1	1	1
Total		9	6	10	8
HY approach					
2005/7/7	2005/8/24	0	1	1	0
2005/11/10	2005/12/30	0	0	1	0
2007/1/16	2007/3/6	0	0	0	1
2007/6/18	2007/8/6	0	0	1	0
2007/10/8	2007/11/23	1	1	0	0
2009/7/15	2009/9/1	1	0	1	0
2010/10/11	2010/11/26	1	0	1	0
2012/6/7	2012/7/25	1	1	1	0
Total		4	3	6	1
RB approach					
2005/7/7	2005/8/24	0	1	1	0
2005/11/10	2005/12/30	0	0	1	0
2007/1/16	2007/3/6	0	0	0	1
2007/6/18	2007/8/6	0	0	1	0
2007/10/8	2007/11/23	1	1	0	0
2009/7/15	2009/9/1	1	0	1	0
2010/10/8	2010/11/25	0	0	1	0
2012/6/7	2012/7/25	1	1	1	0
Total		3	3	6	1
DT approach					
2004/9/3	2004/10/26	1	0	0	0
2005/7/7	2005/8/24	0	1	1	0
2005/11/10	2005/12/30	0	1	1	0
2006/4/4	2006/5/26	0	0	1	0
2007/1/16	2007/3/6	0	0	0	1
2007/6/18	2007/8/6	0	0	1	0
2007/10/8	2007/11/23	1	1	1	0
2009/5/22	2009/7/13	0	0	1	0
2009/7/17	2009/9/3	1	0	1	0
2010/10/6	2010/11/23	1	0	1	0
2012/6/8	2012/7/26	1	1	1	1
2012/9/27	2012/11/16	0	0	1	0
Total		5	4	10	2

References

[1] T.N. Bulkowski, Encyclopedia of Chart Patterns, vol. 225, John Wiley & Sons, 2011.

[2] A. Zapranis, P. Tsinaslanidis, Identification of the head-and-shoulders technical analysis pattern with neural networks, in: Artificial Neural Networks – ICANN 2010, Springer, 2010, pp. 130–136.

[3] M. Zhou, M.-H. Wong, K.-W. Chu, A geometrical solution to time series searching invariant to shifting and scaling, Knowl. Inf. Syst. 9 (2) (2006) 202–229.

[4] A.W. Lo, H. Mamaysky, J. Wang, Foundations of technical analysis: computational algorithms, statistical inference, and empirical implementation, J. Finance 55 (4) (2000) 1705–1770.

[5] Y.N. Rao, J.C. Principe, Time series segmentation using a novel adaptive eigendecomposition algorithm, J. VLSI Signal Process. Syst. Signal Image Video Technol. 32 (1–2) (2002) 7–17.

[6] X. Ge, P. Smyth, Deformable Markov model templates for time-series pattern matching, in: Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2000, pp. 81–90.

[7] J. Lin, E. Keogh, L. Wei, S. Lonardi, Experiencing SAX: a novel symbolic representation of time series, Data Min. Knowl. Discov. 15 (2) (2007) 107–144.

[8] P.M. Barnaghi, A.A. Bakar, Z.A. Othman, Enhanced symbolic aggregate approximation (EN-SAX) as an improved representation method for financial time series data, Int. J. Soft Comput. 8 (4) (2013) 261–268.

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012

[9] F.J. Damerau, A technique for computer detection and correction of spelling errors, Commun. ACM 7 (3) (1964) 171–176.

[10] V.I. Levenshtein, Binary codes capable of correcting deletions, insertions, and reversals, in: Soviet Physics Doklady, vol. 10, 1966, pp. 707–710.

[11] S. Kullback, R.A. Leibler, On information and sufficiency, Ann. Math. Stat. (1951) 79–86.

[12] J. Lin, Divergence measures based on the Shannon entropy, IEEE Trans. Inf. Theory 37 (1) (1991) 145–151.

[13] A. Bhattacharyya, On a measure of divergence between two multinomial populations, Sankhyā: Indian J. Stat. (1946) 401–406.

[14] T.-C. Fu, A review on time series data mining, Eng. Appl. Artif. Intell. 24 (1) (2011) 164–181.

[15] Z. Xing, J. Pei, E. Keogh, A brief survey on sequence classification, ACM SIGKDD Explor. Newslett. 12 (1) (2010) 40–48.

[16] W. Hachicha, A. Ghorbel, A survey of control-chart pattern-recognition literature (1991–2010) based on a new conceptual classification scheme, Comput. Ind. Eng. 63 (1) (2012) 204–222.

[17] D.J. Berndt, J. Clifford, Using dynamic time warping to find patterns in time series, in: KDD Workshop, vol. 10, Seattle, WA, 1994, pp. 359–370.

[18] H. Li, C. Guo, W. Qiu, Similarity measure based on piecewise linear approximation and derivative dynamic time warping for time series mining, Expert Syst. Appl. 38 (12) (2011) 14732–14743.

[19] L. Junkui, W. Yuanzhen, Early abandon to accelerate exact dynamic time warping, Int. Arab. J. Inf. Technol. 6 (2) (2009) 144–152.

[20] Y. Chen, B. Hu, E. Keogh, G.E. Batista, DTW-D: time series semi-supervised learning from a single example, in: Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2013, pp. 383–391.

[21] S. Vishwanathan, A.J. Smola, Fast kernels for string and tree matching, Kernel Methods Comput. Biol. (2004) 113–130.

[22] C.H. Teo, S. Vishwanathan, Fast and space efficient string kernels using suffix arrays, in: Proceedings of the 23rd International Conference on Machine Learning, ACM, 2006, pp. 929–936.

[23] M.I. Abouelhoda, S. Kurtz, E. Ohlebusch, Replacing suffix trees with enhanced suffix arrays, J. Discrete Algorithms 2 (1) (2004) 53–86.

[24] T.-C. Fu, F.-L. Chung, R. Luk, C.-M. Ng, Stock time series pattern matching: template-based vs. rule-based approaches, Eng. Appl. Artif. Intell. 20 (3) (2007) 347–364.

[25] Z. Zhang, J. Jiang, X. Liu, R. Lau, H. Wang, R. Zhang, A real time hybrid pattern matching scheme for stock time series, in: Proceedings of the Twenty-First Australasian Conference on Database Technologies – vol. 104, Australian Computer Society, Inc., 2010, pp. 161–170.

[26] F.-L. Chung, T.-C. Fu, R. Luk, V. Ng, Flexible time series pattern matching based on perceptually important points, in: International Joint Conference on Artificial Intelligence Workshop on Learning from Temporal and Spatial Data, 2001, pp. 1–7.

[27] E.J. Keogh, M.J. Pazzani, A simple dimensionality reduction technique for fast similarity search in large time series databases, in: Knowledge Discovery and Data Mining. Current Issues and New Applications, Springer, 2000, pp. 122–133.

[28] E. Keogh, S. Chu, D. Hart, M. Pazzani, An online algorithm for segmenting time series, in: Proceedings IEEE International Conference on Data Mining, 2001. ICDM 2001, IEEE, 2001, pp. 289–296.

[29] Y.-W. Si, J. Yin, OBST-based segmentation approach to financial time series, Eng. Appl. Artif. Intell. 26 (10) (2013) 2581–2596.

[30] C.-H. Chen, V.S. Tseng, H.-H. Yu, T.-P. Hong, Time series pattern discovery by a PIP-based evolutionary approach, Soft Comput. 17 (9) (2013) 1699–1710.

[31] L. Kobbelt, S. Campagna, H.-P. Seidel, A general framework for mesh decimation, in: Graphics Interface, vol. 98, 1998, pp. 43–50.

[32] N.Q.V. Hung, D.T. Anh, Combining sax and piecewise linear approximation to improve similarity search on financial time series, in: International Symposium on Information Technology Convergence, 2007. ISITC 2007, IEEE, 2007, pp. 58–62.

[33] T. Rakthanmanon, B. Campana, A. Mueen, G. Batista, B. Westover, Q. Zhu, J. Zakaria, E. Keogh, Searching and mining trillions of time series subsequences under dynamic time warping, in: Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, 2012, pp. 262–270.

[34] H. Hyyrö, A bit-vector algorithm for computing levenshtein and damerau edit distances, Nord. J. Comput. 10 (1) (2003) 29–39.

[35] M. Ramoni, P. Sebastiani, P. Cohen, Bayesian clustering by dynamics, Mach. Learn. 47 (1) (2002) 91–121.

[36] W.-K. Ching, M.K. Ng, E.S. Fung, Higher-order multivariate Markov chains and their applications, Linear Algebra Appl. 428 (2) (2008) 492–507.

[37] J. Lin, E. Keogh, S. Lonardi, B. Chiu, A symbolic representation of time series, with implications for streaming algorithms, in: Proceedings of the 8th ACM SIGMOD Workshop on Research Issues in Data Mining and Knowledge Discovery, ACM, 2003, pp. 2–11.

Please cite this article in press as: Y. Wan, et al., Effect of segmentation on financial time series pattern matching, Appl. Soft Comput. J. (2015), http://dx.doi.org/10.1016/j.asoc.2015.10.012