Information Sciences 442–443 (2018) 220–234

Contents lists available at ScienceDirect

Information Sciences

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

Concept drift detection based on Fisher's Exact test

Danilo Rafael de Lima Cabral, Roberto Souto Maior de Barros*

Centro de Informática, Universidade Federal de Pernambuco, Cidade Universitária, Recife, 50740-560, Brazil

ARTICLE INFO

Article history:
Received 5 August 2017
Revised 19 February 2018
Accepted 20 February 2018
Available online 21 February 2018

Keywords:
Concept drift
Data streams
Drift detection
Online learning
Statistical tests

ABSTRACT

Concept drift detectors are software that usually attempt to estimate the positions of concept drifts in large data streams in order to replace the base learner after changes in the data distribution and thus improve accuracy. Statistical Test of Equal Proportions (STEPD) is a simple, efficient, and well-known method which detects concept drifts based on a hypothesis test between two proportions. However, statistically, this test is not recommended when sample sizes are small or data are sparse and/or imbalanced. This article proposes an ingeniously efficient implementation of the statistically preferred but computationally expensive Fisher's Exact test and examines three slightly different applications of this test for concept drift detection, proposing FPDD, FSDD, and FTDD. Experiments run using four artificial dataset generators, with both abrupt and gradual drift versions, as well as three real-world datasets, suggest that the new methods improve the accuracy results and the detections of STEPD and other well-known and/or recent concept drift detectors in many scenarios, with little impact on memory and run-time usage.

1. Introduction

Data streams are environments that frequently contain very large (possibly infinite) amounts of data, flowing rapidly and continuously. Thus, methods that learn from these streams are usually online and under restrictions regarding the usage of memory and run-time. Moreover, reading the same data instance more than once is normally not possible. In addition, this scenario considers that the target distribution of the data may change over time, a situation known as concept drift [23].

A very common categorization of concept drift is based on the speed of change. When the changes between concepts are sudden and/or rapid, they are called *abrupt* and, when the transitions from one concept to another occur over a number of instances, they are called *gradual* [23].

There are many examples of online learning applications which may be affected by concept drift [49], including filtering spam in e-mail messages [30], monitoring data from sensors [33], intrusion detection [32], as well as sentiment analysis [47], among others.

Different directions have been proposed to learn from data streams with concept drift. A common approach is based on concept drift detection methods [25] which are implemented as lightweight software that usually monitor the prediction results of a base classifier and focus on identifying possible changes in the data distribution.

Several researchers have proposed ensembles using a base classifier, sometimes with more sophisticated strategies, and/or adopting different weighting functions to compute the resulting classification, e.g. Dynamic Weighted Majority (DWM) [31], Diversity for Dealing with Drifts (DDD) [38], Adaptable Diversity-based Online Boosting (ADOB) [46], Boosting-

* Corresponding author.
E-mail addresses: drlc@cin.ufpe.br (D.R.d.L. Cabral), roberto@cin.ufpe.br (R.S.M.d. Barros).

https://doi.org/10.1016/j.ins.2018.02.054

like Online Learning Ensemble (BOLE) [7], and Fast Adaptive Stacking of Ensembles (FASE) [21]. Other methods concentrate on detecting concepts that recur to reuse previously trained classifiers, e.g. Recurring Concept Drifts (RCD) [24]. Moreover, some of these ensemble methods also rely on an auxiliary drift detection method [7,10,21,24,38,46].

Several concept drift detectors have been proposed over the years and the most well-known methods are Drift Detection Method (DDM) [22], Early Drift Detection Method (EDDM) [3], Adaptive Windowing (ADWIN) [8], Statistical Test of Equal Proportions (STEPD) [39], Paired Learners (PL) [2], and EWMA for Concept Drift Detection (ECDD) [43]. Of the above-mentioned methods, DDM and STEPD are among the most simple ones and, despite their simplicity, present good all-round performance [25]. Other more recent drift detection methods have also been proposed, including Sequential Drift (SeqDrift) [40], SEED Drift Detector (SEED) [28], Drift Detection Methods based on Hoeffdings Bounds (HDDM) [20], Fast Hoeffding Drift Detection Method (FHDDM) [41], Reactive Drift Detection Method (RDDM) [4,5], and Wilcoxon Rank Sum Test Drift Detector (WSTD) [4,6].

One problem of STEPD is that Nishida et al. [39] adopted a statistical test of equal proportions to detect concept drifts, even when the number of samples is small. The authors acknowledged the problem and claimed the reason for their decision not to use Fisher's Exact test [19] (when samples were small) was its high computational cost.

Besides being commonly used in the medical literature for statistical analysis [37], Fisher's Exact test was also effectively applied in data mining in order to discover dependencies between attributes [26]. Ross et al. [44], in their sequential monitoring of binomial variables, presented another application of this test for drift detection, implemented at an acceptable computational cost. Thus, an efficient implementation of Fisher's Exact test is a worthy contribution.

This work takes advantage of specific details of the intended application to provide an ingeniously efficient simple implementation of the computationally expensive Fisher's Exact test in order to propose three different applications of this test in the concept drift detection problem. More specifically, we propose Fisher Proportions Drift Detector (FPDD), which is a variation of STEPD using Fisher's Exact test when samples are small, Fisher Square Drift Detector (FSDD), which is similar to FPDD but uses the chi-square test instead of the test of equal proportions, and Fisher Test Drift Detector (FTDD), which always detects drifts using Fisher's Exact test.

In addition, using the Massive Online Analysis (MOA) framework [9], we tested the three proposed detectors against DDM, ECDD, SEED , FHDDM, and STEPD in quite a large number of scenarios, with both artificial and real-world datasets, using two different base classifiers, and also performed statistical evaluations and drift identification analysis of the results.

The rest of this article is organized as follows: Section 2 briefly surveys related work, with special attention given to STEPD ; Section 3 describes Fisher's Exact statistical test and its given implementation; Section 4 presents the three proposed detection methods and their respective abstract pseudo-codes; Section 5 details the configuration of the experiments, also including brief descriptions of the datasets used in the tests; Section 6 shows the results obtained, performs evaluations of accuracy, run-time, and memory consumption, statistically comparing accuracies, and analyses the drift identifications; and, finally, Section 7 summarizes our conclusions.

2. Related work - drift detection methods

In data stream environments, a common organization of the learning process is to use a concept drift detector together with a base learner. In general, the concept drift detection method analyses the prediction results of the base classifier and applies some decision model to attempt to detect changes in the data distribution. Methods that follow this approach include DDM [22], EDDM [3], and STEPD [39].

Different concept drift detection methods surveil the performance of the base learner using distinct strategies and/or statistics to decide when concept drifts have occurred. Also, a lower confidence level is usually set to indicate warning levels, signaling that concept drifts may take place. At these points, the detectors create a new instance of the base learner to be trained in parallel. Whenever a concept drift is confirmed, this new instance replaces the original classifier; and if the warning is found to be a false alarm, the new instance is discarded.

DDM detects concept drifts in a sequence of examples by analysing the error rate (the probability of making an incorrect prediction) and its corresponding standard deviation. On the other hand, EDDM is similar but uses the distance between two classification errors rather than the error rate.

DDM assumes the error rate decreases with more examples when the distribution is stationary. Also, when the error increases, DDM presumes that the data distribution has changed and the base classifier has become inefficient. In EDDM, the distance between two consecutive errors tends to increase and drifts are detected when it decreases.

Both methods use parametrized thresholds for the detection of *warnings* (w) and *drifts* (d). The parameters of DDM and their defaults are $w = 2.0$, $d = 3.0$, and $n = 30$, where n is the minimum number of instances before the detection of drifts is permitted. The parameters of EDDM and their defaults are $w = 0.95$, $d = 0.9$, and the minimum number of errors before drift detection is permitted, $e = 30$.

ADWIN [8] uses a variable sized sliding window, which is reduced when drifts occur and becomes larger with longer concepts. Two dynamic sub-windows store older and recent data. Drifts are detected when the difference in the averages between these sub-windows is higher than a given threshold. The parameters of ADWIN and their default values in its MOA implementation are the confidence level to reduce the window size ($\delta = 0.002$) and the minimum frequency of instances needed to reduce the window size ($f = 32$).