Zhang and Chen *EURASIP Journal on Wireless Communications and Networking* (2019) 2019:86
https://doi.org/10.1186/s13638-019-1408-2

EURASIP Journal on Wireless Communications and Networking

RESEARCH Open Access

Research on detection and integration classification based on concept drift of data stream

Baoju Zhang* and Yidi Chen

Abstract

As a new type of data, data stream has the characteristics of massive, high-speed, orderly, and continuous and is widely distributed in sensor networks, mobile communication, financial transactions, network traffic analysis, and other fields. However, due to the inherent problem of concept drift, it poses a great challenge to data stream mining. Therefore, this paper proposes a dual detection mechanism to judge the drift of concepts, and on this basis, the integration classification of data stream is carried out. The system periodically detects data stream with the index of classification error and uses the features of the essential emerging pattern (eEP) with high discrimination to help build the integrated classifiers to solve the classification mining problems in the dynamic data stream environment. Experiments show that the proposed algorithm can obtain better classification results under the premise of effectively coping with the change of concepts.

Keywords: Data stream, Concept drift detection mechanism, Essential emerging pattern, Integration classification

1 Introduction

With the continuous advancement of information technology and the rapid development of computer networks, the real world has generated a large number of data stream, such as weather monitoring data, stock trading data, and network access logs, etc. And as time goes on, the amount of data is constantly expanding, resulting in unstable data distribution, which is easy to generate drifting of concepts. At this point, timely identification data stream with concept changes and accurate classification has become a research hotspot of data mining.

In recent years, the problems of concept drift has attracted more and more scholars' attention, and it has also proposed more reasonable solutions. In general, the mainstream algorithms for dealing with concept drift can be summarized as two types: direct algorithms and indirect algorithms. Initially, the most popular algorithms use a number of detection metrics to directly judge concept drift, such as the most commonly used entropy values [1] and error rates, and judging these metrics can measure concept changes, and even to estimate the degree of drift.

In addition to the above, other scholars indirectly judge the drift by the process of classification. In 2000, Street proposed the SEA (Streaming Ensemble Algorithm) [2], which introduced the integration learning to classification of data stream with concept drift for the first time. This method achieved a rapid response to the change of concepts and proved that it can adapt to any size of data stream. In 2007, the DWM (Dynamic Weighted Majority) algorithm was proposed in [3], which dynamically adjusted the weight of each base classifier for integration and effectively tracking the abrupt concept drifts. Sun et al. [4, 5] proposed an online integration classification algorithm, which updated the weight of the base classifier online and added or deleted the base classifier by weights, thus solving the classification problem of dynamic data stream while adapting to concept drift.

Based on the research progress [6, 7] of related scholars, this paper firstly proposes to use the dual detection mechanism based on classification error to monitor the concept drift, mainly by multi-dimensional comprehensive judgment of the Mahalanobis distance and μ value of the data stream samples. Secondly, under the background of

* Correspondence: wdxyzbj@163.com
Tianjin Key Laboratory of Wireless Mobile Communications and Power Transmission, Tianjin Normal University, Tianjin 300387, China

concept drift, a classification algorithm [8, 9] based on EP is proposed to improve the accuracy of overall integration classifiers. Finally, the drift detection can be achieved while adjusting the performance of the classifier itself. The remainder of this paper is organized as follows. Section 2 presents a mechanism for detecting concept drift. Section 3 introduces an integration classification algorithm based on emerging patterns, and in Section 4, we proposes our major algorithm. Section 5 shows the experimental results of the proposed algorithm and analyzes them. Finally, it is summarized in Section 6.

2 A dual concept drift detection mechanism based on error rate

2.1 Mahalanobis distance detection standard based on error rate

As for high-dimensional datasets, Mahalanobis distance has a more significant advantage in calculation than the Euclidean distance. It is fully recognized by considering the correlation between different attributes of the dataset and independent to measurement scale.

Suppose $\mathbf{A} = (a_1, a_2, \ldots, a_i, \ldots a_n)$, $a_1 \neq a_j$, then the Mahalanobis distance between a_i and a_j is defined as

$$d(a_i, a_j) = \left[(a_i - a_j)^T S^{-1}(a_i - a_j)\right]^{1/2} \tag{1}$$

The calculation formula of the covariance matrix S is

$$S_{ij} = \operatorname{cov}(a_i, a_j) = E\left[(a_i - \mu_i)\left(a_j - \mu_j\right)\right] \tag{2}$$

Among them, $\mu_i = E(a_i)$ is used to represent the expectation value of each vector.

Therefore, for dataset $A = (A_1, A_2, \ldots, A_i, \ldots A_n, \ldots)^T$, data stream is sequentially processed in blocks for the convenience of the experiment. Where A_i represents the ith data block, the classification error rate on this data block is error_i, and the error rate on each data block refers to the average classification error rate of all data on the data block. Then the Mahalanobis distance can be represented by a set of mean values $\mu = (\mu_1, \mu_2, \ldots \mu_n)^T$ and a covariance matrix S, as shown in Eq. (3):

$$D_M(A) = \sqrt{(A - \mu)^T S^{-1}(A - \mu)} \tag{3}$$

After calculation, the degree of error rate change on each data block can be obtained, which indirectly reflects the similarity of adjacent data blocks and compares with the experimental threshold value to conclude whether the drift actually occurs. The further the $D_M(A)$ deviates from the threshold, the greater the possibility of concept drift, indicating that the warning state is entered at this time.

2.2 μ detection standard based on error rate

The principle of μ test in statistics: Let X be an arbitrary sample set, and there are first and second order matrix, which are respectively recorded as EX = μ, DX = σ^2 (σ is unknown). A unilateral assumption on X is as follows: the null hypothesis H_0: $\mu \leq \mu_0$ (μ_0 is a constant) and the alternative hypothesis H_1: $\mu > \mu_0$.The test level α is 0.05 or 0.01, and the value of $\bar{X}$ is to be tested. When the number of samples is large, that is, the value of n is large, the statistic $U = \frac{\bar{X} - \mu_0}{S/\sqrt{n}}$, where $\bar{X}$ is the average of the samples and S is the standard deviation of the samples. The statistic U obeys the standard normal distribution N (0, 1). According to the given test significance level α, there is μ_α that satisfies $P\{U > \mu_\alpha\} \approx \alpha$.

Suppose X have n samples, in which the number of misclassified samples is m, the average value of the misclassified subsamples $\bar{X} = m/n$, and the subsample standard deviation $S\text{^}2 = \bar{X}(1 - \bar{X})$. At this point, the statistic U can be described to the following form:

$$U = \frac{\bar{X} - \mu_0}{\sqrt{\frac{\bar{X}(1 - \bar{X})}{n}}} \tag{4}$$

The μ test method in the data stream environment is implemented on the basis of a certain model. Due to the particularity of data stream, the classification error rate on each data block is mainly tested and the initialization is the average of the classification error rates on the first i data blocks when the data distribution is stable. Therefore, the statistic U can be expressed as $U = \frac{\text{err} - \mu_0}{\sqrt{\text{err}(1 - \text{err})/n}}$. After each data block arrives, the change of the statistical U value is monitored. When $U \geq \mu_\alpha$, the classification error rate is considered to rise significantly and the concept drift occurs. Otherwise, the concepts in the current data stream remain stable.

The dual detection mechanism proposed in this part is mainly to classify each data block with the classifiers and measure the corresponding error rate. Bringing the classification error rate into two different dimensions of Mahalanobis distance and μ test is for calculation. The conclusion of concept drift can only be made when the two-dimensional requirement is reached at the same time. The workflow of the dual concept drift detection mechanism is as follows.

Input: Dataset A, the length of the data block is L; threshold ε, significance level α.

Output: classification error rate err_i on the ith data block, Mahalanobis distance $D_M(A)$,

μ test statistic U, the judgment of whether concept drift occurs.

Process:

1: Data preprocessing ←Data blocks A_1, A_2, … A_i, A_{i+1}, … A_n…
2: Initialization: $\mathrm{err}_i \leftarrow 0$, $D_M(A) \leftarrow 0$, $U \leftarrow 0$.
3: For the arriving data block A_i
4: Enter dual concept drift detection mechanism
5: Apply the basic classification algorithm based on eEP to learn, return err_i;
6: Enter the Mahalanobis distance detection part
7: Calculate the Mahalanobis distance by the formula of (3)
8: If $D_M(A) > \epsilon$, a warning appears, marked as Re1;
9: Enter the μ hypothesis test module
10: The statistic of the current data block is obtained by the formula $U = \frac{\mathrm{err}-\mu_0}{\sqrt{\mathrm{err}(1-\mathrm{err})/n}}$
11: If $U \geq \mu_\alpha$, indicating that the μ test hypothesis is not true, denoted as Re2;
12: Take the intersection of the detection results of the two parts, Result = Re1 ∩ Re2;
13: The system determines that the concepts drift.

3 Integration classification algorithm based on EP

3.1 Basic concepts

Suppose the training data set DB consists of n samples, each of which contains m-dimensional attributes. It is assumed that n samples are divided into K categories C_1, C_2, … C_k. The duality of the attribute name and its corresponding value, that is, property name and attribute value constitutes a data item. $I = \{ i_1, i_2 \ldots, i_n \}$, which denote a set of all data items, then any subset X is called an item set.

Definition 1: Suppose D is a subset of training set DB and records the support of item set X on D as $\mathrm{Sup}_D(X)$, which is defined as $\mathrm{Sup}_D(X) = \mathrm{Count}_D(X)/|D|$, where $\mathrm{Count}_D(X)$ represents the number of samples containing X of D, and $|D|$ represents the total number of samples of D.

Definition 2: For the two datasets D and D', the change of the item set X from D' to D is the growth rate, marked as $\mathrm{GR}_{D'\to D}(X)$.

$$\mathrm{GR}_{D'\to D}(X) = \begin{cases} 0 & \mathrm{Sup}_D(X) = 0, \mathrm{Sup}_{D'}(X) = 0 \\ \infty & \mathrm{Sup}_D(X) \neq 0, \mathrm{Sup}_{D'}(X) = 0 \\ \mathrm{Sup}_D(X)/\mathrm{Sup}_{D'}(X) & \mathrm{Sup}_D(X) \neq 0, \mathrm{Sup}_{D'}(X) \neq 0 \end{cases} \tag{5}$$

Definition 3: Set the growth rate threshold $\rho > 1$, if the growth rate of the item set X from D' to D satisfies $\mathrm{GR}_{D'\to D}(X) \geq \rho$, then X is called emerging patterns (EP) from D' to D and is referred as $\mathrm{GR}_D(X)$.

Definition 4: If the item set X satisfies:

1) X is the EP of D;
2) The support of X in D is not less than the minimum support threshold ξ;
3) Any true subset of X does not meet the conditions 1 and 2;

then X is called essential an emerging pattern (eEP), which is the basic EP.

3.2 Using eEP to establish base classifier

For large databases, especially high-dimensional datasets, eEP has more obvious advantages in terms of time and space complexity than EP. And eEP is the shortest EP, which greatly reduces the redundancy problem of EP in classification.

Taking the sample S as an example, we try to use the relevant theory of eEP to judge. Let D_i be the set of C_i class training samples, D_i' be the set of non-C_i class training samples, and X be the eEP of C_i class. If X does not appear in S, it cannot be judged whether S belongs to the C_i class. If X appears in S, X will have the probability of $\frac{\mathrm{GR}(X,D_i',D_i)}{\mathrm{GR}(X,D_i',D_i)+1}$ to determine that S belongs to the C_i class and that S does not belong to the C_i class by the probability of $\frac{1}{\mathrm{GR}(X,D_i',D_i)+1}$. If $\mathrm{GR}(X,D_i',D_i) = \infty$, $\frac{\mathrm{GR}(X,D_i',D_i)}{\mathrm{GR}(X,D_i',D_i)+1} = 1$, and $\frac{1}{\mathrm{GR}(X,D_i',D_i)+1} = 0$.

At the same time, the eEPs of the non-C_i class also contributes to determining whether S belongs to the C_i class. Let Y be an eEP of the non-C_i class, which appears in S. If the growth rate of Y is large, the effect of Y on determining that S belongs to the C_i class is negligible. However, when the growth rate of Y is not too large (such as $\mathrm{GR}(X,D_i',D_i) < 5$), Y has a considerable influence on determining that S belongs to the C_i class. In general, we take the probability that S belonging to the C_i class is $\frac{1}{GR(Y,D_i,D_i')+1}$.

In order to classify the sample S, it is necessary to consider the effects of the eEPs of the C_i class and non-C_i class. Therefore, the concept of membership is introduced, and the possibility that S belongs to the C_i class is called the membership of S to C_i, denoted as Bel(S).

For $i = 1,2,\ldots K$, let $\mathrm{PS}(S, C_i) = \{X | X$ is eEP of D_i, and X appears in $S\}$, $\mathrm{NS}(S, C_i) = \{Y | Y$ is eEP of D_i', and Y appears in $S\}$. The membership value of S belonging to the C_i class is calculated by:

$$\mathrm{Bel}(S, C_i) = \sum_{X \in PS(S,C_i)} \frac{\mathrm{GR}(X, D_i', D_i)}{\mathrm{GR}(X, D_i', D_i) + 1} + \sum_{Y \in NS(S,C_i)} \frac{1}{\mathrm{GR}(Y, D_i, D_i') + 1} \tag{6}$$

The probability of S belonging to each class is calculated by the above formula, and then S is classified by the following rules. S is classified as the class with the largest degree of membership. If the class with the highest degree of membership is not unique, it is determined by a majority voting strategy.

3.3 Integrate base classifier based on eEP

Considering the temporality and fluidity of data stream, the research in this paper is carried out in the sliding window. Suppose SW is a fixed-size sliding window, K is the number of basic windows in the sliding window. BW is the basic window, labeled as bw, and its length is |BW|. The trained base classifier of basic window bw_i is E_i.

In order to reflect the classification contribution of each base classifier in the integration classifier to the test dataset, we need to assign a weight to each classifier and introduce a weighting method based on classification error. For samples of (x, c), where c is a real class label, the classification error of E_i is $1-f_c^i(x)$, where $f_c^i(x)$ is determined by E_i that the probability of x being class c. Therefore, the mean square error of E_i is

$$\mathrm{MSE}_i = \frac{1}{|\mathrm{BW}|}\sum_{(x,c)\in B_K}\left(1-f_c^i(x)\right)^2 \tag{7}$$

The mean square error of the classifier when making random predictions is $\mathrm{MSE}_r = \sum_c p(c)(1-p(c))^2$

It can be obtained from prior knowledge that MSE_r is used as the threshold for weighting the classifier. To simplify the calculation, the weight w_i is calculated using the following formula.

$$w_i = \mathrm{MSE}_r - \mathrm{MSE}_i \tag{8}$$

The integration algorithm is as follows:

Input: Sup, GR, K total number of base classifiers; D data contained in the basic window bw_{k+1}; E set of K-base classifiers before adjusting weights;

Output: the top K-base classifiers with the highest weight in $E\cup\{E_{k+1}\}$

(1) Initialize K, Sup, GR;
(2) While(bw_{k+1} arrives) {
(3) Train (D, Sup, GR); / / training base classifier E_{k+1}
(4) Calculate the error rate of E_{k+1} on D (10-fold cross-validation);
(5) Calculate the weight w_{i+1} corresponding to E_{k+1} using Eqs. (7) and (8);
(6) for($E_i \in E$) {
(7) $E_i \leftarrow$ Train(E_i, D);
(8) Calculate the MSE_i of E_i on D; //Formula (1)
(9) Calculate E_i corresponding weight w_i; //Formula (2)

4 Integration system under the environment of data stream with concept drift

In order to deal with the integration classification problem in the data stream environment, this paper proposes a weighted classification and update algorithm of data stream based on concept drift detection (WUDCDD) to better adapt to the change of concept. The specific process is described as follows:

(1) Building an integration classifier

It constructs the base classifier on the basic window with eEP as the classification factor and then constructs the K-base classifiers to form the integrated classifier E. When the sliding window reaches the $(K+1)$th basic window, training the base classifier E_{k+1} and calculating the classification error rate of each base classifier E_i. Then weighting and selecting the K-base classifiers with the highest weight as the output according to the weighting method proposed in Section 3.3.

(2) Concept drift detection

The data stream in each basic window is divided into data blocks, and then the classification algorithm established by eEP as a classification factor is used to learn the model to obtain its classification error rate, and when a new data block is reached, the current integration model is utilized to classify it. The Mahalanobis distance from the classification error rate of the previous data block and the current block is calculated. If the distance exceeds a certain threshold condition ε, it is judged that there is a high probability that a concept change will occur, and the warning state is entered at this time. On this basis, the next hypothesis verification is carried out. If the classification error rate on the new data block is significantly increased, the system comprehensively judges the concept drifts.

(3) Updating classifiers

This part performs integration of classifiers by weighting each base classifier, and the weight of each base classifier uses the classification error rate. If the concept drift detection module determines that concept drift occurs, the data block in the current window is used as a training set, and each base classifier is relearned. And comparing the weights of the learned base classifiers, selectively eliminating or retaining the old base classifiers while keeping the total number of base classifiers remains unchanged, so that the updated system is more suitable for the current data stream environment.

5 Experimental results and discussion

5.1 Dataset

Artificial data stream is a simulation of changing concepts by rotating hyperplanes. The hyperplane on the d-dimensional space is a set of points x that satisfy the following conditions:

$$\sum_{i=1}^{d} w_i x_i = w_0$$

Where x_i is the ith coordinate of point x. The samples satisfy $\sum_{i=1}^{d} w_i x_i > w_0$ and are marked as positive samples, and other samples satisfy $\sum_{i=1}^{d} w_i x_i < w_0$ and are marked as negative samples. When simulating the time-varying concepts, we adjust the orientation of the hyperplane smoothly by adjusting the corresponding weight w_i, so the hyperplane is very important. In the experiment, the training set size is 10,000, the test set size is 1000, a total of 10 dimensions, the number of different values in each dimension is 4, and the noise rate is 5%.

6 Results and discussion

In the following section, we mainly compare the accuracy of the proposed algorithm WUDCDD, G_K (representing a single classifier trained on a sliding window that the size is K) and EC4.5 (integration classifiers based on a single classifier of C4.5) under different conditions. The accuracy is mainly compared from four aspects: (1) the influence of the size of the basic window on the change of classification accuracy, (2) the impact of the size of the sliding window on the change of accuracy, (3) the effect of the dimension of the drift on the change of accuracy, (4) the influence of the dimension of data stream on the change of accuracy.

Experiment 1 Testing the variation of classification accuracy with the basic window size (|BW|). The experiment sets 2 dimension drifts and changed weights every 1000 samples. Figure 1 shows the average classification accuracy of the algorithm under different basic windows when the basic window is [250, 1500].

It can be seen from Fig. 1 that the proposed algorithm is more effective than the corresponding single classifier G_K. When $|BW| \le 250 \times 3$, the accuracy of WUDCDD is higher than EC4.5. When $250 \times 3 \le |BW| \le 250 \times 6$, it is comparable to EC4.5.The accuracy decline of WUDCDD is not as obvious as EC4.5 when the range is [250 × 4, 250 × 6], because we incrementally update each model before calculating the weight of each base classifier. It can better adapt to the concept drift.

When the basic window is small, each algorithm has better classification performance. Because the window contains less concept drift, the distribution of data is more stable. However, if it is too small, the accuracy is reduced because there is not enough data to train the base classifier. When the window is too large, it is difficult to detect whether the drift occurs, which also affects its performance. When the window is too small, we can improve the base classifier performance by reducing support.

Experiment 2: Testing the variation of classification accuracy with sliding window size and the parameters of the dataset are the same as experiment 1. Figure 2 shows the average accuracy under different basic windows for sliding windows of 2, 4, 6, and 8, respectively.

Figure 2 shows that as the sliding window increases, and the accuracy of WUDCDD and EC4.5 increases continuously and has better performance than G_K with the reason of G_K not adapting well to concept drift. Moreover, the performance of WUDCDD and EC4.5 increases rapidly at the beginning and then the increase is gradually reduced. Because of the better detection of drift, the increase of base classifiers will have a weak effect on the classification performance. When $|SW| < 8$, WUDCDD is slightly better than EC4.5, because the former single classifier performance is better than C4.5. When $K > 8$, the performance of both is close.

Experiment 3: Testing the effect of the dimension of the drift on the accuracy. When the 2, 4, 6, and 8 dimensions are set to drift, the result is shown in Fig. 3. It can be seen that with the increase of drift, the accuracy of each algorithm drops sharply and then stabilizes. G_K is

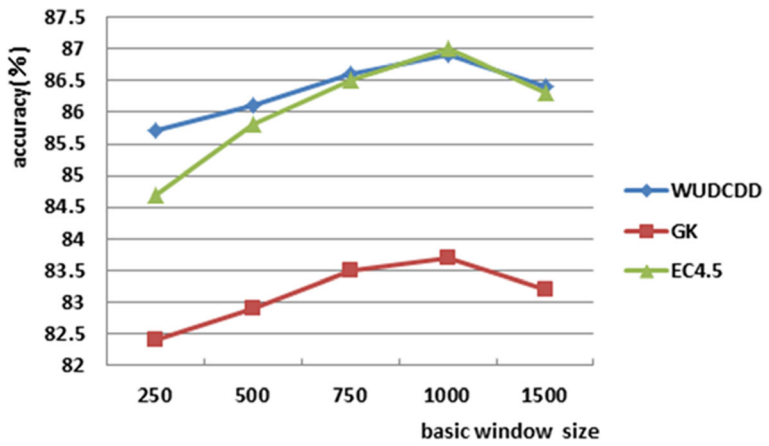

Fig. 1 Comparison of accuracy rates under different basic windows

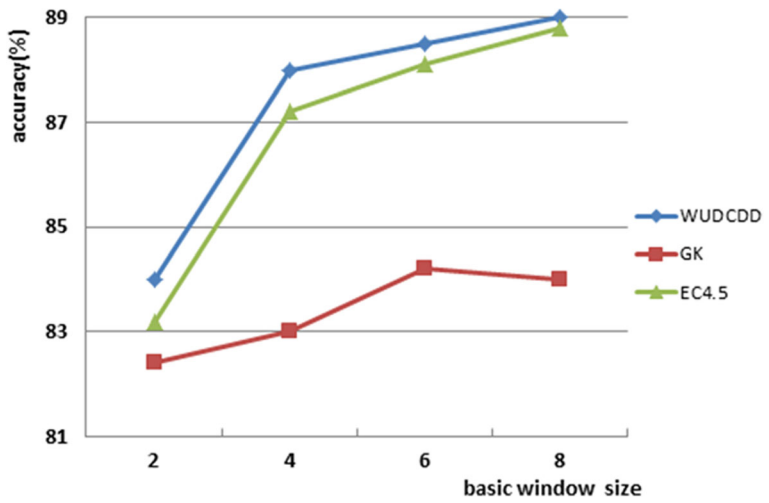

Fig. 2 Comparison of accuracy rates under different sliding windows

most affected because there is no mechanism for processing drift. When the range of varying dimension is [2,4], the performance difference between WUDCDD and EC4.5 is very small. When the range of varying dimension is [4,8], the accuracy of the latter decreases more obviously. Because there is no incremental adjustment decision tree with new data arriving, WUDCDD always maintains the most discriminating eEP and constantly adjusts the obtained EP to reflect the characteristics of the data.

Experiment 4: Testing the effect of the total number of dimensions on accuracy, $|BW| = 250$ and $K = 6$. The experimental results are shown in Fig. 4.

According to the trend of the curve in the figure, it can be seen that as the dimension of the dataset increases and the accuracy rate decreases. This is because of the increase of dimensions leading to the large number of eEPs. But the support and growth rate generally decreases, thereby reducing the discrimination of eEPs and resulting in the decline of classification ability. So the accuracy of WUDCDD also decreases. As the number of dimensions increases, the number of classification rules of EC4.5 increases, which also causes the decrease in accuracy. When the accuracy of WUDCDD drops, we can adjust by lowering the support threshold.

7 Conclusions

How to train models from massive data to effectively predict future data stream has become a hot topic. The traditional data classification algorithm can not be directly applied to the data stream environment, therefore, this paper innovatively introduces the eEP classification algorithm into the data stream classification field and proposes the algorithm of detection and integration classification based on the data stream with concept drift. By comparing with the other two algorithms, it is proved

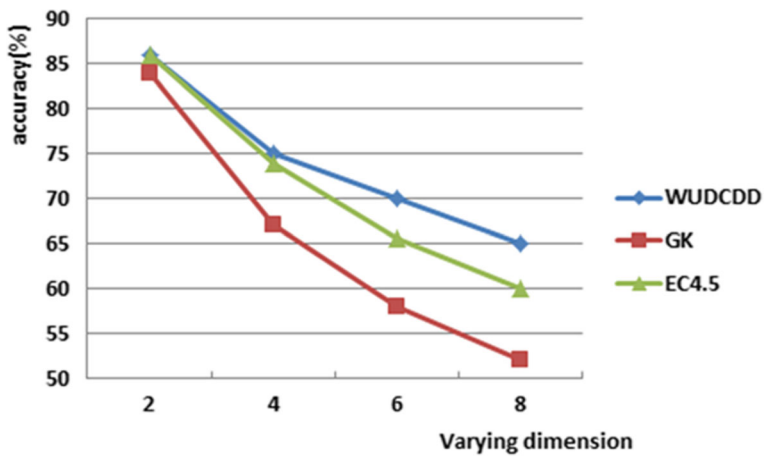

Fig. 3 The effect of the number of dimensional changes on the accuracy

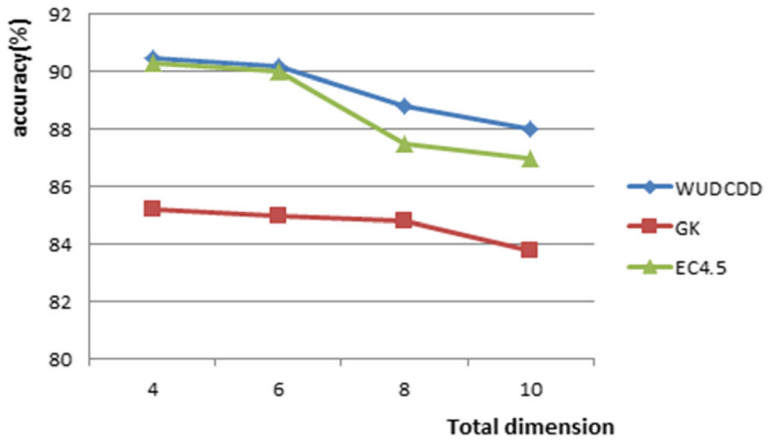

Fig. 4 The effect of the total number of dimensions on the accuracy

that the proposed algorithm can better adapt to the data stream with concept drift and has better classification accuracy, which is also sufficient to compare with the integration algorithm based on the C4.5. Finally, through the experimental result, it can be seen that the update strategy of the algorithm in the sliding window needs further research and improvement in order to apply to more specific fields such as data mining.

Abbreviations
eEP: Essential emerging pattern; WUDCDD: Weighted classification and update algorithm of data stream based on concept drift detection

Acknowledgements
Not applicable

Funding
This paper is supported by Natural Youth Science Foundation of China (61401310) and Tianjin Science Foundation (18JCYBJC86400).

Availability of data and materials
All data generated or analyzed during this study are included in this published article.

Authors' contributions
BJZ analyzed and proposed the significance of current data mining and data analysis and carried out the experimental verification. YDC analyzed the experimental data and become a major contributor to the writing of manuscripts. The final draft was read and approved by both authors.

Competing interests
The authors declare that they have no competing interests.

Publisher's Note

Received: 13 January 2019 Accepted: 19 March 2019
Published online: 03 April 2019

References
1. P. Vorburger, A. Bernstein, in *International Conference on Data Mining*. Entropy-based concept shift detection (2006), pp. 1113–1118
2. W.N. Street, in *ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. A streaming ensemble algorithm (SEA) for large-scale classification (2001), pp. 377–382
3. J.Z. Kolter, M.A. Maloof, in *Proceedings of the 3rd IEEE International Conference on Data Mining*. Dynamic weighted majority: A new ensemble method for tracking concept drift (2003), pp. 123–130
4. Y. Sun, G.J. Mao, X. Liu, C.N. Liu, Concept drift mining in data stream based on multi-classifier. J. Autom. **34**(1), 93–97 (2008)
5. L.L. Minku, A.P. White, X. Yao, The impact of diversity on online ensemble learning in the presence of concept drift. IEEE Trans. Knowl. Data Eng. **22**(5), 730–742 (2010)
6. K. Nishida, K. Yamauchi, in proceedings of *International Conference on Discovery Science*. Detecting concept drift using statistical testing (Springer-Verlag, Berlin Heidelberg 2007), pp. 264–269
7. R. Elwell, R. Polikar, Incremental learning of concept drift in nonstationary environments. IEEE Trans. Neural Netw. **22**(10), 1517–1531 (2011)
8. M. Fan, M.X. Liu, H.L. Zhao, A classification algorithm based on basic exposure mode. Comput. Sci. **31**(11), 211–214 (2004)
9. L. Duan, C.J. Tang, N. Yang, C. Gou, Research and application progress of contrast mining based on revealing mode. J. Comput. Appl. **32**(02), 304–308 (2012)