A comparison of techniques for Virtual Concept Drift detection

Manuel L. González[1], Javier Sedano[1], Ángel M. García-Vico[2], and José R. Villar[3]

[1] Instituto Tecnológico de Castilla y León, Burgos, SPAIN
{manuel.gonzalez,javier.sedano}@itcl.es,
WWW home page: http://www.itcl.es
[2] Andalusian Research Institute in Data Science and Computational Intelligence (DaSCI), University of Granada, Granada, SPAIN, agvico@decsai.ugr.es
[3] University of Oviedo, Oviedo, SPAIN, villarjose@uniovi.es

Abstract. Concept Drift is one of the main problems presents in data stream processing for Data Mining and Machine Learning. This study focuses on Virtual Concept Drift. A common approach includes i) the detection of the drift with a specialized algorithm, and ii) the adaptation of the model to the current scenario. This work studies how well-known pre-processing methods affect abrupt Virtual Concept Drift detection in data streams. The proposed pre-processing techniques are: i) deleting the trend and ii) transforming the data stream from time to spectral domain. Moreover, three Virtual Concept Drift detection methods are compared over three publicly available data sets. According to the results, a slight improvement in the detection of Virtual Concept Drift is achieved when the trend is deleted. In contrast, no detection of Virtual Concept Drift is reported on the spectral domain.

Keywords: Data Stream Mining, Concept Drift Detection, Pre-processing methods

1 Introduction

Nowadays, the exponential increase of IoT devices and sensors is generating a continuous information flow. This continuous flow of data is commonly known as a data stream [1]. The general characteristics of data streams imply a challenge for Data Mining and Machine Learning [2]. The main constraints imposed by data stream mining are the processing time, system memory and the adaptability of the algorithms. In this context of adaptability, algorithms have to deal with the constantly evolving nature of data. This phenomenon is known as Concept Drift (CD) [3] and leads to a decrease in model performance over time for any given task [4–6]. Conceptually, CD happens when the joint probability distribution, $p(X, y)$, for the same pair of input and output data streams, X and y, changes in time: $p_t(X, y) \neq p_{t+1}(X, y)$ [7].

Based on its source, CD can be classified into two different types [8]. *i*) Real Concept Drift (RCD), where the change over time is in the relationships between input and output data, represented by the evolution of conditional probability distributions $p_t(y|X) \neq p_{t+1}(y|X)$. *ii*) Virtual Concept Drift (VCD), where the change over time is in the distributions of the input data, $p_t(X) \neq p_{t+1}(X)$. Another dimension of CD is the type and velocity of the changes over time. Four classes of CD can be distinguished in this regard [8]: abrupt, gradual, incremental and recurring.

Two common approaches are developed to deal with CD. In one of them, the model is continuously updated by the new incoming data. Whereas in the other, the CD is firstly detected, with a specialized algorithm, and then the model has adapted consequently. In this study, is considered the second approach, mainly in CD detection.

Algorithms specialized in CD detection can be divided into two categories, depending on the CD source. *i*) RCD detection methods, which are mainly focused on the model's accuracy change [5, 6, 13, 14]. *ii*) VCD detection methods (VCD-DMs), focused on the change in input data stream statistical properties [11, 12, 17]. There is a vast amount of research papers about CD detection [2, 4, 8–10]. Depending on the CD type and velocity of change over time, some detection methods show better performance than others [2].

The majority of studies in the literature are focused on RCD detection because these algorithms directly measure the decrease in model accuracy. VCD-DMs become interesting since the real output variable y is, in many cases, unknown -thus RCD is not possible-. Furthermore, the study and detection of VCD allow measuring changes in the input data distributions over time, favouring the model to update and tune. Therefore, interest in VCD is justified.

In Data Mining methodology, it is common to apply a pre-processing method (PPR-M) to the input data stream before any processing algorithm [15]. These techniques aim to remove noise or unwanted properties from the data, to add information from other sources or to adapt the input data stream for the processing algorithms.

This study focuses on the detection of abrupt VCD on data streams. This research aims to compare some of the most known techniques for VCD detection, together with different PPR-Ms. This study tries to answer the following questions:

1. How does the VCD-DMs vary if the trend is filtered by a PPR-M?
2. How does the VCD-DMs work in the spectral domain?
3. Which are the differences among VCD-DMs when using the same PPR-M?
4. How is the performance of the VCD-DMs techniques affected by the nature of the sliding window?

The rest of the paper is organized as follow: the next section describes the PPR-Ms and VCD-DMs techniques to be studied in the comparison. The data sets and experimental setup are detailed in section 3. Section 4 depicts the obtained results and discusses the answers to the questions previously formulated. Finally, the study ends with the conclusions and future work.

2 A description of the technologies

As shown in Fig. 1, the most common approach in data stream mining to extract knowledge from drifting data is described below.

1. The arriving data is buffered creating chunks of data of a pre-defined length.
2. PPR-M is applied to the data chunk.
3. When selected, a CD detection method is performed; if CD is detected, then the model is adapted in consequence.
4. Run the adapted model to obtain some knowledge from evolving data.

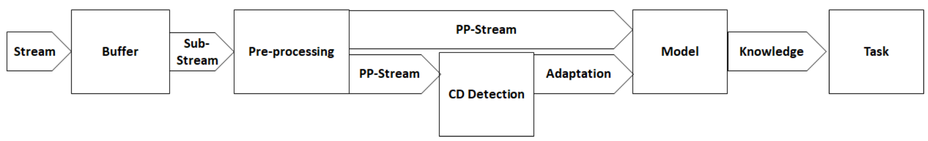

Fig. 1. Scheme of a system adapted to Concept Drift

This section outlines the PPR-Ms that are considered in this study, as well as the VCD-DMs to be compared.

2.1 Pre-processing alternatives

Let $X(t)$ be the input data stream. Up to four different standard and well-known PPR-Ms are used in this study, which are listed below.

1. **Identity (ID)**: $X(t) \longrightarrow X(t)$, in essence, it does not transform the original data stream.
2. **Calculate Local Derivative (LDV)**: $X(t) \longrightarrow (X(t) - X(t-1))/\Delta t$, where Δt is the local increment in time defined as $\Delta t = t_i - t_{i-1}$.
3. **Subtract Simple Moving Average (S-SMA)**: $X(t) \longrightarrow X(t) - SMA(X(t))$, where SMA stands for Simple Moving Average.
4. **Calculate Spectral Distribution (SDT)**: $X(t) \longrightarrow \tilde{X}(f)$, Discrete Fourier Transform (DFT) [16] will be used to transform data stream from time-domain -$X(t)$- to frequency-domain -$\tilde{X}(f)$-.

The first method has been proposed to have a comparison baseline. The second and third ones aim to remove the trend from the data stream. It is worth noticing that the latter transformation normalizes seasonal components of the data stream by changing to the spectral domain. For a comprehensive study on PPR-Ms for CD see [15].

2.2 Virtual Concept Drift detection methods

The majority of the VCD-DMs proposed in the literature are based on the comparison of the statistical properties between different portions of the data stream. These portions are called windows of data. A window of data W on input data stream $X(t)$ is considered a time-ordered subset of consecutive elements $x \in X(t)$ that acts as a buffer. These windows can have fixed or variable length and their elements may vary over time. Depending on the number of windows and their properties, different VCD-DMs have been developed [2, 9]. According to the results in [2], two-windows-based VCD-DMs show better performance for abrupt VCD detection than other alternatives. Therefore, the proposed methods in this study belong to this category. These methods are: Adaptive sliding window algorithm (ADWIN) [17], Kolmogorov-Smirnoff Test (KS-Test) [19] and Fourier Inspired Windows for Concept Drift detection (FIWCD) [21].

Let W_t be a window of data at time t on $X(t)$. Let W_t^O and W_t^R be two subwindows of data on $X(t)$. Their lengths are n^O and n^R, respectively, and W_t^R is the most recent of the two. We denote $W_t = W_t^O \cup W_t^R$ with length n, where n accomplishes Eq. 1, which means, there could be an overlap between the two subwindows. The lengths of W_t^O and W_t^R may vary according to the current VCD-DM, but Eq. 1 must be always satisfied.

$$n \leq n^O + n^R \tag{1}$$

In these conditions, it is considered that VCD has happened whenever remarkable differences are found among the descriptive statistics calculated for each subwindow [17]. The chosen methods follow two different strategies to detect the existence of a remarkable difference: to set some specific bound base on confidence level α, like ADWIN or KS-Test, or to set a similarity threshold λ like FIWCD.

It should be noted that no underlying model is needed for these detection methods, so it can be applied to any system that computes data streams. In the following sections, the proposed methods are going to be explained. It should be highlighted that the notation is slightly different from that of the original papers to unify it.

Adaptive sliding window algorithm. ADWIN method [17] compares the mean values calculated for W_t^O and W_t^R. Non-overlapping subwindows are considered, so $W_t^O \cap W_t^R = \emptyset$ holds. Their lengths are varied until the absolute difference of calculated means is higher than a given threshold ϵ_{cut} (see Eq. 2).

$$|\mu_t^O - \mu_t^R| \geq \epsilon_{cut} \tag{2}$$

The value of ϵ_{cut} is firstly calculated by computing the harmonic mean (Eq. 3) of two subwindows.

$$m = \frac{n^O \cdot n^R}{n^O + n^R} \tag{3}$$

Let us assume a confidence level $\alpha \in (0, 1)$ is given. It must be adjusted by Bonferroni-Dunn correction [18], Eq. 4, where c is the number of considered pairs of subwindows for each W_t. Then, threshold ϵ_{cut} is computed following Eq. 5, with σ_t the standard deviation of data in window W_t.

$$\alpha \leftarrow \frac{\alpha}{c} \tag{4}$$

$$\epsilon_{cut} = \sqrt{\frac{2}{m} \cdot \sigma_t^2 \cdot \ln \frac{2}{\alpha}} + \frac{2}{3m} \ln \frac{2}{\alpha}. \tag{5}$$

Whenever Eq. 2 is met, the data from W_t^O and W_t^R come from different distributions under α confidence level and VCD is detected. ADWIN method can be also applied when the lengths of the subwindows remain constant. In this case, the Bonferroni-Dunn correction, Eq. 4, does not apply.

Kolmogorov-Smirnoff Test. KS-Test is a statistical non-parametric test with no assumptions of underlying distributions. This test compares two samples of data by the supreme distance D between their cumulative distributions $F(x)$. The null hypothesis is that the distributions of two samples are the same. Applying KS-Test for VCD detection has been reported in [19], [20], and is valid for non-overlapping subwindows.

The supreme distance at time t is computed by Eq. 6. $F_t^O(x)$ and $F_t^R(x)$ are cumulative distributions at time t of W_t^O and W_t^R, respectively, m is the harmonic mean given by Eq. 3 and α is the confidence level. Similarly to ADWIN, when KS-Test uses variable length subwindows, Bonferroni-Dunn correction, Eq. 4, must be applied. Finally, VCD is detected if Eq. 7 is met.

$$D_t = \sup_x |F_t^O(x) - F_t^R(x)| \tag{6}$$

$$D_t > \sqrt{\frac{1}{2m} \cdot \ln \frac{1}{\alpha}}, \tag{7}$$

Fourier Inspired Windows for Concept Drift detection. FIWCD method [21] is based on the Bhattacharyya coefficient [22] to determinate the similarity between the distributions of W_t^O and W_t^R. These two subwindows may have an overlapping region.

Lengths of W_t^O and W_t^R are fixed and calculated by a method based on DFT. This transformation is applied to different samples of historical data, as it is specified in the original study [21]. Once n^O and n^R have been determined, μ_t^R and σ_t^R of W_t^R are computed. These statistics are compared against the next element in the data stream x_{t+1}, leading to three possible scenarios. First, if $|x_{t+1} - \mu_t^R| \leq 0.5 \cdot \sigma_t^R$, then update the mean and the standard deviation of W_t^R with x_{t+1} and without x_{t-n^R}. Second, if $0.5 \cdot \sigma_t^R \leq |x_{t+1} - \mu_t^R| \leq 2 \cdot \sigma_t^R$, then just pass to the next element without any update. Third, if $|x_{t+1} - \mu_t^R| \geq 2 \cdot \sigma_t^R$, then VCD is suspected at time $t + 1$. Finally, when VCD is suspected, let pass elements of the data stream until $W^O_{t+1} = W^R_t$ and compute Bhattacharyya coefficient (bc) by Eq. 8 between probability distributions of subwindows W^O_{t+1} and W^R_{t+1}. If similarity threshold λ is upper bc, then VCD event is detected.

$$bc = \sum_{x \in W_{t+1}} \sqrt{p^O_{t+1}(x) \cdot p^R_{t+1}(x)}. \tag{8}$$

3 Materials and methods

In this section, the performance on abrupt VCD detection of the proposed methods is analyzed in three data sets. For this purpose, three common classification metrics are compared. Below, the details of these data sets and the experimental set-up are presented.

3.1 Virtual Concept Drift Data sets

Data sets used in this research are those reported in [21]. They represent abrupt VCD in real scenarios related to financial processes that have been manually segmented and labelled by experts. These data sets are:

1. **BSE**: Bombay Stock Exchange. The closing values from 19-Feb-1999 to 18-Feb-2019 formed the data set. In this dataset, 24 dates are labelled as VCD.
2. **GOOGLE**: The data is the daily closing value of Google stock on trading days from 27-Mar-2014 to 07-Dec-2018. In this data set, 9 dates are labelled as VCD.
3. **USD-SGD**: USD to SGD exchange rate. The weekly data from 08-Jan-1988 to 15-Oct-2015 formed the data set. In this dataset, 19 dates are labelled as VCD.

3.2 Experimental set up

The experimentation is schematized as follows. Each data set is analyzed individually with each possible combination of PPR-M and VCD-DM. The same performance measurement will be calculated for each case, so the results can be compared.

Table 1. Values for free parameters that will be established in the experimental set-up.

Parameter	Values	Parameter	Values
n	$100, 150$	Δn	$1, 5, 10$
p^R	$0.5, 0.7$	α	$0.01, 0.05$
$(p_{min}, \Delta s)$	$(0.3, 10), (0.4, 5)$	λ	$0.6, 0.7, 0.75$

Four relevant parameters of the VCD-DMs implementation are analyzed:

1. The length of window W_t, n.
2. The step of variation for W_t, Δn.
3. The subwindow parameter. If fixed length is chosen, this parameter is the percentage of elements of W_t^R, p^R. Instead, if variable length is chosen, this parameter is the tuple of the minimum percentage of elements for each subwindow and the step of variation, $(p_{min}, \Delta s)$.
4. The statistical bound, according to desired tolerance of the method, α or λ.

These four parameters apply to the implementations of ADWIN and KS-Test. However, for FIWCD, only the fourth parameter λ is applied since the first, second and third ones are fixed by the method.

Table 1 shows the set of parameters that have been established for the different VCD techniques. These values have been extracted from the literature [17, 19–21]. A grid search is applied using all possible combinations of parameters, to optimize each PPR-M plus VCD-DM performance. The performance condition to select the best method is: higher Specificity meanwhile Sensibility is non-zero, if possible. The Accuracy (ACC), the Sensitivity (SENS) and the Specificity (SPEC) (see Eqs. 9, 10 and 11) have been selected as performance measures because they have been used thoroughly in the literature. TP, TN, FP and FN are the True Positive, True Negative, False Positive and False Negative counters.

$$ACC = (TP + TN)/(TP + FN + FP + TN) \tag{9}$$

$$SENS = TP/(TP + FN) \tag{10}$$

$$SPEC = TN/(TN + FP) \tag{11}$$

4 Results and discussion

Results from the experimentation are shown in Table 2. Some interesting issues arose when studying these results. A slight improvement in VCD detection is observed for ADWIN and KS-Test when the trend is deleted with LDV and S-SMA. In contrast, VCD is not detected in the spectral domain, independently of the method and data set. Moreover, a high dispersion in the performance of ADWIN has been observed, while KS-Test seems to be a robust method, with more stable and better performance when the trend is deleted. Surprisingly, FI-WCD performed better when no PPR-M was used; this behaviour is not normal and raises suspicion about the method. Besides, the reported results in [21] were not obtained even though we reproduced their experimentation. Results after a grid search also show that there are not better parameters for a given method and they have to be tuned for each specific problem.

According to the experimental results, the initial questions of this study remain unanswered due to a lack of evidence. These issues motivate future work.

Table 2. Results of the different combinations of PPR-M and VCD-DM for each data set (DS). Parameters for methods that meet performance condition are specified along its ACC, SENS and SPEC values

DS	PPR-M	VCD-DM	n	p^R	$(p_{min}, \Delta s)$	Δn	α	λ	ACC	SENS	SPEC
BSE	ID	ADWIN	100	0.5	–	10	0.05	–	0.1203	0.9167	0.1164
		KS-Test	100	0.5	–	10	0.01	–	0.1143	0.9167	0.1104
		FIWCD	98	0.5	–	1	–	0.6	**0.2942**	**0.7917**	**0.2918**
	LDV	**ADWIN**	100	0.5	–	10	0.01	–	**0.6039**	**0.2500**	**0.6057**
		KSTest	100	–	(0.4, 5)	10	0.01	–	0.2079	0.8333	0.2048
		FIWCD	98	0.5	–	1	–	0.6	0.9952	0.0000	1.0000
	S-SMA	ADWIN	100	0.5	–	10	0.01	–	0.3376	0.5417	0.3366
		KS-Test	100	0.5	–	10	0.01	–	0.1396	0.8333	0.1362
		FIWCD	98	0.5	–	1	–	0.6	**0.9369**	**0.0833**	**0.9410**
	SDT	**ADWIN**	150	0.7	–	5	0.05	–	**0.9655**	**0.0417**	**0.9699**
		KS-Test	100	0.5	–	10	0.01	–	0.9558	0.0833	0.9600
		FIWCD	98	0.5	–	1	–	0.6	0.9952	0.0000	1.0000
GOOGLE	ID	ADWIN	150	0.7	–	10	0.05	–	0.1909	0.8889	0.1855
		KS-Test	150	0.5	–	5	0.05	–	0.1824	0.8889	0.1770
		FIWCD	51	0.5	–	1	–	0.6	**0.4088**	**0.7778**	**0.4060**
	LDV	ADWIN	150	0.5	–	10	0.01	–	0.8632	0.1111	0.8689
		KSTest	100	–	(0.4, 5)	10	0.01	–	0.2652	0.7778	0.2613
		FIWCD	51	0.5	–	1	–	0.6	**0.8649**	**0.1111**	**0.8706**
	S-SMA	ADWIN	100	0.5	–	10	0.01	–	0.3243	0.7778	0.3209
		KS-Test	100	0.5	–	10	0.01	–	0.2044	0.6667	0.2009
		FIWCD	51	0.5	–	1	–	0.6	**0.7373**	**0.2222**	**0.7413**
	SDT	ADWIN	150	0.7	–	5	0.05	–	0.9924	0.0000	1.0000
		KS-Test	150	0.7	–	5	0.05	–	**0.8674**	**0.1111**	**0.8732**
		FIWCD	51	0.5	–	1	–	0.6	0.9924	0.0000	1.0000
USD_SGD	ID	**ADWIN**	100	–	(0.4, 5)	10	0.01	–	**0.7298**	**0.5263**	**0.7304**
		KS-Test	100	0.5	–	10	0.01	–	0.1059	0.7895	0.1040
		FIWCD	97	0.5	–	1	–	0.6	0.4393	0.6316	0.4388
	LDV	ADWIN	150	0.7	–	5	0.05	–	0.9973	0.0000	1.0000
		KSTest	100	–	(0.4, 5)	10	0.01	–	0.1966	0.7368	0.1959
		FIWCD	97	0.5	–	1	–	0.75	**0.9294**	**0.1053**	**0.9316**
	S-SMA	ADWIN	150	0.5	–	5	0.05	–	0.9973	0.0000	1.0000
		KS-Test	100	0.5	–	10	0.01	–	0.1282	1.0000	0.1258
		FIWCD	97	0.5	–	1	–	0.6	**0.9297**	**0.1579**	**0.9318**
	SDT	ADWIN	150	0.7	–	5	0.05	–	0.9973	0.0000	1.0000
		KS-Test	150	0.7	–	5	0.05	–	0.9973	0.0000	1.0000
		FIWCD	97	0.5	–	1	–	0.6	0.9973	0.0000	1.0000

5 Conclusions and future work

In this study, several Virtual Concept Drift methods have been compared with different pre-processing methods on three publicly available data sets. This research aimed to determine which method can be considered the best and whether a pre-processing method performs better than others. It was found that trend suppression before applying a Virtual Concept Drift detection method has the capability to improve detection performance. This is reported for different data sets and methods but, results do not show how this improvement on performance works. Furthermore, it was found that when the spectral domain was used, by applying Discrete Fourier Transform, it was not possible to obtain valid results in the Virtual Concept Drift detection.

Future work should follow two different approaches. First, study how deleting the trend affects VCD detection. Second, understand how VCD is transformed from time to the spectral domain and develop a method that can detect it in this domain. After this work, the questions raised by this study could be answered.

6 Acknowledgments

This project has been founded by ICE (Junta de Castilla y León) under project CCTT3/20/BU/0002 and by the Spanish Ministry of Science and Innovation under project MINECO-TIN2017-84804-R, PID2020-112726RB-I00 and by the Regional Government of Andalusia, program "Personal Investigador Doctor", reference DOC_00235.

References

1. Gama J., Medas P., Castillo G. and Rodrigues P.: Learning with drift detection. Advances in Artificial Intellingence (SBIA), vol.3171 pp. 286-295, Springer (2004) doi:10.1007/978-3-540-28645-5_29
2. Khamassi I., Sayed-Mouchaweh M., Hammami M. and Ghédira K.: Discussion and review on evolving data streams and concept drift adapting. Evolving Systems, vol. 9, pp. 1-23, Springer (2018) doi:10.1007/s12530-016-9168-2
3. Webb, G., Lee, L., Goethals, B. and Petitjean, F.: Analyzing concept drift and shift from sample data. Data Mining and Knowledge Discovery, vol. 32, pp. 1179–1199, Springer (2018) doi:10.1007/s10618-018-0554-1
4. Gama J.., Žliobaite I., Bifet, A., Pechenizkiy M. and Bouchachia A.: A survey on concept drift adaptation. ACM Computing Surveys, vol. 46 no.4, pp. 1–44, (2014) doi:10.1007/s10618-018-0554-1
5. Gama J. and Castillo G.: Learning with local drift detection. LNAI, vol. 4093 pp. 42-55, Springer (2006) doi:10.1007/978-3-540-28645-5_29
6. Baena-Garcia M., Del Campo-Ávila J., Fidalgo R., Bifet A., Gavaldà R. and Morales-Bueno, R.: Early drift detection method. International Workshop on Knowledge Discovery from Data Streams, pp. 77-86, (2006)
7. Gao, J., Fan W., Han J. and Yu, P.: A general framework for mining concept-drifting data streams with skewed distributions. Proceedings of the 2007 SIAM International Conference on Data Mining, pp. 3-14, (2007) doi:10.1137/1.9781611972771.1

8. Lu, J., Liu, A., Dong, F., Gu, F., Gama, J. and Zhang, G.: Learning under concept drift: a review. IEEE Transactions on Knowledge and Data Engineering, vol. 31, no. 12, pp. 2346-2363, (2019) doi:10.1109/TKDE.2018.2876857
9. De Barros, R., Garrido, S. and Santo, C.: An overview and comprehensive comparison of ensembles for concept drift. Information Fusion, vol. 52, pp. 213-244, El Sevier (2019) doi:10.1016/j.inffus.2019.03.006
10. Žliobaite I.: Learning under concept drift: an overview. ArXiv, (2010) https://arxiv.org/abs/1010.4784.
11. Sobolewski, P. and Wozniak, M.: Comparable study of statistical tests for virtual concept drift detection. Proceedings of the 8th International Conference on Computer Recognition Systems, pp. 329–337, (2013) doi:10.1007/978-3-319-00969-8_32
12. Souza, V., Parmezan, A., Chowdhury, F. and Mueen, A.: Efficient unsupervised drift detector for fast and high-dimensional data streams. Knowledge and Information Systems, Springer (2021) doi:10.1007/s10115-021-01564-6
13. Oliveira, G., Cavalcante, R., Cabral, G., Minku, L. and Oliveira, A.: Time series forecasting in the presence of concept drift: a pso-based approach. 2017 IEEE 29th International Conference on Tools with Artificial Intelligence (ICTAI), pp. 239-246, (2017) doi:10.1109/ICTAI.2017.00046
14. Baier, L., Hofmann, M., Kuhl, N., Mohr, N. and Satzger, G.: Handling concept drifts in regression problems–the error intersection approach. Proceedings of 15th International Conference on Wirtschaftsinformatik, (2020)
15. Ramírez-Gallego, S., Krawczyk, B., García, S., Woźniak, M. and Herrera, F.: A survey on data preprocessing for data stream mining: current status and future directions. Neurocomputing, vol. 239 pp. 39–57, (2017) doi:10.1016/j.neucom.2017.01.078
16. Cooley J., Lewis P. and Welch P.: The finite Fourier transform. IEEE Transactions on Audio and Electroacoustics, vol. 17, no. 2, pp. 77-85, (1969) doi:10.1109/TAU.1969.1162036
17. Bifet, A. and Gavaldà, R.: Learning from time-changing data with adaptive windowing. SIAM International Conference on Data Mining, pp. 443–448, (2007) doi:10.1137/1.9781611972771.42
18. Dunn, O.: Multiple Comparisons among Means. Journal of the American Statistical Association, vol. 56, no. 293, pp 52-64 (1961) doi:10.1080/01621459.1961.10482090
19. Wang, Z., and Wang, W.: Concept drift detection based on Kolmogorov-Smirnov test. Artificial Intelligence in China, (LNEE) vol. 572, pp. 273–280, (2020) doi:10.1007/978-981-15-0187-6_31
20. Raab, C., Heusinger, M. and Schleif, F.: Reactive soft prototype computing for concept drift streams. Neurocomputing, vol. 416, pp. 340-351, (2020) doi:10.1016/j.neucom.2019.11.111
21. Misra, S., Biswas, D., Saha, S. and Mazumdar, C.: Applying fourier inspired windows for concept drift detection in data stream. Proceedings of 2020 IEEE Calcutta Conference (CALCON), pp. 152-156, (2020) doi:10.1109/CALCON49167.2020.9106537
22. Bhattacharyya, A.: On the measure of divergence between two statistical populations defined by their probability distributions. Bull. Calcutta Math. Soc, vol. 35, pp. 99-109, (1943)