A New Study of Two Divergence Metrics for Change Detection in Data Streams

Abdulhakim Qahtan[1] and **Suojin Wang**[2] and **Raymond Carroll**[2] and **Xiangliang Zhang**[1]

Abstract. Streaming data are dynamic in nature with frequent changes. To detect such changes, most methods measure the difference between the data distributions in a current time window and a reference window. Divergence metrics and density estimation are required to measure the difference between the data distributions. Our study shows that the Kullback-Leibler (KL) divergence, the most popular metric for comparing distributions, fails to detect certain changes due to its asymmetric property and its dependence on the variance of the data. We thus consider two metrics for detecting changes in univariate data streams: a symmetric KL-divergence and a divergence metric measuring the intersection area of two distributions. The experimental results show that these two metrics lead to more accurate results in change detection than baseline methods such as Change Finder and using conventional KL-divergence.

1 Introduction

Discovering changes in data streams is a widely studied problem, which covers applications such as intrusion detection in networking and suspicious motion detection in vision systems. Change detection methods generally fall into two categories. The first detecting strategy is based on comparing the distribution in current stream window with a reference distribution [4]. Density estimation and divergence metrics are designed to evaluate and compare the distributions. The second type of approaches are based on prediction [8]. Changes are reported when samples deviate from a predictive model.

Kifer et al. [4] is an example of a window-based change detection framework. At each time step, the distance of the data distribution in the *reference window* (first m_1 samples arrived after a reported change point) and *test window* (newest m_2 samples in the stream) is measured. A change is reported if the distance is above a threshold.

The prediction-based approach is less popular than the window-based ones. In [8], changes are detected through checking if there are a large number of outliers in the time series. The detection accuracy highly depends on the prediction algorithm used for reporting outliers that deviate from the predicted value.

In this paper, we study change detection method that falls under the first category, where divergence metrics are the essential component for measuring the difference between reference and test windows. The most popular distribution divergence metric is the Kullback-Leibler (KL) divergence [5]. KL-divergence is asymmetric non-negative metric that is affected by the type of change in data variance (from large to small or from small to large). If the distribution P has larger variance value than the distribution Q, then $D_{KL}(P||Q)$ is much larger than $D_{KL}(Q||P)$. Therefore, algorithms employing the KL-divergence fail to detect some changes with decreasing variance or detect them with a large delay.

To overcome the asymmetric property of KL-divergence, two metrics for valid distribution comparisons are used for the first time for online change detection: a modified symmetric KL-divergence and a measure of intersection area of two distributions. Our approaches are named accordingly CD-MKL and CD-Area. Note that these two metrics are used for the first time for online change detection.

We validated the proposed change detection framework on several synthetic datasets, including various changes. We also compared them with two baseline methods, Change Finder [8] and KDE-KL [3]. The experimental results show that both CD-MKL and CD-Area are more accurate in detecting changes. In addition, CD-Area is generally better than CD-MKL as it detects most of the changes.

2 CD-MKL and CD-Area Algorithm

Algorithm 1 presents our framework for change detection in streaming data. The symbol D_M denotes any divergence metric.

Algorithm 1 Framework of CD-MKL and CD-Area

Parameters: window size w
Online flow in: streaming data $S = \{x_1, ..., x_t, ...\}$
Online output: time t when detecting a change
Procedure:
1: Initialize $t_c = 0$
2: Set reference window $S_1 = \{x_{t_c+1}, ..., x_{t_c+w}\}$
3: Estimate $\hat{f}_1$ using data in S_1
4: Clear S_1
5: Set test window $S_2 = \{x_{t_c+w+1}, ..., x_{t_c+2w}\}$
6: Estimate $\hat{f}_2$ using data in S_2
7: **while** a new sample x_t arrives in the stream **do**
8: remove x_{t-w} from S_2
9: update $\hat{f}_2$ using x_t and x_{t-w}
10: **if** reference window $< 2w$ **then**
11: update $\hat{f}_1$ using x_{t-w}
12: **end if**
13: **if** $mod(t, 0.05w) = 0$ **then**
14: Score = $D_M\left(\hat{f}_2||\hat{f}_1\right)$
15: Compute the threshold τ_t
16: **if** $Score > \tau_t$ **then**
17: Report a change at time t
18: Clear S_2 and GOTO step 2
19: **end if**
20: **end if**
21: **end while**

Line 2 in Algorithm 1 sets the reference window S_1 to be the first w samples arriving after the change point t_c. Intuitively, when a data distribution shifts to a new one, the reference window should be updated to represent the new distribution. This update also enables the

[1] King Abdullah University of Science and Technology, Thuwal 23955, Jeddah, KSA
[2] Department of Statistics, Texas A&M University, College Station, TX 77843, USA.

detection of further changes. Line 5 sets the test window S_2 as a collection of w samples after the reference window. This S_2 will slide along the data stream to include the newest w samples (lines 7, 8).

The window size w is usually set according to the application problems. A small window size will allow for detecting short term changes and reducing the delay but may lead to false positives. A large window size will make the algorithm more robust but may miss alarms. The setting of this parameter is usually left to the user in order to give them the ability for monitoring the long/short term changes, depending on their interests and the application sensitivity.

Change scores are computed by using a divergence metric on two density functions $\hat{f}_1$ and $\hat{f}_2$ (line 14), which are updated upon each sample arrival. The divergence metrics are crucial for computing change scores. In this study, we focus on three important divergence metrics. The Kullback-Leibler (KL) divergence [5] is defined as:

$$D_{KL}\left(\hat{f}_2||\hat{f}_1\right) = \int_x \hat{f}_2(x) \log\left(\frac{\hat{f}_2(x)}{\hat{f}_1(x)}\right) dx, \quad (1)$$

where $\hat{f}_i$ is the Probability Density Function (PDF) estimated from S_i, $i = 1, 2$. The D_{KL} is a nonnegative (≥ 0) and nonsymmetric measure. It is 0 when the two distributions are completely identical, and becomes larger as the two distributions deviate from each other. The nonsymmetry property of the D_{KL} complicates the procedure of setting the threshold for detecting changes in data streams.

To overcome the problem of the KL-divergence, we use a modified symmetric KL-divergence [6]

$$D_{MKL} = \max\left(D_{KL}\left(\hat{f}_1||\hat{f}_2\right), D_{KL}\left(\hat{f}_2||\hat{f}_1\right)\right). \quad (2)$$

The metric was used in [6] for evaluating the correlation between two matching scores in optimal feature selection and shown to be more robust than $SKL = D_{KL}\left(\hat{f}_1||\hat{f}_2\right) + D_{KL}\left(\hat{f}_2||\hat{f}_1\right)$ used in [2].

The second divergence metric is the intersection area under the curves of two density functions [1]. This test can be formulated as:

$$D_A\left(\hat{f}_2||\hat{f}_1\right) = 1 - \int_x \min\left(\hat{f}_1(x), \hat{f}_2(x)\right) dx. \quad (3)$$

This D_A takes values in $[0, 1]$, where the value one means completely different distributions and zero means two identical distributions.

The PDFs required by the divergence metrics must be accurately and timely estimated. KDE-Track, a dynamic density estimator we studied in [7], adapts KDE to handle the evolving underlying distribution in data streams. It gains linear time complexity by adopting linear interpolation and adaptive resampling.

The threshold at time t is set to be $\tau_t = \bar{D}_M^t + 3e(w)$, where $\bar{D}_M^t$ is the accumulated mean of change scores, and $e(w)$ is a function of w that approximates the error in computing the change score. Most of the change score values are in the interval $(\bar{D}_M - 3e(w), \bar{D}_M + 3e(w))$. Values outside this interval are extreme points and indicate changes. This adaptive threshold automates the monitoring of changes under different settings of window w. Therefore, our threshold setting is superior to the fixed setting of threshold, which requires users' prior knowledge for different application data sets.

Table 1. Evaluation results in terms of precision (P) and recall (R) of the four methods on the five datasets in percentage.

Dataset	CF		KDE-KL		CD-MKL		CD-Area	
	P	R	P	R	P	R	P	R
Data1	100	66.7	100	88.9	100	88.9	100	88.9
Data2	100	58.3	100	91.7	100	91.7	100	100
Data3	100	55.6	100	100	100	100	100	100
Data4	100	79.6	100	83.7	100	89.8	100	91.8
Data5	98.2	70.5	100	69.2	100	71.8	100	94.9

3 Experimental Evaluation

The performance of CD-MKL and CD-Area is compared with the performance of Change Finder (CF) [8] and KDE-KL [3] on several synthetic datasets. KDE-KL employs the traditional KL-divergence to compare the distributions in the reference and test windows, which is similar to our framework but uses fixed threshold value and the traditional kernel density estimator. The sliding window size is set to $w = 2 * 10^3$ for the three window-based methods.

Data1 is an example of a *mean shift* with varying amount of change $\mu_k = \mu_{k-1} + 9 - k$. The dataset was generated from the normal distribution ($\mathcal{N}(\mu_k, 1)$) with $\mu_1 = 0$. Changes happen at the time points $k * 10^5, k = 1, 2, \cdots, 9$. Data2 was generated based on Data1 by adding changes of variance. Data3 is an example of *jumping variance*, where the standard deviation changes from 1 to 3 and back to 1. Data4 consists of 10^6 samples, where the mean and variance randomly change every $2 * 10^4$ samples. Data5 contains 100 data segments of size $2 * 10^4$. Each data segment was extracted from normal distribution $\mathcal{N}(0, \sigma_i^2)$ with $\sigma_i \in \{1, 3, 5, 7, 9\}$. The dataset has 78 change points as the data distribution is not changing at the starting of each segment.

The performance of the four methods is compared in Table 1 in the terms of the precision $P = \frac{TP}{TP+FP}$ and the recall $R = \frac{TP}{TP+FN}$, where TP = true positives, FP = false positives and FN = false negatives. The results show that KDE-KL, CD-MKL and CD-Area produce better results than CF on Data1-3 when the changes are easy to be detected. CD-Area outperforms the other methods on Data4 and Data5, which contain changes that are harder to detect. Note that KDE-KL is computationally much more expensive. For example, when obtaining the results for Data5, KED-KL took 5260 seconds, compared with 4429, 65, and 50 seconds for CF, CD-MKL, and CD-Area, respectively.

4 Conclusion

In this paper, we present a framework for detecting changes in data streams with two more effective metrics, a symmetric version of the KL-divergence (D_{MKL}) and the area metric (D_A). These two metrics also enables the automatic setting of the detection threshold τ_t. Evaluation results show that the presented CD-Area and CD-MKL approaches perform better than the baseline methods for reporting changes accurately.

REFERENCES

[1] S. Cha, 'Comprehensive survey on distance/similarity measures between probability density functions', *Intl. J. of Math. Models and Methods in App. Sci.*, **1**, 300–307, (2007).

[2] J. Inglada and G. Mercier, 'A new statistical similarity measure for change detection in multitemporal sar images and its extension to multiscale change analysis', *IEEE Transaction on Geoscience and Remote Sensing*, **45**, 1432–1445, (2007).

[3] Y. Kawahara and M. Sugiyama, 'Change-point detection in time-series data by direct density-ratio estimation', in *SDM*, (2009).

[4] D. Kifer, S. Ben-David, and J. Gehrke, 'Detecting change in data streams', in *VLDB*, (2004).

[5] S. Kullback and R. A. Leibler, 'On information and sufficiency', *Annals of Mathematical Statistics*, **22**, 79–86, (1951).

[6] D. Liu, D. Sun, and Z. Qiu, 'Feature selection for fusion of speaker verification via maximum kullback-leibler distance', in *ICIP*, (2010).

[7] A. Qahtan, X. Zhang, and S. Wang, 'Efficient estimation of dynamic density functions with an application to outlier detection', in *CIKM*, (2012).

[8] J. Takeuchi and K. Yamanishi, 'A unifying framework for detecting outliers and change points from time series', *TKDE*, **18**, 482–492, (2006).