Change-point Detection in Multivariate Time-series Data by Recurrence Plot

MIN HU[1,2], SHENGCHEN ZHOU[1,2], JIAJIE WEI[1], YUYAN DENG[1], WENTING QU[2,3]
1. SHU-UTS SILC Business School, Shanghai University
2. SHU-SUCG Research Center for Building Industrialization, Shanghai University
3. Department of Civil Engineering, Shanghai University
Shanghai, China
minahu@shu.edu.cn, zhoushengchen@shu.edu.cn

Abstract: - Change-point detection in time-series is an important data mining task with applications to abnormity diagnosis, events monitoring, climate change analysis, and other domains. This paper presents a novel method based on recurrence plot for detecting multiple change-points in multivariate time series. Bhattacharyya distance function is applied to improve the recurrence plot generation so as to capture the dependency change among variables. A window-based detection algorithm is proposed to capture the change-points quickly and automatically. With experiments on artificial and real datasets, we show that the algorithm has made improvement to traditional recurrence plot, is able to handle noisy data with optimized parameter, and can cope with complex situation like human activity and micro-blog events monitoring.

Key-Words: Change-point Detection, Multivariate Time Series, Recurrence Plot, Bhattacharyya Distance

1 INTRODUCTION

Change-point detection is the problem of finding abrupt changes in time-series, and it has become one of the research focuses in the field of statistics and data mining [1-3]. Recently, change point detection in multivariate time-series has been attracting increasing attention from researchers [4-8]. The techniques can be used in a variety of applications, such as detecting twitter events [6], human activities [7], financial events [8], music signal processing [9] and etc. Unlike traditional change-point detection with only one dimension, it is more challenging in consideration of the dependency among variables and complicated noise interference from more time series [10].

In this paper, we propose a direct change-point detection method based on recurrence plot, which is a visual representation of time series introduced by Eckmann et al. [11]. We introduce hybrid distance functions in recurrence plots generation, which can describe the dissimilarity between each pair of multivariate time stamps more effectively. Also, we propose an algorithm to detect change points almost automatically with the optimization parameters.

The remainder of this paper is organized as follows: In section 2, we give a brief literature review of the existing methods. In Section 3, we formulate our change-point detection problem. In section 4, we describe our proposed change-point detection algorithms based on recurrence plot. In section 5, we report empirical comparative results on various artificial and real-world datasets. Finally, we conclude this paper and state future works in section 6.

2 Related Works

In this section, we review several major approaches to change-points detection methods and research focuses on recurrence plot.

2.1 Change-point Detection

The existing methodologies for change-point detection are mainly based on statistical framework. From the perspective of probability distributions, the change-point refers to time stamp t where two interval distributions before and after t are significantly different. A common strategy comparing the difference between distributions is to compute the probability density ratio. Since probability density estimation is known to be a difficult problem, direct density-ratio based change-point detection approaches have been proposed [12]. Methods like Kullback-Leibler importance estimation procedure (KLIEP) [12], relative unconstrained least-squares importance fitting (RuLSIF) [6] and etc. have been applied to estimate the density ratio. This kind of method can be applied to multivariate time-series data.

Another group of methods is based on subspace analysis. For instance, singular-spectrum analysis is applied to sequential detection of changes in time series [13]. The main idea of singular-spectrum analysis is reconstructing the original time series subsequently into a trajectory matrix and then performing singular value decomposition of the matrix. In [14], subspace identification is used to detect change-points, which is a geometric approach comparing the subspaces spanned by the columns of an extended observability matrix and estimating linear state-space models behind time-series data. For multivariate time-series change-point detection, stationary subspace analysis is used to reduce the dimensionality of data to the most nonstationary directions, which are most informative for detecting state changes in the time series [15].

In consider of the dependency changes among variables for multivariate time-series data, paper [16] assume that the covariance structure of the series changes abruptly at change-point times, and an adaptive method is proposed to detect those change-points. Recently, researchers pay attention to visualization based method. For example, a novel method named “state transition graph” is proposed to identify the hazard (similar to change-point concept) [17]. The graph is generated by mapping state transition trajectory on Cartesian coordinate system. After extracting features from the graphs, the hazard can be identified through classification algorithms.

2.2 Recurrence Plot

Recurrence plot (RP) is used to plot trajectory of a time-series dynamical system in phase space [11]. With recurrence quantification analysis (RQA) [18] as a powerful tool, recurrence plot based methods have been used to study changes and transitions in the dynamics of a time-series system [19].

In data mining community, RP is being concerned by researchers. Existing algorithms can be categorized into two types:

- RPs similarity based methods. In [20], unthresholded RPs are used for similarity-based time series classification using the Campana-Keogh (CK-1) distance, and the authors plan to investigate their method in time-series anomaly detection task.

- Local feature based methods. In [21], the authors have already applied RPs to search for discord (abnormal subsequence) in time series. Discord is derived from distance vector that stores for each subsequence the distance to its nearest non-overlapping neighbor. In this paper, we will focus on the local feature of RP as well.

For multivariate time-series analysis, researchers have already extended recurrence plots to detect damage-induced changes to the structural dynamics in multivariate time-series data [22].

3 Problem Formulation

In this section, we formulate basic ideas of our change-point detection problem.

Let $x(t) \in \mathbb{R}^d$ be a d-dimensional time-series sample at time t. Let

$$X(t) = (x(t), x(t+\tau), \dots, x(t+(m-1)\tau)) \quad (1)$$

be reconstructed dynamics with embedding dimension m and delay τ. The recurrence plot is constructed by forming the recurrence matrix

$$R_{ij} = \Theta(\varepsilon - D_{ij}) \quad (2)$$

where ε is a threshold parameter representing the specific length scale of focus and D_{ij} measures the distance between $X(i)$ and $X(j)$. Heaviside function $\Theta(\cdot)$ decides whether the values of R_{ij} are 1 or 0 depending on D_{ij} greater or less than ε. Then, the recurrence matrix can be transformed into an image where R_{ij} represent a pixel point, which called recurrence plot.

As shown in Figure 1, different state patterns of multivariate time series can be easily observed by differentiating color areas. Abrupt changes often take place when one system state transit to another state [18], so change-points on Figure 1 can be obviously caught at time stamps around 250 and 500, which are boundaries of different state patterns.

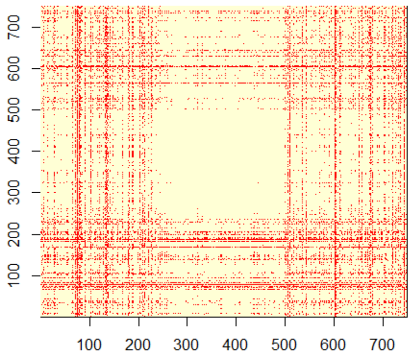


Fig.1. An illustrative example of recurrence plot generated by three-dimensional time-series data with two change-points ($\varepsilon = 0.75, m = 1, \tau = 1$)

Now the problems that need to be addressed are how to characterize the dissimilarity between multivariate timestamps effectively, and how to extract the change-points from recurrence plots quickly. We will discuss these issues in the next section.

4 Methodology

In this section, we first define our dissimilarity measure, and then show details in processes of methods to capture the change-points from recurrence plot.

4.1 Dissimilarity Measure

The key element of generating recurrence plot is the dissimilarity measure used to determine the distance between two given observations. Although there are numerous distance measures proposed in the literature [23], few of them are applicable to multivariate time series. In this paper, we measure the distance between two timestamps using two distance functions: Euclidean distance function and Bhattacharyya distance function.

The Euclidean distance is widely used to measure the dissimilarity between two points in a Euclidean space, shown in Eq. (3):

$$D_E(X,X') = \sqrt{\sum_{i=1}^{N}(X_i - X'_i)^2} \tag{3}$$

where X and X' are two timestamps with N variables respectively. Though the Euclidean distance is very useful, it cannot capture the changes of dependency among variables. To compensate, Bhattacharyya distance is applied in this paper, which can successfully measure the dissimilarity of variable distributions at two timestamps, shown in Eq. (4):

$$D_B(X,X') = \sqrt{1 - \sum_{i=1}^{N}\sqrt{X_i X'_i/(Y_i Y'_i)}} \tag{4}$$

where Y_i and Y'_i equal the sum of variable values at each timestamp.

The two functions mentioned above can be used to construct two dissimilarity matrices by computing the dissimilarity between every pair of timestamps in given time series.

4.2 Generation of Recurrence Plot

As described above, we need to convert the dissimilarity matrices into recurrence plot. We use Eq. (2) to transform dissimilarity matrices into symmetric boolean matrices E and B. For simplicity, here ε is computed as the 75th percentile of the distance value, which can make significant representation of dynamical structure of recurrence plots, and delay τ is fixed to be 1. Then, we integrate them into one matrix by Eq. (5).

$$R_{ij} = E_{ij} \wedge B_{ij} \tag{5}$$

where R is meet of E and B. $R_{ij} = 1$ if and only if $E_{ij} = B_{ij} = 1$. Note, R can retain the characters of both Bhattacharyya and Euclidean distance functions' effects. Finally, the resulting matrix R can be converted into recurrence plot by pixelate.

4.3 Window-based Detection Algorithm

This section presents our proposed multivariate time series change-point detection algorithm. A high-level summary of the algorithm is shown in Algorithm 1. We first normalize the original multivariate time series $\mathcal{D}$, the aim is to make data suitable for distance function computing, shown in Eq. (6):

$$\mathcal{S}(i) = 1 + \frac{\mathcal{D}(i) - \mathcal{D}(i)_{min}}{\mathcal{D}(i)_{max} - \mathcal{D}(i)_{min}} \tag{6}$$

where $\mathcal{D}(i)$ is the ith time series of $\mathcal{D}$. We rescale the range of each variable in $[1,2]$.

Algorithm 1 Window-based Change-points Detection on Recurrence Plot

Input: Multivariate time series $\mathcal{D}$

Output: Change-point vector c

Method:

1. $\mathcal{S} \leftarrow \text{Normalize}(\mathcal{D})$
2. $\mathcal{R} \leftarrow \text{RPGenerate}(\mathcal{S})$
3. $\boldsymbol{v} \leftarrow \text{WindowSlide}(\mathcal{R})$
4. $c \leftarrow \text{CPCapture}(\boldsymbol{v})$

Next, we generate RP matrix $\mathcal{R}$ as discussed in 4.2. It is possible for us to capture the change-points directly from RP by observation, while locating the multiple change-points automatically and quickly is still in demand. Here we implement a window-based method to reach the objective. Unlike the generic sliding window algorithm used in time series analysis, we defined sliding window on recurrence plot matrix as a $n \times n$ small matrix which slides along the diagonal. Then, we compute the color value $\boldsymbol{v}(t)$ of the sliding window at each time stamp.

$$\boldsymbol{v}(t) = \frac{1}{n^2}\sum_{i,j=t-n}^{t} R_{ij}\ , t \geq n \tag{7}$$

To estimate the change-points, we set a threshold β (usually equals zero) to transform $\boldsymbol{v}$ into 0-1 Boolean sequence, as shown in Eq. (8).

$$\boldsymbol{v}'(t) = \begin{cases} 1, & \boldsymbol{v}(t) \geq \beta \\ 0, & \boldsymbol{v}(t) < \beta \end{cases} \tag{8}$$

Finally, change-points vector c can be captured by computing the difference of $\boldsymbol{v}'$ at each two time stamps, and finding where the diff. is significant. It is worth mentioned that the derived change-points should be adjusted according to situation whether the color value of sliding window is increasing or descending. Eq. (9) gives the strategy to make an adjustment.

$$\hat{c_i} = \begin{cases} c_i + 2, & \boldsymbol{v}'(c_i + 1) = 0 \\ c_i + 2 + n, & \boldsymbol{v}'(c_i + 1) = 1 \end{cases} \tag{9}$$

where n is the window width.

4.4 Selection of Embedding Dimension

Embedding dimension is closely related to phase space reconstruction of multivariate time-series data, which influences the clearness of generated recurrence plot. Since our method is defined on the region edges of recurrence plot, it is essential to make plot clear enough with a proper embedding dimension. Here, we propose a novel method. Eq. (10) shows the optimization problem, where m indicates the estimated embedding dimension, μ is the mean of color value at different timestamps and k is the length of color value vector.

$$\max_{m \in \mathbb{R}^d} \sqrt{\frac{1}{k}\sum_{t=1}^{k}[v(t)-\mu]^2} \qquad (10)$$
$$\text{s.t. } m_{1,\dots,}m_d \in N^+$$

The above equation can be explained as finding a proper embedding dimension to maximize the standard deviation of color value vector. Thus, the estimated m could improve the clearness of recurrence plot.

5 Experiments

In this section, we perform some experiments on both synthetic and real-world datasets to test the performance of the proposed method. All the experiments were conducted on a Windows 7 machine with 2.30GHz CPU and 4.00 GB RAM. Note, we regard the correct detection at t with true alarm is between t-10 and t+10, and window width n used in change-point capture is 50.

5.1 Artificial Datasets

Here, we use the following two artificial time-series datasets that contain manually inserted change-points.

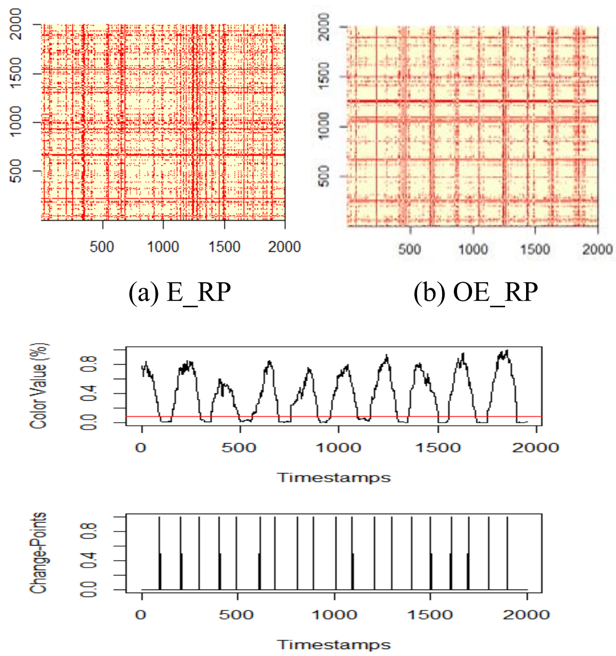
Dataset 1 (Switching covariance): The following covariance matrix Σ at time t (borrowed from the paper [6]) is used for generating 2000 2-dimensional samples

$$\Sigma = \begin{cases} \begin{pmatrix} 1 & -\frac{4}{5}-\frac{N-2}{500} \\ -\frac{4}{5}-\frac{N-2}{500} & 1 \end{pmatrix} \\ \begin{pmatrix} 1 & \frac{4}{5}+\frac{N-2}{500} \\ \frac{4}{5}+\frac{N-2}{500} & 1 \end{pmatrix} \end{cases} \qquad (11)$$

where N is a natural number such that $100(N-1)+1 \le t \le 100N$.

Dataset 2 (Tail behavior): 750 2-dimensional data are drawn from a bivariate normal distribution and a bivariate t-distribution with 2 degrees of freedom. Change-points are located in timestamp 250 and 500.

Firstly, we compare the performance of the proposed method (OE_RP) with traditional recurrence plot method (E_RP) using Euclidean function as dissimilarity measure between every two timestamps. Figure 2-(a), (b) show the generated recurrence plots by two methods from Dataset 1 ($m=1, \tau=1$). With the same length scale of focus ε, OE_RP is less sensitive to noise than E_RP, leading to a much clearer plot. Therefore, OE_RP is better than E_RP to identify and capture the change-points from plots by running algorithm 1. Figure 2-(c) gives the capture results of dataset1 by OE_RP, with accuracy of 79% under $\beta = 0.08$.

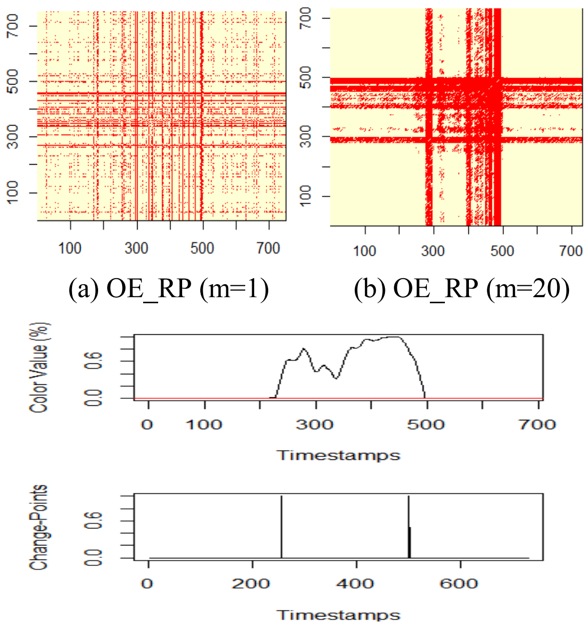


(a) E_RP (b) OE_RP

(c) Color value & change-points of OE_RP (β=0.08)

Fig.2. Results for Dataset 1($m = 1$)

Secondly, we illustrate an example to evaluate section 4.4. We use Eq. (10) to estimate proper embedding dimension of dataset 3 with a small training set, including 100 observations. We call this is a focusing process, just like finding a focal distance before taking a photo. As shown in Figure 4, the estimated embedding dimension is 20, with the highest standard deviation of color value vector. The resulting OE_RP [m=20, Figure 3-(b)] is much clearer than the one with common one embedding dimension [m=1, Figure 3-(a)]. We can easily obtain two change-points at timestamp 255 and 502, with 100% accuracy [m=1, Figure 3-(c)].



(a) OE_RP (m=1) (b) OE_RP (m=20)

(c) Color value & change-points (m = 20, β=0)

Fig.3. Results for Dataset 2

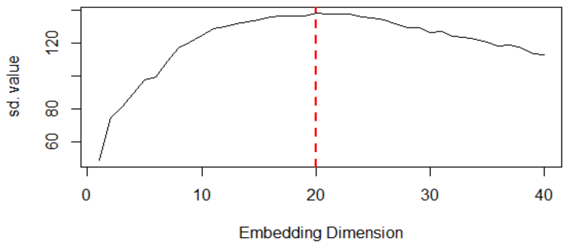


Fig.4. Selection of Embedding Dimension on Dataset 2

5.2 Real-World Datasets

Next, we evaluate the performance of our method using two real-world datasets.

We include the following methods in our comparison.

RuLSIF [6]: RuLSIF is a statistical change-point detection algorithm based on non-parametric divergence estimation between time-series samples from two retrospective segments. The divergence measure is relative Pearson divergence, estimated by direct density-ratio estimation. The authors provide website address to download codes in the paper. Therefore, it is convenient for us to compare their methods with ours.

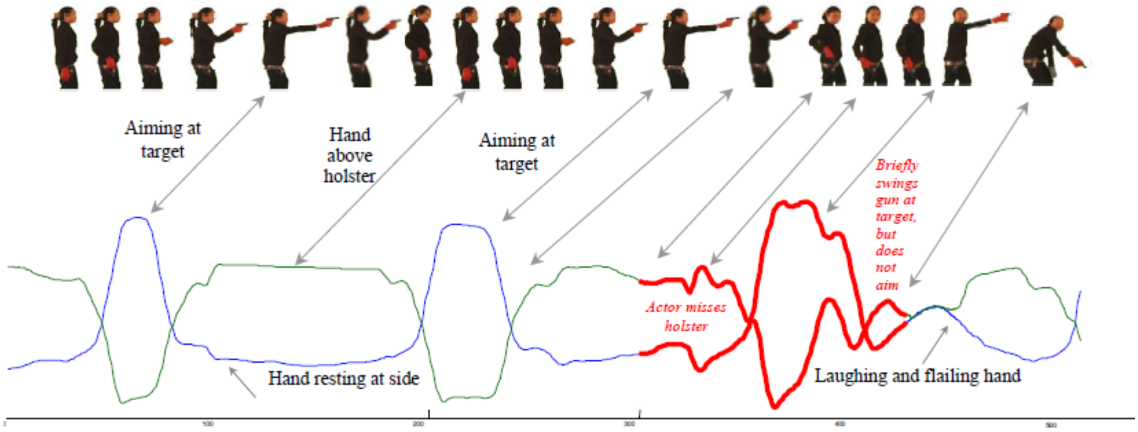
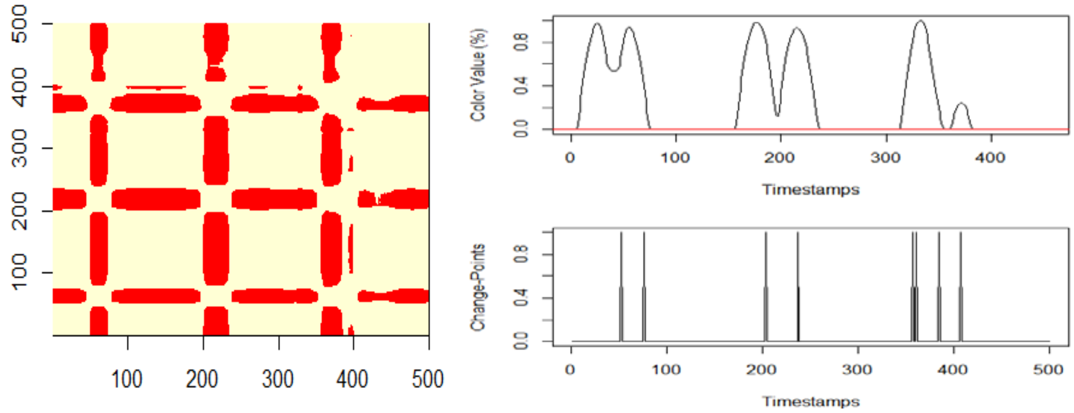
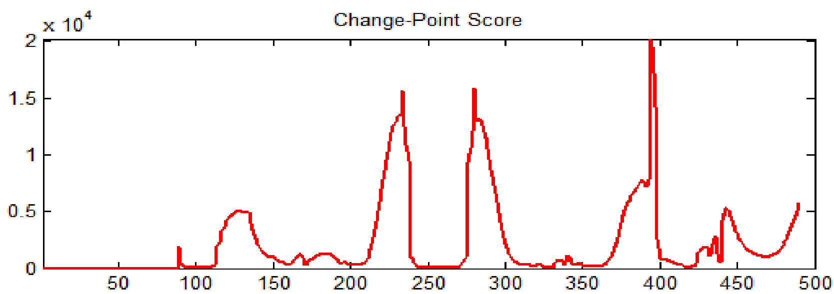


Fig.5. Description of video surveillance dataset [24]



(a) Change-points obtained by OE_RP-based method ($m = 1, \beta = 0, n = 50$)



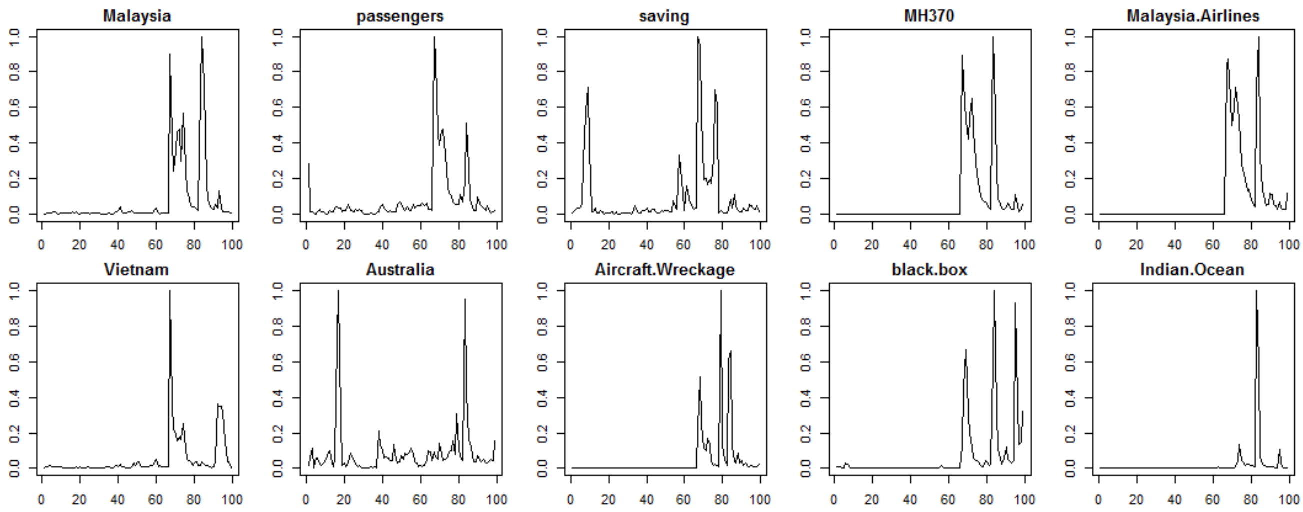
(b) RuLSIF (n =45, k=10,α= 0)

Fig.6. Results for video surveillance dataset

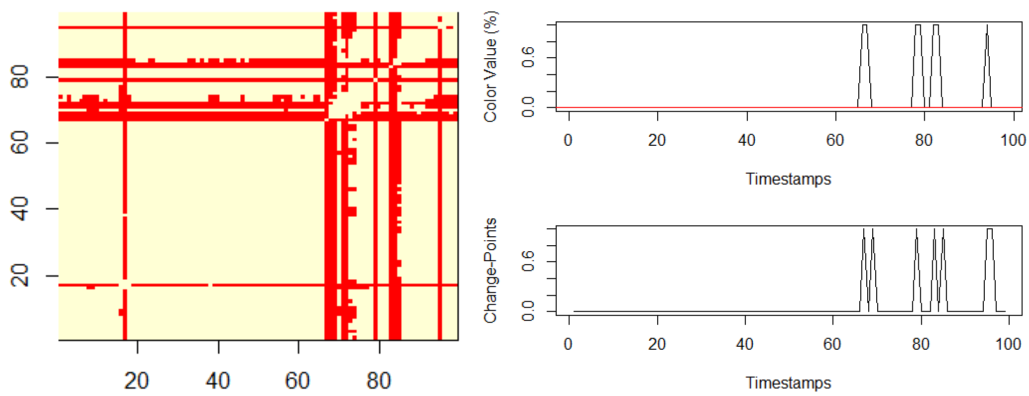
5.2.1 Video Surveillance Dataset

We consider a 2D UCR [25] time-series dataset extracted from a video of an actor performing various actions with and without a replica gun, as shown in Figure 5. The two time-series measure the X and Y coordinates of the actor's right hand. This dataset has several subjects performing activities such as: aiming at target, hand above holster, laughing and flailing hand, etc. Here, the intersection points are regarded as change-points.

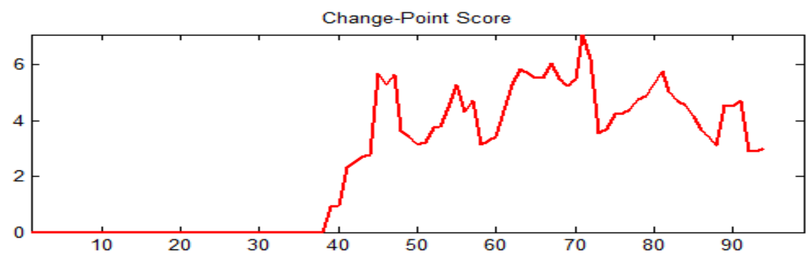
As shown in Figure 6-(a), seven defined change-points have been successfully captured, while RuLSIF-based method, shown in Figure 6-(b), cannot locate all the defined change-points accurately. From the OE_RP in Figure 6-(a), we can also identify the abnormal subsequence near timestamp 400, when the actor didn't aim at target as before. In addition, we found that tuning parameters for our proposed method is much easier than RuLSIF-based method. In the RP generation step, the Eq. (10) can help us to select the best parameter m, while in many cases, $m = 1$ is sufficient. By observing the recurrence plot, we can roughly determine the parameters n and β. However, selecting the parameters is not intuitive for RuLSIF-based method.



(a)Normalized Frequencies of 10 keywords



(b)Change-points obtained by OE_RP-based method ($m = 1$, β=0, n=2)



(c) Change-point Score obtained by RuLSIF -based method

Fig.7. Results for Micro-blog dataset

5.2.2 Micro-blog Dataset

The second real-world dataset we use here is the micro-blog hot keywords dataset, which collected from January 2014 to April 2014 via the Weibo.com application programming interface (inspired by paper [6]). Here we monitor the frequencies of selected ten keywords related to “Malaysia Airlines Flight 370” that lost contact on March 8th, 2014. The ten keywords are “Malaysia”, “passengers”, “saving”, “MH370”, “Malaysia Airlines”, “Vietnam”, “Australia”, “Aircraft Wreckage”, “black box” and “Indian Ocean” in Chinese. We perform our method directly on the 10-dimensional data. Figure 7-(a) shows the normalized frequencies of the 10 keywords.

Considering the timeline of the event development, Figure 7-(b) based on OE_RP method gives us several change-points corresponding to important news respectively. For instance, the detected timestamp 67 (March 8th) was the initial time of event, most of the keywords had unexpected growth. On March 20th (detected timestamp 79), aircraft wreckage was suspected to find. Shortly after Malaysian Prime Minister announced that MH370 had lost in southern Indian Ocean, the frequencies of related keywords are erupted on Micro-blog at timestamp 83. Finally, captured timestamp 95 (April 5th) illustrates when China claimed they had found the signal of black box of MH370.

However, it is difficult to explain the event development by RuLSIF-based method in this case. OE_RP-based method can locate the change-points more directly and accurately, with less false alarm rate.

The above two experiments have shown the ability of OE_RP based method to detect change-points in complex datasets to some extent. In comparison with RuLSIF-based method, we did not use the common receiver operating characteristic (ROC) curve or confusion matrix measurement, as the performance of our OE_RP based method is significant. Nevertheless, we will test our method on more real-world datasets and compare it with other existing methods. The aim of this paper is to give a quick introduction to improved recurrence plot for multivariate time-series data, and its effective application in change-point detection problem for data mining purpose.

6 CONCLUSION

In this paper, we proposed a change-point detection method based on recurrence plot for multivariate time-series data. Bhattacharyya distance function was introduced to improve the generation of recurrence plot in consider of capturing the dependency change among different variables. We proposed a novel window-based algorithm for change-points detection and demonstrated on artificial and real-world datasets that our approach is significantly effective. We have also applied our methods in micro-blog event analysis.

Following the current line of research, there are several issues to be pursued if we are to improve the performance and robustness. For example, the time complexity of our algorithm is $O(n^2)$, which is not efficient enough. Thus, we need to develop a faster algorithm. We have already taken into account optimizing critical parameters and verified experimentally, while it is in demand to prove the optimization method in theory. In future works, we plan to investigate the ability of our proposed method in outlier detection, which is also a hot topic of data mining in recent years.

Acknowledgments

This work is supported by Shanghai Science and Technology Development Funds (13511504803). The authors are grateful to Prof. Josip Music and the anonymous reviewers to improve this paper.

REFERENCES:

[1] M. Basseville and I. V. Nikiforov, Detection of abrupt changes: theory and application: Prentice-Hall, Inc., Englewood Cliffs, N. J., 1993.

[2] Y. Kawahara, T. Yairi, and K. Machida. Change-point detection in time-series data based on subspace identification. In Proceedings of the 7th IEEE International Conference on Data Mining, pages 559–564, 2007.

[3] Yamanishi and 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, 2002, pp. 676-681.

[4] A. Lung-Yut-Fong, C. Lévy-Leduc, and O. Cappé, "Homogeneity and change-point detection tests for multivariate data using rank statistics," arXiv preprint arXiv:1107.1971, 2011.

[5] H. Cho and P. Fryzlewicz, “Multiple change-point detection for high-dimensional time series via sparsified binary segmentation,” The London School of Econ. and Political Sci., London, U.K., Tech. Rep., 2014.

[6] S. Liu, M. Yamada, N. Collier, and M. Sugiyama, "Change-point detection in time-

series data by relative density-ratio estimation," Neural Networks, vol. 43, pp. 72-83, 2013.
[7] M. Yamada, A. Kimura, F. Naya, and H. Sawada, "Change-point detection with feature selection in high-dimensional time-series data," in Proceedings of the Twenty-Third international joint conference on Artificial Intelligence, 2013, pp. 1827-1833.
[8] D. S. Matteson and N. A. James, "A nonparametric approach for multiple change point analysis of multivariate data," Journal of the American Statistical Association, vol. 109, pp. 334-345, 2014.
[9] F. Desobry, M. Davy, and C. Doncarli, "An online kernel change detection algorithm," Signal Processing, IEEE Transactions on, vol. 53, pp. 2961-2974, 2005.
[10] H. Cheng, P.-N. Tan, C. Potter, and S. A. Klooster, "Detection and Characterization of Anomalies in Multivariate Time Series," in SDM, 2009, pp. 413-424.
[11] J.-P. Eckmann, S. O. Kamphorst, and D. Ruelle, "Recurrence plots of dynamical systems," Europhys. Lett, vol. 4, pp. 973-977, 1987.
[12] Y. Kawahara and M. Sugiyama. Change-point detection in time-series data by direct density-ratio estimation. In SDM, pages 389–400, 2009.
[13] V. Moskvina and A. Zhigljavsky. Change-point detection algorithm based on the singularspectrum analysis. Communications in Statistics: Simulation and Computation, 32: 319–352, 2003b.
[14] Y. Kawahara, T. Yairi, and K. Machida. Change-point detection in time-series data based on subspace identification. In Proceedings of the 7th IEEE International Conference on Data Mining, pages 559–564, 2007.
[15] D. A. Blythe, P. von Bunau, F. C. Meinecke, and K. Muller, "Feature extraction for change-point detection using stationary subspace analysis," Neural Networks and Learning Systems, IEEE Transactions on, vol. 23, pp. 631-643, 2012.
[16] M. Lavielle and G. Teyssiere, "Detection of multiple change-points in multivariate time series," Lithuanian Mathematical Journal, vol. 46, pp. 287-306, 2006.
[17] M. Hu, F. F. Wu, B. Zhu, B. Lu, and J. L. Pu, "A New Hazard Identification Method-State Transition Graph," Applied Mechanics and Materials, vol. 48, pp. 71-78, 2011.
[18] C. Webber and J. P. Zbilut, "Dynamical assessment of physiological systems and states using recurrence plot strategies," Journal of Applied Physiology, vol. 76, pp. 965-973, 1994.
[19] N. Marwan, S. Schinkel, and J. Kurths, "Recurrence plots 25 years later—Gaining confidence in dynamical transitions," EPL (Europhysics Letters), vol. 101, p. 20007, 2013.
[20] V. M. A. d. S. Diego F. Silva, Gustavo E. A. P. A. Batista, "Time Series Classification Using Compression Distance of Recurrence Plots," in ICDM, 2013, pp. 687-696.
[21] L. Wei, M. Gallagher, and J. Wiles, "Parameter-free Search of Time-series Discord," Journal of Computer Science and Technology (English Language Edition), vol. 28, pp. 300-10, March 2013
[22] J. Nichols, S. Trickey, and M. Seaver, "Damage detection using multivariate recurrence quantification analysis," Mechanical systems and signal processing, vol. 20, pp. 421-437, 2006.
[23] X. Wang, A. Mueen, H. Ding, G. Trajcevski, P. Scheuermann, and E. Keogh, "Experimental comparison of representation methods and distance measures for time series data," Data Mining and Knowledge Discovery, vol. 26, pp. 275-309, 2013.
[24] E. Keogh, Zhu, Q., Hu, B., Hao. Y., Xi, X., Wei, L. & Ratanamahatana, C. A. (2011). The UCR Time Series Classification/Clustering Homepage. Available: www.cs.ucr.edu/~eamonn/time_series_data/