The Journal of Open Source Software

TSInterpret: A Python Package for the Interpretability of Time Series Classification

Jacqueline Höllig [1¶], **Cedric Kulbach** [1], **and Steffen Thoma** [1]

1 FZI Research Center for Information Technology, Karlsruhe, Germany ¶ Corresponding author

DOI: 10.21105/joss.05220

Software

- Review
- Repository
- Archive

Editor: Britta Westner

Reviewers:

- @ulf1
- @jwwthu
- @siebert-julien

Submitted: 22 December 2022
Published: 02 May 2023

Summary

`TSInterpret` is a python package that enables post-hoc interpretability and explanation of black-box time series classifiers with three lines of code. Due to the specific structure of time series (i.e., non-independent features (Ismail et al., 2020)) and unintuitive visualizations (Siddiqui et al., 2019), traditional interpretability and explainability libraries like Captum (Kokhlikyan et al., 2020), Alibi Explain (Klaise et al., 2021), or tf-explain (Meudec, 2021) find limited usage. `TSInterpret` specifically addresses the issue of black-box time series classification by providing a unified interface to state-of-the-art interpretation algorithms in combination with default plots. In addition, the package provides a framework for developing additional easy-to-use interpretability methods.

Statement of need

Temporal data is ubiquitous and encountered in many real-world applications ranging from electronic health records (Rajkomar et al., 2018) to cyber security (Susto et al., 2018). Although deep learning methods have been successful in the field of Computer Vision (CV) and Natural Language Processing (NLP) for almost a decade, application on time series data has only occurred in the past few years (e.g., Fawaz et al., 2019; Rajkomar et al., 2018; Ruiz et al., 2021; Susto et al., 2018). Deep learning models have achieved state-of-the-art results on time series classification (e.g., Fawaz et al., 2019). However, those methods are black boxes due to their complexity which limits their application to high-stake scenarios (e.g., in medicine or autonomous driving), where user trust and understandability of the decision process are crucial. In such scenarios, post-hoc interpretability is useful as it enables the analysis of already trained models without model modification. Much work has been done on post-hoc interpretability in CV and NLP, but most developed approaches are not directly applicable to time series data. The time component impedes the usage of existing methods (Ismail et al., 2020). Thus, increasing effort is put into adapting existing methods to time series (e.g., LEFTIST based on SHAP/Lime (Guillemé et al., 2019), Temporal Saliency Rescaling for Saliency Methods (Ismail et al., 2020), or Counterfactuals (Ates et al., 2021; Delaney et al., 2021; Höllig et al., 2022)). Compared to images or textual data, humans cannot intuitively and instinctively understand the underlying information in time series data. Therefore, time series data, both uni- and multivariate, have an unintuitive nature, lacking an understanding at first sight (Siddiqui et al., 2019). Hence, providing suitable visualizations of time series interpretability becomes crucial.

Features

Explanations can take various form (see Figure 1). Different use cases or users need different types of explanations. While for a domain expert, counterfactuals are useful, a data scientist or machine learning engineer prefers gradient-based approaches (Ismail et al., 2020) to evaluate

the model's feature attribution.

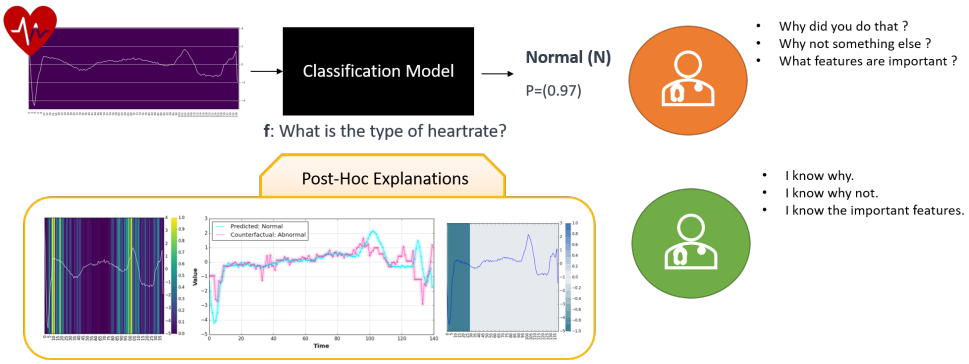

Figure 1: Explanations.

Counterfactual approaches calculate counterexamples by finding a time series close to the original time series that is classified differently, thereby showing decision boundaries. The intuition is to answer the question 'What if?'. `TSInterpret` implements Ates et al. (2021), a perturbation-based approach for multivariate data, Delaney et al. (2021) for univariate time series, and Höllig et al. (2022) an evolutionary based approach applicable to uni-and multivariate data. Gradient-based approaches (e.g., GradCam) were adapted to time series by Ismail et al. (2020) who proposed rescaling according to time step importance and feature importance. This is applicable to both gradient and perturbation-based methods and based on tf-explain (Meudec, 2021) and captum (Kokhlikyan et al., 2020). LEFTIST by Guillemé et al. (2019) calculates feature importance based on a variety of Lime based on shapelets.

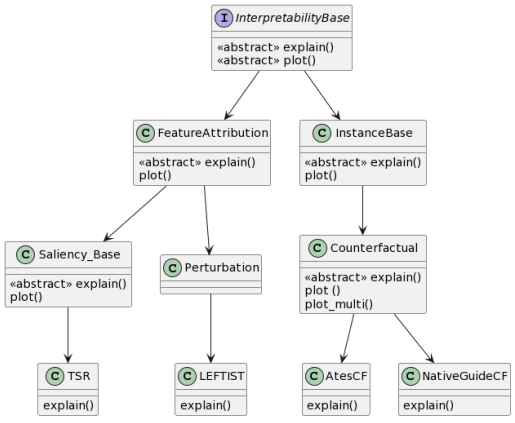

Figure 2: Architecture of TSInterpret.

`TSInterpret` implements these algorithms according to the taxonomy shown in Figure 2. The interpretability methods are sorted according to a) the model output (e.g., is a feature map returned or an example time series) and b) the used mechanism (e.g., based on gradients). All implemented objects share the interface `InterpretabilityBase` to enforce unified access (i.e., all methods must implement the functions `explain` and `plot`). The `plot` function is implemented on the level below the interface based on the output structure provided by the interpretability algorithm to provide a unified visualization experience (e.g., in the case of Feature Attribution, the `plot` function visualizes a heatmap on the original sample). If necessary, those plots are refined by the Mechanism layer. The `explain` function is implemented on the method level. This high reusability ensures the consistency and extensibility of the framework.

Acknowledgements

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

References

Ates, E., Aksar, B., Leung, V. J., & Coskun, A. K. (2021). Counterfactual Explanations for Machine Learning on Multivariate Time Series Data. *2021 International Conference on Applied Artificial Intelligence (ICAPAI)*, 1–8. https://doi.org/10.1109/ICAPAI49758.2021.9462056

Delaney, E., Greene, D., & Keane, M. T. (2021). Instance-based counterfactual explanations for time series classification. In *International Conference on Case-Based Reasoning* (pp. 32–47). Springer. https://doi.org/10.1007/978-3-030-86957-1_3

Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L., & Muller, P.-A. (2019). Deep learning for time series classification: A review. *Data Mining and Knowledge Discovery*, *33*(4), 917–963. https://doi.org/10.1007/s10618-019-00619-1

Guillemé, M., Masson, V., Rozé, L., & Termier, A. (2019). Agnostic Local Explanation for Time Series Classification. *2019 IEEE 31st International Conference on Tools with Artificial Intelligence (ICTAI)*, 432–439. https://doi.org/10.1109/ictai.2019.00067

Höllig, J., Kulbach, C., & Thoma, S. (2022). TSEvo: Evolutionary counterfactual explanations for time series classification. *2022 21st IEEE International Conference on Machine Learning and Applications (ICMLA)*, 29–36. https://doi.org/10.1109/icmla55696.2022.00013

Ismail, A. A., Gunady, M., Bravo, H. C., & Feizi, S. (2020). Benchmarking Deep Learning Interpretability in Time Series Predictions. *arXiv:2010.13924*. https://doi.org/10.48550/arXiv.2010.13924

Klaise, J., Looveren, A. V., Vacanti, G., & Coca, A. (2021). Alibi Explain: Algorithms for Explaining Machine Learning Models. *Journal of Machine Learning Research*, *22*(181), 1–7.

Kokhlikyan, N., Miglani, V., Martin, M., Wang, E., Alsallakh, B., Reynolds, J., Melnikov, A., Kliushkina, N., Araya, C., Yan, S., & Reblitz-Richardson, O. (2020). Captum: A unified and generic model interpretability library for PyTorch. *arXiv:2009.07896*. https://doi.org/10.48550/arXiv.2009.07896

Meudec, R. (2021). *tf-explain*. Zenodo. https://doi.org/10.5281/ZENODO.5711704

Rajkomar, A., Oren, E., Chen, K., Dai, A. M., Hajaj, N., Hardt, M., Liu, P. J., Liu, X., Marcus, J., Sun, M., Sundberg, P., Yee, H., Zhang, K., Zhang, Y., Flores, G., Duggan, G. E., Irvine, J., Le, Q., Litsch, K., … Dean, J. (2018). Scalable and accurate deep learning with electronic health records. *Npj Digital Med*, *1*(1), 18. https://doi.org/10.1038/s41746-018-0029-1

Ruiz, A. P., Flynn, M., Large, J., Middlehurst, M., & Bagnall, A. (2021). The great multivariate time series classification bake off: A review and experimental evaluation of recent algorithmic advances. *Data Min Knowl Disc*, *35*(2), 401–449. https://doi.org/10.1007/s10618-020-00727-3

Siddiqui, S. A., Mercier, D., Munir, M., Dengel, A., & Ahmed, S. (2019). TSViz: Demystification of Deep Learning Models for Time-Series Analysis. *IEEE Access*, *7*, 67027–67040. https://doi.org/10.1109/access.2019.2912823

Susto, G. A., Cenedese, A., & Terzi, M. (2018). Time-Series Classification Methods: Review and Applications to Power Systems Data. In *Big Data Application in Power Systems* (pp. 179–220). Elsevier. https://doi.org/10.1016/b978-0-12-811968-6.00009-7