Enabling the Visualization of Distributional Shifts using Shapley Values[*]

Chiara Balestra[‡]
TU Dortmund University[†]

Bin Li [‡]
TU Dortmund University

Emmanuel Müller
TU Dortmund University

Abstract

In streaming data, distributional shifts can appear both in the univariate dimensions and in the joint distributions with the labels. However, in many real-time scenarios, labels are often either missing or delayed; Unsupervised drift detection methods are desired in those applications.

We design slidSHAPs, a novel representation method for unlabelled data streams. Commonly known in machine learning models, Shapley values offer a way to exploit correlation dependencies among random variables; We develop an unsupervised sliding Shapley value series for categorical time series representing the data stream in a newly defined latent space and track the feature correlation changes.

The slidSHAP series allows us to track how distributional shifts affect the correlations among the input variables independently from any kind of labeling. We show how abrupt distributional shifts in the input variables are transformed into smoother changes in the slidSHAP series, allowing for an intuitive visualization of the shifts when they are not observable in the original data.

1 Introduction

Time series analysis includes forecasting, anomaly detection, and concept drift identification. Concept drifts in time series refer both to distributional changes in the labels and in the input variables of the time series; Distributional shifts in the input variables gain traction in the *unsupervised* case when labels are not available, delayed and when unreliable or expensive to obtain.

Commonly used in machine learning applications, Shapley values gained increased popularity [25]. Not yet widely spread in time series, they are recently timidly appearing for anomaly detection, label prediction, and interpretability [5, 31, 26, 30]. The challenge to face in streaming data is the pairing of Shapley values as importance scores with the time-dependence. Distributional shifts have not yet been fairly explored using Shapley values; While some attempts refer to drifts in label-input variables distribution, to the best of our knowledge, none contextualize them for shifts in the input variables. Due to the chaotic structure of the data streams, changes in correlation among input variables can be hard to visualize, making the detection itself a jump of trust in the concept drift detector. Hence, we introduce th slidSHAP series, a time-dependent series representing the distributions and correlations among subsets of the streaming data's input variables. Based on Shapley Values, we obtain an unsupervised tool for visualizing and detecting distributional shifts in the original N-dimensional time series through a new N-dimensional latent space. This preliminary work is a representation method for unlabeled discrete time series; Although we now focus on time series whose dimension assumes a finite number of values, future work generalizes the approach to a broader application.

[*] This research was supported by the research training group *Dataninja* funded by the German federal state of North Rhine-Westphalia and by the Research Center Trustworthy Data Science and Security, an institution of the University Alliance Ruhr.

[†] Corresponding author `chiara.balestra@cs.uni-dortmund.de`

[‡] These authors contributed equally to this work

Workshop on Distribution Shifts, 36th Conference on Neural Information Processing Systems (NeurIPS 2022).

2 Methods

A cooperative game is a pair (P, ν) where $P = \{X_1, \dots, X_N\}$ is the set of players and ν a value function, i.e., a set function $\nu : \mathcal{P}(P) \to \mathbb{R}_+$. Shapley values SV [27] are computed for each player $X_i \in P$ as

$$\phi(X_i) = \sum_{A \subseteq P \setminus \{X_i\}} k_A \cdot (\nu(A \cup \{X_i\}) - \nu(A)) \tag{1}$$

where k_A depends on N and the size of A. SVs have been used as a mean for achieving interpretability of black-box models [17, 28] but, more generally, they represent a way of distributing resources among players in a game. Balestra et al. [4] proposed SVs within an unsupervised feature selection method; the authors used them to encode the structure and the correlations within subsets of features of an unlabelled tabular data set with categorical entries. Given a set of N discrete random variables $F = \{X_1, \dots, X_N\}$, the authors propose to interpret F as a set of players; In order to encode the data structure and the correlations within subsets of F, they argued in favor of using as value function a correlation metric. They proposed for their categorical context the total correlation, i.e.,

$$\nu(A) = H(A) - \sum_{X \in A} H(X) \tag{2}$$

where $H(\cdot)$ is the discrete Shannon entropy [4]. The encoding of the correlations using the total correlation and the use of Shapley values enables to extract information from the data set based on the correlations' structure; the result is that features obtaining high Shapley values are highly correlated with subsets of the other variables while features with lower Shapley values are uncorrelated with the other variables.

2.1 Time series and sliding windows

Let $X = (X_1, \dots, X_N)$ be a multivariate N-dimensional discrete time series, X_i the i-th univariate dimension of the time series. We indicate with t_1 the first timestamp on which the time series is defined. For each timestamp $t_k > t_1$, $X(t_k)$ is a N-dimensional vector of discrete values, i.e., $X_i(t_k) \in D_i$ and the cardinality of D_i is finite.

We define *overlapping sliding windows* as a series of time windows $\{w_s\}_{s \in \mathbb{N}}$ through a window length d and overlap a among adjacent windows, i.e.,

$$w_s = \{t_{s(d-a)}, \dots, t_{s(d-a)+d-1}\}. \tag{3}$$

Each window w_s contains d timestamps, and a is the number of timestamps lying in the overlap among adjacent windows, i.e., $|w_s \cap w_{s+1}| = a$ for each $s \in \mathbb{N}$. After fixing a and d, at the current timestamp t_T we have created $M(T) = \left\lfloor \frac{T-d+1}{d-a} \right\rfloor$ time windows; hence, for a fixed windows' length d, the larger is the overlap a, the higher the number of windows created and for fixed overlap a, the number of windows decreases for increasing d. Note that $a, d \in \mathbb{N}_+$.

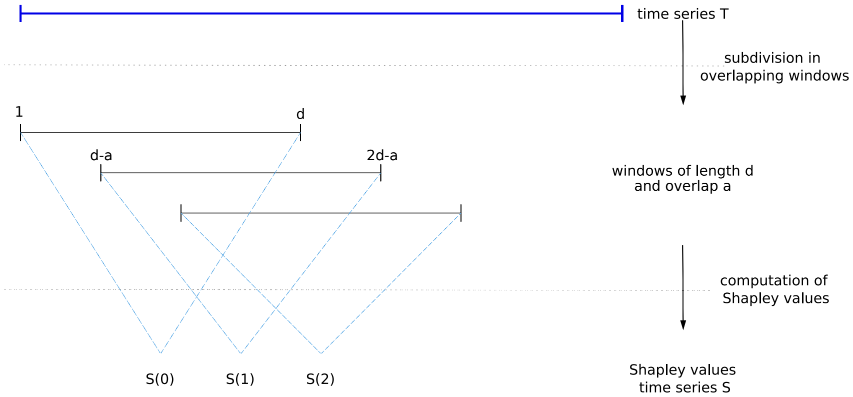

time series T
subdivision in overlapping windows
1
d
d-a
2d-a
windows of length d and overlap a
computation of Shapley values
S(0)
S(1)
S(2)
Shapley values time series S

Figure 1: Construction of the slidSHAPs.

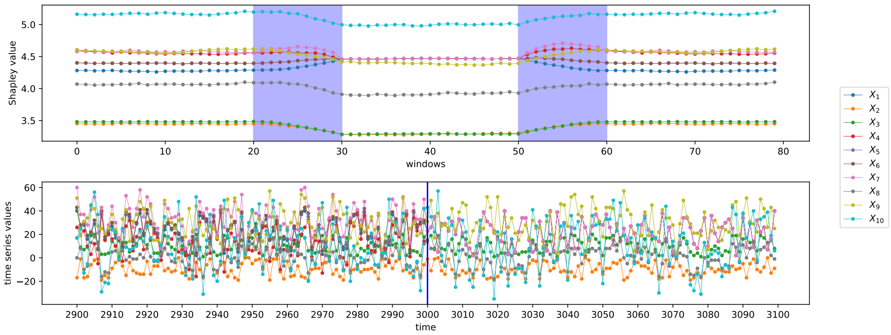


Figure 2: COR3: In the upper plot, the slidSHAP series; in the lower plot, the original time series in the interval $\{2900, \dots, 3100\}$ where the distributional shifts happen.

2.2 slidSHAPs

Given a multivariate time series X with N-dimensions we can interpret the value the i-th dimension at timestamp t_k $X_i(t_k)$ as the realization of a discrete (categorical) random variable X_i; hence, given the set of timestamps $\{t_1, \dots, t_T\}$, $\{X_i(t_1), \dots, X_i(t_T)\}$ are T independent realizations of the random variable X_i. Similarly, we can interpret $\{X(t_1), \dots, X(t_T)\}$ as the set of realizations of a N-dimensional discrete random variable. This interpretation allows us to study the correlations among the univariate dimensions of X.

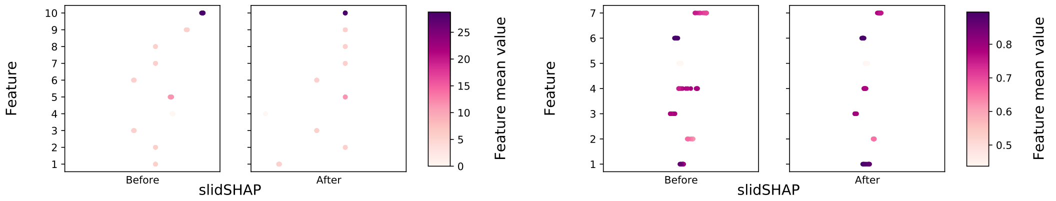
We interpret the realizations of a time series on a time window w_s as a discrete tabular data set with N columns and d rows; this allows us to compute a Shapley value for each column, i.e., for each univariate dimension of the time series using [4]. Our goal remains the one of visualizing the correlations' structure among dimensions of streaming data; thus, we want to keep a trace of the distributional changes over time. We use the sliding time windows $\{w_s\}_{s \in \mathbb{N}}$ defined in Section 2.1 and compute Shapley values of the univariate dimensions of X when restricted to the w_ss; Hence, we compute the Shapley values for each univariate dimension X_i of the streaming data in the time window w_s, i.e., $S_i(s) = \phi(X^i_{w_s})$. For each time window w_s, we obtain a vector of Shapley values $S(s) = [S_1(s), \dots, S_N(s)] \in \mathbb{R}^N$ and each $S_i(s)$ consider the correlations of X_i with the other dimensions of the time series in w_s.

As described in Section 2.1, the computation of the Shapley values inherits from the w_s the same time-dependency. Figure 1 represents a visual schema for the slidSHAPs series construction process. From the original discrete time series X assuming finite values, we extrapolate information about the univariate dimensions' correlations and transfer the structure of the data stream to a new N-dimensional real-valued series, i.e., each slidSHAP value is a N-dimensional real-valued vector. We interpret the space on which the slidSHAP values are defined as a latent space where we have projected the correlation structure of the original time series. As the sliding windows are partly overlapping, given two close-by indices s_1, s_2, the information conveyed by $S(s_1)$ and $S(s_2)$ relate to partly overlapping time windows of the original time series X; Hence, the set up of the parameters a and d for the windows' creation is essential to set up the *granularity* for the Shapley values' computations.

Finally, we underline that the slidSHAP series is not dependent on the same time notion of the original time series; when we write $S(s)$, s represents the index of the time window on which the Shapley values have been computed, i.e., $S(s)$ is the vector of Shapley values in the time window $w_s = \{t_{s(d-a)}, \dots, t_{s(d-a)+d-1}\}$ while $X(t_k)$ is the value of the time series at the time stamp t_k and it is a N-dimensional discrete-valued vector.

2.3 Distributional shifts in the univariate dimensions

When dealing with real-world time series, often only a few of the input variables are subject to distributional changes; on the other hand, those changes could affect the correlation structure of the whole set of input variables. We use slidSHAPs as an unsupervised tool for unlabelled time series obtained *sliding* over time windows of fixed amplitude to visualize the correlation structure in the time series. Targeting to detect distributional shifts of the input variables in an unsupervised manner,



(a) COR20

(b) LED

Figure 3: Feature importance before and after the first concept drift in each dataset; The color represents the average value of each sliding window corresponding to the slidSHAP value.

we employ the non-parametric and distribution-free two-sample Kolmogorov-Smirnov Test (K-S Test) and the classical t-test. We slide windows on the slidSHAP series and perform the statistical testing on each of the univariate variables following the approach proposed in [23].

3 Experiments

Datasets: We use synthetic datasets and a common benchmarking dataset to show the performances of slidSHAPs. We first create two 10-dimensional synthetic datasets; in COR3, the correlations among the input variable change every 3000 data instances, while in COR20 the changes happen every 5000 instances. Furthermore, we employ the standard benchmark dataset LED where distributional shifts happen every 9000 instances. The concepts underlying the input variables' distributions in all datasets change abruptly at specific time stamps. More details on the dataset construction and their characteristics can be found in Appendix C.

Results: We used the datasets to show our representation of the time series correlations on the latent space. Figure 2 shows how the slidSHAP series are visually more appealing than the original data streams in visualizing distributional shifts. In COR3 the distributional shifts happen at the time stamps 3000 and 6000. Using slidSHAPs ($a = 900$, $d = 1000$), we see that the abrupt changes are smoothed out in the upper plot; in the lower plot, the original time series data, where, although containing a significant distributional shift, it is not possible to observe any. Figure 5 (in the Appendix) shows the slidSHAP series for the other two datasets; we plotted dashed lines, where the shifts happen, and where they have been detected using statistical testing.
We checked for changes in the slidSHAP series and plotted them against the changes in the range of values in which the dimensions of the time series vary. The values of the sliSHAPs in the time windows are averaged and color-coded in the plots. Intuitively, a distributional drift in the input space causes a change in the slidSHAP values, which can be detected as concept drift (e.g., X_4 and X_7 in LED). Figure 3 shows this comparison in the LED and COR20 datasets before and after the first distributional shift. Furthermore, the slidSHAP series also clearly show the shifts of some non-observable input space drifts, where the amplitude of features stays in the same range while the feature correlation changes. Significant changes can still be observed in the slidSHAP values (e.g., X_{10} in COR20).

4 Discussion and conclusions

In real-time applications, streaming data appear with or without labels. In this second case, distributional shifts can appear in one or more of the input dimensions. We propose the slidSHAP series, a new representation of the time-dependent correlations among input variables of unlabelled time series; slidSHAPs allow for visualizing distributional changes in the input features as well as keeping track of the correlations changes among the input dimensions. We base our approach on a feature correlation-based value function, hence being completely unsupervised. The visualization of the slidSHAP series provides additional understanding of the often non-observable correlational shifts in the input variables. The experimental results show our approach's effectiveness in various synthetic data sets. Future work will include the study slidSHAPs in real-world scenarios.

References

[1] Detecting Concept Drift In Medical Triage | Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval.

[2] Zahra Ahmadi and Stefan Kramer. Modeling recurring concepts in data streams: a graph-based framework. *Knowledge and Information Systems*, 55(1):15–44, 2018.

[3] Liat Antwarg, Ronnie Mindlin Miller, Bracha Shapira, and Lior Rokach. Explaining anomalies detected by autoencoders using Shapley Additive Explanations. *Expert Systems with Applications*, 2021.

[4] Chiara Balestra, Florian Huber, Andreas Mayr, and Emmanuel Müller. Unsupervised Features Ranking via Coalitional Game Theory for Categorical Data. In *Big Data Analytics and Knowledge Discovery: 24th International Conference, DaWaK 2022, Vienna, Austria, August 22–24, 2022, Proceedings*, pages 97–111, Berlin, Heidelberg, August 2022. Springer-Verlag.

[5] João Bento, Pedro Saleiro, André F. Cruz, Mário A. T. Figueiredo, and Pedro Bizarro. TimeSHAP: Explaining Recurrent Models through Sequence Perturbations. In *Proceedings of the 27th ACM SIGKDD Conference on Knowledge Discovery & Data Mining*, 2021.

[6] Ayne A. Beyene, Tewelle Welemariam, Marie Persson, and Niklas Lavesson. Improved concept drift handling in surgery prediction and other applications. *Knowl Inf Syst*, 44(1):177–196, July 2015.

[7] Albert Bifet and Ricard Gavalda. Learning from time-changing data with adaptive windowing. In *Proceedings of the 2007 SIAM international conference on data mining*, pages 443–448. SIAM, 2007.

[8] Rodolfo C Cavalcante, Leandro L Minku, and Adriano LI Oliveira. Fedd: Feature extraction for explicit concept drift detection in time series. In *2016 International Joint Conference on Neural Networks (IJCNN)*, pages 740–747. IEEE, 2016.

[9] Shay Cohen, Gideon Dror, and Eytan Ruppin. Feature Selection via Coalitional Game Theory. *Neural Computing*, 2007.

[10] Fausto G da Costa, Felipe SLG Duarte, Rosane MM Vallim, and Rodrigo F de Mello. Multidimensional surrogate stability to detect data stream concept drift. *Expert Systems with Applications*, 87:15–29, 2017.

[11] Tamraparni Dasu, Shankar Krishnan, Suresh Venkatasubramanian, and Ke Yi. An information-theoretic approach to detecting changes in multi-dimensional data streams. In *In Proc. Symp. on the Interface of Statistics, Computing Science, and Applications*. Citeseer, 2006.

[12] Gregory Ditzler and Robi Polikar. Hellinger distance based drift detection for nonstationary environments. In *2011 IEEE symposium on computational intelligence in dynamic and uncertain environments (CIDUE)*, pages 41–48. IEEE, 2011.

[13] Christopher Duckworth, Francis P. Chmiel, Dan K. Burns, Zlatko D. Zlatev, Neil M. White, Thomas W. V. Daniels, Michael Kiuber, and Michael J. Boniface. Using explainable machine learning to characterise data drift and detect emergent health risks for emergency department admissions during COVID-19. *Sci Rep*, 11(1):23017, November 2021. Number: 1 Publisher: Nature Publishing Group.

[14] Björn Friedrich, Taishi Sawabe, and Andreas Hein. Unsupervised statistical concept drift detection for behaviour abnormality detection. *Appl Intell*, May 2022.

[15] Joao Gama, Pedro Medas, Gladys Castillo, and Pedro Rodrigues. Learning with drift detection. In *Brazilian symposium on artificial intelligence*, pages 286–295. Springer, 2004.

[16] Ben Halstead, Yun Sing Koh, Patricia Riddle, Mykola Pechenizkiy, Albert Bifet, and Russel Pears. Fingerprinting concepts in data streams with supervised and unsupervised meta-information. In *2021 IEEE 37th International Conference on Data Engineering (ICDE)*, pages 1056–1067. IEEE, 2021.

[17] Scott M. Lundberg and Su-In Lee. A Unified Approach to Interpreting Model Predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, 2017.

[18] Stefano Moretti, Vito Fragnelli, Fioravante Patrone, and Stefano Bonassi. Using coalitional games on biological networks to measure centrality and power of genes. *Bioinformatics*, 2010.

[19] Quoc Phong Nguyen, Kar Wai Lim, Dinil Mon Divakaran, Kian Hsiang Low, and Mun Choon Chan. GEE: A Gradient-based Explainable Variational Autoencoder for Network Anomaly Detection. In *2019 IEEE Conference on Communications and Network Security (CNS)*, 2019.

[20] Karlson Pfannschmidt, Eyke Hüllermeier, Susanne Held, and Reto Neiger. Evaluating Tests in Medical Diagnosis: Combining Machine Learning with Game-Theoretical Concepts. In Joao Paulo Carvalho, Marie-Jeanne Lesot, Uzay Kaymak, Susana Vieira, Bernadette Bouchon-Meunier, and Ronald R. Yager, editors, *Information Processing and Management of Uncertainty in Knowledge-Based Systems*, volume 610, pages 450–461. Springer International Publishing, Cham, 2016. Series Title: Communications in Computer and Information Science.

[21] Karlson Pfannschmidt, Eyke Hüllermeier, Susanne Held, and Reto Neiger. Evaluating Tests in Medical Diagnosis: Combining Machine Learning with Game-Theoretical Concepts. In *Information Processing and Management of Uncertainty in Knowledge-Based Systems*. 2016.

[22] Abdulhakim A Qahtan, Basma Alharbi, Suojin Wang, and Xiangliang Zhang. A pca-based change detection framework for multidimensional data streams: Change detection in multidimensional data streams. In *Proceedings of the 21th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 935–944, 2015.

[23] Stephan Rabanser, Stephan Günnemann, and Zachary Lipton. Failing loudly: An empirical study of methods for detecting dataset shift. *Advances in Neural Information Processing Systems*, 32, 2019.

[24] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. " why should i trust you?" explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.

[25] Benedek Rozemberczki, Lauren Watson, Péter Bayer, Hao-Tsung Yang, Olivér Kiss, Sebastian Nilsson, and Rik Sarkar. The Shapley Value in Machine Learning. *arXiv:2202.05594 [cs]*, 2022.

[26] Rohit Saluja, Avleen Kaur Malhi, Samanta Knapic, Kary Främling, and Cicek Cavdar. Towards a Rigorous Evaluation of Explainability for Multivariate Time Series. Technical report, arXiv, 2021.

[27] Lloyd S Shapley. A value for n-person games. *Contributions to the Theory of Games*, 1953.

[28] Erik Strumbelj and Igor Kononenko. An Efficient Explanation of Individual Classifications Using Game Theory. *J Mach Learn Res*, 2010.

[29] Min Sun, Stefano Moretti, Kelley Paskov, Nate Stockham, Maya Varma, Brianna Chrisman, Peter Washington, Jae-Yoon Jung, and Dennis Wall. Game theoretic centrality: a novel approach to prioritize disease candidate genes by combining biological networks with the Shapley value. *BMC Bioinformatics*, 2020.

[30] Naoya Takeishi. Shapley Values of Reconstruction Errors of PCA for Explaining Anomaly Detection. In *2019 International Conference on Data Mining Workshops (ICDMW)*, 2019.

[31] Shuhan Yuan and Xintao Wu. Trustworthy Anomaly Detection: A Survey. Technical report, arXiv, 2022.

[32] Di Zhao and Yun Sing Koh. Feature drift detection in evolving data streams. In *International Conference on Database and Expert Systems Applications*, pages 335–349. Springer, 2020.

[33] Shihao Zheng, Simon B van der Zon, Mykola Pechenizkiy, Cassio P de Campos, Werner van Ipenburg, Hennie de Harder, and Rabobank Nederland. Labelless concept drift detection and explanation. In *NeurIPS 2019 Workshop on Robust AI in Financial Services: Data, Fairness, Explainability, Trustworthiness, and Privacy*, 2019.

.

A Related work

Several approaches are proposed in the state-of-the-art literature to deal with concept drifts in time series. Many approaches are based on prediction error rate, where the distributional shift appears in $\mathbb{P}(y|X)$ instead of in the input variables $\mathbb{P}(X)$ [16, 7, 15]. However, the increasing number of scenarios where time series data are collected and no labels are provided increased fast; the necessity of dealing with distributional shifts on input variables forced the development of unsupervised concept drift detection methods. Among the several representation learning approaches used for detecting drift, we recall the most common. Bifet et al. [7] use the mean values of defined adaptive windows to represent the univariate dimensions of the time series, while Cavalcante et al. [8] employ a few linear and non-linear features to represent the whole time series. Da Costa et al. [10] propose to apply multidimensional Fourier transformation to get information about the frequency domain. More recently, a combination of multiple statistical features of the time series data has been proposed as meta-information vectors [16]. Other approaches measure the distributional discrepancy between data from different periods: It is the case in HDDDM [12] that measures the Hellinger distance between two distributions and Dasu et al. [11] that partition data via constructing a Kdq-Tree and generalize the Kulldorff's spatial scan statistic, allowing to identify regions in the Kdq-Tree with the most considerable changes quantitatively. Unfortunately, all the methods above do not explicitly monitor the correlation changes between features and may miss some drifts appearing in this domain. In the state-of-the-art literature, feature correlation in time series is mainly studied using covariance. Attempts to extend the limitation of using only covariance appeared in [2], where both mean, and covariance is used to represent the concepts in multivariate data streams. Qahtan et al. [22] detect concept drifts using the covariance matrix, tracking covariance changes in a transformed artificial low-dimensional space obtained applying PCA on the original time series data; thus, their approach does not explicitly reflect the correlation change in the original input space. A few works are worth to be mentioned where Shapley values have been applied for drift detection: Zheng et al. [33] show that the traditional Shapley value in a classification context can also be used for drift detection. Zhao et al. [32] employ Wasserstein distance and Energy distance to detect feature drifts without label and apply Shapley value and LIME [24] as post-hoc interpretation for the detected drifts. Moreover, supervised and unsupervised distributional shift detection are of great importance in many critical contexts, for example, in time series for health applications. Applications vary among surgery prediction, medical triage, heart diseases prediction [1, 6, 13]. Also, in the medical context, unsupervised distributional shifts are getting more and more importance [14].

Introduced by Shapley in 1953 [27], Coalitional Game Theory became popular in machine learning in the early 2000s. Cohen et al. [9] introduced Shapley values as a fair evaluation for features' contributions in order to achieve supervised feature selection. Later, Shapley values were applied to interpret black-box models by Lundberg et al. [17]. The success of their use in machine learning, brought to further extensions and applications both in the machine learning community [20, 21] and in bioinformatics [29, 18]. In recent years, Shapley values also started being applied to time series data. TimeSHAP [5] being one of the first extensions of Shapley values to time series is based on KernelSHAP [17]; The authors propose a method to compute Shapley values to get event- and feature-level explanations and provide insights on reducing Shapley computation's computational complexity. Saluja et al. [26] propose an application for prediction and forecasting of the income of a consulting company; Antwarg et al.[3] introduce two different methods. The first approach looks for anomalies through the reconstruction error and the autoencoder; the method is based on comparing the Shapley values of the original features and the reconstructed ones. Taikeish [30] studies the difference among Shapley values of single instances before and after a change in one feature that makes the instance itself anomalous. Nguyen et al. [19] explain the anomalies by analyzing the gradients to identify the main features affecting the anomalies.

B Experiments

Evaluation metric For each synthetic dataset, the slidSHAP series are computed with various combinations of sliding window lengths and overlaps, i.e., the parameters a and d. In order to evaluate the concept drift detection on the newly defined slidSHAP series, we need to include labels from the ground truth from the original time series; since we built the time series with drifts happening at specific timestamps, we need to transfer the induced labeling to the new slidSHAP series.

B.1 Distributional shifts detection

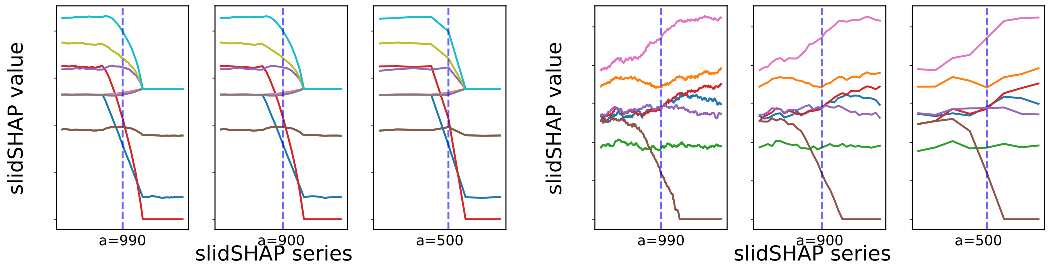
We apply a sliding window with window length $d = 1000$ over each time series dataset, and we examine the detection performance under different sliding window ovrerlap sizes $a \in \{990, 900, 500\}$. We conduct the K-S tests at significance level $\alpha = 0.05$ with buffers of length 5. As shown in Table 1, higher a leads to more false positive detection and slightly better accuracy. We are able to figure out that the model becomes more sensitive to drifts due to the increase of slidSHAP curve smoothness.

Table 1: Drift detection performance ($\alpha = 0.05$)

	$a = 990$				$a = 900$				$a = 500$			
	TP	FP	FN	ACC	TP	FP	FN	ACC	TP	FP	FN	ACC
COR3	0	6	2	0.990	1	2	1	0.958	1	1	1	0.714
COR20	7	115	12	0.987	19	43	0	0.956	17	1	2	0.984
LED	0	22	9	0.997	3	5	6	0.988	2	2	7	0.947

The step size is considered a factor of granularity in the slidSHAPs. For each concept drift, a smaller d leads to more sliding windows, thus more slidSHAP instances containing the same drift event so that the trend in slidSHAPs becomes smoother. As in Figure 4, we plot the nearby slidSHAP values of the first distributional shift in each dataset. As the overlap size a decreases, the shift becomes more abrupt in the slidSHAP space.

Note that even if only a few dimensions are affected by distributional shifts, all slidSHAPS are affected. This happens as Shapley values evaluate how the dimensions are correlated; when some dimension X_i shifts to another distribution, all the other input dimensions which were correlated with X_i or that are correlated with X_i after the shift has happened are affected (cf. Appendix C).



(a) COR20

(b) LED

Figure 4: slidSHAP evolution of the first concept drift in each dataset: Each solid line denotes the slidSHAP values on a feature dimension. The dashed blue line depicts the first concept drift position.

B.2 sildSHAP visualization

In Figure 5, we visualize complete slidSHAPS series.

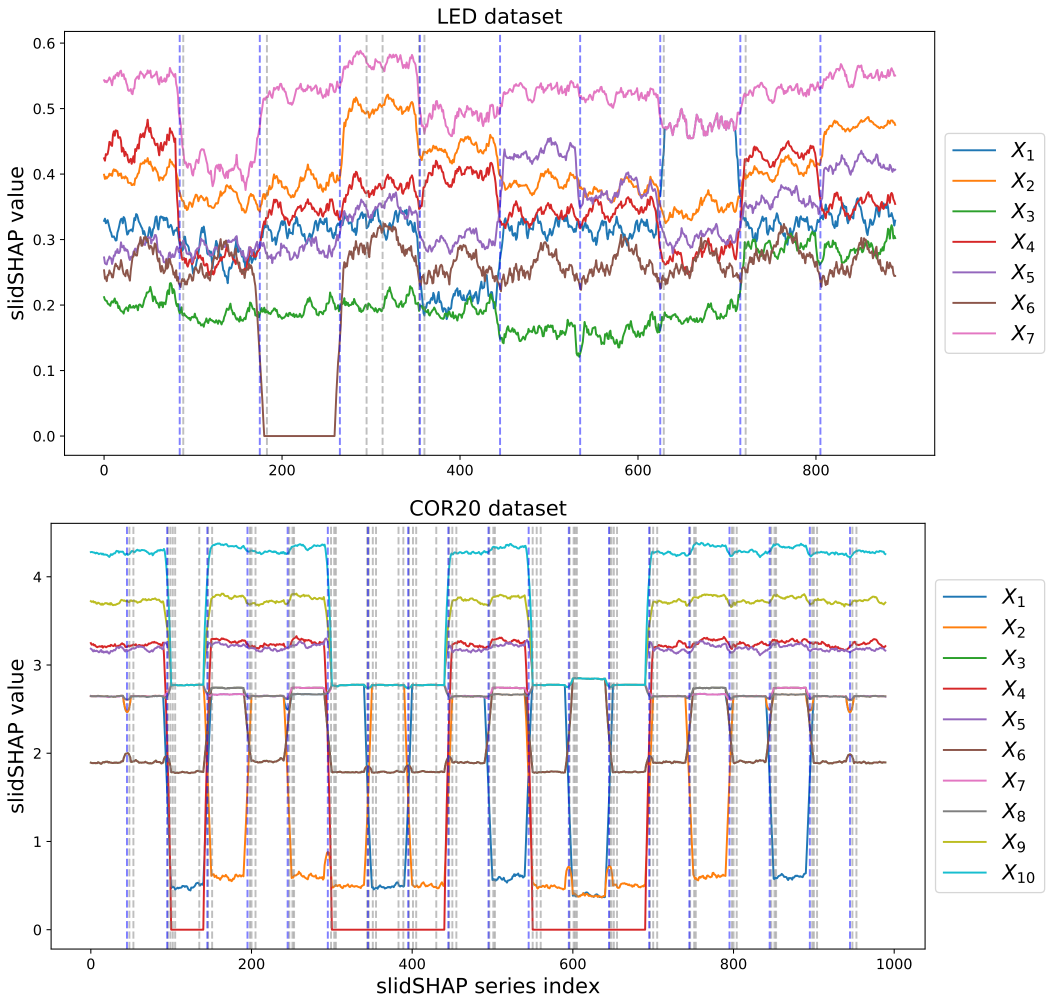


Figure 5: slidSHAPs of the LED dataset and the COR20 dataset both using windows' length 1000 and overlap = 100. The blue dashed lines indicate the true position of the distributional drifts, while the grey dashed lines are where they have been detected using the t-test on the slidSHAPs.

C Dataset details

COR3 contains 9000 instances and 10 dimensions; two distributional shifts happen respectively at timestamp 3000 and 6000, involving only 3 of the dimensions of the time series data. In COR3, the first 3 dimensions are randomly and individually sampled integers, $X_1 \in [1, 30]$, $X_2 \in [-20, -10]$, $X_3 \in [1, 10]$. Moreover, we define an integer random noise $\epsilon \in [1, 3]$. The remaining dimensions are correlated with the first dimensions,

- $X_4 = X_1 - X_3 + \epsilon$
- $X_5 = X_1 + \epsilon$
- $X_6 = X_1 + \epsilon$
- $X_7 = X_1 + X_3 + \epsilon$
- $X_8 = X_2 + X_3$
- $X_9 = 2 \times X_3 - X_2$
- $X_{10} = 2 \times X_2 + 3 \times X_3$

3000 data instances are generated in this way in the first 3000 timestamps. To simulate a distrubutional shift, we change the following four dimensions to,

- $X_4 = X_1$
- $X_5 = X_1$
- $X_6 = X_1$
- $X_7 = X_1$

The new concept also lasts 3000 timestamps. Finally, the first concept reappears again as the third distributional shift. The three concepts are concatenated directly at the timestamps 3000 and 6000, respectively.

COR20 contains 20 different 10-dimensional concept, each concept lasts 10000 timestamps. The first 3 dimensions are also randomly and individually sampled integers, $d1, X_2, X_3 \in [1, 10]$. To generate each concept, we randomly select $X_a, X_b \in \{X_1, X_2, X_3\}$, and define the remaining dimensions as,

- $X_4 = X_a - X_b$
- $X_5 = X_a + X_b$
- $X_6 = X_3$
- $X_7 = X_a$
- $X_8 = X_b$
- $X_9 = 2 \times X_a - X_b$
- $X_{10} = 3 \times X_a + 2 \times X_b$

All concepts are concatenated directly as abrupt concept drift. To be noticed, if $X_a = X_b$, X_4 becomes constant. We enforce that adjacent concepts cannot have the same X_a and X_b. The first concept reappears at every $5th$ concept.