Pattern Recognition 48 (2015) 659–669

Contents lists available at ScienceDirect

Pattern Recognition

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

EWMA model based shift-detection methods for detecting covariate shifts in non-stationary environments

Haider Raza*, Girijesh Prasad, Yuhua Li

Intelligent Systems Research Centre, School of Computing and Intelligent Systems, University of Ulster, Magee Campus, Londonderry, Northern Ireland, UK

ARTICLE INFO

Article history:
Received 18 December 2013
Received in revised form
20 June 2014
Accepted 26 July 2014
Available online 5 August 2014

Keywords:
Non-stationary environments
Dataset shift-detection
Covariate shift
EWMA

ABSTRACT

Dataset shift is a very common issue wherein the input data distribution shifts over time in non-stationary environments. A broad range of real-world systems face the challenge of dataset shift. In such systems, continuous monitoring of the process behavior and tracking the state of shift are required in order to decide about initiating adaptive corrections in a timely manner. This paper presents novel methods for covariate shift-detection tests based on a two-stage structure for both univariate and multivariate time-series. The first stage works in an online mode and it uses an exponentially weighted moving average (EWMA) model based control chart to detect the covariate shift-point in non-stationary time-series. The second stage validates the shift-detected by first stage using the Kolmogorov–Smirnov statistical hypothesis test (K–S test) in the case of univariate time-series and the Hotelling *T*-Squared multivariate statistical hypothesis test in the case of multivariate time-series. Additionally, several orthogonal transformations and blind source separation algorithms are investigated to counteract the adverse effect of cross-correlation in multivariate time-series on shift-detection performance. The proposed methods are suitable to be run in real-time. Their performance is evaluated through experiments using several synthetic and real-world datasets. Results show that all the covariate shifts are detected with much reduced false-alarms compared to other methods.

1. Introduction

In the non-stationary environments (NSEs), assessing the stationarity of the data generating process i.e., checking whether shifts and drifts have affected the data generating process, is an important challenge. Most of the pattern classification methods are built upon the common assumption that the data distribution remains stationary during classifier training and testing or operating stages, hereafter called the stationary hypothesis. Hence, monitoring the validity of the stationary hypothesis over time can be an advantageous step as it allows one to do the following: (a) verify the system stationarity, as not only classification, but system identification, and fault-detection methods as well are mostly designed under the common assumption that the process is stationary; and (b) take appropriate corrective actions, e.g., adaptation by updating the parameters of the classifier. Particularly in the streaming data applications, the input data distribution may shift over time during the operating phase due to the presence of myriads of environmental non-stationarities. In literature, various types of commonly occurring shifts and drifts have been defined [23]. The shift in the joint distribution of the multi-class data from the training to test stages is termed as dataset shift. The difference in the input distribution at different time periods is called as covariate shift [1]. Classifying the time-series data in the NSEs requires a learning model which should be computationally efficient and able to detect the dataset shift-point in the underlying distribution of the data stream in real-time, so that the on-line learning remains unaffected from spurious changes or white noise. Such a learning process in NSEs is sometimes also called as non-stationary learning (NSL) [2].

In NSL, an efficient and effective stationarity evaluation test is required to deal with the large class of applications without any a priori information about the data generating process, which is hardly available in the real-world problems. In NSEs various forms of shifts can be found such as abrupt, transient and gradual shifts. Detecting these shifts in the data may form part of a change (shift)-point detection method [3]. Moreover, the correctness of the shift-detection test can be described in terms of time delay and detection accuracy and hence, these are the important issues in the shift-detection literature. Based upon the time delay in detection, shift-detection methods can be categorized into retrospective detection and online (or real-time) detection; for more detail see [4–6]. In retrospective shift-detection, a window of the data is used and therefore it has a time-delay of at least one time-window in shift detection. In online shift-detection, a single pass method is used to process the data. The online shift-detection techniques may need to be performed in several key areas for the

* Corresponding author.
E-mail address: raza-h@email.ultser.ac.uk (H. Raza).

http://dx.doi.org/10.1016/j.patcog.2014.07.028
0031-3203/© 2014 Elsevier Ltd. All rights reserved.

monitoring of streaming data, such as electroencephalography (EEG) based brain–computer interface [7], spam-filtering [8], and network intrusion detection [9].

Moreover, the shift-detection algorithms can be active or passive [2]. The active shift-detection method detects the time and severity of the shift and initiates classifier learning, if needed. In passive shift-detection, the learner accepts that the environment may shift at any time or may be continuously shifting. So, the passive learning algorithm then continually learns from the environment by updating their knowledge-base (KB). If the shift has occurred, the shift is learned. If the shift has not occurred, the model continues learning and the existing knowledge is reinforced. In NSEs there are several types of dataset shift, a brief review of dataset shift and the types of dataset shift are presented in the next section.

A relatively large literature addresses the statistical shift-detection methods. In the shift-detection methods, there are two types of tests: parametric and non-parametric statistical tests. Examples of parametric tests are the Student *t*-test and the Fisher *f*-test [10,11]. These tests mainly address the shift in the mean and the variance. A parametric test generally requires the availability (or an estimate) of probability density function (pdf) of the data generating process before and after the shift-detection. Whereas in the non-parametric test, no strong a priori information is required. The Mann–Whitney *U*-test for independent samples and the Wilcoxon signed-rank test are the two examples of the non-parametric tests [12]. A range of methods have demonstrated very good shift point-detection performance by monitoring the moving control charts and comparing the probability distributions of the time-series samples over past and present intervals. These methods follow different strategies such as the Shewhart charts [13] that monitors the quality characteristics of measurements for one observation at a time. In Cumulative SUM (CUSUM) [14] a sequential cumulative sum technique is used and when the sum reaches above the pre-defined threshold the shift is detected. Both, the Shewhart and CUSUM charts only detect the large shift in the data. However, using extended CUSUM and Computational-Intelligence CUSUM (CI-SUSUM) in [15], the shift-detection is presented for the univariate and multivariate input data, which is deployed with a just-in-time (JIT) adaptive classifier in the NSEs. This method may suffer from the time-delay and large number of false alarms in the shift-detection, which may impact the classification accuracy of the system. The Intersection of Confidence Interval (ICI) rule [16] is a more advanced work in the shift point-detection test based on a hierarchical structure. The performance of ICI is shown to be better in terms of less false alarms in comparison to other methods, but it suffers from the time-delay in the shift-detection. Other reported approaches are statistical ones [17], neural network based approaches [18] and the generalized logarithm-ratio method [19], which also suffer from similar limitations. Moreover, to overcome the weaknesses discussed above, recently some researchers have proposed a different strategy which estimates the ratio of two probability densities without direct density estimation [20], but it also suffers from some delay in shift-detection. Recently in [21], the shift-detection is performed in parallel, on both the input data distribution and the classification error i.e., covariate shift and concept shift respectively, for monitoring recurrent concepts in the NSEs.

However, most of the aforementioned methods depend on pre-designed parametric models such as underlying probability distribution, auto-regressive models and state-space models, for tracking some specific statistics such as the mean, variance, and spectrum. Thus, they are not robust against different types of shifts because of the delay in shift-detection on account of the need for identifying models from the past data, which may significantly limit their range of applications in fast data streaming problems. Moreover, most of the systems also tend to generate excessive number of false-alarms, which is an obstacle in the real-world applications.

In order to reduce the time delay and false alarms, a two-stage shift detection strategy is proposed in this paper. The paper advances the work presented in [4,5] by proposing a complete general formulation for the covariate shift-detection for both univariate and multivariate data. The shift-detection method is built on an exponentially weighted moving average (EWMA) chart. It is demonstrated to outperform other approaches in terms of non-stationarity detection with significantly reduced time delay and false alarms. The approach is computationally efficient because of low computational cost and less memory requirements during online processing. So, this scheme can be deployed along with any classifier such as *k*-nearest neighbor (*k*NN), linear discriminant analysis (LDA), artificial neural networks (ANNs), or support vector machines (SVMs) in an adaptive online learning framework.

The novel contributions of the paper can be summarized as follows:

- A complete framework for the covariate shift-detection test is introduced for both univariate and multivariate processes. The covariate shift-detection test is based on an EWMA model. This shift-detection test assesses the stationarity of input data generating process, i.e., only on the input data distribution, disregarding the associated output labels. The approach is particularly promising in covariate shift-detection with much reduced time-delay.
- A procedure to reduce the false-alarms is proposed by a two-stage shift-detection test structure. The two-stage test uses a Kolmogorov–Smirnov statistical hypothesis test (K–S test) at the second stage to validate the shifts detected by the first stage of the test in a univariate case. In a multivariate case, Hotelling's *T*-Squared statistical test is proposed. The approach is particularly promising in reducing the false-alarms.
- A novel contribution in multivariate covariate shift-detection is in counteracting adverse effect of cross-correlation among multiple input processes on detection performance. To this end, orthogonal transformation and blind source separation techniques are investigated.

This paper proceeds as follows: Section 2 presents background information behind dataset shift and EWMA control chart. Section 3 deals with the shift-detection algorithms for univariate and multivariate data. Section 4 presents the datasets used in the experiment. Finally, Section 5 presents the results and discussion.

2. Background

2.1. Dataset shift

Assume a pattern classification problem is described by a set of features or inputs x, a target variable y, the joint distribution $P(y,x)$, the prior probability $P(x)$ and conditional probability. The term dataset shift [22,23] was first well-defined in the workshop of neural information processing systems (NIPS, 2006). The dataset shift is a "*case where the joint distribution of inputs and outputs differs between training and test stage, i.e., when* $(P_{train}(y,x) \neq P_{test}(y,x))$" [24]. Dataset shift was previously defined by various authors giving different names to the same concept such as, concept shift or drift [25], changes of classification [26], changing environment [27], contrast mining [28], and fracture point [29]. In pattern classification problems, the dataset shift is now mainly categorized into three different types that usually occur in the real-world applications such as (i) covariate shift, (ii) prior probability shift, and (iii) concept shift.

2.1.1. Covariate shift

The covariate shift has been defined by different terms in the literature. Several authors defined covariate shift as, "population drift", "a case where the population distribution may change over time" [30]. In a generic way, it is defined as "covariate shift appears only in $X \rightarrow Y$ problems, and the case where the conditional probability in training and testing remains same ($P_{train}(y|x) = P_{test}(y|x)$), but the input distribution $P(x)$ changes between training and testing, i.e., ($P_{train}(x) \neq P_{test}(x)$)" [23]. Let us take an example of a process where covariate shift can be seen. Assume a training input data distribution $P_{train}(x)$ is a normal distribution with mean and standard deviation as 2 and 1.5 respectively, i.e. [$x_{train} = \mathcal{N}(x;\ 2, 1.5)$] and the test input data distribution $P_{test}(x)$ is also a normal distribution with mean and the standard deviation as 4 and 1.5 respectively, i.e. [$x_{test} = \mathcal{N}(x;\ 4, 1.5)$]. Fig. 1 shows the covariate shift as is given in the example above where only the mean has changed between the training and test stages.

The problem of covariate shift can be easily found in the real-world applications. Some of the common examples are spam filtering [31], brain–computer interfaces (BCIs) [7], and network intrusion detection [9]. For other types of dataset shift such as prior probability and concept shift, readers may refer to [4,5]. There exists other shifts that could happen in theory, but we are not discussing those as they appear rarely; for more details see [23]. In this paper, our main focus is on the covariate shift-detection, because the pattern classification problem is based on the predictive model, i.e., $X \rightarrow Y$.

2.2. EWMA control chart

An exponentially weighted moving average (EWMA) control chart [32] is a member of the family of control charts within the statistical process control (SPC) theory. Control charts are a graphical representation of sample statistics for SPC. The EWMA is used in detecting small shifts in the mean of a time-series data. The EWMA control chart overtakes other control charts because it pools present and past data in such a way that a small shift in the time-series can be detected more easily and quickly. Other charts, such as the Shewhart chart [13], only consider the most current observations by forgetting the past data. The EWMA uses a weighting constant, lambda (λ), which decides the importance of current and historical observations.

The EWMA model is defined as

$$z_{(i)} = \lambda x_{(i)} + (1-\lambda) z_{(i-1)} \tag{1}$$

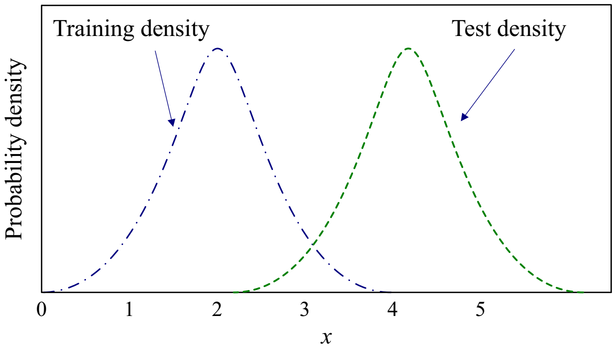

Fig. 1. Schematic diagram for covariate shift: training dataset has normal distribution with $\mathcal{N}(x;\ 2, 1.5)$, and test dataset also has normal distribution with $\mathcal{N}(x;\ 4, 1.5)$. Thus the mean of the testing data distribution has changed from that of training, resulting in covariate shift.

where λ is the smoothing constant ($0 < \lambda \leq 1$), z is the exponentially weighted moving average (EWMA) and x is the observation. Moreover, the EWMA charts are used for both uncorrelated and auto-correlated data. We are only considering the auto-correlated data in our study and simulation because the data obtained from NSEs in the real-world applications are often correlated.

2.2.1. EWMA model for auto-correlated data

If data contains a sequence of auto-correlated observations $x_{(i)}$, then the EWMA statistic in Eq. (1) can be used to provide a 1-step-ahead prediction model of auto-correlated data. Here, it is assumed that the process observations $x_{(i)}$, can be defined as Eq. (2) below, which is a first-order auto-regressive integrated moving average (ARIMA) model. In time series analysis, an ARIMA model is a generalization of an auto-regressive moving average (ARMA) model. These models are fitted to time-series data either to better understand the data characteristics or to predict the future points in the series (forecasting) [33]. Moreover, these models can represent system dynamics wherein data show evidence of non-stationary behavior. In particular, Eq. (2) describes a non-stationary behavior, wherein the covariate $x_{(i)}$ shifts as if there is no fixed value of the process mean.

$$x_{(i)} = x_{(i-1)} + \varepsilon_i - \theta \varepsilon_{i-1} \tag{2}$$

where ε_i is a sequence of independent and identically distributed (i.i.d.) random signal with zero mean and constant variance. It can be easily shown that the EWMA with ($\lambda = 1 - \theta$) is the optimal 1-step-ahead prediction for this process [11,34].

According to [34], if $\hat{x}_{i+1}(i)$ is the forecast of the observation for the period $(i+1)$ made at the end of period i, then, the 1-step-ahead prediction for $x_{(i)}$ is the EWMA $z_{(i-1)}$, in Eq. (1). Fig. 2 explains it more clearly through a state diagram. For more detail about 1-step-ahead prediction by EWMA see Appendix A where we have derived a relationship between the EWMA and the ARIMA models. The 1-step-ahead prediction errors $err_{(i)}$ are calculated as

$$err_{(i)} = x_{(i)} - z_{(i-1)} \tag{3}$$

Assume, that the 1-step-ahead prediction errors $err_{(i)}$ are normally distributed with mean equals to zero. It is given in [11] that it is possible to combine information about the statistical control and process dynamics on a single control chart. Then, the control limits of the chart on these errors satisfy the following probability statement by substituting the right hand side of Eq. (3) in the formulation below. Therefrom, the EWMA control chart on $x_{(i)}$ [34] can be derived as given below

$$P[-L\sigma_{err} \leq err_{(i)} \leq L\sigma_{err}] = 1 - \alpha$$
$$P[-L\ \sigma_{err} \leq x_{(i)} - z_{(i-1)}(i) \leq L\sigma_{err}] = 1 - \alpha$$
$$P[z_{(i-1)} - L\sigma_{err} \leq x_{(i)} \leq z_{(i-1)} + L\sigma_{err}] = 1 - \alpha$$

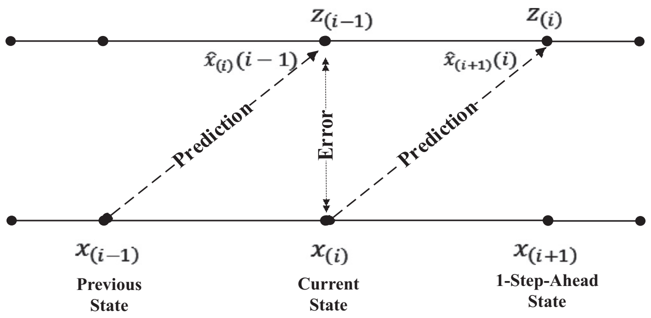

Fig. 2. Schematic state diagram for the 1-step ahead prediction by EWMA model. The $z_{(i-1)}$ is the 1-step-ahead prediction for the observation $x_{(i)}$ from the state $(i-1)$. The 1-step-ahead prediction error is given as $err_{(i)} = x_{(i)} - z_{(i-1)}$.

where σ_{err} is the standard deviation of the errors, L is the control limit multiplier, $(1-\alpha)$ is the confidence interval and the α is the 5% level of significance. If the EWMA is a suitable 1-step-ahead predictor, then one could use $z_{(i-1)}$ as the center line for the period i with Upper Control Limit (UCL) and Lower Control Limit (LCL) [11], defined as

$$UCL_{(i)} = z_{(i-1)} + L\sigma_{err_{(i-1)}} \tag{4a}$$

$$LCL_{(i)} = z_{(i-1)} - L\sigma_{err_{(i-1)}} \tag{4b}$$

Whenever, the $x_{(i)}$ moves out of $UCL_{(i)}$ and $LCL_{(i)}$, the process is said to be out of control. This method is also known as a moving center-line EWMA control chart. The standard deviation of the 1-step-ahead error or model residuals σ_{err} may be estimated in several ways such as the mean absolute deviation (MAD) or a directly calculated smoothed variance [9,11].

The EWMA control chart is robust to the normality assumption if properly designed for the t and gamma distributions [35]. So, the EWMA chart can be employed when there is a concern about the normality assumption. Following the above formulation, we have designed an algorithm for the covariate shift-detection based on the EWMA of the process observation of auto-correlated data, as discussed in the next section.

3. Methodology

In the statistical process control theory, control charts are tools used to determine if a process is in a state of statistical control. Generally it is represented by three lines plotted along the horizontal axis. The center line and two control lines (control limits) are plotted on a control chart, which correspond to the target value (μ) and acceptable deviation ($L\sigma$) from either side of the target value respectively, where L is the control limit multiplier and σ is the standard deviation of the data generating process. The aim of the control chart is to monitor a process behavior, such as the shift in the data generating process. This work employs an EWMA control chart for the covariate shift-detection. When the process observation falls outside the EWMA control limits, the process is said to be out of control and so the covariate shift is detected.

The proposed method works in two phases, the first phase is a retrospective (training) phase and the second phase is an operation (testing) phase. In the training phase, the parameters are calculated to decide the null hypothesis and it is assumed that the training data obtained are in stationary state. In the testing phase, the process observations are continuously monitored by the EWMA chart and when an observation falls outside the control limits of the control chart, the point is said to be a point of covariate shift. In other words, the process observation falling outside the control limit is not in the stationary state and the null hypothesis is rejected in favor of an alternative hypothesis and the shift is detected. An important point to note here is that we have assumed that non-stationarity occurs due to changes in the input distribution only. So, it is said to be a covariate shift-detection in a non-stationary time-series. In the following sub-sections, the designed algorithms for univariate and multivariate processes are discussed in detail.

3.1. Shift-detection based on EWMA (SD-EWMA)

Our algorithm works in two different phases, the first phase is a retrospective/training phase in which the parameters $(\lambda, z_{(0)}, \sigma^2_{err_{(0)}})$ are calculated followed by the operation/test phase for the covariate shift-detection. The pseudo code of the algorithm is given in Table 1.

The step-1 of the training phase is to obtain the sequence of observations and the step-2 is used to calculate the mean, and set it to $z_{(0)}$. Next, the EWMA statistics by Eq. (1) at step-3 is obtained. The EWMA smoothing constant λ is then estimated by minimizing the sum of the squared one-step-ahead prediction error on the training dataset, as given in the SD-EWMA algorithm at steps 4 and 5 of the training phase. Finally, at step-6 the sum of the squared one-step-ahead prediction error divided by the length of the training dataset is used as an initial value of $\sigma^2_{err_{(0)}}$, for the testing data.

Table 1
Algorithm SD-EWMA.

```
Input: Submit the training dataset to the training phase and compute the parameters for testing.
  Receive new data in testing phase sample-by-sample and perform the check as follows.
    IF (Shift detected)
    THEN (Report the point of shift and initiate an appropriate corrective action)
    ELSE (Continue and integrate the upcoming information).
Output: Shift-detection points.
Training Phase
1. Assign training data to $x_{(i)}$ for $i = 1$ to $n$, $n$ is the size of training data.
2. Calculate the mean of input data ($\bar{x}$) and assign it to $z_{(0)}$.
3. Compute the $z$-statistics for each observation $x_{(i)}$ in training data for a range of $\lambda$ values.
    $z_{(i)} = \lambda x_{(i)} + (1-\lambda) z_{(i-1)}$
4. Compute 1-step-ahead prediction errors $err_{(i)} = x_{(i)} - z_{(i-1)}$
5. Estimate $\lambda$ by minimizing the sum of the squared prediction error on the training data.
6. Finally compute the sum of the square of 1-step-ahead prediction error divided by the number of observations and use it as the initial value of the variance ($\sigma_{err^2_{(0)}}$) for
   the testing phase.
Testing Phase
1. For each data point $x_{(i)}$ in the operation/testing phase,
2. Compute $z_{(i)} = \lambda x_{(i)} + (1-\lambda) z_{(i-1)}$
3. Compute $err_{(i)} = x_{(i)} - z_{(i-1)}$
4. Compute the estimated variance $\hat{\sigma}^2_{err_{(i)}} = \vartheta\, err^2_{(i)} + (1-\vartheta)\hat{\sigma}^2_{err_{(i-1)}}$
5. Compute $UCL_{(i)}$ and $LCL_{(i)}$:
6.    $UCL_{(i)} = z_{(i-1)} + L\hat{\sigma}_{err_{(i-1)}}$
7.    $LCL_{(i)} = z_{(i-1)} - L\hat{\sigma}_{err_{(i-1)}}$
8.      IF ($LCL_{(i)} < x_{(i)} < UCL_{(i)}$)
9.        THEN (Continue processing with new sample)
10.     ELSE (Covariate Shift detected and Initiate an appropriate corrective action)
```

In the testing phase, for each observation, use Eq. (1) to obtain the EWMA statistics and follow the steps given in Table 1 and then compute the $UCL_{(i)}$ and $LCL_{(i)}$ using Eqs. (4a) and (4b). Next, check if each observation $x_{(i)}$ falls within the control limits $[UCL_{(i)}, LCL_{(i)}]$, otherwise a shift is detected and an alarm is raised. The standard deviation of 1-step-ahead error can be estimated in various ways such as by directly estimating the smoothed variance [11] as given in step 4 of the testing phase in the SD-EWMA algorithm in Table 1, where ϑ is an error smoothing constant; reference [34] suggests that smaller values of ϑ are preferred. As this method mainly accounts for the shift in the mean, excessive number of false-positives (i.e., false alarm) maybe observed in some cases. So, to reduce the number of false-positives, a two-stage based shift-detection test is proposed in the next section.

3.2. Two-stage shift-detection based on EWMA (TSSD-EWMA)

The proposed two-stage shift-detection based on the EWMA test works in two stages. Using the SD-EWMA [5] method discussed in the previous section, stage-I works in the online mode, continuously processing the upcoming data from the data stream. Stage-II uses a statistical hypothesis test to validate the shift detected by stage-I. Stage-II operates in retrospective mode and starts validation once the shift is detected by stage-I. The two-stage based structure for shift-detection is given in Fig. 3. The pseudo code of the algorithm is given in Table 2.

3.2.1. Stage-I

As discussed earlier, stage-I works in two different phases: training phase and operation or testing phase. In the first phase, the parameters $(\lambda, z_{(0)}, \sigma^2_{err_{(0)}})$ are calculated to decide the null hypothesis. In the testing phase, as in the TSSD-EWMA algorithm (Table 2), it is checked if each observation $x_{(i)}$ falls within the control limits $[UCL_{(i)}, LCL_{(i)}]$, otherwise the shift is detected and alarm is raised at stage-I. Furthermore, the shift detected by stage-I is passed to stage-II for validation in order to reduce the number of false positive alarms.

3.2.2. Stage-II

Stage-II works in retrospective mode only when a shift is detected at stage-I. In particular, to validate the shift detected by stage-I, the available information need to be partitioned into two-disjoint subsequences and then the statistical hypothesis test is applied. The two-sample Kolmogorov–Smirnov test [10] is used to validate the stationarity in the sub-sequences because of its non-parametric nature. This test returns a test decision of null hypothesis if the data in the subsequences are stationary with equal means and equal but unknown variances. The Kolmogorov–Smirnov statistics is briefly described as follows:

$$D_{n1,n2} = sup_x |F_{1,n1}(x) - F_{2,n2}(x)| \tag{5}$$

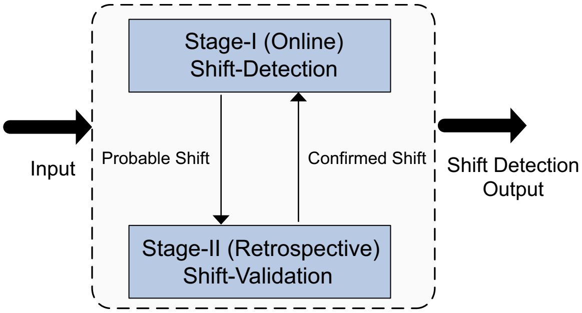

Fig. 3. Schematic diagram for the two-stage based structure for the shift-detection test. The first stage detects covariate shift in an online mode. The detected covariate shifts are validated at stage-II.

where sup_x is the supremum and $F_{1,n1}(x)$ and $F_{2,n2}(x)$ are the empirical cumulative distribution functions on the first and second sub-sequences respectively. The $n1$ and $n2$ are the two sub-sequences of length p and q as $n1 = ((i-(p-1)) : i)$ and $n2 = ((i+1) : (i+q))$ where i is the current observation. The null hypothesis is rejected at level α and $(H=1)$ is returned if

$$\sqrt{\frac{pq}{p+q}} D_{n1,n2} > K_\alpha \tag{6}$$

where K_α is the critical value and can be found in [36].

3.3. Multivariate shift-detection based on EWMA (MSD-EWMA)

In real-world systems, there are many situations in which the parallel monitoring or the control of two or more co-related input processes is necessary. In the following formulation, the input $x^d_{(i)}$ is therefore extended to the d-dimensional case. Monitoring of such processes independently maybe very misleading, e.g., if the probability that a variable exceeds three-sigma control limits is 0.0027 then a false-detection rate of 0.27% is expected. However, the joint probability that d such variables exceed their control limits simultaneously is $(0.0027)^d$, which is considerably smaller than 0.0027. So, the use of d-independent charts may provide highly distorted outcomes.

In [37] a multivariate version of EWMA control chart is presented. The multivariate EWMA is a logical extension of the SD-EWMA and is defined as follows:

$$z_{(i)} = \lambda x^d_{(i)} + (1-\lambda) z_{(i-1)} \tag{7}$$

where the EWMA $z_{(i-1)}$ is a vector of dimension d and λ is the smoothing constant $(0 < \lambda \leq 1)$. The quantity plotted on the control chart is

$$T_i^2 = z'_{(i)} \sum_{z_{(i)}}^{-1} z_{(i)} \tag{8}$$

where the covariance matrix for T_i^2 statistics is

$$\sum_{z_{(i)}} = \frac{\lambda}{2-\lambda}[1-(1-\lambda)^{2i}]\Sigma \tag{9}$$

where Σ is a covariance matrix obtained from the training dataset. The control limit $H > 0$ is chosen to achieve a specified in-control (on-target) average run length (ARL). The value of control limit H is chosen based upon a table presented in [38]. Moreover, the value of H depends upon the number of variables to be monitored, as the number increases the value of H grows simultaneously. When, the T_i^2 statistics falls outside the control limit H, the shift-point is detected. The criteria of optimizing the smoothing parameter (λ) are related to the detection capability. In [37,39], to select the optimal smoothing parameter λ, two methods are suggested. The first method assumes that each variable of $x^d_{(i)}$ is similar and has the same smoothing constant. The second method assumes that each variable of $x^d_{(i)}$ may have different characteristics requiring appropriately matched parameters λ_i selected based on the optimization rule of univariate EWMA. However, the components of $x^d_{(i)}$ may often be correlated very closely and it will be very difficult to find λ_i independently. So, to make the components uncorrelated, an orthogonal transformation is used such that all new components are independent of each other.

The principal component analysis (PCA) is often used to reduce the dimensionality of the data. Moreover, the PCA is used to select a small number of uncorrelated components, containing most of the variability in the data [40,41]. On the other hand, a popular class of algorithms to separate independent sources, called independent component analysis (ICA), make the simplification that finding independent sources out of such data can be reduced to finding maximally non-Gaussian components. For a meaningful

Table 2
Algorithm-TSSD-EWMA.

```
Input: Submit the training dataset to the training phase and compute the parameters for testing.
  Receive new data in the testing phase sample-by-sample and perform the check as follows.
    IF (Shift detected)
    THEN (Report the point of shift and move to stage-II for validation)
    ELSE (Continue and integrate the upcoming information).
Output: Shift-detection points.

Stage-I
Training Phase
    1. Assign training data to $x_{(i)}$ for $i=1{:}n$, where $n$ is the number of observations in training data
    2. Calculate the mean of $x_{(i)}$ and set as $z_{(0)}$.
    3. Compute the $z$-statistics for each observation $x_{(i)}$ in training data for a range of $\lambda$ values.
       $z_{(i)}=\lambda x_{(i)}+(1-\lambda)z_{(i-1)}$
    4. Estimate $\lambda$ by minimizing over the training dataset the square of 1-step-ahead prediction error: $err_{(i)}=x_{(i)}-z_{(i-1)}$.
    5. Finally estimate the variance of error for the testing phase.

Testing Phase
    1. For each data point $x_{(i)}$ in the operation/testing phase
    2. Compute $z_{(i)}=\lambda x_{(i)}+(1-\lambda)z_{(i-1)}$
    3. Compute $err_{(i)}=x_{(i)}-z_{(i-1)}$
    4. Estimate the variance $\hat{\sigma}^2_{err_{(i)}}=\vartheta\, err^2_{(i)}+(1-\vartheta)\hat{\sigma}^2_{err_{(i-1)}}$
    5. Compute $UCL_{(i)}$ and $LCL_{(i)}$:
    6. $UCL_{(i)}=z_{(i-1)}+L\hat{\sigma}_{err_{(i-1)}}$
    7. $LCL_{(i)}=z_{(i-1)}-L\hat{\sigma}_{err_{(i-1)}}$
    8. IF ($LCL_{(i)}<x_{(i)}<UCL_{(i)}$)
        THEN (Continue processing)
        ELSE (Go to Stage-II)

Stage-II
    1. For each $x_{(i)}$
    2. Wait for $m$ observations after the time $i$, organize the sequential observations around time $i$ into two partitions, one containing $x_{((i-(m-1)):i)}$, another $x_{((i+1):(i+m))}$.
    3. Execute the hypothesis test on the partitioned data
    4. IF ($H=1$)
        THEN (test rejects the null hypothesis): Alarm is raised
        ELSE (The detection received by stage-I is a false and discarded)
```

representation of the multivariate data, linear transformation of the original data is required. Using independent component analysis (ICA) [42], a linear representation of the data is obtained in the form of statistically independent components. Some popular ICA algorithms are Fast-ICA [42], Infomax-ICA [43] and Fully Blind Source Separation (FBSS) [44].

The number of uncorrelated components obtained from an orthogonal transformation such as PCA or ICA determines the value of the smoothing constant λ. Based on the number of components, the value for the smoothing constant is selected from the table given in [38]. The table presents the value of λ for a process (x^d), where d is the number of variables to be monitored. It is suggested that the smaller values of λ are preferred for detecting small shifts and vice versa. Moreover, as the number of variables d increases, the value of control limit H increases.

As we have discussed above, the EWMA control chart is robust to the normality assumption if properly designed for the t and gamma distributions [35]. So, the PCA is used as a pre-processing step to reduce the dimensionality of data. The ICA is used to identify the maximally separated independent sources and then the multivariate shift-detection tests are used to detect the shift in the process. We have compared performance of both the PCA and ICA based algorithms in the experiments. As in the case of univariate shift-detection, a two-stage based structure is developed to address the issue of false alarms.

3.4. Two-stage multivariate shift-detection based on EWMA (TSMSD-EWMA)

The proposed two-stage multivariate shift-detection test based on EWMA (TSMSD-EWMA) works in two-stages as in the case of TSSD-EWMA. In stage-I, the method employs a control chart to detect the dataset shift in the data stream. Stage-I works in an online mode, which continuously processes the upcoming data from the data stream. Stage-II uses a multivariate statistical hypothesis test to validate the shift detected by stage-I and operates in retrospective mode. A stage-wise algorithmic formulation is discussed below.

3.4.1. Stage-I

In stage-I, the test works in two different phases. The first phase is a training phase and the second phase is an operation or testing phase. In the first phase, the parameter ($\Sigma_{z_{(i)}}$) defined in Eq. (7) is calculated to decide the null hypothesis that there is no shift in the data. In the testing phase, for each observation, Eq. (8) is used to obtain the T_i^2 statistics. Next, it is checked if each T_i^2 statistics for each observation $x^d_{(i)}$ falls below the control limits H, otherwise the shift is detected and alarm is raised at stage-I. Once, the shift is detected by stage-I, it is passed on to stage-II for validation in order to reduce the number of false-positive alarms.

3.4.2. Stage-II

This phase works in a retrospective mode and it executes only when a shift is detected in stage-I. In particular, to validate the shift detected by stage-I, the available information need to be partitioned into two-disjoint subsequences and then the statistical hypothesis test is applied. The Hotelling T-Squared test for two multivariate independent samples [45] is used to validate the stationarity in the sub-sequences. The reason for choosing this test is because it is a non-parametric method and it returns a test decision whether the data in the subsequences are stationary with similar means. Hotelling's T-Squared test is defined by the following equation:

$$HT^2=(\mu_1-\mu_2)\left\{\left(\frac{\Sigma_1}{|n1|}+\frac{\Sigma_2}{|n2|}\right)\right\}^{-1}(\mu_1-\mu_2)' \tag{10}$$

where μ_1, μ_2 and Σ_1, Σ_2 are the means and the covariances of sample 1 and sample 2 (i.e., two sub-sequences here), respectively. The HT^2 is distributed as $F(d, n1+n2-d-1)$, where $n1$ and $n2$ are the two sub-sequences of lengths $|n1|$ and $|n2|$ for sample 1 and sample 2, respectively. Moreover, sample 1 was assumed to be stationary; the data sub-sequence $n_1 = ((i-(m-1)) : i)$; the data sub-sequence $n_2 = ((i+1) : (i+m))$; here i is the current observation, d is the number of columns (i.e., variables), and F is the F-distribution. The null hypothesis is rejected, if the HT^2 test statistic is greater than the critical value from the F-distribution.

4. Datasets and feature analysis

To validate the effectiveness of the proposed algorithms, a series of experimental evaluations have been performed on four synthetic datasets and one real-world dataset. The datasets are described as follows.

4.1. Synthetic data

Dataset 1 – abrupt shift (D1)

The dataset consists of 2000 data-points and the non-stationarity occurs in the middle of the data stream, shifting from $\mathcal{N}(x;\ 1,1)$ to $\mathcal{N}(x;\ 3,1)$, where $\mathcal{N}(x;\ \mu,\sigma)$ denotes the normal distribution with mean μ and standard deviation σ respectively.

Dataset 2 – jumping mean (D2)

The dataset used here is the same as the toy dataset given in [6] for detecting shift point in time-series data. The dataset is defined as $x(t)$ in which 5000 samples are generated (i.e. $t = 1, \ldots, 5000$)

$$x(t) = 0.6x(t-1) - 0.5x(t-2) + \varepsilon_t$$

where ε_t is a noise with mean μ and standard deviation 1.5. The initial values are set as $x(1) = x(2) = 0$. A shift point is inserted at every 100 time steps by setting the noise mean μ at time t as

$$\mu_N = \begin{cases} 0 & N = 1 \\ \mu_{N-1} + \frac{N}{16} & N = 2, \ldots, 49 \end{cases}$$

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

Dataset 3 – multivariate normal shift (D3)

This dataset is a 10-dimensional normal distribution $\mathcal{N}(x;\ M, \Sigma)$, where M is the mean vector and Σ is the covariance matrix. The stream consists of 300 data points, in which the non-stationarity occurs after generating 100 points; the mean vector M of each variable is shifted from 0 to 1 and then back to its initial position at 201, while the covariance matrix remains fixed Σ as

$$\Sigma = \begin{bmatrix} 0.45 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.45 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.45 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.45 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.45 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.45 & 0.3 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.45 & 0.3 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.45 & 0.3 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.45 & 0.3 \\ 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.3 & 0.45 \end{bmatrix}$$

Dataset 4 – multivariate non-normal shift (D4)

This dataset is a 10-dimensional t-distribution, where M is the mean vector and Σ is the covariance matrix as given in D3 dataset. The stream consists of 300 data points, in which the non-stationarity occurs after generating 100 points; the mean vector M of each variable is shifted from 0 to 1 and then back to its initial position at position 201, while the degree of freedom remains fixed at 10.

4.2. Real-world dataset

Dataset 5 – EEG-based brain signals (D5)

The real-world data used here are from the BCI competition-III dataset (Section 4.2) [46]. This dataset, contains 2 classes, 118 EEG channels (0.05–200 Hz), 1000 Hz sampling rate which is down-sampled to 100 Hz, 210 training trials, and 420 test trials. This dataset was recorded from one healthy subject. He sat in a comfortable chair with arms resting on armrests. The training dataset consists of the first 3 (non-feedback) sessions. Visual cues (letter presentation) indicated for 3.5 s required the subject to perform (L) left hand or (F) right foot motor imageries. The presentation of the target cues was intermitted by periods of random length, 1.75–2.25 s, in which the subject could relax. The test data was recorded more than 3 h after the training data. The experimental setup was similar to the training sessions, but the motor imagery had to be performed for 1 s only, compared to 3.5 s in the training sessions. The intermitting periods ranged from 1.75 to 2.25 s as before. For the training purposes, the data from session-1 is used and it is assumed that it is in stationary state. For testing phase, there are 4 sessions and blocks of 10 trials are selected from each session of the experiment. Further, the selected blocks are merged and passed as a data stream for testing the shift in the non-stationary EEG time-series. It is clear from Fig. 4 that EEG data is non-stationary over multiple sessions of a BCI experiment. Moreover, from Fig. 4 we can easily mark the shift in the mean and therefore the change in the feature distribution in the EEG data obtained from different trainings and test sessions of a BCI experiment. If these continuous shifts in the distribution are detected at the time of the occurrence, then an appropriate action could be taken to account for the dataset shifts. This real-world dataset is a good example to validate if the proposed methods are able to detect the covariate shifts in the data generating process. To perform the test on the univariate and multivariate data, the dataset has been divided into two categories, single and multi-channels respectively.

(A) *Univariate*: A single channel (C3) was selected and the band-pass filtering was performed over mu (μ) band (8–12 Hz) to extract band-power features.

(B) *Multivariate*: Total five channels (C3, C1, Cz, C2, and C4) are selected and band-pass filtering was performed over the mu

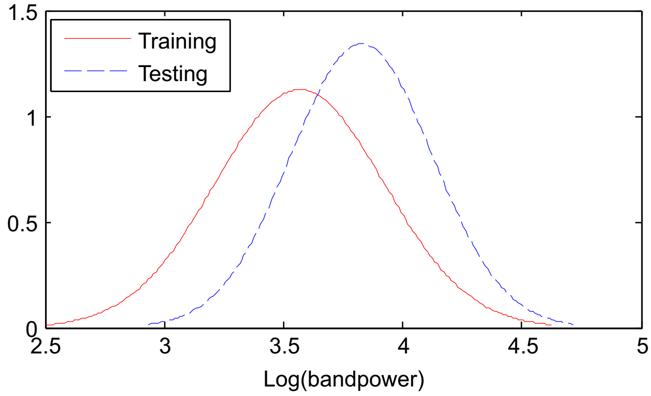

Fig. 4. Probability density plot of the data taken from the training (red solid-line) and testing session (blue doted-line). It is clear from the plot that, in different sessions the distribution is changed by shifting the mean from session-to-session transfer. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

(μ) band (8–12 Hz) in each channel to extract band-power features. Next, the data pre-processing step was performed (i.e., data normalization, making it to zero mean by removing the average).

5. Experimental evaluation

5.1. Evaluation metrics

On each dataset, the covariate shift-detection techniques are evaluated by the number of false alarms and number of misses. A false alarm is the signal of false detection or type-I error. A false-negative is a miss and measured as type-II error. The following keys are suggested to measure the performance of the tests:

- *False positive* (*FP*): it occurs when a test detects a shift in the sequence when it is not there, (i.e. false alarm) or type-I error.
- *False negative* (*FN*): it occurs when a test does not detect a shift in the sequence when it is there, (i.e. a miss) or type-II error.
- *Recognition Capability Index* (*RCI*): measures the delay in the shift-detection process, (i.e. the number of observations processed before reporting of the shift).
- *Computation Time* (*CT*): provides the execution time needed to perform the shift-detection test on the entire dataset, (machine configuration: Intel Xeon, 3-GHz, 16-GB RAM, Windows 7 and all unnecessary processes are terminated).

Table 3
Simulation results of the univariate datasets.

	SD-EWMA	TSSD-EWMA	ICI-CDT
Synthetic dataset			
D1			
FP (%)	0.05	0	0
FN (%)	0	0	0
RCI	0	10	60
CT (s)	0.198	0.245	0.172
D2			
FP (%)	0.3	0	0
FN (%)	0	11.1	55.4
RCI	0	10	40
CT (s)	0.189	0.224	0.196
Real-dataset			
D5-A			
FP (%)	0.08	0.0002	NA
FN (%)	0	0	NA
RCI	0	10	NA
CT (s)	0.296	0.283	NA

Results are given in Tables 3 and 4 for univariate and multivariate datasets, respectively. The NA denotes a “Not Applicable” situation and the test cannot be run because of the lack of available information. To assess the performance of the proposed shift-detection tests against other shift-detection methods, the intersection of confidence interval change detection test ICI-CDT [47] is chosen because it is a state-of-the-art non-parametric sequential change-point detection test, shown to provide good performance in recent literature.

The choice of the smoothing constant λ is an important issue in the EWMA based shift-detection tests. In [48] a method for selecting λ is already suggested for uncorrelated observations. Moreover, for correlated data, it is suggested in [34] to select λ that minimizes the sum of the squares of the 1-step ahead prediction errors. In the experiments, we have used several techniques to select the values of λ such as minimizing the sum of squares of 1-step-ahead prediction errors and heuristic methods, for example, a trial-and-error approach so as to minimize the occurrences of FPs and FNs. The values of λ selected by minimizing the sum of squares of 1-step-ahead prediction errors has been discussed in our papers [4,5] and these values are quite close to the values obtained empirically using heuristic methods. In this paper, for univariate case, the values of λ are selected by minimizing the sum of squares of 1-step-ahead prediction errors. In the case of multivariate data, the value of λ has been selected based on the design of multivariate control chart to achieve a specified in-control (on-target) average run length (ARL). The value of control limit H is chosen based upon a table presented in [38]. The design parameters of the charts depend upon the number of variables to be monitored and some a-priori knowledge about the expected shift in the process.

5.2. Results and discussion

5.2.1. Univariate shift-detection

For the dataset D1, the value of smoothing constant is obtained as $\lambda=0.50$, by minimizing the sum of squares of the prediction errors on training data. Table 3 shows that the performances of SD-EWMA, TSSD-EWMA and ICI-CDT are the same in terms of FN rate (all zeroes). The TSSD-EWMA and ICI-CDT are better by reducing the percentage of FP. The delay in TSSD-EWMA shift-detection is shorter than ICI-CDT. Moreover, the TSSD-EWMA is

Table 4
Simulation results of the multivariate datasets

	MSD-EWMA	MSD-EWMA-PCA	MSD-EWMA-ICA	TSMSD- EWMA	TSMSD-EWMA-ICA
Synthetic dataset					
D3					
FP (%)	5	5.5	5	2	2.5
FN (%)	8	6	8	5	5
RCI	9	8	8	25	25
CT (s)	0.228	0.272	0.528	0.201	0.594
D4					
FP (%)	4	5.5	5	5	4.5
FN (%)	15	5	15	7	9
RCI	7	8	7	25	25
CT (s)	0.213	0.251	0.677	0.202	0.647
Real-dataset					
D5-B					
FP (%)	44.65	25.80	37.70	37.70	28.40
FN (%)	62.40	71.16	15.81	70.50	18.13
RCI	NA	NA	NA	NA	NA
CT (s)	0.347	0.698	16.612	0.597	16.462

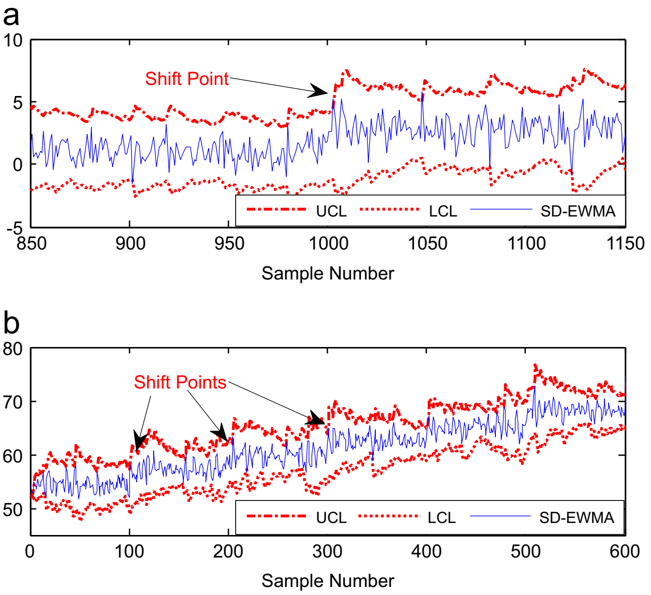

Fig. 5. Univariate Dataset: (a) D1: detects the covariate shift by the SD-EWMA based test, the shifts are detected after producing every 1000 observations, and few false-positives can also be marked near 900, 1050, 1085, and 1125 data points. (b) D2: the covariate shift point is detected after every 100 observations.

slightly more computationally expensive than other tests. Fig. 5(a) shows the plot of covariate shift-detection.

For the dataset D2, the value of the smoothing constant is obtained as $\lambda = 0.40$, by minimizing the sum of squares of the prediction errors on training data. The performance of SD-EWMA is better than that of TSSD-EWMA and ICI-CDT in terms of FN, whereas there are some FPs with SD-EWMA. These FPs have been reduced by the TSSD-EWMA test because of its two-stage structure. This result demonstrates the effectiveness of the TSSD-EWMA algorithm in reducing the FP. Also, the computational delay in TSSD-EWMA is better than ICI-CDT, which shows the advantage of the test. The FN rate for TSSD-EWMA is large because it missed a single shift out of total nine shifts. Fig. 5(b) shows the plot of covariate shift-detection.

The dataset D5-A is a single channel EEG data. Now, to perform the covariate shift-detection test, we have taken data from session one; it contains 70 trials and the parameters are calculated in the training phase to be used in the testing/operational phase. Next, the test is applied on the set of trials from rest of the sessions as discussed in Section 4.2 and the results are given in Table 3. In the case of D5-A, the value of the smoothing constant was obtained as $\lambda = 0.05$ by minimizing the sum of squares of the prediction errors on training data which is obtained from the session-1 of the BCI experiment. The TSSD-EWMA provided better performance than that of SD-EWMA and ICI-CDT is not applicable due to lack of a prior information. Fig. 7(a) shows a window of 10 s for covariate shift-detection on the EEG data. In this window, no shift is detected, as the solid line never crosses the control limits plotted as dotted lines.

5.2.2. Multivariate shift-detection

In the dataset D3, the data are from a multivariate normal distribution. The value of the H (i.e., control limit) is chosen as suggested in [38]. For the MSD-EWMA and MSD-EWMA (ICA), the value of H is 22.67 as there are ten variables to be monitored (i.e., the dimensions of the dataset) for the shift-detection. For MSD-EWMA (PCA), the value of H is 10.58 because in PCA only first few components are used and we have selected first two components, so the value of control limit is smaller. For the multivariate tests

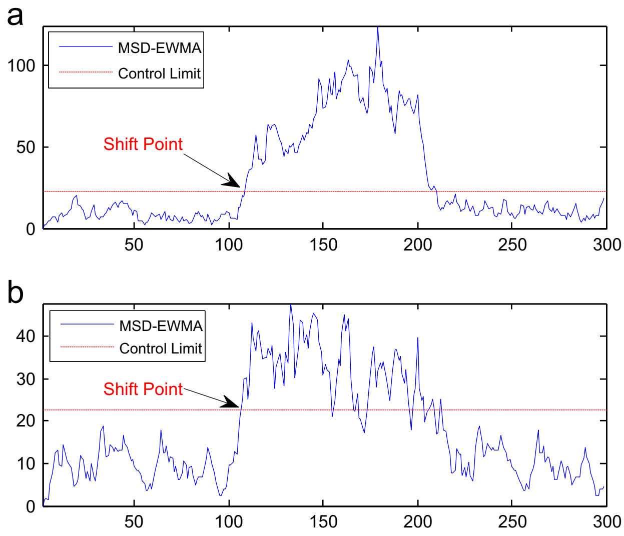

Fig. 6. Multivariate Dataset: (a) D3: detects the covariate shift by the MSD-EWMA based test, the shift is detected after the 100th observation, and (b) D4: the covariate shift point is detected after 100th observations. The normal distributed data can be easily be detected over non-normal data.

such as MSD-EWMA, MSD-EWMA (PCA) and MSD-EWMA (ICA), the performances are very closely similar as there are nearly the same number of FPs, whereas the performances of TSMSD-EWMA and TSSMD-EWMA with ICA are better in terms of having less number of FPs. In the TSMSD-EWMA and TSSMD-EWMA with ICA, the delay in shift-detection is increased. The delay is increased due to the cardinality of the sample set used for validation in the second stage of the test. Fig. 6(a) shows the detected shift. Lastly, the computational time for tests with ICA has slightly increased.

For the dataset D4, the data are from a multivariate t-distribution. The limit of H is the same as in D3 for all the multivariate tests. The performances of MSD-EWMA, MSD-EWMA-ICA, and TSMSD EWMA-ICA are well-meaning with nearly the same number of FPs, whereas the MSD-EWMA-PCA is less accurate in reducing the number of FPs. However, the TSMSD-EWMA with ICA has shown a better performance over other methods with slightly less FPs. Furthermore, the ICA approaches are slightly more computationally expensive because of the computational cost of the ICA algorithm. Fig. 6(b) shows the detected shift.

For the dataset D5-B the data are 5-channels of EEG data; each channel is treated as a variable. The value of $H = 16.27$ is selected based upon the number of variables to be monitored. Now, to perform the shift-detection test, we have taken data from session-1 and assumed that it is stationary; it contains 70 trials and the parameter $\Sigma_{z_{(i)}}$ is calculated using Eq. (9). Next, the test is applied to the rest of the sessions on a specific set of trials. From each session, 10 trials are selected and combined so as to form a data stream. The results of the test are given in Table 4. It has been discussed previously that when the number of variables is increased the performance of the system will degrade. So, the MSD-EWMA has the worst performance with high FN and FP rates as expected. Moreover, by using PCA, the performance of the test has slightly improved by reducing the rate of FP. The MSD-EWMA with ICA improves the accuracy of the result with almost 50% less FN as compared to other methods such as MSD-EWMA and MSD-EWMA-PCA given in Table 4. The issue of FPs again was handled by the two-stage test and the number of FPs has been reduced. Whereas, the TSMSD-EWMA without ICA suffers from high rate of FNs and with ICA it shows better performance. Fig. 7(b) shows the MSD-EWMA with ICA. This shows the advantage of using ICA for the EEG data, as it makes the components independent and more appropriate for the covariate shift-detection test.

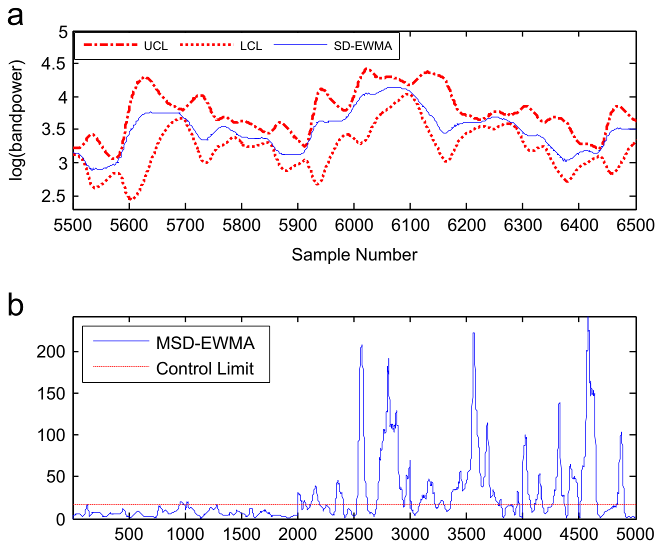

Fig. 7. Real-world dataset: EEG-based brain signals (a) D5-A: univariate shift-detection (single channel). The window shows a single trial and no shift is detected as the solid line never crosses the red line, (b) D5-B: multivariate shift-detection (multi-channel) with ICA. The solid line is the multivariate T^2 statistics, after 2000 observations, the data are from different sessions; so it has crossed the control limit *H*. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

It is important to note that SD-EWMA is based on the current observation of the data to detect the covariate-shift, so it detects without delay and this is the main advantage of this method. Fig. 5(a) and (b) represents the SD-EWMA based covariate shift-detection test results for datasets D1 and D2 respectively. The solid line is the observation plotted on the chart and the two dotted lines are the ULC and LCL; whenever the solid line crosses the dotted line (control limits), it is the covariate shift-point detection. The value of smoothing constant λ is another important issue, which we have discussed earlier. In D5-A, for the real-world dataset, (i.e. EEG based BCI), the smaller value of λ is a better choice for the shift-detection and is obtained by minimizing the sum of squares of one-step ahead prediction errors. The smaller values of λ avoids covariate shift-detection resulting from noise or spurious changes through much more intense smoothing of EEG signal. Moreover, for correlated data, the smaller values of λ produce smaller prediction errors, thereby resulting in smaller estimated standard error. The SD-EWMA test shows the issue of the occurrence of large number of FPs, which is a concern for most of the shift-detection tests. So, the TSSD-EWMA is used to validate the shift using the two-stage structure. It provides promising results by reducing the number of false alarms at the second stage. However, there is a small delay in the TSSD-EWMA based shift-detection test.

For the multivariate shift-detection, the classical multivariate shift-detection (MSD-EWMA) shows that if the number of variables in the process increases, it has an adverse effect on the performance of the test. The adverse effect leads to large rate of FN and FP. Moreover, in the real-time applications getting high rate of false alarms is painful. However, using PCA with MSD-EWMA the performance of the test has slightly improved as the PCA has reduced the dimensionality of the data. The MSD-EWMA with ICA shows good performance for the real-world dataset. The ICA works well by identifying the independent components in EEG based brain signals.

As a final note, we recommend TSMSD-EWMA with ICA (Infomax) configuration as the most suitable covariate shift-detection test for identifying a possible shift in an EEG-based brain signals for multiple channels, as other ICA algorithms such as fast-ICA and FBSS were found much less successful in reducing false alarms.

6. Conclusion

In NSL, the covariate shift-detection is an important aspect for initiating the corrective adaptive action. This paper presented novel methods of covariate shift-detection in the NSEs based on a two-stage structure for both univariate and multivariate time-series. The first stage uses an exponentially weighted moving average (EWMA) model based control chart to detect the covariate shift-point in non-stationary time-series. At first stage, to choose the smoothing parameter λ optimally, the minimization of prediction of error is used for the univariate case, and for the multivariate case the orthogonal transformation is suggested to make the data uncorrelated. The second stage validates the shift-detected by first stage using the Kolmogorov–Smirnov (K–S test) statistical hypothesis test in the case of univariate time-series and Hotelling's *T*-Squared multivariate statistical hypothesis test in the case of multivariate time-series. The two-stage structure helped reduce the rate of false alarm for all example data-sets. Also the ICA (Infomax) algorithm has been found to be most effective in counteracting the adverse effect of cross-correlation in multivariate time-series and further improving the shift-detection performance. The methods are found computationally more efficient in terms of both computation time and storage size. Experimental analysis shows that the proposed approaches perform well in a range of non-stationary situations. This work is planned to be extended further by employing it into pattern recognition problems along with an appropriate classifier.

Appendix A. Relationship between EWMA and ARIMA

An autoregressive integrated moving average (ARIMA) model is the generalization of an autoregressive moving average (ARMA) model. These models are fitted to time series data either to better understand the data characteristics or to predict future points in the series (forecasting). Many non-stationary series are found to be fitted quite well with ARIMA (0,1,1) [33].

The exponentially weighted moving average (EWMA) Eq. (A1) can also be written in the form of an autoregressive integrated moving average (ARIMA (0,1,1)) Eq. (2) as described in Box and Jenkins [49].

$$z_{(i)} = \lambda x_{(i)} + (i-1) z_{(i-1)} \tag{A1}$$

Eq. (A2) below is usually denoted as IMA (1,1), where the first 1 in the parentheses denotes difference in the series one time and the second 1 denotes fitting a moving average parameter. Eq. (A2) can also be represented in an autoregressive moving average form ARIMA (0,1,1) where the number 0 indicates that the order of the autoregressive part is zero. The general form of the IMA (1,1) model is

$$x_{(i-1)} = x_{(i)} + \varepsilon_{i+1} - \theta \varepsilon_i \tag{A2}$$

where ε_i represents a random shock referred to as white noise. The white noise is with mean zero and some variance σ^2.

Note that for predicting x at time period $(i+1)$ given all the information up to and including time period i, you would obtain Eq. (A3), since the prediction for ε_{i+1} is zero, as it is assumed to be purely white noise.

$$\hat{x}_{(i-1)} = x_{(i)} - \theta \varepsilon_i \tag{A3}$$

Since $\varepsilon_i = x_{(i)} - \hat{x}_{(i)}$, and putting it into Eq. (A3) you will get Eq. (A4)

$$\begin{aligned} \hat{x}_{(i-1)} &= x_{(i)} - \theta (x_{(i)} - \hat{x}_{(i)}) \\ \hat{x}_{(i-1)} &= (1-\theta)\, x_{(i)} - \theta \hat{x}_{(i)} \end{aligned} \tag{A4}$$

Let us denote $\lambda = 1-\theta$, $\hat{x}_{(i)} = z_{(i-1)}$ and $z_{(i)} = \hat{x}_{(i-1)}$. Then

$$\hat{x}_{(i-1)} = (\lambda)\, x_{(i)} - (i-1)\hat{x}_{(i)}$$
$$\hat{x}_{(i-1)} = (\lambda)\, x_{(i)} - (1-\lambda) z_{(i-1)}$$
$$z_{(i)} = \lambda x_{(i)} + (1-\lambda) z_{(i-1)}$$

Note that in comparing the equation for EWMA and IMA (1,1), we have proved that the ARIMA (0,1,1) can be written as EWMA, and it will provide an optimal 1-step ahead prediction.

References

[1] H. Shimodaira, Improving predictive inference under covariate shift by weighting the log-likelihood function, J. Stat. Plan. Inference 90 (2) (2000) 227–244.

[2] R. Elwell, R. Polikar, Incremental learning of concept drift in non-stationary environments, IEEE Trans. Neural Netw. 22 (10) (2011) 1517–1531.

[3] Y. Kawahara, M. Sugiyama, Sequential change-point detection based on direct density-ratio estimation, Stat. Anal. Data Min. 5 (2) (2012) 114–127.

[4] H. Raza, G. Prasad, Y. Li, EWMA based two-stage dataset shift-detection in non-stationary environments, in: Proceedings of the Artificial Intelligence Applications and Innovations, 2013, pp. 625–635.

[5] H. Raza, G. Prasad, Y. Li, Dataset shift detection in non-stationary environments using EWMA charts, in: Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics, 2013, pp. 3151–3156.

[6] S. Liu, M. Yamada, N. Collier, M. Sugiyama, Change-point detection in time-series data by relative density-ratio estimation, Neural Netw. 43 (2013) 72–83, Retrieved from http://www.sciencedirect.com/science/article/pii/S0893608013000270.

[7] B. Blankertz, G. Curio, K. Müller, N. Group, K.B. Franklin, Classifying single trial EEG: towards brain computer interfacing, Adv. Neural Inf. Process. Syst. 14 (2002) 157–164.

[8] S. Bickel, T. Scheffer, Dirichlet-enhanced spam filtering based on biased samples, Adv. Neural Inf. Process. Syst. 19 (2007) 161–168.

[9] N. Ye, S. Vilbert, Q. Chen, Computer intrusion detection through EWMA for autocorrelated and uncorrelated data, IEEE Trans. Reliab. 52 (1) (2003) 75–82.

[10] G.W. Snedecor, W.G. Cochran, Statistical Methods, eight ed., Iowa State University Press, Ames, Iowa, 1989.

[11] C.M. Douglas, Introduction to Statistical Quality Control, 5th ed., John Wiley & Sons, USA, 2007.

[12] H. Mann, D. Whitney, On a test of whether one of two random variables is stochastically larger than the other, Ann. Math. Stat. 18 (1) (1947) 50–60.

[13] W.A. Shewhart, Statistical Method from the Viewpoint of Quality Control, Dover Publications, Mineola, NY, 1939.

[14] M. Basseville, I. Nikiforov, Detection of Abrupt Changes: Theory and Application, Prentice-Hall, Cliffs, Englewood, 1993.

[15] C. Alippi, M. Roveri, Just-in-time adaptive classifier – Part I: detecting nonstationary changes, IEEE Trans. Neural Netw. 19 (7) (2008) 1145–1153.

[16] C. Alippi, G. Boracchi, M. Roveri, A just-in-time adaptive classification system based on the intersection of confidence intervals rule, Neural Netw. 24 (8) (2011) 791–800.

[17] M. Markou, S. Singh, Novelty detection: a review – Part 1: statistical approaches, Signal Process. 83 (12) (2003) 2481–2497.

[18] M. Markou, S. Singh, Novelty detection: a review – Part 2: neural networks, Signal Process. 83 (12) (2003) 2499–2521.

[19] F. Gustafsson, The marginalize likelihood ratio test for detecting abrupt changes, IEEE Trans. Autom. Control 41 (1) (1996) 66–78.

[20] M. Sugiyama, T. Suzuki, T. Kanamori, Density Ratio Estimation in Machine Learning, Cambridge University Press, New York, USA, 2012.

[21] C. Alippi, G. Boracchi, M. Roveri, Just-in-time classifiers for recurrent concepts, IEEE Trans. Neural Netw. Learn. Syst. 24 (4) (2013) 620–634.

[22] A.J. Storkey, When training and test sets are different: characterising learning transfer, Dataset Shift in Machine Learning, Section 12, MIT Press, Cambridge, Massachusetts (2010) 3–28.

[23] M.J.G. Torres, T. Raeder, R. Alaiz-Rodríguez, N.V. Chawla, F. Herrera, A unifying view on dataset shift in classification, Pattern Recognit. 45 (1) (2012) 521–530.

[24] M. Sugiyama, A. Schwaighofer, N.D. Lawrence, Dataset Shift in Machine Learning, The MIT Press, Cambridge, Massachusetts, 2009.

[25] G. Widmer, M. Kubat, Learning in the presence of concept drift and hidden contexts, Mach. Learn. 101 (23) (1996) 69–101.

[26] K. Wang, C.A. Fu, J.X. Yu, Mining changes of classification by correspondence tracing, in: Proceedings of the 3rd SIAM International Conference on Data Mining (SDM-03), 2003, pp. 95–106.

[27] N. Japkowicz, Assessing the impact of changing environments on classifier performance, in: Proceedings of the 21st Canadian Conference in Artificial Intelligence, 2008, pp. 13–24.

[28] Y. Yang, X. Wu, X. Zhu, Conceptual equivalence for contrast mining in classification learning, Data Knowl. Eng. 67 (3) (2008) 413–429.

[29] D.a. Cieslak, N.V. Chawla, A framework for monitoring classifiers' performance: when and why failure occurs? Knowl. Inf. Syst. 18 (1) (2008) 83–108.

[30] M.G. Kelly, D.J. Hand, N.M. Adams, The impact of changing populations on classifier performance, in: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, vol. 32, no. 2, 1998, pp. 367–371.

[31] S. Bickel, Learning under Differing Training and Test Distributions, University of Potsdam, 2009.

[32] S.W. Roberts, Control chart tests based on geometric moving averages, Technometrics 1 (3) (1959) 239–250.

[33] T.C. Mills, Time Series Techniques For Economists, Cambridge University Press, New York, USA, 1991.

[34] D. Montgomery, C. Mastrangelo, Some statistical process control methods for autocorrelated data, J. Qual. Technol. 23 (3) (1991) 179–204.

[35] M.B. Connie, C.M. Douglas, C.R. George, Robustness of the EWMA control chart to non-normality, J. Qual. Technol. 31 (3) (1999) 309.

[36] Table of Critical Values for The Two-Sample Test. Available from: 〈http://www.soest.hawaii.edu/wessel/courses/gg313/Critical_KS.pdf〉 (online).

[37] C.A. Lowry, W.H. Woodall, C.W. Champ, S.E. Rigdon, A multivariate exponentially weighted moving average control chart, Technometrics 34 (1) (1992) 46–53.

[38] S. Sharad, C. George, Designing a multivariate EWMA control chart, J. Qual. Technol. 29 (1) (1997) 8.

[39] L. Yumin, An improvement for MEWMA in multivariate process control, Comput. Ind. Eng. 31 (3) (1996) 779–781.

[40] A. Miguel, A Review of Dimension Reduction Techniques, 1997.

[41] K. Pearson, LIII. On lines and planes of closest fit to systems of points in space, The London, Edinburgh, and Dublin Philosophical … (1901) 559–572.

[42] A. Hyvärinen, E. Oja, Independent Component Analysis: Algorithms and Applications, Neural Netw. 13 (4–5) (2000) 411–430.

[43] A.J. Bell, T.J. Sejnowski, An information-maximization approach to blind separation and blind deconvolution, Neural Comput. 7 (6) (1995) 1129–1159.

[44] X.-L. Li, Blind spatiotemopral separation of second and/or higher-order correlated sources by entropy rate minimization, in: Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), Dallas, TX, no. 1, 2010, pp. 1934–1937.

[45] R.A. Johnson, D.W. Wichern, Applied Multivariate Statistical Analysis, 6th ed., Prentice Hall, New York (2007) 210–258.

[46] G. Dornhege, B. Blankertz, G. Curio, K.-R. Müller, Boosting bit rates in noninvasive EEG single-trial classifications by feature combination and multi-class paradigms, IEEE Trans. Biomed. Eng. 51 (6) (2004) 993–1002.

[47] C. Alippi, G. Boracchi, M. Roveri, Change detection tests using the ICI rule, in: Proceedings of the International Joint Conference on Neural Networks, July 2010, pp. 1–7.

[48] J.M. Lucas, M.S. Saccucci, Exponentially weighted moving average control schemes: properties and enhancements, Technometrics 32 (1) (1990) 1.

[49] G.E.P. Box, G.M. Jenkins, Time Series Analysis, Forecasting and Control, Series in. Holden-Day, San Francisco: John Wiley & Sons, USA, 1970.

Haider Raza received the B.Tech. degree in Computer Science and Engineering from the Integral University, Lucknow, India, in 2008, the M.Tech. degree in Computer Engineering from the Manav Rachna International University, Delhi-NCR, India, in 2011. From 2012, he is pursuing his Ph.D. degree from the University of Ulster, UK. His Ph.D. research is focused on adaptive learning for non-stationary systems.

Mr. Raza worked as an Assistant Professor with the School of Mathematical and Computer Science, Dilla University, Ethiopia, from 2011 to 2012. In 2008–2009, he worked as a lecturer in Cosmic Business School, New Delhi, India. Since, 2008, he is Microsoft Certified Technology Specialist (MCTS) in C# for web and command based applications. His current research interests include shift-detection and learning in non-stationary environments, machine learning, pattern recognition and artificial intelligence.

Girijesh Prasad received the B.Tech. degree in Electrical Engineering from NIT (formerly REC), Calicut, India, in 1987, the M.Tech. degree in Computer Science and Technology from IIT (formerly UOR), Roorkee, India, in 1992, and the Ph.D. degree from Queen's University of Belfast, UK, in 1997. He has been an academic staff member with the University of Ulster, Magee Campus, Derry, UK, since 1999, and is currently professor of intelligent systems. He is an executive member of Intelligent Systems Research Centre, where he leads the Brain–Computer Interface and Assistive Technology Team. He has published over 160 research papers in international journals, books, and conference proceedings. His current research interests include self-organizing hybrid intelligent systems, statistical signal processing, adaptive predictive modeling and control with applications in complex industrial and biological systems including brain modeling, brain–computer interfaces and neuro-rehabilitation, assistive robotic systems, biometrics, and energy systems.

Prof. Prasad is a chartered engineer and a fellow of the IET. He is a founding member of IEEE SMC TCs on Brain–Machine Interface Systems and Evolving Intelligent Systems.

Yuhua Li received the Ph.D. degree in general engineering from the University of Leicester. He worked at the Manchester Metropolitan University and then the University of Manchester from June 2000 to September 2005 as a Senior Research Fellow and a Research Associate, respectively. Since October 2005, he has been a lecturer at the School of Computing and Intelligent Systems, the University of Ulster. His research interests include pattern recognition, machine learning, knowledge-based systems, signal processing, and condition monitoring and fault diagnosis.