XTSC-Bench: Quantitative Benchmarking for Explainers on Time Series Classification

Jacqueline Höllig*, Steffen Thoma*, Florian Grimm†
* FZI Research Center for Information Technology, Karlsruhe, Germany, {hoellig, thoma}@fzi.de
† ESB Business School, Reutlingen, Germany, Florian.Grimm@Reutlingen-University.de

2023 International Conference on Machine Learning and Applications (ICMLA) | 979-8-3503-4534-6/23/$31.00 ©2023 IEEE | DOI: 10.1109/ICMLA58977.2023.00168

Abstract—**Despite the growing body of work on explainable machine learning in time series classification (TSC), it remains unclear how to evaluate different explainability methods. Resorting to qualitative assessment and user studies to evaluate explainers for TSC is difficult since humans have difficulties understanding the underlying information contained in time series data. Therefore, a systematic review and quantitative comparison of explanation methods to confirm their correctness becomes crucial. While steps to standardized evaluations were taken for tabular, image, and textual data, benchmarking explainability methods on time series is challenging due to a) traditional metrics not being directly applicable, b) implementation and adaption of traditional metrics for time series in the literature vary, and c) varying baseline implementations. This paper proposes XTSC-Bench, a benchmarking tool providing standardized datasets, models, and metrics for evaluating explanation methods on TSC. We analyze 3 perturbation-, 6 gradient- and 2 example-based explanation methods to TSC showing that improvements in the explainers' robustness and reliability are necessary, especially for multivariate data.**

Index Terms—**Explainable AI, Time Series Classification, XAI Metrics**

I. Introduction

As the use of machine learning models, especially deep learning, increases in various domains ranging from health care [1] to predictive maintenance [2], the need for reliable model explanations is also growing. An increasing number of methods providing a variety of explanation types (e.g., example-based methods like counterfactuals [3], or feature attribution methods like SHAP [4]) on different data types (e.g., images [5], tabular data [6]) are available. However, measuring the performance of such explanation methods is still a challenge. There are no generally agreed upon-metrics measuring the quality of explanations, and comparisons between different implementations and metrics are difficult (e.g., [7]–[9]). The first steps to standardize the metrics notion and implementation have been taken by different frameworks implementing explainability algorithms (e.g., Captum [10], AIX360 [11]) and Quantus [12], a framework dedicated to the evaluation of explanations. The main focus of those frameworks is to provide explainability to image, tabular, and textual classification tasks. Although time series classification (TSC) is a ubiquitous task, it has been neglected. Due to the different structure and properties of time-ordered data, the application of non-time-specific explanation algorithms is not advisable, leading to a new subfield in Explainable Artificial Intelligence (XAI) - Explainable Time Series Classification (XTSC) [13].

While the first step to standardize the explanation benchmarking process for the time series domain has been taken by TSInterpret [14] - a framework implementing explanation methods for time series classification in a unified interface - standardized metrics for evaluating the quality of explanation methods are still missing [15]. Similar to the explanation methods implemented in the different explainability frameworks, transferring metrics to the time series domain is complex. Using metrics from traditional frameworks (e.g., [12]) can lead to erroneous assumptions in the time series domain. This lack of specific and standardized metric and baseline implementations lead to a high variety of proposed metrics, metric implementations, proposed baselines, and baseline implementations.

In this paper, we propose XTSC-Bench, a benchmarking tool implementing a variety of metrics for a standardized and systematic evaluation of explainers for TSC. Its connection to TSInterpret [14] ensures a unified implementation of benchmarking algorithms. We utilize XTSC-Bench to evaluate 3 gradient- and 6 perturbation-based feature importance methods and 2 example-based approaches. Our contribution is twofold:

- A thorough investigation of existing approaches.
- An easy-to-use benchmarking tool compatible with TSInterpret [14].

II. Related Work

Several researchers stress the need for formal evaluation metrics and a more systematic evaluation of explainability methods [29] [15]. For image, tabular, and textual data, a standard is slowly emerging [38] [39] with easy-to-adopt frameworks and the inclusion of some general metrics into explanation frameworks (e.g., [11], [10]) as well as a framework dedicated to quantization [12]. Nonetheless, due to the relative newness of explainability to Deep Learning for TSC[1], standardization for benchmarking explainability algorithms on time series is still missing. Table I shows the evaluation settings of various explanation algorithms for TSC. While the data basis is mostly standardized, i.e., most algorithms use a subset of data included in the UCR [17] or UEA

This work was carried out with the support of the German Federal Ministry of Education and Research (BMBF) within the project "MetaLearn" (Grant 02P20A013).

[1] First XAI approaches on time series were only emerging in the last decade. For a detailed survey, we refer the reader to [13].

1946-0759/23/$31.00 ©2023 IEEE
DOI 10.1109/ICMLA58977.2023.00168

TABLE I: Evaluation settings of explainers for TSC. Bold are the explanation algorithms evaluated in Section V. If no metric source is provided, the paper authors did not specify which notation was used. The * indicates that only a subset of the dataset was used.

Explainer	Dataset	Metrics / Target	Baselines	TS-Baselines
LEFTIST [16]	UCR Archive [17]*	Faithfulness, Understandable [16]	-	-
NG [18]	UCR Archive [17]*	Proximity, Sparsity, Plausibility, Diversity [19]	W-CF [3]	NUN-CF [18]
TSEvo [20]	UCR Archive [17]*	Proximity, Sparsity, Plausibility [19]	W-CF [3]	NG [18], COMTE [21]
TSR [22]	Synthetic Data [22]	Reliability [22]	GradCam [23], Integrated Gradients [24], Feature Occlusion [25] & more	-
COMTE [21]	Hpas, Taxonomist, Cori, NATOPS	Complexity [26], Faithfulness [6], Robustness [27]	SHAP [4], Lime [6]	-
LASTS [28]	UCR/UEA Archive	Faithfullness [29], [30], Robustness	SHAP [4]	SHAPS [31]
SETS [32]	Solar Flare Dataset	Proximity, Sparsity, Plausibility [19]		NG [18], COMTE [21],
TSInsight [33]	UEA Dataset [34]*	Faithfullness [35]	GradCam [23], Gradient x Input [36], Feature Occlusion [25] & more	-
TSViz [37]		Complexity, Faithfullness, Robustness [37]	-	-

Archive [34], all of the algorithms included in Table I rely on comparing the newly developed algorithm with a time series unspecific algorithm. However, it has been shown that time-series unspecific explanation algorithms are not able to capture the time component sufficiently as they rely heavily on independent feature assumption and cannot uncouple the feature and time domain [22]. Although most metrics used in these evaluations have the same evaluation target, i.e., faithfulness, robustness, and reliability[2], their definitions and implementations differ. Often, metrics are directly transferred from image classification [33]. However, many metrics rely on replacing input parts with uninformative information (e.g., to measure if the explanation method shows the same behavior as the classifier) or on comparisons to segmentation masks (e.g., to measure if the explanation method was able to localize relevant features). While providing uninformative features is trivial for e.g., images by replacing parts of an image with black or white pixels [38], replacing features with standard techniques (class means or zeros) might be relevant information in time series.

III. Problem Definition

We study a supervised TSC problem. Let $x = [x_{11}, ..., x_{NT}] \in \mathbb{R}^{N \times T}$ be a uni- or multivariate time series, where T is the number of time steps, and N is the number of features. Let $x_{i,t}$ be the input feature i at time t. Similarly, let $X_{:,t} \in \mathbb{R}^N$ and $X_{i,:} \in \mathbb{R}^T$ be the feature vector at time t, and the time vector for feature i, respectively. Y denotes the output, and $f : x \rightarrow Y$ is a classification model returning a probability distribution vector over classes $Y = [y_1, ..., y_C]$, where C is the total number of classes (i.e., outputs) and y_i the probability of x belonging to class i. An explanation method E_f finds an explanation $E_f(X) \in \mathbb{R}^{N \times T}$. In the case of feature attribution methods, the explainer E_f assigns an attribution a_{it} to explain the importance of a feature i a time step t, resulting in $E_f(X) = (a_{11}, ..., a_{NT})$. For example-based methods E_f provides an example with the same prediction or a counterexample resulting in $E_f(X) = (x'_{11}, ..., x'_{NT})$.

For an Explainer E_f to provide good explanations, those explanations need to be:

- Reliable: An explanation should be centered around the region of interest, the ground truth GT.

$$E_f(x) \cong GT$$

- Faithful: The explanation algorithm E_f should replicate the models f behavior.

$$E_f(x) \sim f(x)$$

- Robust: Similar inputs should result in similar explanations.

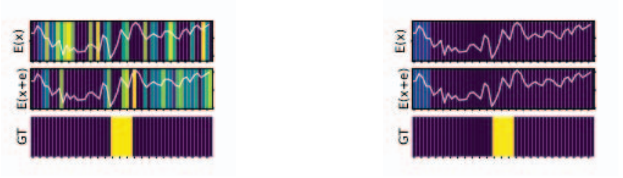
$$E_f(x) \approx E_f(x + \epsilon)$$

- Complex: Explanations using a smaller number of features are preferred. It is assumed that explanations using a large number of features are difficult for the user to understand [26].

$$\min \mathbb{1}_{E_f(x) > 0}$$

Figure 1 visualizes the implications of the requirements above on explanations obtained from an gradient-based explainer (an explanation based on the classifiers gradient estimations) and a perturbation-based explainer (an explanation based on observing the influence of input modifications). The top images show the original time series x with an explanation $E(x)$ visualized as a heatmap. The middle image shows the perturbed time series $x + \epsilon$ with the explanation $E(x + \epsilon)$. The bottom image shows the known ground truth GT. In case of this specific time series: The complexity is high for Figure 1a, resulting from the many attributions (highlights). For Figure 1b the complexity is low. Although Figure 1b performs better on complexity taking the ground truth GT into account, the attributions obtained on the sample are inconsistent with GT. The explanation in Figure 1b is more robust than Figure 1a as the explanations $E(x)$ and $E(x + \epsilon)$ are identical. Faithfulness quantifies the consistency between the decision-making process of f and the explanations E. The consistency of Figure 1a is higher than the one from Figure 1b as Figure 1a relies on the gradients of f while Figure 1b fits a surrogate model. Overall, in this case, although Figure 1b performs better on complexity and robustness than Figure 1a, due to the limited reliability (i.e., consistency with GT), Figure 1a should be the preferred explainer.

[2] Proximity, sparsity, diversity and plausibility are counterfactual-specific evaluation metrics and therefore not applicable to all explainer types.



(a) Gradient-Based E (b) Perturbation-Based E

Fig. 1: Visualization of metric implications on a sample explanation $E(x)$.

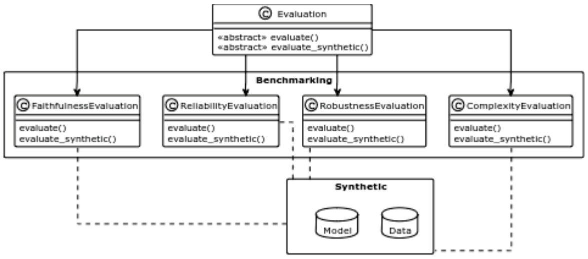


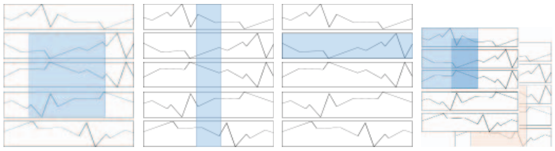
Fig. 2: Architecture of XTSC-Bench.

IV. XTSC-Bench: A Benchmarking Tool

The goal of XTSC-Bench is to provide a simple and standardized framework to allow users to apply and evaluate different State-of-the-Art explanation models in a standardized and replicable way on the notions of complexity, reliability, robustness, and faithfulness. Figure 2 visualizes the architecture. The benchmarking tool is split according to Section III into different classes for benchmarking reliability, faithfulness robustness, and complexity. As some of the notions (e.g., reliability) rely on a fairly accurate definition of an explanation ground truth GT or the iterative masking of parts of the original input with known uninformative features, we include uni- and multivariate synthetic data and pre-trained models in the benchmarking tool (see Section IV-A). Each class follows the evaluation interface, providing a method *evaluate* and a method *evaluate_synthetic*. The function *evaluate* allows the usage of non-synthetic data and models as well as the evaluation of a single explanation on-the-fly. For all metrics we use a wrapper build around Quantus [12] and added some time-specific tweaks.

A. Synthetic Data and Pretrained Models

XTSC-Bench provides 60 uni- and 60 multivariate synthetic datasets with 50 time steps generated according to Ismail et al. [22][3]. The 'base' dataset is generated based on various time series processes (Gaussian, Autoregressive, Continuous Autoregressive, Gaussian Process, Harmonic, NARMA and Pseudo Periodic). For each 'base' dataset obtained from the time series process, multiple synthetic datasets are obtained by adding various Informative Features ranging from Rare Features (less than 5% of features) and time steps (less than 5% of time steps) mimicking an anomaly detection task to boxes covering over 30% of features and time steps (see Figure 3). A binary label is added for each dataset (time process × informative feature) by highlighting informative features with the addition of a constant for positive classes and subtraction for negative classes. For all synthetic uni- and multivariate datasets, we train a 1D-Convolutional Network with ResNet Architecture (CNN) and Long Short Term Memory (LSTM) with a hidden layer of size 10. We train the networks with a cross-entropy loss for 500 epochs with a patience of 20 and Adam with a learning rate of 0.001. The trained networks are also provided in XTSC-Bench.



(a) Box (b) Rare Time (c) Rare Feat. (d) Moving

Fig. 3: Visualization of Informative Features types. The rectangle indicates the informative features.

B. Robustness

Robustness measures the stability of an explanation method's output subjected to a slight input perturbation $\bar{x} = x + \epsilon$ under the assumption that the model's output approximately stays the same $f(x) \approx f(\bar{x})$. Small, unmeaningful changes around x should lead to a consistent explanation. XTSC-Bench employs two metrics measuring the robustness of an explanation algorithm E:

- Max Sensitivity [38] measures the maximum change in the explanation with a small perturbation of the input x. r denotes the input neighborhood ratio.

$$Sens_{max}(E, f, x, r) = max_{\bar{x}-x \leq r} ||E_f(\bar{x}) - E_f(x)|| \quad (1)$$

- Average Sensitivity [38] denotes the average sensitivity in the neighborhood of x with $\bar{x} - x \leq r$.

$$Sens_{mean}(E, f, x, r) = \frac{1}{|x|} \sum ||E_f(\bar{x}) - E_f(x)|| \quad (2)$$

C. Faithfulness

Faithfulness quantifies the consistency between the prediction model f and explanation model E. Most faithfulness metrics rely on so called reference baselines consisting of non-informative features. In literature, those reference baselines are often training data means or zeros (e.g., [24]). However, for time series data those baselines might contain information (e.g., 0 might be an informative anomaly). Therefore, on the proposed synthetic data the reference baseline $\tilde{x}$ is sampled from the generation process. XTSC-Bench employs faithfulness correlation [38] to measure the correlation between the sum of attributions $\sum_{s \in S} E_f(x_{x_s=\tilde{x}_s})$ and the difference in output $f(x) - f(x_{x_s=\tilde{x}_s})$ when setting those features to a reference baseline $x_{x_s=\tilde{x}_s}$. S is a subset of input features, $\tilde{x}_S$ denotes a

[3] Find details on the Data Generation in [22] or in the Appendix B in our GitHub Repository: https://github.com/JHoelli/XTSC-Bench/blob/main/Appendix.pdf.

subset of the reference baseline $\tilde{x}$ and x_s the corresponding subset for the original instance x.[4]

$$Faith(f,E,x) = corr(\sum_{s \in S} E_f\ (x_{x_s=\tilde{x}_s}), f(x) - f(x_{x_s=\tilde{x}_s})) \tag{3}$$

D. Complexity

Complexity [38] measures the number of features used in an explanation with a fractional contribution distribution $\mathbb{P}_g$: the fractional contribution of feature x_i to the total magnitude of the attribution: $\mathbb{P}_g(i) = \frac{E_f(x)_i}{\sum |E_f(x)|}; \mathbb{P}_g \in \{\mathbb{P}_g(1) \ldots, \mathbb{P}_g(d)\}$. The maximum value of complexity is $log(|E_f(x)|)$, where $|.|$ denotes the vector length.

$$cpx(f;E;x) = -\sum_{i=1}^{d} \mathbb{P}_g(i) ln(\mathbb{P}_g(i)) \tag{4}$$

E. Reliability

Explanation methods should distinguish important from unimportant features at each time step and note changes over time. “Major” parts of an explanation should lie inside the ground truth mask $GT(x)$. XTSC-Bench includes the ground truth based measures relevance rank accuracy and relevance mask accuracy from [40].

- Relevance Rank Accuracy [40]: The relevance rank accuracy measures how much of the high intensity relevance lies within the ground truth. We sort the top K values of $E_f(x)$ in decreasing order $X_{topK} = \{x_1, ..., x_k | E_f(x)_1 > ... > E_f(x)_K\}$.

$$RACC = \frac{|X_{topK} \cap GT(x)|}{|GT(x)|} \tag{5}$$

- Relevance Mass Accuracy [40]: The relevance mass accuracy is computed as the ratio of the sum of the Explanation values lying within the ground truth mask over the sum of all values.

$$MACC = \frac{\sum_{E_f(x)_i \in GT(x)} E_f(x)_i}{\sum E_f(x)} \tag{6}$$

V. Empirical Evaluation

This section compares the performance of 6 gradient- with 3 perturbation-based feature attribution methods and 2 example-based methods across Recurrent Neural Networks and Temporal Convolutional Networks for both the multi- and univariate synthetic time series (Section IV-A). The results are reported on a before unseen test set. As gradient-based methods, we include Saliency (GRAD) [41], Gradient Shap (GS) [4], and Smooth Gradient (SG) [5] with and without Temporal Saliency Rescaling (TSR) [22]. As perturbation-based, we include Feature Occlusion (FO) [25] with and without Temporal Saliency Rescaling (TSR) [22] and LEFTIST [16], an approach based on Lime adapted to time series. TSEvo [20], and Native Guide (NG) [18] represent the example-based methods. For all methods, we use the implementation in TSInterpret [14]. By employing XTSC-Bench, we evaluate the explainers' capabilities on complexity, reliability, robustness, and faithfulness for all classifiers with an accuracy of over 90%[5]. Additional information regarding the setting and the results can be found in our GitHub[6].

Figure 4 summarizes the explainer-wise results on complexity, reliability, faithfulness, and robustness, averaged over all datasets and classifier models. On complexity (Figure 4a and Figure 4b), gradient- and perturbation-based methods provide less complex explanations than example-based methods. The results obtained by TSR contain slightly fewer attributions than the plain gradient- and perturbation-based methods, indicating that the explanations obtained after Temporal Saliency Rescaling are slightly easier to grasp. Averaging the obtained relevance scores on both the feature and time domain with TSR leads to a complexity decrease by eliminating areas with less relevance (e.g., single and small relevance scores on certain time steps).

The reliability (Figure 4c and Figure 4d) on univariate data is higher than on multivariate data showing a decreasing capability of centering the explanation around the, in this case, known ground truth of all explainers with increasing data complexity. The on average lower relevance mask than rank indicates that while relevant features are found, the contribution of the found informative features to the overall relevance is low. Interestingly for both dataset types, the plain gradient- and perturbation-based methods (without TSR) perform slightly better on the Relevance Rank. On Relevance Mass, the difference between TSR and the plain approaches diverge (e.g., on univariate GRAD, TSR results in an improvement, on univariate GS, TSR results in a deterioration).

The faithfulness (Figure 4e and Figure 4f) of the explainers to the classification models' behavior is similar for most explainers on the uni- and multivariate data. Least faithful is LEFTIST, as LEFTIST is the only approach relying on a local surrogate model instead of frequent classifier calls or the classifiers' inner workings (i.e., gradients).

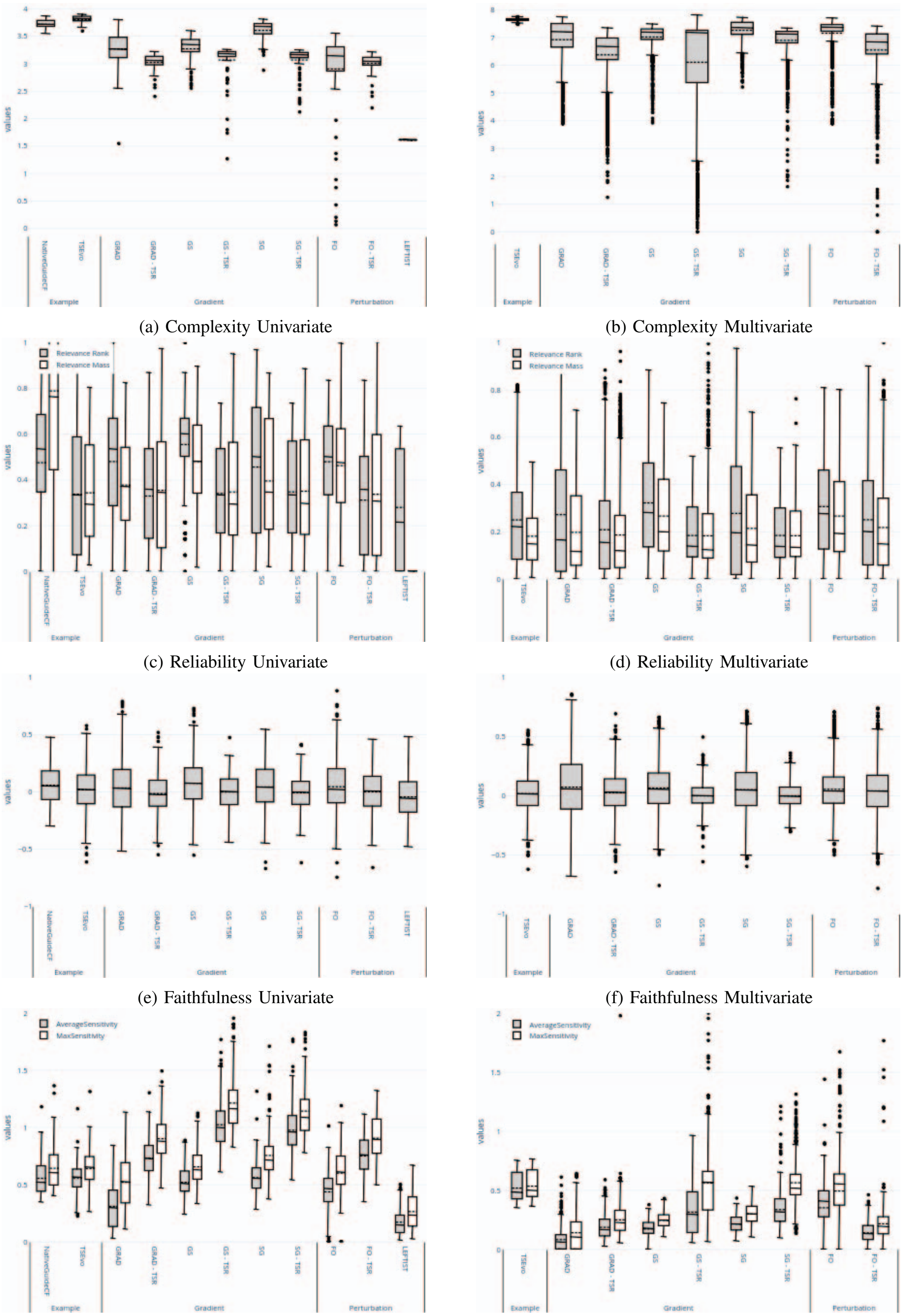
The results on robustness (Figure 4g and Figure 4h) indicate that on univariate data, perturbation-based approaches are less sensitive to small changes than example-based approaches. This results from perturbation-based approaches only relying on the perturbation function (which is constantly the same) and the classification model's output, while gradient-based approaches rely on a model's inner workings that possibly change with varying the input.

Summarizing the results, no clear indication can be given on which explanation approaches should be preferred. No approach was able to dominate the plain gradient, and perturbation-based methods, which are included as baselines. Both, traditional and time-series specific explainers show potential for improvement in all aspects. With increasing data complexity (univariate vs. multivariate), the metric performances diverge further, indicating a need for less complex, more reliable, and robust explainers, especially for multivariate time series classification.

[4] In case of using our benchmarking tool with non-synthetic data we provide the possibility to provide a custom baseline. As default, baselining is done uniformly.

[5] Explainers are usually used to validate the inner-workings of well-performing classifiers. Classifiers with low accuracy cannot be expected to learn sufficient features to ensure an explainers reliability and classifier consistency.

[6] https://github.com/JHoelli/XTSC-Bench



(a) Complexity Univariate

(b) Complexity Multivariate

(c) Reliability Univariate

(d) Reliability Multivariate

(e) Faithfulness Univariate

(f) Faithfulness Multivariate

(g) Robustness Univariate

(h) Robustness Multivariate

Fig. 4: Explainer Performance on complexity, reliability, faithfulness, and robustness averaged over all datasets. The line denotes the median and the dotted line the mean. The start and end of the boxes are the first and third quartiles. Note, that Native Guide and LEFTIST only apply to univariate data and are therefore missing in the multivariate evaluation.

VI. Conclusion

In this work, we propose XTSC-Bench, a benchmarking tool for the standardized evaluation of explainers for time series classifiers. XTSC-Bench aims to dissolve existing ambiguities and enable more comparability by providing synthetic datasets with informative features, from analogies to anomaly detection to moving features, trained models for the synthetic data, and options to evaluate custom data. A first empirical evaluation of the explainers implemented in TSInterpret [14] showed that the current time series explainers leave potential for improvement, especially in providing reliable explanations for multivariate TSC.

References

[1] A. F. Markus, J. A. Kors, and P. R. Rijnbeek, "The role of explainability in creating trustworthy artificial intelligence for health care: a comprehensive survey of the terminology, design choices, and evaluation strategies," *Journal of Biomedical Informatics*, vol. 113, p. 103655, 2021.

[2] S. Vollert, M. Atzmueller, and A. Theissler, "Interpretable machine learning: A brief survey from the predictive maintenance perspective," in *2021 26th IEEE international conference on emerging technologies and factory automation (ETFA)*. IEEE, 2021, pp. 01–08.

[3] S. Wachter, B. Mittelstadt, and C. Russell, "Counterfactual explanations without opening the black box: Automated decisions and the gdpr," *Harv. JL & Tech.*, vol. 31, p. 841, 2017.

[4] S. M. Lundberg and S.-I. Lee, "A unified approach to interpreting model predictions," *NeurIPS*, vol. 30, 2017.

[5] D. Smilkov, N. Thorat, B. Kim, F. Viégas, and M. Wattenberg, "Smoothgrad: removing noise by adding noise," *arXiv preprint arXiv:1706.03825*, 2017.

[6] M. T. Ribeiro, S. Singh, and C. Guestrin, "Why should i trust you?: Explaining the predictions of any classifier," in *Proceedings of the 22nd ACM SIGKDD*. ACM, 2016, pp. 1135–1144.

[7] D. V. Carvalho, E. M. Pereira, and J. S. Cardoso, "Machine learning interpretability: A survey on methods and metrics," *Electronics*, vol. 8, no. 8, p. 832, 2019.

[8] Z. C. Lipton, "The mythos of model interpretability: In machine learning, the concept of interpretability is both important and slippery." *Queue*, vol. 16, no. 3, pp. 31–57, 2018.

[9] P. Schmidt and F. Biessmann, "Quantifying interpretability and trust in machine learning systems," in *Proceedings of the AAAI-19 Workshop on Network Interpretability for Deep Learning*, 2019.

[10] N. Kokhlikyan, V. Miglani, M. Martin, E. Wang, B. Alsallakh, J. Reynolds, A. Melnikov, N. Kliushkina, C. Araya, S. Yan *et al.*, "Captum: A unified and generic model interpretability library for pytorch," *arXiv preprint arXiv:2009.07896*, 2020.

[11] V. Arya and et al, "One explanation does not fit all: A toolkit and taxonomy of ai explainability techniques," 2019. [Online]. Available: https://arxiv.org/abs/1909.03012

[12] A. Hedström and et al, "Quantus: An explainable ai toolkit for responsible evaluation of neural network explanations and beyond," *JMLR*, vol. 24, no. 34, pp. 1–11, 2023.

[13] T. Rojat, R. Puget, D. Filliat, J. Del Ser, R. Gelin, and N. Díaz-Rodríguez, "Explainable artificial intelligence (xai) on timeseries data: A survey," *arXiv preprint arXiv:2104.00950*, 2021.

[14] J. Höllig, C. Kulbach, and S. Thoma, "Tsinterpret: A python package for the interpretability of time series classification," *JOSS*, vol. 8, no. 85, p. 5220, 2023.

[15] A. Theissler, F. Spinnato, U. Schlegel, and R. Guidotti, "Explainable ai for time series classification: A review, taxonomy and research directions," *IEEE Access*, 2022.

[16] M. Guillemé, V. Masson, L. Rozé, and A. Termier, "Agnostic local explanation for time series classification," in *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*. IEEE, 2019, pp. 432–439.

[17] H. A. Dau, A. Bagnall, K. Kamgar, C.-C. M. Yeh, Y. Zhu, S. Gharghabi, C. A. Ratanamahatana, and E. Keogh, "The ucr time series archive," *IEEE/CAA Journal of Automatica Sinica*, vol. 6, no. 6, pp. 1293–1305, 2019.

[18] E. Delaney, D. Greene, and M. T. Keane, "Instance-based counterfactual explanations for time series classification," in *ICCBR 2021*. Springer, 2021, pp. 32–47.

[19] M. Pawelczyk, S. Bielawski, J. v. d. Heuvel, T. Richter, and G. Kasneci, "Carla: a python library to benchmark algorithmic recourse and counterfactual explanation algorithms," *NeurIPS 2021*, 2021.

[20] J. Höllig, C. Kulbach, and S. Thoma, "Tsevo: Evolutionary counterfactual explanations for time series classification," in *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2022, pp. 29–36.

[21] E. Ates, B. Aksar, V. J. Leung, and A. K. Coskun, "Counterfactual explanations for multivariate time series," in *2021 ICAPAI*, 2021, pp. 1–8.

[22] A. A. Ismail, M. Gunady, H. Corrada Bravo, and S. Feizi, "Benchmarking deep learning interpretability in time series predictions," *NeurIPS*, vol. 33, pp. 6441–6452, 2020.

[23] R. R. Selvaraju, M. Cogswell, A. Das, R. Vedantam, D. Parikh, and D. Batra, "Grad-cam: Visual explanations from deep networks via gradient-based localization," in *Proceedings of the IEEE international conference on computer vision*, 2017, pp. 618–626.

[24] M. Sundararajan, A. Taly, and Q. Yan, "Axiomatic attribution for deep networks," in *International conference on machine learning*. PMLR, 2017, pp. 3319–3328.

[25] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *ECCV 2014*. Springer, 2014, pp. 818–833.

[26] T. Miller, "Explanation in artificial intelligence: Insights from the social sciences," *Artificial intelligence*, vol. 267, pp. 1–38, 2019.

[27] D. Alvarez Melis and T. Jaakkola, "Towards robust interpretability with self-explaining neural networks," *NeurIPS*, vol. 31, 2018.

[28] R. Guidotti, A. Monreale, F. Spinnato, D. Pedreschi, and F. Giannotti, "Explaining any time series classifier," in *2020 IEEE Second International Conference on Cognitive Machine Intelligence (CogMI)*. IEEE, 2020, pp. 167–176.

[29] R. Guidotti, A. Monreale, S. Ruggieri, F. Turini, F. Giannotti, and D. Pedreschi, "A survey of methods for explaining black box models," *ACM computing surveys (CSUR)*, vol. 51, no. 5, pp. 1–42, 2018.

[30] A. A. Freitas, "Comprehensible classification models: a position paper," *ACM SIGKDD*, vol. 15, no. 1, pp. 1–10, 2014.

[31] U. Schlegel, H. Arnout, M. El-Assady, D. Oelke, and D. A. Keim, "Towards a rigorous evaluation of xai methods on time series," in *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*. IEEE, 2019, pp. 4197–4201.

[32] O. Bahri, S. F. Boubrahimi, and S. M. Hamdi, "Shapelet-based counterfactual explanations for multivariate time series," *ACM SIGKDD Workshop on Mining and Learning from Time Series (KDD-MiLeTS 2022)*, 2022.

[33] S. A. Siddiqui, D. Mercier, A. Dengel, and S. Ahmed, "Tsinsight: A local-global attribution framework for interpretability in time series data," *Sensors*, vol. 21, no. 21, p. 7373, 2021.

[34] A. J. Bagnall and et al, "The UEA multivariate time series classification archive, 2018," *CoRR*, vol. abs/1811.00075, 2018.

[35] R. Fong, M. Patrick, and A. Vedaldi, "Understanding deep networks via extremal perturbations and smooth masks," in *Proceedings of the IEEE/CVF international conference on computer vision*, 2019, pp. 2950–2958.

[36] A. Shrikumar, P. Greenside, and A. Kundaje, "Learning important features through propagating activation differences," in *Proceedings of the 34th International Conference on Machine Learning - Volume 70*. JMLR.org, 2017, p. 3145–3153.

[37] S. A. Siddiqui, D. Mercier, M. Munir, A. Dengel, and S. Ahmed, "Tsviz: Demystification of deep learning models for time-series analysis," *IEEE Access*, vol. 7, pp. 67 027–67 040, 2019.

[38] U. Bhatt, A. Weller, and J. M. F. Moura, "Evaluating and aggregating feature-based model explanations," in *Proceedings of the Twenty-Ninth International Joint Conference on Artificial Intelligence*, 2021.

[39] A.-p. Nguyen and M. R. Martínez, "On quantitative aspects of model interpretability," *arXiv preprint arXiv:2007.07584*, 2020.

[40] L. Arras, A. Osman, and W. Samek, "Clevr-xai: a benchmark dataset for the ground truth evaluation of neural network explanations," *Information Fusion*, vol. 81, pp. 14–40, 2022.

[41] K. Simonyan, A. Vedaldi, and A. Zisserman, "Deep inside convolutional networks: Visualising image classification models and saliency maps," *arXiv preprint arXiv:1312.6034*, 2013.