Available online at www.sciencedirect.com
ScienceDirect
Journal of Computational and Applied Mathematics 214 (2008) 227 – 237
JOURNAL OF COMPUTATIONAL AND APPLIED MATHEMATICS
www.elsevier.com/locate/cam

Multi-scale anomaly detection algorithm based on infrequent pattern of time series

Xiao-yun Chen*, Yan-yan Zhan

College of Mathematics and Computer Science, Fuzhou University, Fuzhou 350002, China

Received 28 November 2006; received in revised form 15 February 2007

Abstract

In this paper, we propose two anomaly detection algorithms PAV and MPAV on time series. The first basic idea of this paper defines that the anomaly pattern is the most infrequent time series pattern, which is the lowest support pattern. The second basic idea of this paper is that PAV detects directly anomalies in the original time series, and MPAV algorithm extraction anomaly in the wavelet approximation coefficient of the time series. For complexity analyses, as the wavelet transform have the functions to compress data, filter noise, and maintain the basic form of time series, the MPAV algorithm, while maintaining the accuracy of the algorithm improves the efficiency. As PAV and MPAV algorithms are simple and easy to realize without training, this proposed multi-scale anomaly detection algorithm based on infrequent pattern of time series can therefore be proved to be very useful for computer science applications.

MSC: 68T10; 62L10

Keywords: Time series; Anomaly detection; Linear pattern; Wavelet transform; Support count

1. Introduction

The mining of the time series data is one of the important research topics in data mining field, especially the problem of anomaly detection in time series data has received much attention [4,10,13,18,5,8,19]. Anomaly detection, or outlier detection [7,2,9,14,16], refers to automatic identification objects that are different from most other objects. In some situations, anomaly data have unusual meaning and can provide lots of useful information. The research on the anomaly detection of time series is not very mature, mainly because it is hard to obtain sufficient knowledge and an accurate representation of "novelty" given a problem [1]. So there has not been an acknowledged definition at present. The terms related with the outlier of time series have novelty [4,13], anomaly [10], surprise [18], deviant [8], change point [19], etc.

Despite on the great challenge, over the past 10 years the research on the anomaly detection of time series has been a topic acquiring increasing attention, and quite a few techniques have been proposed. These techniques were experimentally proven to be effective in some cases, while they can fail in other cases. In some other studies [13,8,19],

* Corresponding author. Tel.: +86 0591 83733907.
E-mail address: c_xiaoyun@fzu.edu.cn (X.-y. Chen).

0377-0427/$ - see front matter © 2007 Elsevier B.V. All rights reserved.
doi:10.1016/j.cam.2007.02.027

anomaly detection was interpreted simply as outlier point detection; however, this method cannot discover novel patterns formed by several continuous points. In particular, the anomaly detection method proposed in [13] is based on a technique called Support Vector Regression (SVR), whose formulation forces it to "identify" some abnormal point no matter how normal the whole data set is. SVR algorithm builds regression model by training the history time series and estimate the confidence of the new series point as novelty according to the degree of the new series point match with the model. A wavelet-based signal trend shift detection method is proposed in [18]. Nevertheless, this method cannot detect short abnormal patterns embedded in normal signals. An interesting idea for novelty detection, inspired by the negative-selection mechanism in the immune system, was proposed in [4]. However, this method can fail when the negative set goes to null with the increasing diversity of the normal set. Another method for target time series novelty detection, called TARZAN [10], is based on converting the time series into a symbolic string. However, the procedure for discrediting and symbolizing real values in time series, as well as the various choices of string-representation parameters, can cause the loss of meaningful patterns in the original time series.

In the research mentioned above, the anomaly is defined as the anomaly point or the pattern that deviates from certain training model. Firstly, we are not interested in finding individually outlier data points, but interested in finding anomaly patterns, i.e., combination of data points whose structure and frequency somehow defies our expectations. Secondly, the anomaly detection algorithm based on training model must build the model by learning from all normal parts of time series and then distinguish anomaly pattern using the model [13].

According to the representation, the anomaly can be divided into point anomaly, pattern anomaly and series anomaly. The point anomaly and the pattern anomaly can be found in a time series; however, the series anomaly must be detected in a time series set which consists of many time series. We research mainly the pattern anomaly, namely the pattern which is obviously different from other pattern in the same series.

In this paper, we propose an anomaly detection algorithm PAV based on infrequent linear patterns, PAV finds directly anomaly pattern using the anomaly value of the pattern, and need not train a model. Furthermore, PAV algorithm can be combined with the Haar wavelet transform to create a new anomaly detection algorithm MPAV. We proposed the multi-scale anomaly detection algorithm MPAV combining Haar wavelet transform and PAV algorithm. MPAV can find anomaly pattern under different scale applying the multi-resolution property of wavelets.

The rest of this paper is organized as follows. The new definition of anomaly pattern is proposed in Section 2. Based on the definition, Section 3 proposes an anomaly detection algorithm PAV. To improve the performance of the detection algorithm, Section 4 presents an algorithm MPAV combining a compress technique called Haar wavelet transform with PAV. Experiments are proposed in Section 5.

2. Anomaly pattern based on support count

The time series is a sequence of measurements of a variable at different time points, that is

$$X = \langle v_1 = (x_1, t_1), v_2 = (x_2, t_2), \ldots, v_n = (x_n, t_n)\rangle,$$

where element $v_i = (x_i, t_i)$ denotes the measurement value x_i at time t_i for the time series X, the time variable t_i is strictly incremental ($i < j \Leftrightarrow t_i < t_j$). $X = \langle v_1 = (x_1, t_1), v_2 = (x_2, t_2), \ldots, v_n = (x_n, t_n)\rangle$ can be denoted simply by $X = \langle x_1, x_2, \ldots, x_n\rangle$, where $x_i (i = 1, 2, \ldots, n)$ is the amplitude at time t_i.

Usually, the sampling interval $\Delta t = t_i - t_{i-1}$ $(i = 2, 3, \ldots, n)$ is changeless, so we let $t_1 = 1$, $\Delta t = 1$. If the sampling interval Δt is variable, the time series is called as unequal interval time series.

Definition 1 (*Linear pattern*). The linear segment joining two neighbor sampling points $\langle x_i, x_{i+1}\rangle$ $(i = 1, 2, \ldots, n-1)$ in the time series X is named as linear pattern, called simply as pattern.

Definition 2 (*Support count*). Let $X = \langle x_1, x_2, \ldots, x_n\rangle$ be a time series and $Y_i = \langle x_i, x_{i+1}\rangle$ $(i = 1, 2, \ldots, n-1)$ be its pattern, if the pattern Y_i occurs δ times in series X, the support count of pattern $Y_i = \langle x_i, x_{i+1}\rangle$ $(i = 1, 2, \ldots, n-1)$ is defined as δ. Formally, the support count, $\delta(Y_i)$ for a pattern Y_i can be stated as follows:

$$\delta(Y_i) = |\{Y_i | Y_i = \langle x_i, x_{i+1}\rangle \wedge x_i \in X \wedge x_{i+1} \in X\}.$$

Anomaly pattern can be defined as infrequent or rare patterns, i.e., the support count of anomaly patterns is lower than other patterns in time series. In other words, the lower support count for the pattern Y_i, the higher the anomaly degree of Y_i.

So the anomaly degree of a pattern can be measured in terms of its anomaly value, the *anomaly value* determines how anomalous a pattern Y_i appears in a given time series. The formal definition of anomaly value can be stated as follows:

Definition 3 (*Anomaly value*). Let δ_i be the support count for the pattern Y_i $(i = 1, 2, \ldots, n-1)$ and δ_i be normalized to unit interval [0,1] using the following formula:

$$\hat{\delta}_i = \frac{\delta_i - \min_j(\delta_j)}{\max_j(\delta_j) - \min_j(\delta_j)}. \tag{1}$$

The anomaly value for Y_i is defined as

$$\mathrm{AV}_i = 1 - \hat{\delta}_i. \tag{2}$$

Definition 4 (*Anomaly pattern*). Given a time series X, the pattern Y_i $(i = 1, 2, \ldots, n-1)$ having $\mathrm{AV}_i \geqslant$ minvalue is called anomaly pattern, where minvalue is the anomaly value thresholds.

The higher the AV, the greater the likelihood of abnormal pattern. In some cases, it is difficult to specify an appropriate threshold, so we rank each patterns on the basis of its AV value decreasing order and declare the greatest or top k patterns in this ranking to be anomaly.

For judging whether a linear pattern is anomaly, one needs to address the problem of how to determine whether the pattern is same with other patterns. Therefore, slope and length of the pattern are introduced to resolve the problem. If the two patterns have the same slope and length, the two patterns will be the same.

3. Time series anomaly pattern mining algorithm PAV

According to the definition of the anomaly pattern in the previous section, we propose an anomaly detection algorithm based on pattern anomaly value (PAV).The algorithm finds the anomaly pattern by calculating AV of the pattern.

The algorithm includes mainly two phases:

(1) Extracting features of patterns in time series. Firstly, the two neighbor points: x_i and x_{i+1} of time series is connected to the linear patterns $Y_i = \langle x_i, x_{i+1} \rangle$ $(i = 1, 2, \ldots, n-1)$. Then extract two features of pattern Y_i: slope s_i and length l_i, where the slope of linear pattern is $s_i = (x_{i+1} - x_i)/(t_{i+1} - t_i)$ and the precision of the slope is denoted by parameter e, namely retain e digits after decimal point when evaluating the slope, the e usually is set as 1 or 2; the length of linear pattern is $l_i = \sqrt{(x_{i+1} - x_i)^2 + (t_{i+1} - t_i)^2}$.

(2) Compute the anomaly value of patterns.
The two patterns are the same when the slope and the length of the two patterns are equal. Therefore, firstly compare all patterns in time series; if the two patterns are same, the support count of the two patterns will be increased by 1. In this way, we can obtain the support count of each pattern and the maximum or the minimum in these support counts. Then map the support counts of patterns to the interval [0, 1] using the formula (1), and compute the anomaly value of each pattern by formula (2). Lastly, the anomaly degree can be estimated ultimately by the anomaly value.

Algorithm. Time series anomaly pattern detection algorithm.

Input: Time series $X = \langle x_1, x_2, \ldots, x_n \rangle$, the precision of the slope e, the number of anomaly patterns k or the minimum threshold *minav*.

Output: The anomaly value of the pattern $Y_i = \langle x_i, x_{i+1} \rangle$ $(i = 1, 2, \ldots, n-1)$ and the anomaly patterns.

Method:

1. Compute the support count δ_i of each pattern Y_i.
   1.1. Compute the slope s_i of the pattern Y_i, and retained e digits after decimal point.

$$s_i = \frac{x_{i+1} - x_i}{t_{i+1} - t_i}. \tag{3}$$

   Compute the length l_i of the pattern Y_i:

$$l_i = \sqrt{(x_{i+1} - x_i)^2 + (t_{i+1} - t_i)^2}. \tag{4}$$

   1.2. Compare pattern Y_i with Y_j $(i = 1, 2, \ldots, n-1;\ \ j = i, i+1, \ldots, n-1)$, if the slope and length of Y_i and Y_j is equal, the support count of Y_i and Y_j will be increased by 1. The vector $\delta = (\delta_1, \delta_2, \ldots, \delta_{n-1})$ consisting of the support counts of the patterns can be generated.
2. Standardize the vector $\delta = (\delta_1, \delta_2, \ldots, \delta_{n-1})$
   2.1. Compute the maximum of support counts of all patterns: $\max = \max_j(\delta_j)$.
   2.2. Compute the minimum of support counts of all patterns: $\min = \min_j(\delta_j)$.
   2.3. Standardize the support count vector $\delta = (\delta_1, \delta_2, \ldots, \delta_{n-1})$, the support count δ_i of each pattern Y_i $(i = 1, 2, \ldots, n-1)$ is standardized by the formula $\hat{\delta}_i = \dfrac{\delta_i - \min}{\max - \min}$, and generate the standardization vector $\hat{\delta} = (\hat{\delta}_1, \hat{\delta}_2, \ldots, \hat{\delta}_{n-1})$.
3. Compute the anomaly value AV_i of pattern Y_i, then generate and output the pattern anomaly vector

$$P = (\mathrm{AV}_1, \mathrm{AV}_2, \ldots, \mathrm{AV}_{n-1}).$$

4. Sort the anomaly value AV_i $(i = 1, 2, \ldots, n-1)$ by the descending order and output the top k patterns or the patterns whose anomaly values are greater than *minav* or equal to *minav*.

Under the worst case, PAV needs to carry out $n(n-1)/2$ times comparison between the patterns, so its time complexity is $O(n^2/2)$ and space complexity is $O(n)$.

As the two patterns can be judged as being same according to the same slope and same length of the pattern in PAV, PAV can also be applied to unequal interval time series. This advantage greatly expanded the scope of the application of the algorithm.

If the time series is in equal interval, and assume $\Delta t = t_{i+1} - t_i = 1$, formula (3) can be simplified as

$$s_i = x_{i+1} - x_i. \tag{5}$$

Formula (4) can be simplified as

$$l_i = \sqrt{(x_{i+1} - x_i)^2 + 1}. \tag{6}$$

From formulas (5) and (6) we can find that if two patterns have the same slope, they have the same length under equal interval case. Therefore, the same two patterns can be determined as long as the slope of the same value.

4. Time series multi-scale anomaly detection based on Haar wavelet transform

Wavelet transform (WT) or discrete wavelet transform (DWT) has been found to be effective in replacing DFT in many applications in computer graphics, image, speech, and signal processing [15,11,12,3]. We apply this technique to dimensionality reduction for time series.

There are a wide variety of popular wavelet algorithms, including Daubechies wavelets, Mexican Hat wavelets and Morlet wavelets [6]. These wavelet algorithms have the advantage of better resolution for smoothly changing time series. But they have the disadvantage of being more expensive to calculate than the Haar wavelets, so we adopt the Haar wavelets.

The time series can be decomposed into two sets of coefficients: approximation coefficients A and detail coefficients D by DWT. Approximation coefficients are generated by a scaling function which is a low-pass filter. The low-pass filter suppresses the high-frequency components of a signal and allows the low-frequency components through. The Haar scaling function calculates the average of an even and an odd element, which results in a smoother that is low-pass signal.

Detail coefficients are generated by a wavelet function which is a high-pass filter. The high-pass filter allows the high-frequency components of a signal through while suppressing the low-frequency components. For example, the differences that are captured by the Haar wavelet function represent high-frequency change between an odd and an even value.

4.1. Related terms on Haar wavelet transform

Scaling function: Given a time series $X = \langle x_1, x_2, \ldots, x_n \rangle$ (n is even), the Haar wavelet scaling function is

$$x'_i = \frac{x_{2i+1} + x_{2i+2}}{\sqrt{2}} \quad \left(i = 1, 2, \ldots, \frac{n}{2}\right), \tag{7}$$

where x'_i is an approximation coefficient, which is a smoothed value of time series. The scaling function produces a smoother version of the data set, which is half the size of the input data set. Wavelet algorithms are recursive and the smoothed data becomes the input for the next step of the wavelet transform.

Wavelet function: Given a time series $X = \langle x_1, x_2, \ldots, x_n \rangle$ (n is even), the Haar wavelet function is

$$d_i = \frac{x_{2i+1} - x_{2i+2}}{\sqrt{2}} \quad \left(i = 1, 2, \ldots, \frac{n}{2}\right), \tag{8}$$

where d_i is a wavelet coefficient, representing the difference between adjacent points of time series, or called as wavelet detail coefficients.

We use the most simple matrix multiplication to describe discrete Haar transform.

Haar wavelet transform: Given a time series $X = \langle x_1, x_2, \ldots, x_n \rangle$, its Haar wavelet transform formula is as follows:

$$W = \begin{bmatrix} x'_1 \\ d_1 \\ x'_2 \\ d_2 \\ \vdots \\ x'_{n/2} \\ d_{n/2} \end{bmatrix} = \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ & \cdots & & \cdots & & \cdots & \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}}_{H} \times \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ \vdots \\ x_{n-1} \\ x_n \end{bmatrix} \tag{9}$$

where H is the transform matrix, W is the wavelet transform coefficient vector consisting of the high-frequency coefficients $(d_1, d_2, \ldots, d_{n/2})$ and the wavelet coefficient $(x'_1, x'_2, \ldots, x'_{n/2})$.

Haar wavelet converse transform: The converse transform can be represented as

$$X = \underbrace{\frac{1}{\sqrt{2}} \begin{bmatrix} 1 & 1 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ & \cdots & & \cdots & & \cdots & \\ 0 & 0 & 0 & 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix}}_{H} \times W \tag{10}$$

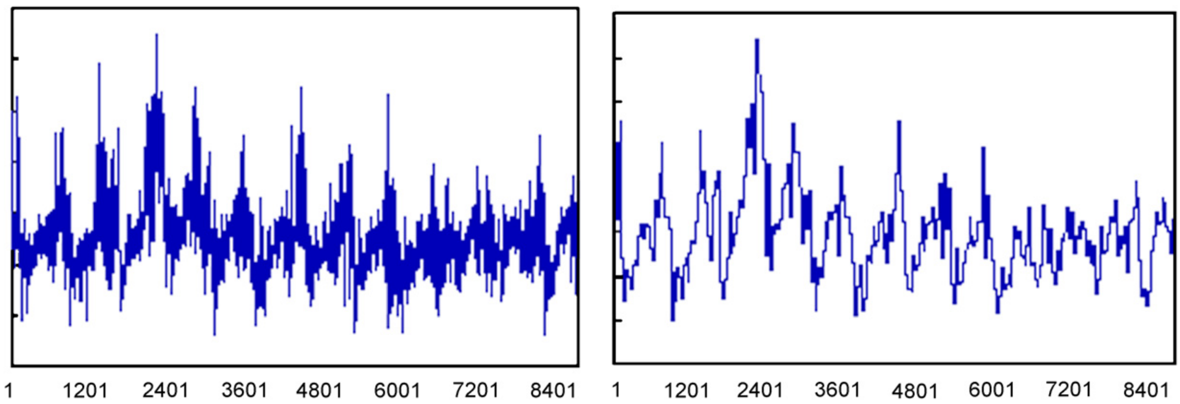

Fig. 1. The compression of time series based on DWT. (a) The original time series, its length is 8746. (b) The compressed time series, its length is 274.

4.2. Multi-sale wavelet decomposition

The set of approximation coefficients A is represented as $A=[x'_1, x'_2, \ldots, x'_{n/2}]^{\mathrm{T}}$, and the detail coefficients D is represented as $D=[d_1, d_2, \ldots, d_{n/2}]^{\mathrm{T}}$. The final transformed vector is $H(X)=\{A, D\}$.

Given a time series X of length n, the DWT consists of $\log_2 n$ stages at most. The first step produces, starting from X, two vectors of coefficients: approximation coefficients A_1 and detail coefficients D_1. The coefficients vectors A_1 and D_1 are of length $n/2$.

The next step splits the approximation coefficients A_1 into two parts using the same scheme, replacing X by A_1, and producing A_2 and D_2, and so on. The wavelet decomposition of the time series X at level k has the following structure: $H(X)=\{A_k, D_1, D_2, \ldots, D_k\}$, and $|A_k|=1/2^k$, $|D_j|=1/2^j$ $(j=1,2,\ldots,k)$, $k \leqslant \log_2 n$.

As the first few wavelet transform coefficient is the strongest after wavelet transform of time series and the highest frequency part of wavelet coefficient contains the most of the noise, the profile of time series remains basically unchanged when only the first few coefficients are retained. Therefore, wavelet transforms can play a role of compressing time series and eliminating noise. The more retained a coefficient, the lower the compression rate, but also the better the approximation for the original time series. We use directly the approximation coefficient A_k to figure original time series, the method is relatively simple, and the compressed time sequence is more smooth.

Fig. 1(a) shows an original time series including 8746 sampling points. After five wavelet decomposition level, all high-frequency coefficients of wavelet decomposition are set zero, and use the approximation coefficients A_5 as the compressed series. Although the compressed series shown in Fig. 1(b) only has 274 points, that is the compression rate is 96.868%, it still retains 72.75% of the energy of the original series and maintains a good shape of the original series. When detecting the anomaly using the compressed series we can obviously reduce the size of data. Moreover, the different levels of wavelet decomposition coefficients reflect the morphological characteristics of time series of the different scales; the short or long term anomalies can be detected through adjusting the scale.

4.3. Multi-scale anomaly pattern detection algorithm MPAV

Based on the algorithm PAV we adopt the multiple levels Haar transform to compress time series and then detect the anomaly pattern using PAV. The new method is called the multi-scale abnormal pattern mining algorithm MPAV.

MPAV algorithm includes the following steps:

(1) The set of approximation coefficients A_k is obtained by k levels wavelet decomposition of the time series $X=\langle x_1, x_2, \ldots, x_n\rangle$.
(2) The anomaly value sequence $\langle \mathrm{AV}_1^k, \mathrm{AV}_2^k, \mathrm{AV}_3^k, \ldots, \mathrm{AV}_{(n/2^k-1)}^k\rangle$ of all patterns of new series A_k was calculated using the PAV algorithm.

(3) Reconstruct the sequence of anomaly values of A_k, that is calculate the pattern anomaly value $\langle \mathrm{AV}_1, \mathrm{AV}_2, \mathrm{AV}_3, \ldots, \mathrm{AV}_{(n-1)} \rangle$ of the original time series using formula $\mathrm{AV}_i = \mathrm{AV}^k_{(\lfloor (i-1)/k \rfloor + 1)}$ $(i = 1, 2, \ldots, n-1)$.
(4) According to the anomaly value descending order, the top k patterns are the anomaly patterns, or the pattern whose abnormal value is larger than given threshold *minav*.

The number or average length of the pattern of time series is different under the different compression rate, the fewer the number of patterns, the longer the average length of patterns, and the smaller the scale of the anomaly. So MPAV algorithm can detect the different scale anomaly pattern of the time series.

MPAV algorithm has the following advantages:

(1) Wavelet transform can eliminate noise and compress time series data, therefore, the accuracy and efficiency of the anomaly detection can be improved; and the space complexity and time complexity of Haar transform is only $O(n)$.
(2) Multi-scale anomaly patterns can be detected from the compressed time series with different compressed ratio.

5. Experimental results and analysis

Among the existing anomaly detection methods, we analyze only the four major ones, and use their experimental data to test our PAV and MPAV algorithms. The experimental results are analyzed as follows.

5.1. Experiment 1: SantaFe Data

5.1.1. Experimental data

Ma et al. in the literature [13] present an anomaly detection algorithm based on support vector regression (support vector regression, called SVR), the algorithm adopts the famous Santa Fe Institute Competition data [17], which is 1000-point time series. These data were recorded from a far-infrared laser in a chaotic state.

5.1.2. Experimental results and analysis

SVR algorithm establishes SVR repression model by training the historical data of the time series, then the matching degree of regression model with the new data points is used to represent the confidence which the data points become anomaly.

In this experiment, we adopt PAV algorithm to detect the anomaly of SantaFe Data, and set the precision of the slope $e = 1$, the pattern with the greatest anomaly value becomes the anomaly. Fig. 2 gives the experimental results for SVR algorithms and PAV algorithm on SantaFe Data.

Fig. 2 shows that the SVR and PAV algorithms can discover the two anomalies after the first 200 points. But SVR needs to select the first 200 points for training the SVR, so it cannot find the abnormal pattern of the entire sequence. Our PAV algorithm can discover directly all three anomaly patterns of the entire time series because PAV does not require training.

5.2. Experiment 2: Ma_Data set

5.2.1. Experimental data

In addition to Santa Fe data set, Ma et al. [3] also use simulation data sets Ma_Data to test SVR algorithm. The data set includes two time series generated from the following stochastic process:

$$X_1(t) = \sin\left(\frac{40\pi}{N} t\right) + n(t),$$

$$X_2(t) = \sin\left(\frac{40\pi}{N} t\right) + n(t) + e_1(t),$$

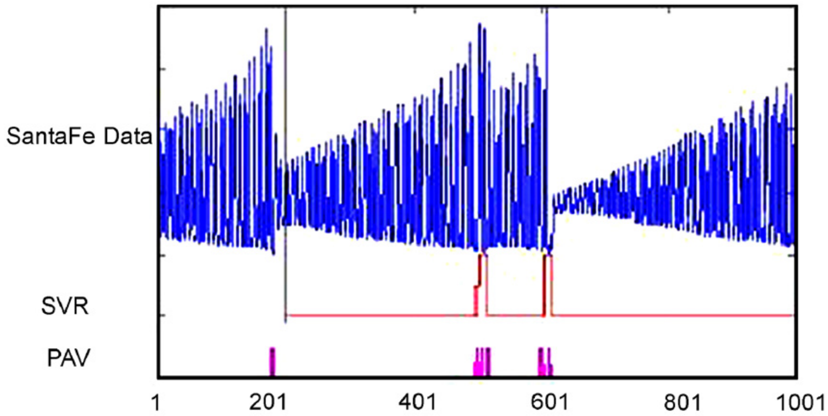

Fig. 2. A comparison for algorithms SVR and PAV on the SantaFe Data.

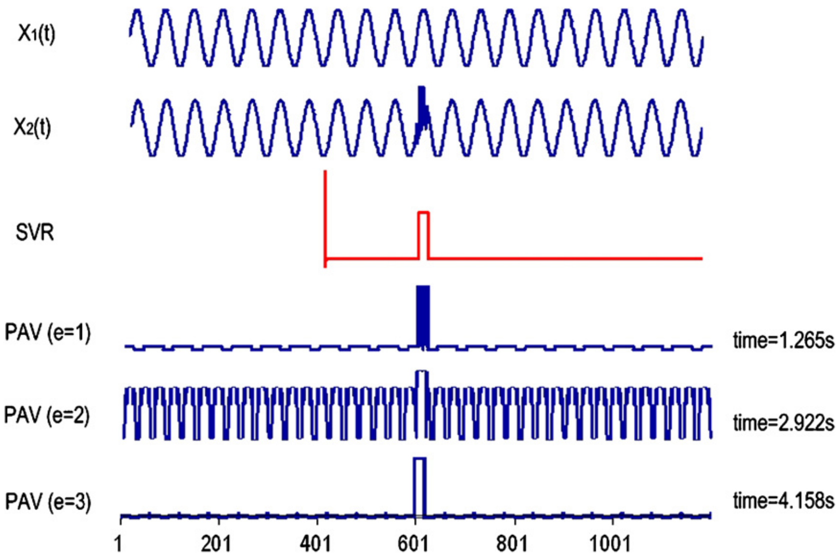

Fig. 3. The experimental results for algorithms SVR and PAV on time series $X_2(t)$.

where $t = 1, 2, \ldots, N$, $N = 1200$, $n(t)$ is an additive Gaussian noise with zero-mean and a standard deviation of 0.1. $e_1(t)$ is a novel event, and defined as follows:

$$e_1(t) = \begin{cases} n_1(t), & t \in [600, 620], \\ 0 & \text{otherwise}, \end{cases}$$

where $n_1(t)$ follows a normal distribution of N (0,0.5).

We can see that the time series $X_1(t)$ is the normal time series with 1200 points, $X_2(t)$ is a time series added on an abnormal event $e_1(t)$ in the [600,620] interval.

5.2.2. *Experimental results and analysis*

Fig. 3 shows the experimental results of SVR and PAV algorithms applied in $X_2(t)$. In Fig. 3, the result of SVR is the corresponding curves with SVR label, and the three curves with PAV label, respectively, denoting the anomaly value when parameters of PAV algorithm $e = 1$, 2 and 3.

As shown in Fig. 3, for the different values of parameter e, anomaly patterns generated by PAV algorithm are not very different, and their anomaly interval all is [600,621], basically coinciding with the actual anomaly interval [600,620]. But the smaller the value of e is, the shorter the algorithm execution time, specifically, if the parameter e value was reduced to 1, then the efficiency of the algorithm can be improved by 30–60%.

SVR and PAV algorithms successfully detect the entire anomaly pattern in $X_2(t)$. However, SVR algorithm must undergo training (Ma et al. used the first 400 points for training, and the last 800 points for detecting anomaly patterns [13]), and the PAV algorithm does not require training.

5.3. Experiment 3: Keogh_Data set

5.3.1. Experimental data

The Keogh_Data is a simulation data set which used to test three anomaly detection algorithms IMM, TSA-Tree and Tarzan by Keogh et al. in the literature [10]. The data set is created by the following formula:

$$Y_1(t) = \sin\left(\frac{50\pi}{N}t\right) + n(t),$$

$$Y_2(t) = \sin\left(\frac{50\pi}{N}t\right) + n(t) + e_1(t),$$

where $t = 1, 2, \ldots, N$, $N = 800$. $n(t)$ is an additive Gaussian noise with zero-mean and a standard deviation of 0.1. $e_1(t)$ is a synthetic "anomaly", defined as follows:

$$e_1(t) = \begin{cases} \sin\left(\frac{75\pi}{N}t\right) - \sin\left(\frac{50\pi}{N}t\right), & t \in [400, 432], \\ 0 & \text{otherwise.} \end{cases}$$

From the above description it is known, Y_1 (t) is a sine wave in 32 cycle with a Gaussian noise. Y_2 (t) is obtained by changing the sine wave cycle in the interval [400,432], that is adding an anomaly event e_3 (t) in the time series $Y_1(t)$.

5.3.2. Experimental results and analysis

On Keogh_Data set, we compare PAV algorithm and its MPAV version based on wavelet multi-scale decomposition with the following three anomaly detection algorithms:

(1) IMM algorithm [4]: uses the negative-selection machine from immunology to detect the novelty in Time Series Data.
(2) TSA-Tree algorithm [18]: the improved TSA-Tree used to achieve the anomaly pattern detection. TSA-Tree algorithm defines the anomaly pattern as the sudden change in time series, and finds these anomalies by calculating the local maxima of wavelet coefficients.
(3) Tarzan algorithm [10]: anomaly patterns are defined as the pattern whose frequency has significant difference with the expected frequency. The algorithm uses a suffix tree to encode the frequency of all observed patterns and allows a Markov model to predict the expected frequency of previously unobserved pattern. Once the suffix tree has been constructed, a measure of surprise for all the patterns in a new time series data set can be determined.

Experimental results of the five algorithms are shown in Fig. 4, MPAV-1, MPAV-2 and MPAV-3 express, respectively, the result of MPAV algorithm using one layer, two layer and three layer wavelet decomposition. The parameter is $e = 2$.

Fig. 4 shows IMM algorithm and ISA-tree algorithm cannot find the right anomaly in time series $Y_2(t)$, while PAV, MPAV and Tarzan can denote correctly and clearly the anomalies. But Tarzan must convert the time series into a symbolic string and built a suffix tree, so its space costs higher than MPAV; PAV algorithms detect the anomaly in the original series, and when compared with MPAV, which find the anomaly on the compressed series using wavelet transform, the efficiency of PAV is obviously lower than that of the latter.

Table 1 shows the change of the performance for MPAV in different compression ratios, when compression ratio is zero, MPAV and PAV is the same algorithm. According to the experimental results shown in Table 1, we can find with the increased levels of decomposition, compression ratio is increased, the efficiency of MPAV algorithm is improved significantly, but it can still find all anomaly patterns (see Fig. 4).

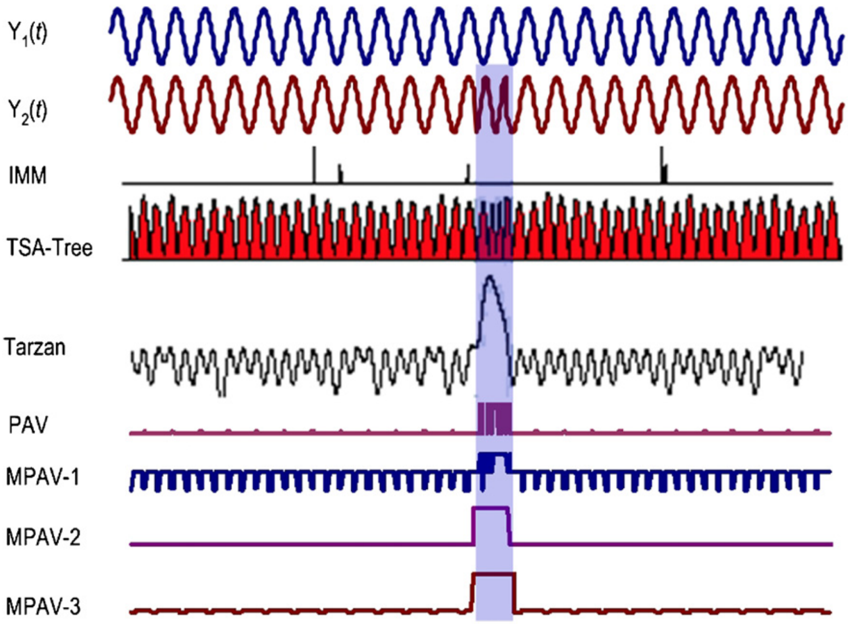

Fig. 4. The comparison of experimental results for five algorithms on time series $Y_2(t)$.

Table 1
The performance comparison between algorithms PAV and MPAV

Algorithm name		Length of original or compressed series	Compression ratio (%)	Time (s)
PAV		800	0	1.203
MPAV	Level = 1	400	50	0.658
	Level = 2	200	75	0.312
	Level = 3	50	93.75	0.297

6. Conclusions

We put forward to PAV and MPAV algorithms for anomaly detection of the time series in the paper. PAV algorithm defined the anomaly of time series as the infrequent linear pattern; MPAV algorithm obtains smaller storage and computational cost using wavelet transform to the time series. The experimental results show that either the MPAV or PAV algorithm can detect effectively the abnormal and needs no training. Especially, MPAV algorithm supports on different scales of the anomaly detection for time series by the wavelet multi-scale decomposition; it makes MPAV to fit a broader field of application.

Acknowledgments

We would like to thank Keogh Doctor for his helping in providing the experimental data sets. This research was partly funded by the Natural Science Foundation of Fujian Province under grant S0650013.

References

[1] Y. Alexander, R.P. Duin, Novelty detection using self-organizing maps, in: Progress in Connectionist-Based Information Systems, Springer, London, 1997, pp. 1322–1325.
[2] V. Barnett, T. Lewis, Outliers in Statistical Data, Wiley, New York, 1994.

[3] K.-P. Chan, W.-c. Fu, C. Yu, Haar wavelets for efficient similarity search of time-series: with and without time warping, IEEE Trans. Knowledge Data Eng. 15 (3) (2003) 686–705.
[4] D. Dasgupta, S. Forrest, Novelty detection in time series data using ideas from immunology, in: Proceedings of the Fifth International Conference on Intelligent Systems, Reno, Nevada, June 19–21, 1996.
[5] D. Decoste, Mining multivariate time-series sensor data to discover behavior envelopes, in: Proceedings of the Third Conference on Knowledge Discovery and Data Mining, AAAI Press, 1997, pp. 151–154.
[6] T. Edwards, Discrete wavelet transforms: theory and implementation, Technical Report, Stanford University, 1991.
[7] D. Hawkin, Identification of Outliers, Chapman & Hall, London, 1980.
[8] V. Jagadish, N. Koudas, S. Muthukrishnan. Mining deviants in a time series database, in: Proceedings of the 25th International Conference on Very Large Data Bases, Morgan Kaufmann Publishers Inc., San Francisco, 1999, pp. 102–113.
[9] M.F. Jiang, S.S. Tseng, C.M. Su, Two-phase clustering process for outliers detection, Pattern Recognition Lett. 22 (6/7) (2001) 691–700.
[10] E. Keogh, S. Lonardi, W. Chiu, Finding surprising patterns in a time series database in linear time and space, in: Proceedings of the Eigth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, New York, 2002, pp. 550–556.
[11] E.J. Keogh, K. Chakrabarti, M.J. Pazzani, S. Mehrotra, Dimensionality reduction for fast similarity search in large time series databases, Knowledge Inform. Systems 3 (3) (2001) 263–286.
[12] J. Lin, M. Vlachos, E. Keogh, D. Gunopulos, Iterative incremental clustering of time series, in: Proceedings of the IX Conference on Extending Database Technology (EDBT 2004), Crete, Greece, March 14–18, 2004.
[13] J. Ma, S. Perkins. Online novelty detection on temporal sequences, in: Proceedings of the International Conference on Knowledge Discovery and Data Mining, ACM Press, New York, 2003, pp. 24–27.
[14] L. Martin, A Mixture Approach to Novelty Detection Using Training Data With Outliers, Lecture Notes in Computer Science, vol. 2167, Springer, Berlin, 2001, pp. 300–310.
[15] K.B. Prat, E. Fink, Search for patterns in compressed time series, Internat. J. Image Graphics 2 (1) (2002) 89–106.
[16] S. Ramaswamy, R. Rastogi, K. Shim, Efficient algorithms for mining outliers from large data sets, in: Proceedings of the 2000 ACM SIGMOD International Conference on Management of Data, ACM Press, New York, 2000, pp. 427–438.
[17] SantaFe data, ⟨ http://www.psych.stanford.edu/∼andreas/Time-Series/SantaFe.html⟩ [EB/OL], 2006-3-20.
[18] C. Shahabi, X. Tian, W. Zhao, TSA-Tree: a wavelet-based approach to improve the efficiency of multi-level surprise and trend queries on time-series data, in: Proceedings of the 12th International Conference on Scientific and Statistical Database Management (SSDBM'00), IEEE Computer Society, Berlin, Germany, 2000, pp. 55–68.
[19] K. Yamanishi, J.-i. Takeuchi, A unifying framework for detecting outliers and change points from non-stationary time series data, in: Proceedings of the Eighth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM Press, New York, 2002, pp. 676–681.