



Article scientifique

Article

2023

Published version

Open Access

This is the published version of the publication, made available in accordance with the publisher's policy.

Evaluation of post-hoc interpretability methods in time-series classification

Turbé, Hugues; Bjelogrić, Mina; Lovis, Christian; Mengaldo, Gianmarco

How to cite

TURBÉ, Hugues et al. Evaluation of post-hoc interpretability methods in time-series classification. In: Nature machine intelligence, 2023, vol. 5, n° 3, p. 250–260. doi: 10.1038/s42256-023-00620-w

This publication URL: <https://archive-ouverte.unige.ch/unige:168612>

Publication DOI: [10.1038/s42256-023-00620-w](https://doi.org/10.1038/s42256-023-00620-w)

© The author(s). This work is licensed under a Creative Commons Attribution (CC BY 4.0)

<https://creativecommons.org/licenses/by/4.0>

Evaluation of post-hoc interpretability methods in time-series classification

Received: 7 February 2022

Accepted: 30 January 2023

Published online: 13 March 2023

 Check for updates

Hugues Turbé^{1,2}✉, Mina Bjelogrić^{1,2}, Christian Lovis^{1,2} & Gianmarco Mengaldo^{3,4}✉

Post-hoc interpretability methods are critical tools to explain neural-network results. Several post-hoc methods have emerged in recent years but they produce different results when applied to a given task, raising the question of which method is the most suitable to provide accurate post-hoc interpretability. To understand the performance of each method, quantitative evaluation of interpretability methods is essential; however, currently available frameworks have several drawbacks that hinder the adoption of post-hoc interpretability methods, especially in high-risk sectors. In this work we propose a framework with quantitative metrics to assess the performance of existing post-hoc interpretability methods, particularly in time-series classification. We show that several drawbacks identified in the literature are addressed, namely, the dependence on human judgement, retraining and the shift in the data distribution when occluding samples. We also design a synthetic dataset with known discriminative features and tunable complexity. The proposed methodology and quantitative metrics can be used to understand the reliability of interpretability methods results obtained in practical applications. In turn, they can be embedded within operational workflows in critical fields that require accurate interpretability results for, example, regulatory policies.

Time-series—sequences of indexed data that follow a specific time order—are ubiquitous. They can describe physical systems¹ such as the state of the atmosphere and its evolution, social and economic systems² such as the financial market, and biological systems³ such as the heart and the brain via electrocardiogram (ECG) and electroencephalogram signals, respectively. The availability of this type of data is increasing, and so is the need for automated analysis tools that are capable of extracting interpretable and actionable knowledge from them. To this end, although established and more interpretable time-series approaches remain competitive for many tasks^{4–6}, artificial intelligence (AI) technologies and neural networks in particular are opening the path towards highly accurate predictive tools for an increasing number of

time-series regression^{7–9} and classification^{10,11} learning tasks. Yet the adoption of AI technologies as black-box tools is problematic in several applied contexts. To address this issue, numerous interpretability methods have been proposed in the literature, especially in the context of neural networks. These different methods usually produce tangibly different results, preventing practitioners from fully unlocking the interpretability of the results, which is increasingly needed. Figure 1 shows four different post-hoc interpretability methods applied to time-series classification, in which the neural network is tasked with identifying the pathology associated with a patient's ECG. The four interpretability methods produce remarkably different results for the same model. Hence the question: which method produced an

¹Division of Medical Information Sciences, University Hospitals of Geneva, Geneva, Switzerland. ²Department of Radiology and Medical Informatics, University of Geneva, Geneva, Switzerland. ³Department of Mechanical Engineering, College of Design and Engineering, National University of Singapore, Singapore, Singapore. ⁴Honorary Research Fellow, Department of Aeronautics, Imperial College London, London, UK. ✉e-mail: hugues.turbe@unige.ch; mpegim@nus.edu.sg

interpretability map closer to the one actually adopted by the neural network to make its prediction? In this paper we answer this question quantitatively while addressing the issues found in the existing literature on the evaluation of interpretability methods. Aside from research purposes, understanding the accuracy of interpretability methods is de facto mandatory in critical sectors (such as healthcare) for legal and ethical reasons¹². Failing to understand the performance of interpretability methods may prevent their adoption and, in turn, lead practitioners to avoid using neural network tools altogether, in favour of more white-box and interpretable tools.

Different definitions of what it means for a neural-network model to be interpretable have been formulated. Most of these definitions can be summarized under two categories: transparency and post-hoc interpretability¹³. Transparency refers to how a model and its individual constituents work, whereas post-hoc interpretability refers to how a trained model makes predictions and uses the input features it is given. In this work we consider post-hoc interpretability applied to time-series classification, as it is seen as a key to meet recent regulatory requirements¹² and translate current research efforts into real-world applications, especially in high-risk areas such as healthcare¹⁴. Post-hoc interpretability methods assign a relevance to each feature of a sample, reflecting its importance to the model for the classification task being performed. The ability to express the specific features used by a neural network to classify a given sample can help humans assess the reliability of the classification produced and allows one to compare the model's predictions with existing knowledge. It also provides a way to understand possible model biases that could lead to the model's failure in a real-world setting.

A range of methods to provide post-hoc interpretability of classification results have been developed in the past few years. These are mainly focused on natural language processing and image classification tasks. With the more recent growing interest for neural-network interpretability, leading actors in the machine learning community built a range of post-hoc interpretability methods. As part of this effort, Facebook recently released the Captum library to group a large number of interpretability methods under a single developmental framework¹⁵. Although these initiatives allow researchers to more easily use the different methods, they do not provide a systematic and comprehensive evaluation of those methods on data with different characteristics and across neural-network architectures. A systematic methodology that provides the accurate evaluation of these methods is of paramount importance to allow their wider adoption, and measure how trustable the results they provide are.

The evaluation of interpretability methods was initially based on a heuristic approach in which the relevance attributed to the different features was compared with the expectation of an observer for common image classification tasks¹⁶, or of a domain expert for more complex tasks^{17,18}. However, these works shared a common pitfall: they assumed the representation of a task learned by a neural network should use the same features as a human expert. The community later moved towards the idea that the evaluation should be independent of human judgement¹⁹. This paradigm shift was supported by the evidence that certain saliency methods—while looking attractive to human experts—produced results independent of the model they aimed to explain, thereby failing the interpretability task²⁰. More recent evaluations were performed by occluding (also referred to as corrupting) the most relevant features identified and comparing the drop in score observed between model predictions on the initial and modified samples²¹. This evaluation method was later questioned, as corrupting the images changes the distribution of the values of the sample and therefore the observed drop in score might be caused by this shift in distribution rather than actual information being removed²². An approach named ROAR was proposed to address this issue²², in which important pixels are removed in both the training and testing sets. The model is then retrained on the corrupted (that is, occluded) samples, with the

drop in score being retained on this newly trained model. This method has the benefit of maintaining a similar distribution across the training and evaluation sets with the modified samples. Yet we argue that it does not necessarily explain which features the initial network used to make its prediction as the similarity between neural network models is only maintained if the models are trained on datasets sampled from the same distribution²³. In their case, the distribution is changed as the model is retrained on a corrupted dataset and therefore the post-hoc interpretability of the retrained model is not constrained to being similar to the one of the initial model. The post-hoc interpretability instead highlights the properties of the dataset in regards to its target, such as the redundancy of the information present in the features that are indicative of a given class—a limitation that was acknowledged by the authors²¹.

Neural network interpretability for time-series data was only recently explored. Initial efforts applied some of the interpretability methods introduced for natural language processing and image classification on univariate time series, and evaluated the drop in score obtained by corrupting the most relevant parts (also referred to as time steps) of the signal²⁴. An evaluation of some interpretability methods was recently proposed²⁵, with a dataset designed to address the issue of retaining equal distribution between the initial and occluded datasets; however, this work may have two crucial drawbacks: the proposed dataset contains static discriminative properties (for example, the mean of the sample) and it is not independent of human judgement. The former issue can lead the model to learn from static properties and thus the dataset might not reflect the complexity of real-world time-series classification tasks, in which time dependencies usually play the discriminative role. The latter is related to the assumption that the model uses:

- all the discriminative information synthetically provided (comprising a static shift applied to a portion of the time series),
- no information outside of it.

We argue that this assumption does not necessarily hold as the model might require just a subset of the discriminative information provided and might use information from outside of the discriminative portion.

In this work we propose an approach for the model-agnostic evaluation of interpretability methods for time-series classification that addresses the various issues just highlighted. The approach consists of two new metrics, namely $AUC\hat{S}_{top}$ and $F1\hat{S}$. The first is the area under the top curve, and aims to measure how the top relevance indeed captures the most important time steps for the neural network, whereas the second—the modified F1 score—is a harmonic mean reflecting the capability of the different interpretability methods to capture both the most and least important time steps. These two metrics evaluate how interpretability methods order time steps according to their importance, referred to as relevance identification. In this paper we also aim to qualitatively evaluate the capacity of the different interpretability methods to reflect the importance of each time step relative to the others. The latter evaluation is referred to as relevance attribution. We note that a key aspect of this work is the training of the models with a random level of perturbation for each batch, in a similar fashion to widely used data-augmentation methods²⁶. This perturbation is later used to corrupt the signal when evaluating interpretability methods such that the distribution is maintained across the training and perturbed datasets used for the evaluation. This addresses one of the main concerns found in the literature (that is, the shift in distribution when occluding samples in the evaluation set) and does not require retraining through the ROAR approach.

The six interpretability methods we considered are: (1) DeepLift²⁷, (2) GradShap¹⁷, (3) Integrated Gradients²⁸, (4) KernelShap¹⁷, (5) DeepLiftShap¹⁷ and (6) Shapley Value Sampling (also referred to as Shapley sampling or simply Shapley)²⁹. These were chosen to capture

a broad range of available interpretability methods while keeping the problem computationally tractable for all of the models presented. These interpretability methods are applied to three neural-network architectures; namely, convolutional (CNN), bidirectional long-short term memory (Bi-LSTM) and Transformer neural networks. The evaluation of the interpretability methods for time-series classification is performed on a new synthetic dataset as well as on two datasets adopted in practical applications. The overall code framework is part of the InterpretTime library freely available at Github (<https://github.com/hturbe/InterpretTime>).

In summary, the approach proposed and the new synthetic dataset we outlined address the following points:

1. The need for a robust and quantifiable approach to evaluate and rank the performance of interpretability methods over different neural-network architectures trained for the classification of time series. Our approach addresses the issues found in the literature by providing novel quantitative metrics for the evaluation of interpretability methods independent of human judgement¹⁹, using an occluded dataset²¹ and without retraining the model²².
2. The lack of a synthetic dataset with tunable complexity that can be used to assess the performance of interpretability methods, and that is able to reproduce time-series classification tasks of arbitrary complexity. We note that our synthetic dataset differs from ref. ²⁵ as the neural network must learn the time dependencies in the data. Furthermore, the dataset encodes a priori knowledge of the discriminative features, analogous to the BlockMNIST synthetic dataset³⁰. Finally, the classification task is multivariate by design, as the neural network must learn at least two features to predict the correct class. This is a desirable property as real-world datasets are commonly multivariate.

We first present an evaluation of six interpretability methods using the proposed framework across different datasets and model architectures. These results are then discussed to highlight the main trends as well as the potential for the developed metrics to build trust in post-hoc interpretability methods. In Methods we outline the new framework to evaluate interpretability methods for time-series classification, including the novel method used to maintain a constant distribution between the training and evaluation sets, the new metrics and the synthetic dataset.

Results

All of the metrics presented in this section are built on the relevance—denoted by \mathbf{R} —that an interpretability method provides along the time series (a more detailed explanation for \mathbf{R} is provided in Table 1 and Methods). An example for the ECG time series is depicted in Fig. 1, in which the countour maps represent \mathbf{R} and the black lines represent the actual time series the neural network is using to make the prediction. The higher the relevance, the more important the portion of the time series associated with it is for the neural network classification task. The metrics are evaluated on three different datasets (synthetic, ECG and FordA) and three different architectures (Bi-LSTM, CNN and Transformer). The hyperparameters and classification metrics for the different models are presented in Supplementary Sections 2 and 3.

We next focus on evaluating the effectiveness of an interpretability method in ordering time steps according to their importance to explaining the neural network’s predictions. This crucial aspect of interpretability methods’ evaluation is also referred to as relevance identification; it is measured by $\text{AUC}\tilde{S}_{\text{top}}$ and $\text{F1}\tilde{S}$, which are described in Table 1 and Methods.

The ordering of the time steps obtained using the relevance is used to corrupt the top- and bottom- k elements with positive relevance. Here, k elements refers to the percentage of time steps in the time series that are corrupted with respect to the total number of time steps with positive relevance. Top- k elements refers to a corruption strategy that

Table 1 | Notations and symbols used in this paper

| | Symbol | Definition |
|------------------|--------------------------------------|---|
| General notation | $\bar{\cdot}$ | Corrupted value |
| | $\hat{\cdot}$ | Normalized value |
| Variables | \mathbf{X} | Sample |
| | $\bar{\mathbf{X}}$ | Corrupted sample |
| | \mathbf{R} | Attributed relevance |
| | $x_{m,t}$ | Feature m at time step t from a sample \mathbf{X} |
| | $r_{m,t}$ | Relevance for feature m at time step t from a relevance matrix \mathbf{R} |
| | $\bar{\mathbf{X}}_k^{\text{top}}$ | Sample with top- k time steps corrupted |
| | $\bar{\mathbf{X}}_k^{\text{bottom}}$ | Sample with bottom- k time steps corrupted |
| | R^+ | Set of positive attributed relevance in a given sample |
| | M | Number of features in \mathbf{X} |
| | N_c | Number of labels |
| | J | Number of samples available |
| | T | Number of ordered time steps per feature |
| | N | Number of total time steps in given sample ($M \times T$) |
| | \bar{N} | Number of time steps corrupted in a given sample |
| | \tilde{N} | Fraction of corrupted time steps in a given sample ($\frac{\bar{N}}{N}$) |
| | k | Percentage of time steps corrupted with respect to the total number of time steps with positive relevance |
| Operators | $\mathcal{S}(\cdot)$ | Post-softmax model’s output |
| | $\tilde{\mathcal{S}}(\cdot)$ | Normalized change in score (equation (5)) |
| | $\tilde{\mathcal{S}}_A(\cdot)$ | Adjusted $\tilde{\mathcal{S}}$ (equation (11)) |
| | $\mathcal{A}(\cdot)$ | Attribution scheme |
| Metrics | $\text{AUC}\tilde{S}$ | Area under the \tilde{S} curve (equation (6)) |
| | $\text{F1}\tilde{S}$ | Modified F1 score (equation (7)) |
| | TIC | Time-series information content (equation (8)) |
| | IR | Information ratio (equation (12)) |

corrupts time steps starting with higher relevance and descending to lower relevance. Similarly, bottom- k elements refers to a corruption strategy starting from time steps with lower relevance and ascending to higher relevance. We note that k is only used for calculating the number of elements to corrupt; however, the evaluation of the interpretability methods is performed with respect to the total number of elements in the sample, denoted by \bar{N} . This was performed such that the evaluation of different interpretability methods is independent of the number of time steps assigned with positive relevance, and instead is based on the total number of time steps. Figure 2 shows \tilde{S} , that is, the normalized change in score (see equation (5) in Methods) for a Transformer trained on the newly created synthetic dataset. Results for the Bi-LSTM and CNN architectures are presented in Extended Data Figs. 1 and 2. These $\tilde{S} - \bar{N}$ curves constitute the basis for computing $\text{AUC}\tilde{S}_{\text{top}}$ and $\text{F1}\tilde{S}$. The figure contains all of the six interpretability methods considered in this work, and a baseline (depicted in black). The baseline illustrates $\tilde{S} - \bar{N}$ for a random assignment of the relevance. Similar figures for the ECG dataset are presented in Extended Data Figs. 3, 4 and 5, whereas the results obtained on the FordA dataset are in

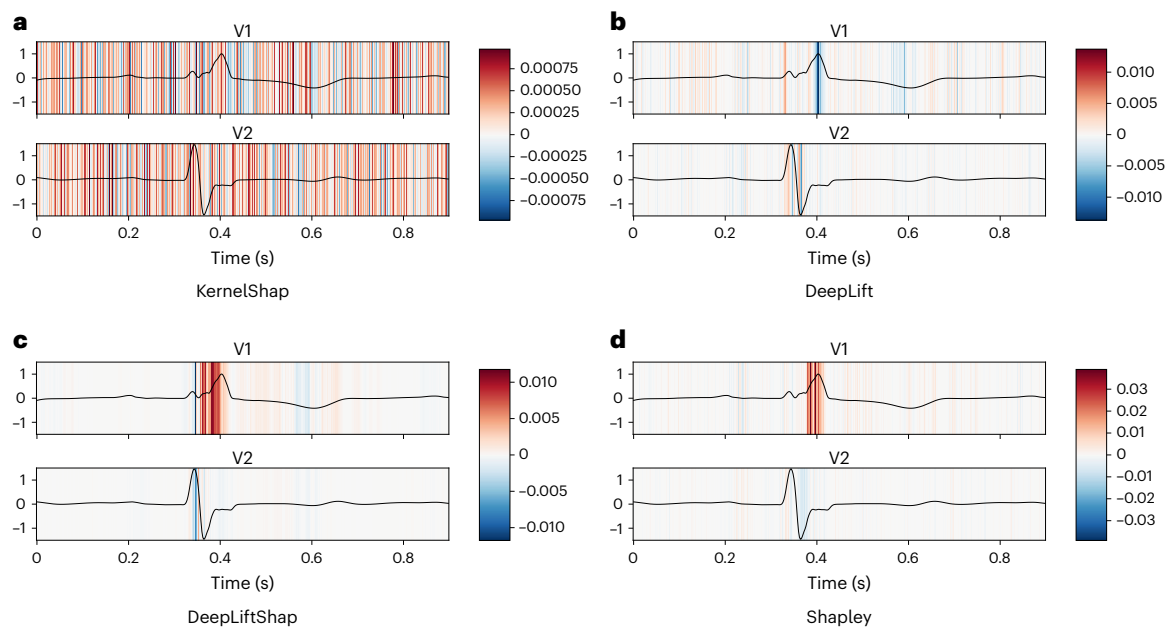


Fig. 1 | Relevance produced by four post-hoc interpretability methods. Relevance produced by four post-hoc interpretability methods, obtained on a time-series classification task, where a Transformer neural network needs to identify the pathology of a patient from ECG data. Two signals (V1 and V2) are depicted in black, and the contour maps represent the relevance produced by

the interpretability method. Red indicates positive relevance, whereas blue indicates negative relevance. The former marks portions of the time series that were deemed important by the interpretability method for the neural-network prediction, whereas the latter marks portions of the time series that were going against the prediction.

Supplementary Section 1.1. Both \tilde{S} and \tilde{N} are detailed in Table 1 and Methods. As mentioned above, the points \tilde{N} are removed in two ways: from the most important to the least important points identified by the interpretability method (top- k strategy), resulting in the top curve and from the least important to the most important points (bottom- k strategy), resulting in the bottom curve in Fig. 2.

The higher the value of $AUC\tilde{S}_{top}$, the better the interpretability method has understood which points were the most important for the model to assign the correct class. The smaller the area under the bottom curve, the better the interpretability method has understood which points were least important for the model to assign the correct class. A good trade-off between the two therefore shows that the interpretability method has identified both the most and least important points. The $F1\tilde{S}$ metric represents the harmonic mean between the capacity to extract the most and least relevant time steps. A higher score—as with the $AUC\tilde{S}_{top}$ metric—represents a better relevance identification performance.

Table 2 shows $AUC\tilde{S}_{top}$ and $F1\tilde{S}$ for all of the datasets, interpretability methods and neural-network architectures considered. Furthermore, the observed drop in accuracy for samples being progressively corrupted is presented in Extended Data Figs. 6 and 7 for the synthetic and ECG datasets, respectively, whereas the results for the FordA datasets are presented in Supplementary Section 1.2.

The relevance identification evaluated through the two metrics presented above focuses on assessing how the relevance produced by interpretability methods allows ordering of time steps to extract the most (or least) relevant time steps for a model. Another important aspect of interpretability methods is their capacity to estimate the relative effect of a given time step on the final prediction. We call this aspect relevance attribution. Developing on properties of the interpretability methods included in this work, the relevance attribution of interpretability methods are evaluated qualitatively using curves of the adjusted normalized change in score \tilde{S}_A (defined in equation (11) in Methods) versus the time-series information content (TIC) index, the latter of which measures the proportion of positive relevance contained

in the corrupted portions of the time series. Figure 3 shows \tilde{S}_A as a function of the TIC index measured on the ECG dataset, and allows a qualitative evaluation of the relevance attribution performance of interpretability methods. If a curve is above the theoretical unit linear slope (depicted as dashed black lines), the interpretability method underestimates the influence of the corrupted time steps with regard to their effect on the model's prediction. The opposite is true if the curve stands below the unit slope. The evaluation of the relevance attribution can therefore be seen as a measure of how well calibrated an interpretability method is in terms of the relevance it assigns to the different time steps with respect to their importance for the model to make its predictions. Similar figures for the synthetic and FordA datasets are presented in Extended Data Fig. 8 and Supplementary Section 1.3, respectively.

Discussion

This paper presents a new evaluation method and a set of evaluation metrics for post-hoc interpretability to answer the question posed in the introduction: which method produced an interpretability map closer to the one actually adopted by the neural network to make its prediction? The two new metrics, $AUC\tilde{S}_{top}$ and $F1\tilde{S}$, allow quantification of the relevance identification performance of an interpretability method and can be used to, for example, rank interpretability methods. These two metrics agree in identifying Shapley as the best performing method (see Table 2).

Focusing on the $AUC\tilde{S}_{top}$ values presented in Table 2, Shapley consistently outperforms the other interpretability methods across the different datasets and architectures (except for the CNN trained on the ECG dataset). The $AUC\tilde{S}_{top}$ metric reflects the capacity of Shapley to extract the most important time steps for a model prediction. Shapley is, however, the most computationally intensive interpretability method of the ones tested in this paper. It is therefore convenient to look for alternatives, which depend on the type of architecture selected. Integrated Gradients is the second best interpretability method for Bi-LSTM networks, whereas DeepLiftShap is ranked second for CNN.

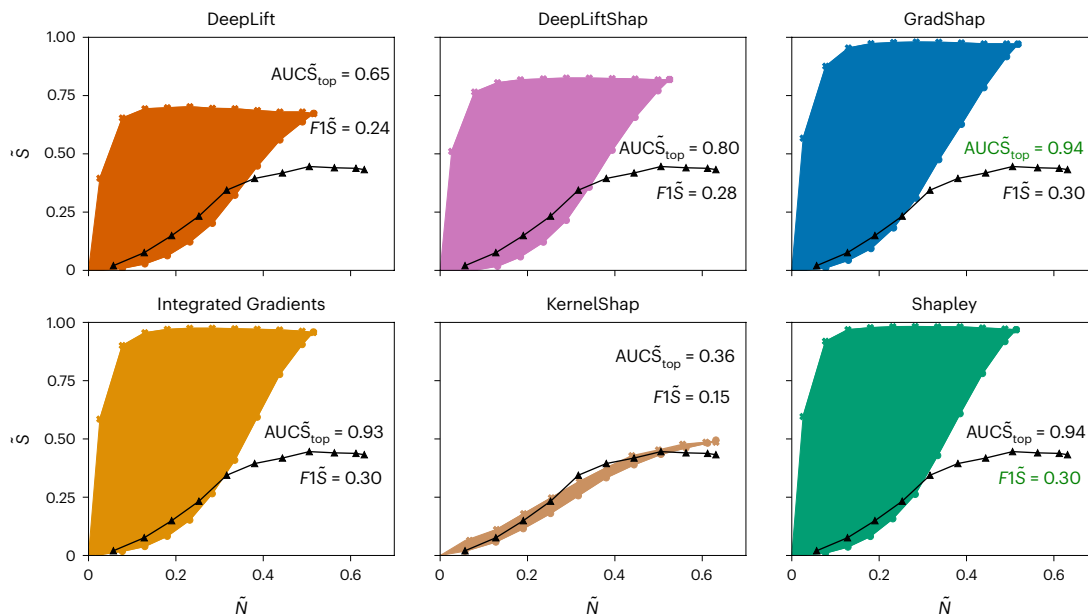


Fig. 2 | \tilde{S} as a function of the ratio of points removed with respect to \tilde{N} . Each subfigure represents one of the six interpretability methods considered for a transformer trained on the synthetic dataset.

Table 2 | $AUC\tilde{S}_{top}$ and $F1\tilde{S}$ metrics for all of the datasets, interpretability methods and neural networks considered in this work

| Network | Method | Synthetic | $AUC\tilde{S}_{top}$ | | | $F1\tilde{S}$ | | |
|--------------------|--------------|--------------|----------------------|--------------|--------------|---------------|--------------|--|
| | | | FordA | ECG | Synthetic | FordA | ECG | |
| Bi-LSTM | DeepLift | 0.381 | 0.248 | 0.165 | 0.165 | 0.105 | 0.068 | |
| | DeepLiftShap | 0.439 | 0.296 | 0.156 | 0.182 | 0.108 | 0.061 | |
| | GradShap | 0.392 | 0.283 | 0.262 | 0.172 | 0.126 | 0.110 | |
| | IG | 0.480 | 0.364 | 0.326 | 0.196 | 0.130 | 0.131 | |
| | KernelShap | 0.291 | 0.207 | 0.099 | 0.131 | 0.084 | 0.028 | |
| | Shapley | 0.554 | 0.380 | 0.348 | 0.210 | 0.173 | 0.139 | |
| | Random | 0.302 | 0.181 | 0.088 | | | | |
| CNN | DeepLift | 0.626 | 0.483 | 0.280 | 0.245 | 0.206 | 0.122 | |
| | DeepLiftShap | 0.717 | 0.487 | 0.465 | 0.284 | 0.259 | 0.200 | |
| | GradShap | 0.673 | 0.419 | 0.357 | 0.252 | 0.171 | 0.156 | |
| | IG | 0.659 | 0.485 | 0.313 | 0.246 | 0.178 | 0.134 | |
| | KernelShap | 0.335 | 0.247 | 0.025 | 0.155 | 0.102 | -0.006 | |
| | Shapley | 0.757 | 0.499 | 0.341 | 0.276 | 0.194 | 0.142 | |
| | Random | 0.312 | 0.214 | 0.047 | | | | |
| Transformer | DeepLift | 0.652 | 0.309 | 0.441 | 0.243 | 0.096 | 0.168 | |
| | DeepLiftShap | 0.800 | 0.330 | 0.539 | 0.277 | 0.107 | 0.217 | |
| | GradShap | 0.943 | 0.382 | 0.584 | 0.301 | 0.141 | 0.222 | |
| | IG | 0.929 | 0.438 | 0.581 | 0.301 | 0.170 | 0.215 | |
| | KernelShap | 0.362 | 0.272 | 0.150 | 0.156 | 0.108 | 0.051 | |
| | Shapley | 0.943 | 0.650 | 0.619 | 0.303 | 0.245 | 0.228 | |
| | Random | 0.407 | 0.248 | 0.156 | | | | |

Bold and italicized values indicate the best and worst performing interpretability methods for the given metric and dataset, respectively. IG, Integrated Gradients.

The results are slightly less clear for Transformer, where GradShap and Integrated Gradients have very similar performances.

In addition to $AUC\tilde{S}_{top}$, the $F1\tilde{S}$ metric measures the ability of different interpretability methods not only to select the most important time steps but also the least important ones. The rankings produced

using the two metrics are consistent with one another for both Transformer and Bi-LSTM, while favouring DeepLiftShap for CNN.

We addressed the issues identified in the literature to obtain reliable results for $AUC\tilde{S}_{top}$ and $F1\tilde{S}$. In particular, we evaluated the interpretability methods avoiding human judgement, and did not retrain

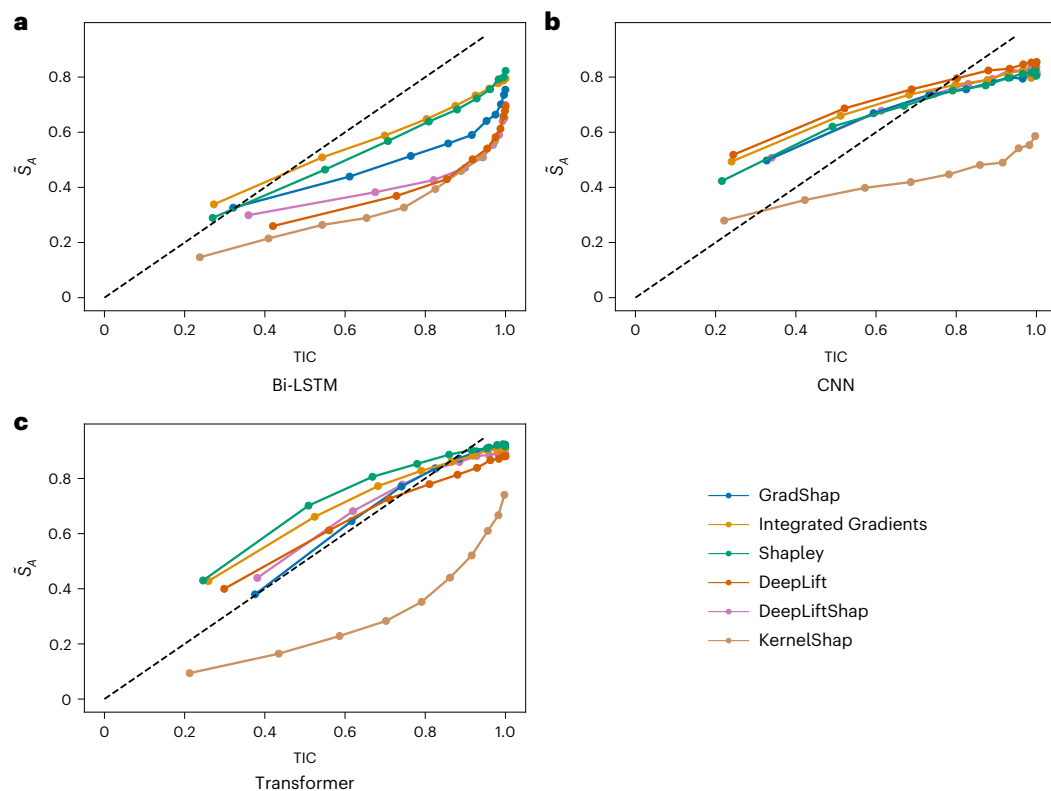


Fig. 3 | \hat{S}_A as a function of the TIC index for the six interpretability methods considered using the ECG dataset. **a–c**, Results depicted for the Bi-LSTM (**a**), CNN (**b**) and Transformer (**c**) architectures.

the model, while also avoiding a distribution shift between the training and occluded sets used to evaluate the interpretability methods. The distribution shift is one of the main concerns found in the interpretability literature. The method proposed in this paper (described in detail in Methods) addresses this issue and thus the drop in score observed in Fig. 2 as the samples are progressively corrupted cannot be attributed to a distribution shift. We also note that a larger drop in score is systematically observed when corrupting the most relevant time steps (identified by the interpretability method) as compared with corrupting a random selection of time steps (black lines in Fig. 2), as expected. The approach presented in this paper allows quantitative evaluation of interpretability methods without retraining, while avoiding a distribution shift between the training and evaluation sets.

The two metrics, $AUC\hat{S}_{top}$ and $F1\hat{S}$, along with the majority of the literature on interpretability methods evaluation, focus on relevance identification (that is, ranking time steps according to their importance). In this work we also make a first step towards evaluating relevance attribution. This evaluates how the relevance reflects the relative importance of each time step compared with the others. The attribution is qualitatively evaluated using the \hat{S}_A – TIC curves (Fig. 3). These curves provide an understanding of the ability of an interpretability method to correctly weigh relevance and are compared with a newly derived theoretical estimation (derived in Supplementary Section 4). The relevance attribution performance consistently varies between the different neural networks tested and it also changes between datasets. The common denominator is the inability of the interpretability methods to follow the theoretical estimate. This indicates that the relevance attributed to each time step does not reflect the relative importance of this time step in the classification task. The attributed relevance instead acts more as a ranking of the most important time steps among themselves. For example, a point with a relevance of 0.1 for a total classification score of 1 might not necessarily account for 10% of the final prediction, but will

be more important than a point with a relevance equal to 0.05. Albeit qualitative, these curves may be used to visually assess whether an interpretability method provides a balanced relevance.

As part of this work, we also provide a new synthetic dataset that can be used to evaluate interpretability methods (a sample of which can be found in Fig. 4). The new dataset forces the neural network to learn time dependencies as opposed to static information, and the discriminative portions of the time series are known a priori. Furthermore, the dataset is multivariate by construction, which is a desirable property especially when trying to mimic real-world (that is, non-synthetic) datasets. The performance of the interpretability methods on the new synthetic dataset is consistent with the performance obtained on the two real-world datasets tested as part of this work, namely FordA and ECG. The designed dataset hence acts as a good proxy for real-world classification tasks with two convenient properties: its complexity and properties are tuneable, and its generation is lightweight. As such, it can complement real-world datasets for a range of different research objectives within the context of evaluating post-hoc interpretability methods in time-series classification, given its known multivariate and time-dependent discriminative properties.

Finally, we assessed the usefulness of interpretability methods validated with our evaluation framework in an operational setting. In particular, we used the ECG clinical dataset because it provides a good example of how interpretability can be used once the interpretability methods have been evaluated. The use of clinical data was favoured because the healthcare sector will probably become highly regulated and thus require accurate interpretability of AI technologies¹². To this end, we interacted with clinicians to understand a common disease that is representative in the context of ECGs, and that can be of interest to them. This turned out to be the well-studied cardiac disease, right bundle branch block (RBBB). In the classification task presented in Fig. 1 for the ECG data, the neural networks were trained to classify RBBB. Shapley is

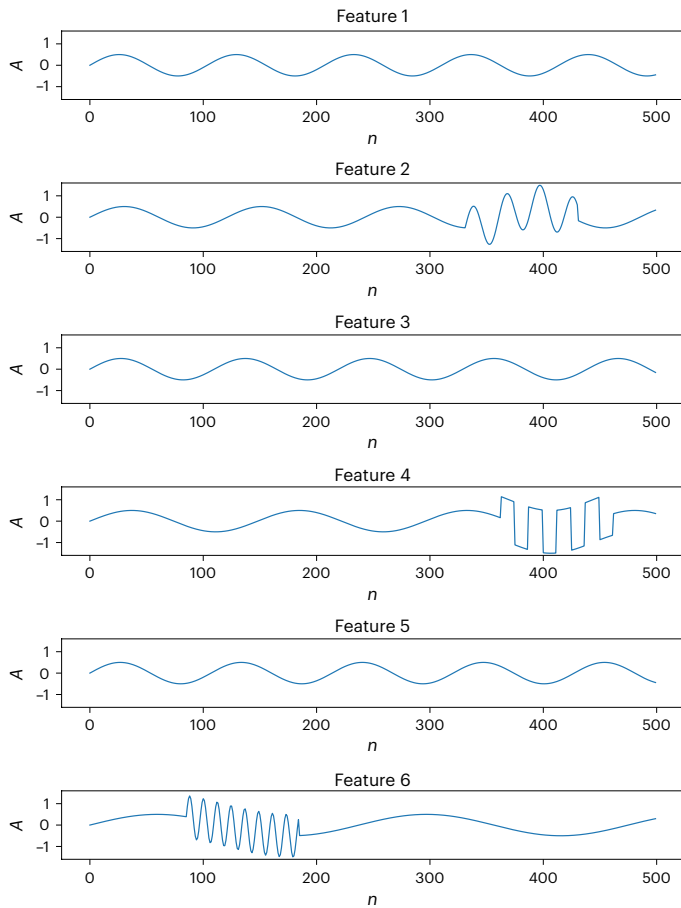


Fig. 4 | Sample from the synthetic dataset. Subfigures show the six features of the sample. The classification task for the synthetic dataset aims to classify whether the sum of the frequencies of two sine waves with a compact support of 100 time steps are above a specific threshold. These waves can be observed in the sample below in features 2 and 6.

able to pinpoint a specific and compact region of interest in the time series, whereas the other methods provide interpretability maps that are less compact (in the case of KernelShap, a sparse map without a clear region). The feature highlighted by the Shapley relevance map corresponds to one of the morphological features cardiologists look at to diagnose the disease of interest³¹. The interpretability method also shows that the trained model relies almost entirely on a single lead to predict the disease in question, namely RBBB; however, other diagnostic criteria focusing on different leads are commonly used. This type of analysis provides practical insights to understand how trained models will perform in an applied operational setting, and may help identifying possible biases, spurious correlations and potential corrective actions. Moving forwards, it is of interest to understand how these analyses could be implemented for regulatory purposes, for example, and deployed as part of AI-based technologies in new high-risk applications.

Methods

Tackling distribution shift

A long-time issue when evaluating interpretability methods has been the shift in distribution between the training and corrupted datasets used for the evaluation. Interpretability methods have been frequently evaluated, comparing the drop in score when the most relevant time steps are corrupted with the score of the initial sample. The ROAR approach was proposed to address this issue²², however, retraining the model of interest comes with its own drawbacks, as discussed in the main text. The training method presented next aims to address this issue, thereby

maintaining a constant distribution between the training dataset and the one used to evaluate the interpretability method. To achieve this task, the models presented in this paper were trained with random perturbations applied to the time series. This method was inspired by data-augmentation strategies commonly used when training models for image classification, object detection and other image-based tasks. On these tasks, random cropping has been shown to improve the classification performance of the developed model as well as its robustness³². In this work, the aim of the perturbations in the training set is not related to improving the performance of the model but to instead maintain an identical distribution between the training dataset and the corrupted samples used to evaluate the interpretability methods.

Similarly to the random cropping applied to images, part of the timeseries is corrupted by substituting the initial time steps with points drawn from a normal distribution $\sim \mathcal{N}(0, 1)$. This distribution follows the normalization applied as a preprocessing step to the samples. In a similar fashion as DropBlock³³, consecutive time steps (or blocks) are corrupted. The augmentation is applied per batch, with the overall fraction of the time series being corrupted (γ) and the size of the blocks (β) being sampled from the following uniform distributions:

$$\gamma \sim \mathcal{U}(0, 0.8) \tag{1}$$

$$\beta \sim \mathcal{U}(1, 7) \tag{2}$$

Given the method described above, when specific time steps are corrupted to evaluate the interpretability methods, the distribution is retained with the samples used when training the model; β is chosen to reflect the range of consecutive time steps above the median positive relevance empirically observed over the used datasets; γ was also empirically chosen to cover most of the samples, where the positive relevance is rarely assigned to more than 80% of the total number of time steps in a given sample. The change in score observed when corrupting time steps can therefore not be attributed to a shift in the distribution, and hence fully reflects a loss of information for the model as measured by the interpretability methods. This approach addresses the distribution shift in the evaluation of interpretability methods without requiring retraining the model. The latter point is important as it is not possible to assert that the retrained model uses the same time steps as the initial one which the interpretability methods aim to explain.

Novel approach for evaluating post-hoc interpretability methods

The time-series classification task considered in this paper can be formalized as follows (the symbols and notation adopted are also reported in Table 1). Given a trained neural-network model, f , we aim to map a set of features $\mathbf{X} \in \mathbb{R}^{M \times T}$ to a labelled target $C \in \mathbb{N}^{N_c}$ for each sample i contained in a given dataset $\mathcal{D}_i = [\mathbf{X}, C]_i$ for $i = 1, \dots, J$, where M is the number of features in \mathbf{X} , T is the number of ordered time steps per feature, N_c is the number of labels and J is the total number of samples available. Typically, a final dense layer will produce logits that are then fed to a softmax layer to output the probability of sample i to belong to a given class $C \in \mathbb{N}^{N_c}$.

To assess quantitatively time-series interpretability methods, we developed novel metrics or indices to evaluate how closely an interpretability method reflects the representation learned by the model of interest. Interpretability methods produce an attribution scheme \mathcal{A} that assigns relevance \mathbf{R} to the input \mathbf{X} for a specific class $c \in C$ such that $\mathcal{A}_c : \mathbf{X} \rightarrow \{\mathbf{R} \in \mathbb{R}^{M \times T}\}$, where $\mathbf{X} = (x_{m,t})$ and $\mathbf{R} = (r_{m,t})$, with m and t being the indices associated with M and T , respectively. For simplicity, the class that the attribution scheme aims to explain is dropped for the rest of the paper and the attribution scheme is denoted by \mathcal{A} . The new metrics are built on the relevance that an interpretability method provides along the time series. The relevance can be positive or negative (except for some interpretability methods, where it is only positive; see for

example, the saliency method³⁴). A positive relevance means that the neural network is using that portion of the time series to make its prediction. A negative relevance indicates that the neural network sees the portion of the time series as going against its prediction. As we are interested in how the network is using data to make its predictions, we use the positive relevance to build the new metrics. Logits have often been used as the input for interpretability methods^{34,35}. However, Srinivas and colleagues³⁶ demonstrate that pre-softmax outputs are related to a generative model that is uninformative of the discriminative model used for the classification task. In this sense, the evaluation of the interpretability methods for the rest of the paper are produced with the post-softmax models' output as well as evaluated with changes in these outputs when corrupting samples, denoted by $S : \mathbb{R}^{M \times T} \rightarrow [0, 1]^N$.

In this work we aim to build a framework to evaluating the performance of interpretability methods. To this end, we chose six interpretability methods that capture a broad range of methods available, while keeping the problem computationally tractable. These are: (1) DeepLift²⁷, (2) GradShap¹⁷, (3) Integrated Gradients²⁸, (4) KernelShap¹⁷, (5) DeepLiftShap¹⁷ and (6) Shapley²⁹. Their implementation uses the Captum library¹⁵.

Relevance identification and attribution

We need to tackle two aspects to assess the performance of interpretability methods: relevance identification and relevance attribution. We next detail these two concepts along with the methods developed to measure them.

Relevance identification. The concept behind relevance identification is that interpretability methods should correctly identify and order, according to their relevance, the set of points (in our case time steps) used by the model to make its predictions. Extending the assumption formulated by Shah and co-workers³⁰, time steps with larger relevance are more relevant for the model to make a prediction than the ones with smaller relevance. The relevance produced by an attribution scheme can be used to create an ordering that ranks feature m at time-step t (namely $x_{m,t}$) from a sample \mathbf{X} according to its importance for the model's prediction. It is important to remember that here we only focus on the positive relevance. The ordering can then be used to define $\mathbf{X}_k^{\text{top}}$ and $\mathbf{X}_k^{\text{bottom}}$, which represent the samples with top- and bottom- k time steps corrupted, respectively. These are ordered using the assigned relevance, corrupted as follows:

$$\mathbf{X}_k^{\text{top}} = \begin{cases} x_{m,t} & \text{if } r_{m,t} < Q_{R^+}(1 - k) \\ \mathcal{N}(0, 1) & \text{otherwise} \end{cases} \quad (3)$$

$$\mathbf{X}_k^{\text{bottom}} = \begin{cases} x_{m,t} & \text{if } r_{m,t} > Q_{R^+}(k) \\ \mathcal{N}(0, 1) & \text{otherwise} \end{cases} \quad (4)$$

where $Q_{R^+}(p)$ denotes the p -quantile (where $p = 1 - k$ for $\mathbf{X}_k^{\text{top}}$ and $p = k$ for $\mathbf{X}_k^{\text{bottom}}$) of the set of positive attributed relevance in a given sample R^+ , measured over sample \mathbf{X} using attribution scheme \mathcal{A} , with $R^+ = \{r_{m,t} | r_{m,t} > 0\}$. The rest of the analysis is performed for the following set of top- and bottom- k percentage of time steps with positive relevance: $k \in [0.05, 0.15, 0.25, \dots, 0.95, 1]$.

In the general case of a modified sample $\tilde{\mathbf{X}}$, where \tilde{N} points along the time series are corrupted, we can define the normalized difference in score:

$$\tilde{S}(\tilde{\mathbf{X}}) = \frac{S(\tilde{\mathbf{X}}) - S(\mathbf{X})}{S(\mathbf{X})} \quad (5)$$

It is possible to build \tilde{S} versus \tilde{N} curves for top- or bottom- k points (or time steps) corrupted, where $\tilde{N} = \frac{\tilde{N}}{N}$ is the fraction of points removed

with respect to the total number of time steps $N = M \times T$ present in the time series. The area under the $\tilde{S} - \tilde{N}$ curve is denoted as:

$$\text{AUC}\tilde{S} = \int_0^1 \tilde{S} \, d\tilde{N} \quad (6)$$

Using equation (6), we can define $\text{AUC}\tilde{S}_{\text{top}}$ and $\text{F1}\tilde{S}$. The former aims to evaluate the ability of an interpretability method to recover the most important time steps for a model's prediction. In this sense, the area under the drop in score when the top- k time steps are progressively corrupted should be maximized. To normalize for the interpretability methods assigning a different number of time steps with positive relevance, the $\text{AUC}\tilde{S}_{\text{top}}$ is measured on a modified $\text{AUC}\tilde{S}$ curve. This modified curve is created by adding an extra point with coordinates $(\tilde{N} = 1; \tilde{S} = \tilde{S}(\mathbf{X}_{k=1}^{\text{top}}))$. Adding this point allows favouring of interpretability methods that are able to achieve a large drop in score with a minimal number of time steps assigned with positive relevance.

The $\text{F1}\tilde{S}$ aims to build an harmonic mean between the ability of an attribution scheme to correctly rank the time steps with the highest and smallest relevance, respectively. Corrupting time steps with high relevance should result in a substantial drop in score (the model's outputs are greatly affected). Corrupting time steps with small relevance should result in a negligible drop in score (the model's outputs are negligibly affected). The best attribution scheme should have maximized $\text{AUC}\tilde{S}_{\text{top}}$ for top- k points corruptions, and minimized $\text{AUC}\tilde{S}_{\text{bottom}}$ for bottom- k points corruptions. Regarding this desired property, we can define the following F1 score:

$$\text{F1}\tilde{S} = \frac{\text{AUC}\tilde{S}_{\text{top}}(1 - \text{AUC}\tilde{S}_{\text{bottom}})}{\text{AUC}\tilde{S}_{\text{top}} + (1 - \text{AUC}\tilde{S}_{\text{bottom}})} \quad (7)$$

Relevance attribution. The idea behind relevance attribution is that the relevance should not only serve to order the time steps but also reflect the individual contribution of each time step relative to the others towards the model's predicted score. All interpretability methods presented in this work are additive feature attribution methods, as defined by Lundberg and colleagues¹⁷. The produced relevance therefore aims to linearly reflect the effect of each feature on the model's outputs. Focusing on the positive relevance, the difference between the model's prediction on an initial given sample $S(\mathbf{X})$ and a version with the top- k points corrupted $S(\mathbf{X}_k^{\text{top}})$ is dependent on the positive relevance corrupted between the two samples. The proportion of relevance attributed to corrupted time steps to the initial one is summarized with the time information content index defined as:

$$\text{TIC}(k) = \frac{\sum_{R^+} r_{m,t}}{\sum_{R^+} r_{m,t} + \epsilon} \quad (8)$$

where the following sets are defined:

$$R^+ = \{r_{m,t} | r_{m,t} > 0\} \quad (9)$$

$$R_k^+ = \{r_{m,t} | r_{m,t} \in R^+ \cap r_{m,t} \geq Q_R(1 - k)\} \quad (10)$$

The TIC index reflects the ratio of the relevance attributed to the top- k set of points to the total positive relevance. Taking the model's output when all of the positive relevance is corrupted as a reference, we can normalize the change in score as follows:

$$\tilde{S}_A(k) = \frac{S(\mathbf{X}) - S(\mathbf{X}_k^{\text{top}})}{S(\mathbf{X}) - S(\mathbf{X}_{k=1}^{\text{top}})} \quad (11)$$

where $S(\bar{\mathbf{X}}_{k=1}^{\text{top}})$ corresponds to the model's output when all time steps with positive relevance of sample \mathbf{X} is corrupted. We name the quantity in equation (11) adjusted normalized drop in score. Given the linear additivity property of the relevance, the index $\bar{S}_A(k)$ should be equal to the TIC(k) index (see Supplementary Section 4) such that the information ratio satisfies

$$\text{IR} = \frac{\bar{S}_A(k)}{\text{TIC}(k)} = 1 \quad (12)$$

Given this theoretical approximation, it is possible to evaluate how different interpretability methods over- or underestimate the role of different time steps in the model's prediction. An information ratio larger than one will indicate the relevance of the points under the quantile of interest was underestimated while the opposite is true for an information ratio smaller than one. An example of $\bar{S}_A(k) - \text{TIC}(k)$ curve is depicted in Fig. 3 where we report the results for every interpretability method considered as well as the theoretical linear line (dashed).

Datasets

The new interpretability evaluation approach has been applied to: a new synthetic dataset created for this work, a standard univariate dataset for anomaly classification and a biomedical dataset based on ECG signals. The three datasets are described below.

A new synthetic dataset

The evaluation of interpretability methods for time-series classification has been lacking a dataset: (1) where the discriminative features are known and (2) that replicates the complexity of common time-series classification tasks with time dependencies across features. The developed dataset is inspired by the BlockMNIST dataset, which is derived from the MNIST dataset³⁰. Each sample in the dataset comprises six features, each with 500 time steps corresponding to $\Delta t = 2$ ms. Each feature comprises a sine wave with its amplitude multiplied by 0.5 and frequency $\sim \mathcal{U}(2, 5)$, which serves as a random baseline. In two of the features—picked randomly for each sample—sine waves with a support of 100 time steps are added to the baseline at a random position in time.

The respective frequency of sine waves f_1 and f_2 are drawn from a discrete uniform distribution $\sim \mathcal{U}(10, 50)$. In the remaining four features, a square wave is included with a probability of 0.5 and frequency $\sim \mathcal{U}(10, 50)$.

The classification task then consists of predicting whether the sum of the two frequencies (f_1 and f_2) is above or below a given threshold τ . For the presented task, τ was set to 60 to balance the classes of the classification target y such that:

$$y = \begin{cases} 0, & \text{if } f_1 + f_2 < \tau \\ 1, & \text{otherwise} \end{cases} \quad (13)$$

The main idea behind the developed dataset is to force the network to learn temporal dependencies, that is, the frequency of the sine wave with closed support, as well as dependencies across features, the sum of the frequencies (f_1 and f_2). An example of a generated sample is presented in Fig. 4. The closed support sine waves used to create the classification target are observed in features 2 and 6. We note that the synthetic dataset proposed here can be regarded as a family of datasets, as the number of features, length of time series, class imbalance and discriminative features are tunable.

FordA

The FordA dataset is part of the UCR Time Series Classification Archive, which aims to group different dataset for time series classification³⁷. FordA is a univariate and binary classification task. The data originate from an automotive subsystem and the classification task aims to find samples with a specific anomaly. The dataset comes with a training ($n = 3,601$) and testing split ($n = 1,320$), which was retained in this paper. The dataset is

of interest as it has often served as a benchmark for classification algorithms³⁸ as well as for benchmarking interpretability methods²⁴.

ECG dataset

To mimic a real-world classification task, we applied the interpretability framework to an ECG dataset. Electrocardiogram records the electrical activity of the heart and typically produces twelve signals, corresponding to twelve sensors or leads. For this task, a subset of the Classification of twelve-lead ECGs (The PhysioNet—Computing in Cardiology Challenge 2020³⁹, published under Creative Commons Attribution 4.0 License) was used. The dataset was narrowed down to the CPSC subset⁴⁰, which included 6,877 ECGs annotated for nine cardiovascular diseases. As part of these annotations, it was chosen to classify the ECGs for the presence/absence of a Right Bundle Branch Block (RBBB). The dataset includes 5,020 cases showing no sign of a RBBB and 1,857 cases annotated as carrying a RBBB; RBBB was found to be associated with higher cardiovascular risks as well as mortality⁴¹.

The data were first denoised using different techniques for low and high-frequency artifacts. The baseline wander as well as low-frequencies artifacts were first removed by performing Empirical mode decomposition (EMD). The instantaneous frequency is computed and averaged across each intrinsic mode resulting from EMD to obtain an average frequency of the modes. Modes with an average frequency below 0.7 Hz are then discarded and the signal is reconstructed with the remaining modes. The threshold is a parameter based on the literature where the thresholds range between 0.5 and 1 Hz (refs. 42–44). Given the difficulty to separate high-frequency noises using EMD, power-line and others high-frequency noises are removed by thresholding the wavelet transform coefficients using the 'universal threshold'⁴⁵.

To obtain an average beat, the R-peaks of each ECG are extracted using the BioSPPy library⁴⁶. The beats centred around the R-peaks are then extracted from the ECG by taking 0.35 before and 0.55 s after the R-peak. The mean of the extracted beat is then computed to obtain an average beat of each lead. An example of the initial signal along the transformed one for a subset of the twelve leads is presented in Extended Data Figs. 9 and 10. The average beat was computed in each lead. The resulting modified twelve leads were used to train the model.

Baseline for interpretability methods

The interpretability methods used as part of this work require setting an uninformative baseline as a reference. Most methods (Integrated Gradients, DeepLift, Shapley, KernelShap) require a single sample set as the baseline. For those methods, the baseline was set as the mean taken across samples for each time step. GradShap and DeepLiftShap uses a distribution of baseline and this baseline was constructed by taking 50 random samples from the test set.

Reporting summary

Further information on research design is available in the Nature Portfolio Reporting Summary linked to this article.

Data availability

All datasets and trained models used in this paper have been made available on Zenodo (<https://zenodo.org/record/7534770#Y8lkkXbMI2w>)⁴⁷. The ECG dataset is based on the public dataset released as part of The PhysioNet/Computing in Cardiology Challenge 2020³⁹ available under the following <https://doi.org/10.13026/f4ab-0814>. The FordA dataset comes from the UEA & UCR Time Series Classification Repository³⁷. The synthetic dataset used as part of this study can be generated using the code shared on github: <https://github.com/hturbe/InterpretTime>.

Code availability

The full code used to perform the analysis is available at <https://github.com/hturbe/InterpretTime>. The specific version of the code used to generate the results presented in this article is archived in Zenodo⁴⁸.

References

1. Weyn, J. A., Durran, D. R. & Caruana, R. Improving data driven global weather prediction using deep convolutional neural networks on a cubed sphere. *J. Adv. Modell. Earth Syst.* Sep **12**, e2020MS002109 (2020).
2. Yang, R. et al. Big data analytics for financial Market volatility forecast based on support vector machine. *Int. J. Inf. Manage.* **50**, 452–462 (2020).
3. Rajkumar, A. et al. Scalable and accurate deep learning with electronic health records. *NPJ Digit. Med.* **1**, 1–10 (2018).
4. Dau, H. A. et al. The UCR time series archive. *IEEE/CAA J. Autom. Sin.* **6**, 1293–1305 (2019).
5. Manibardo, E. L., Laña, I. & Del Ser, J. Deep learning for road traffic forecasting: does it make a difference? *IEEE Trans. Intell. Transp. Syst.* **23**, 6164–6188 (2021).
6. Ye, L. & Keogh, E. Time series shapelets: a new primitive for data mining. In *Proc. 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* 947–956 (ACM, 2009).
7. Hewamalage, H., Bergmeir, C. & Bandara, K. Recurrent neural networks for time series forecasting: current status and future directions. *Int. J. Forecast.* **37**, 388–427 (2021).
8. Lim, B., Arık, S. Ö., Loeff, N. & Pfister, T. Temporal fusion transformers for interpretable multi-horizon time series forecasting. *Int. J. Forecast.* **37**, 1748–1764 (2021).
9. Tang, B. & Matteson, D. S. Probabilistic transformer for time series analysis. In *Advances in Neural Information Processing Systems* Vol. 34, 23592–24608 (NeurIPS, 2021).
10. Fawaz, H. I., Forestier, G., Weber, J., Idoumghar, L. & Muller, P. A. Deep learning for time series classification: a review. *Data Min. Knowl. Discov.* **33**, 917–963 (2019).
11. Hong, S., Zhang, W., Sun, C., Zhou, Y. & Li, H. Practical lessons on 12-lead ECG classification: meta-analysis of methods from PhysioNet/computing in cardiology challenge 2020. *Front. Physiol.* <https://doi.org/10.3389/fphys.2021.811661> (2022).
12. *Proposal for a Regulation of the European Parliament and of the Council Laying Down Harmonised Rules on Artificial Intelligence (Artificial Intelligence Act) and Amending Certain Union Legislative Acts COM/2021/206 final* (European Commission, Directorate-General for Communications Networks, Content and Technology, 2021); <https://eur-lex.europa.eu/legal-content/EN/ALL/?uri=CELEX:52021PC0206>
13. Lipton, Z. C. The mythos of model interpretability: in machine learning, the concept of interpretability is both important and slippery. *Queue* **16**, 31–57 (2018).
14. Shad, R., Cunningham, J. P., Ashley, E. A., Langlotz, C. P. & Hiesinger, W. Designing clinically translatable artificial intelligence systems for high-dimensional medical imaging. *Nat. Mach. Intell.* **3**, 929–935 (2021).
15. Kokhlikyan, N. et al. Captum: a unified and generic model interpretability library for PyTorch. Preprint at <https://arxiv.org/abs/2009.07896> (2020).
16. Montavon, G., Bach, S., Binder, A., Samek, W. & Müller, K. R. Explaining nonlinear classification decisions with deep Taylor decomposition. *Pattern Recogn.* **65**, 211–222 (2017).
17. Lundberg, S. & Lee, S.-I. A unified approach to interpreting model predictions. In *Advances in Neural Information Processing Systems* (eds Guyon, I. et al.) Vol. 30 (NeurIPS, 2017).
18. Neves, I. et al. Interpretable heartbeat classification using local model-agnostic explanations on ECGs. *Comput. Biol. Med.* **133**, 104393 (2021).
19. Jacovi, A. & Goldberg, Y. Towards faithfully interpretable NLP systems: How should we define and evaluate faithfulness? In *Proc. of the 58th Annual Meeting of the Association for Computational Linguistics* 4198–4205 (Association for Computational Linguistics, 2020).
20. Adebayo, J. et al. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems* Vol. 31 (2018).
21. Samek, W., Binder, A., Montavon, G., Lapuschkin, S. & Müller, K. R. Evaluating the visualization of what a deep neural network has learned. *IEEE Trans. Neural Netw. Learn. Syst.* **28**, 2660–2673 (2016).
22. Hooker, S., Erhan, D., Kindermans, P. J. & Kim, B. A benchmark for interpretability methods in deep neural networks. In *Advances in Neural Information Processing Systems* Vol. 32 (NeurIPS, 2019).
23. Hacoheh, G., Choshen, L. & Weinshall, D. Let's agree to agree: neural networks share classification order on real datasets. In *International Conference on Machine Learning* 3950–3960 (PMLR, 2020).
24. Schlegel, U., Arnout, H., El-Assady, M., Oelke, D. & Keim, D. A. Towards a rigorous evaluation of XAI methods on time series. In *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)* 4197–4201 (IEEE, 2019); <https://doi.org/10.1109/ICCVW.2019.00516>
25. Ismail, A. A., Gunady, M., Corrada Bravo, H. & Feizi, S. Benchmarking deep learning interpretability in time series predictions. In *Advances in Neural Information Processing Systems* Vol. 33, 6441–6452 (2020).
26. Liu, B., Wang, X., Dixit, M., Kwitt, R. & Vasconcelos, N. Feature space transfer for data augmentation. In *Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* 9090–9098 (IEEE, 2018).
27. Shrikumar, A., Greenside, P. & Kundaje, A. PMLR. Learning important features through propagating activation differences. In *International Conference on Machine Learning* 3145–3153 (ICML, 2017).
28. Sundararajan, M., Taly, A. & Yan, Q. Axiomatic attribution for deep networks. In *International Conference on Machine Learning* 3319–3328 (PMLR, 2017).
29. Castro, J., Gómez, D. & Tejada, J. Polynomial calculation of the Shapley value based on sampling. *Comput. Oper. Res.* **36**, 1726–1730 (2009).
30. Shah, H., Jain, P. & Netrapalli, P. Do input gradients highlight discriminative features? In *Advances in Neural Information Processing Systems* Vol. 34, 2046–2059 (NeurIPS, 2021).
31. Surawicz, B., Childers, R., Deal, B. J. & Gettes, L. S. AHA/ACCF/HRS recommendations for the standardization and interpretation of the electrocardiogram: part III: intraventricular conduction disturbances: a scientific statement from the American Heart Association Electrocardiography and Arrhythmias Committee, Council on Clinical Cardiology; the American College of Cardiology Foundation; and the Heart Rhythm Society Endorsed by the International Society for Computerized Electrocardiology. *J. Am. College Cardiol.* **53**, 976–981 (2009).
32. Cubuk, E. D., Zoph, B., Shlens, J. & Le, Q. RandAugment: practical automated data augmentation with a reduced search space. In *Advances in Neural Information Processing Systems* (eds Larochelle H. et al.) Vol. 33, 18613–18624 (NeurIPS, 2020).
33. Ghiasi, G., Lin, T. Y. & Le, Q. V. Dropblock: a regularization method for convolutional networks. In *Advances in Neural Information Processing Systems* Vol. 31 (NeurIPS, 2018).
34. Simonyan, K., Vedaldi, A. & Zisserman, A. Deep inside convolutional networks: visualising image classification models and saliency maps. In *Proc. of the 2nd International Conference on Learning Representations* (eds Bengio, Y. & LeCun, Y.) (ICLR, 2014).
35. Selvaraju, R. R. et al. Grad-CAM: visual explanations from deep networks via gradient-based localization. In *2017 IEEE International Conference on Computer Vision (ICCV)* 618–626 (IEEE, 2017).

36. Srinivas, S. & Fleuret, F. Rethinking the role of gradient-based attribution methods for model interpretability. In *2021 International Conference on Learning Representations (ICLR, 2021)*.
37. Bagnall, A., Lines, J., Bostrom, A., Large, J. & Keogh, E. The Great Time Series Classification Bake Off: a review and experimental evaluation of recent algorithmic advances. *Data Min. Knowl. Discov.* **31**, 606–660 (2017).
38. Yang, C. H. H., Tsai, Y. Y. & Chen, P. Y. Voice2Series: Reprogramming acoustic models for time series classification. In *Proc. 38th International Conference on Machine Learning (eds Meila M. & Zhang, T.) Vol. 139, 11808–11819 (PMLR, 2021)*; <https://proceedings.mlr.press/v139/yang21j.html>
39. Perez Alday, E. A. et al. *Classification of 12-lead ECGs: The PhysioNet/Computing in Cardiology Challenge 2020* (PhysioNet, 2022); <https://physionet.org/content/challenge-2020/1.0.2/>
40. Liu, F. et al. An open access database for evaluating the algorithms of electrocardiogram rhythm and morphology abnormality detection. *J. Med. Imaging Health Inform.* **8**, 1368–1373 (2018).
41. Bussink, B. E. et al. Right bundle branch block: prevalence, risk factors, and outcome in the general population: results from the Copenhagen City Heart Study. *European Heart J.* **34**, 138–146 (2012).
42. Thakor, N. V. & Zhu, Y. S. Applications of adaptive filtering to ECG analysis: noise cancellation and arrhythmia detection. *IEEE Trans. Biomed. Eng.* **38**, 785–794 (1991).
43. Van Alste, J. A. & Schilder, T. S. Removal of base-line wander and power-line interference from the ECG by an efficient FIR filter with a reduced number of taps. *IEEE Trans. Biomed. Eng.* **BME-32**, 1052–1060 (1985).
44. van Alsté, J. A., van Eck, W. & Herrmann, O. E. ECG baseline wander reduction using linear phase filters. *Comput. Biomed. Res.* **19**, 417–427 (1986).
45. Donoho, D. L. & Johnstone, I. M. Ideal spatial adaptation by wavelet shrinkage. *Biometrika* **81**, 425–455 (1994).
46. Carreiras, C. et al. *BioSPPy: Biosignal Processing in Python* (GitHub, 2018); <https://github.com/PIA-Group/BioSPPy/>
47. Turbé, H., Bjelogrić, M., Lovis, C. & Mengaldo, G. *Dataset: Evaluation of Post-Hoc Interpretability Methods in Time-Series Classification* (Zenodo, 2023); <https://doi.org/10.5281/zenodo.7534770>
48. Turbé, H., Bjelogrić, M., Lovis, C., Mengaldo, G. *hturbe/InterpretTime: Initial Release to Replicate Results of the Submitted Article* (Zenodo, 2023); <https://doi.org/10.5281/zenodo.7560836>

Acknowledgements

G.M acknowledges Singapore's Ministry of Education support through MOE Tier 1 grant 22-4900-A0001-0. We also thank A. Gualandi for fruitful discussions and precious feedback he provided as part of

this research. We thank the anonymous reviewers for their insightful comments, which helped considerably improve the paper.

Author contributions

H.T. conceived the initial research idea with input from all of the authors. M.B., G.M. and H.T. performed the experiments and data analysis. M.B., G.M. and H.T. wrote the paper with input from all of the authors. C.L. and G.M. supervised the project.

Funding

Open access funding provided by University of Geneva.

Competing interests

The authors declare no competing interests.

Additional information

Extended data is available for this paper at <https://doi.org/10.1038/s42256-023-00620-w>.

Supplementary information The online version contains supplementary material available at <https://doi.org/10.1038/s42256-023-00620-w>.

Correspondence and requests for materials should be addressed to Hugues Turbé or Gianmarco Mengaldo.

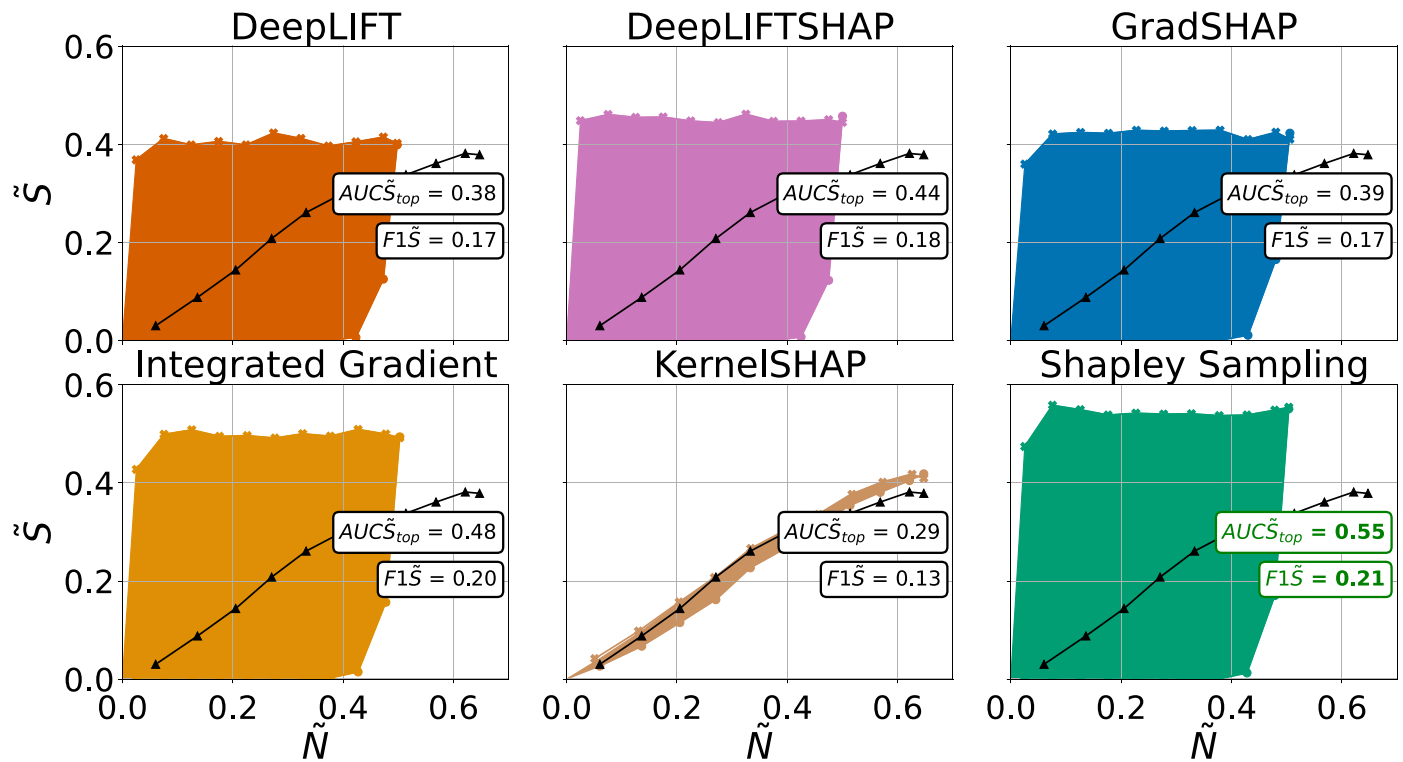
Peer review information *Nature Machine Intelligence* thanks Massimo Rivolta, Eamonn Keogh and the other, anonymous, reviewer(s) for their contribution to the peer review of this work.

Reprints and permissions information is available at www.nature.com/reprints.

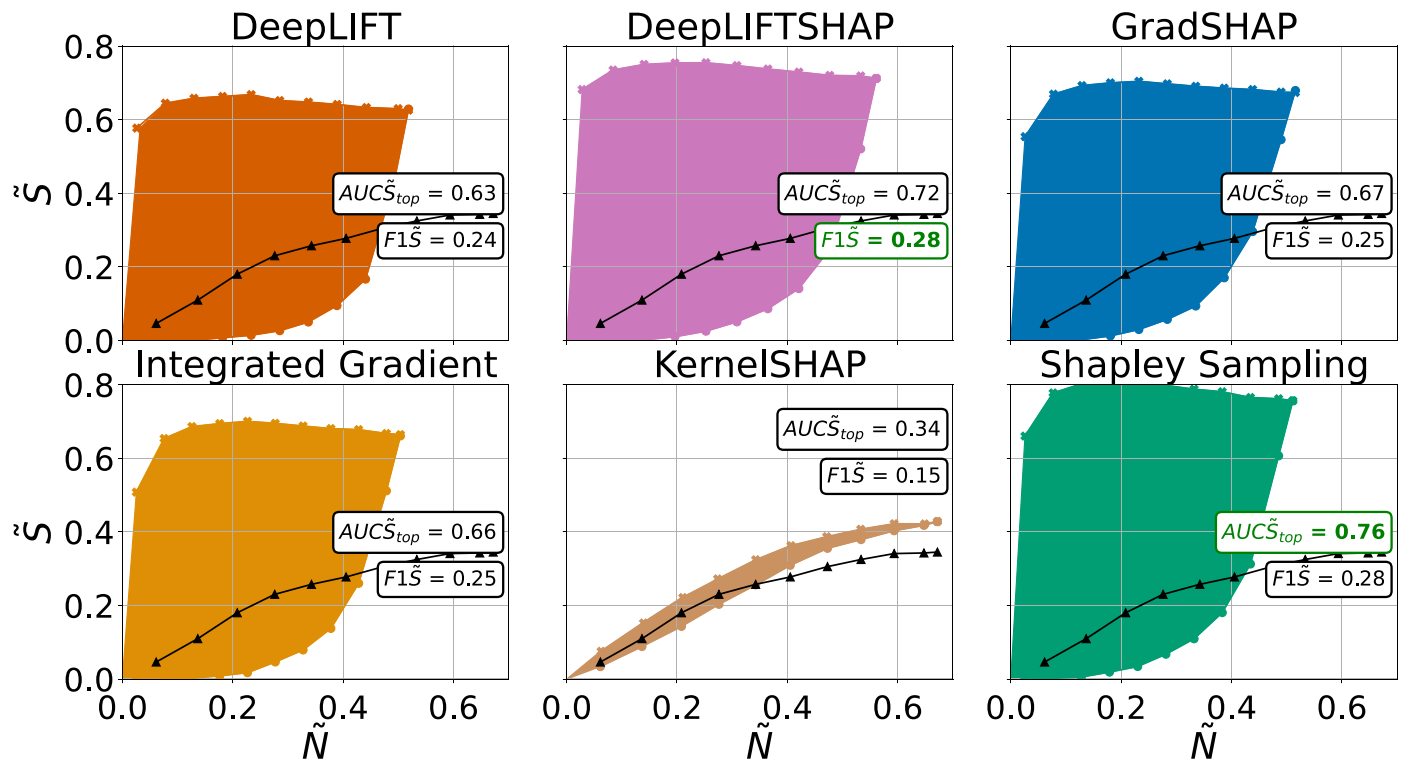
Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Open Access This article is licensed under a Creative Commons Attribution 4.0 International License, which permits use, sharing, adaptation, distribution and reproduction in any medium or format, as long as you give appropriate credit to the original author(s) and the source, provide a link to the Creative Commons license, and indicate if changes were made. The images or other third party material in this article are included in the article's Creative Commons license, unless indicated otherwise in a credit line to the material. If material is not included in the article's Creative Commons license and your intended use is not permitted by statutory regulation or exceeds the permitted use, you will need to obtain permission directly from the copyright holder. To view a copy of this license, visit <http://creativecommons.org/licenses/by/4.0/>.

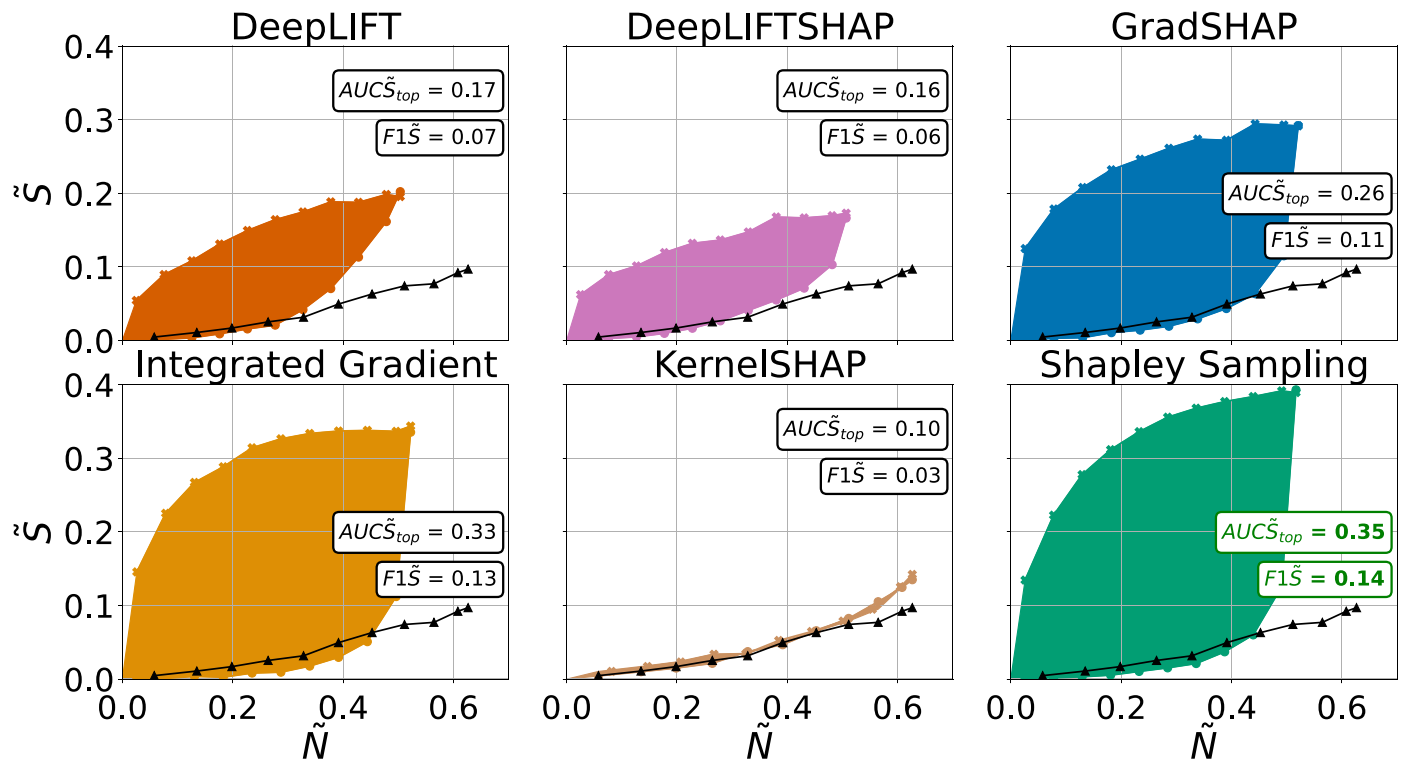
© The Author(s) 2023



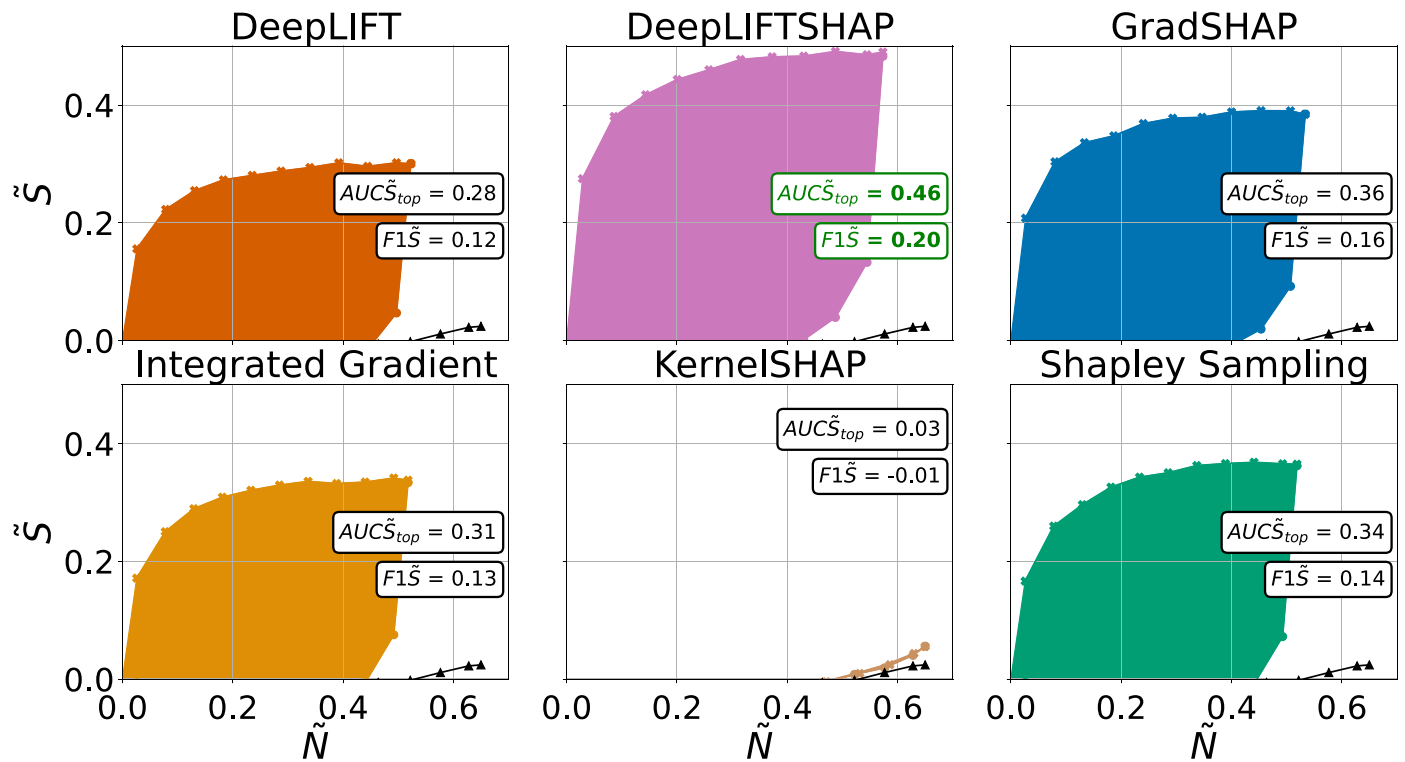
Extended Data Fig. 1 | \tilde{S} as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} . Each subfigure represents one of the six interpretability methods considered for a Bi-LSTM trained on the synthetic dataset.



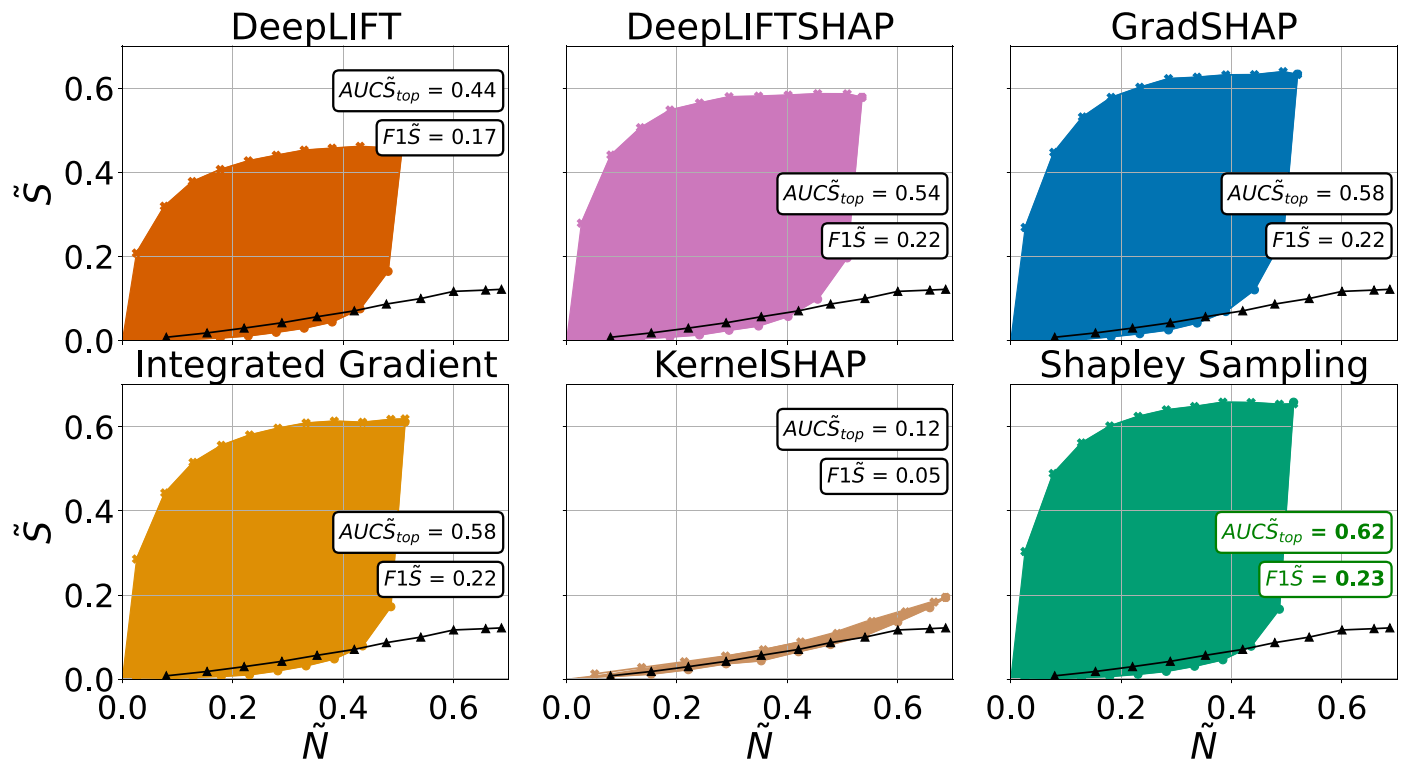
Extended Data Fig. 2 | \tilde{S} as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} . Each subfigure represents one of the six interpretability methods considered for a CNN trained on the synthetic dataset.



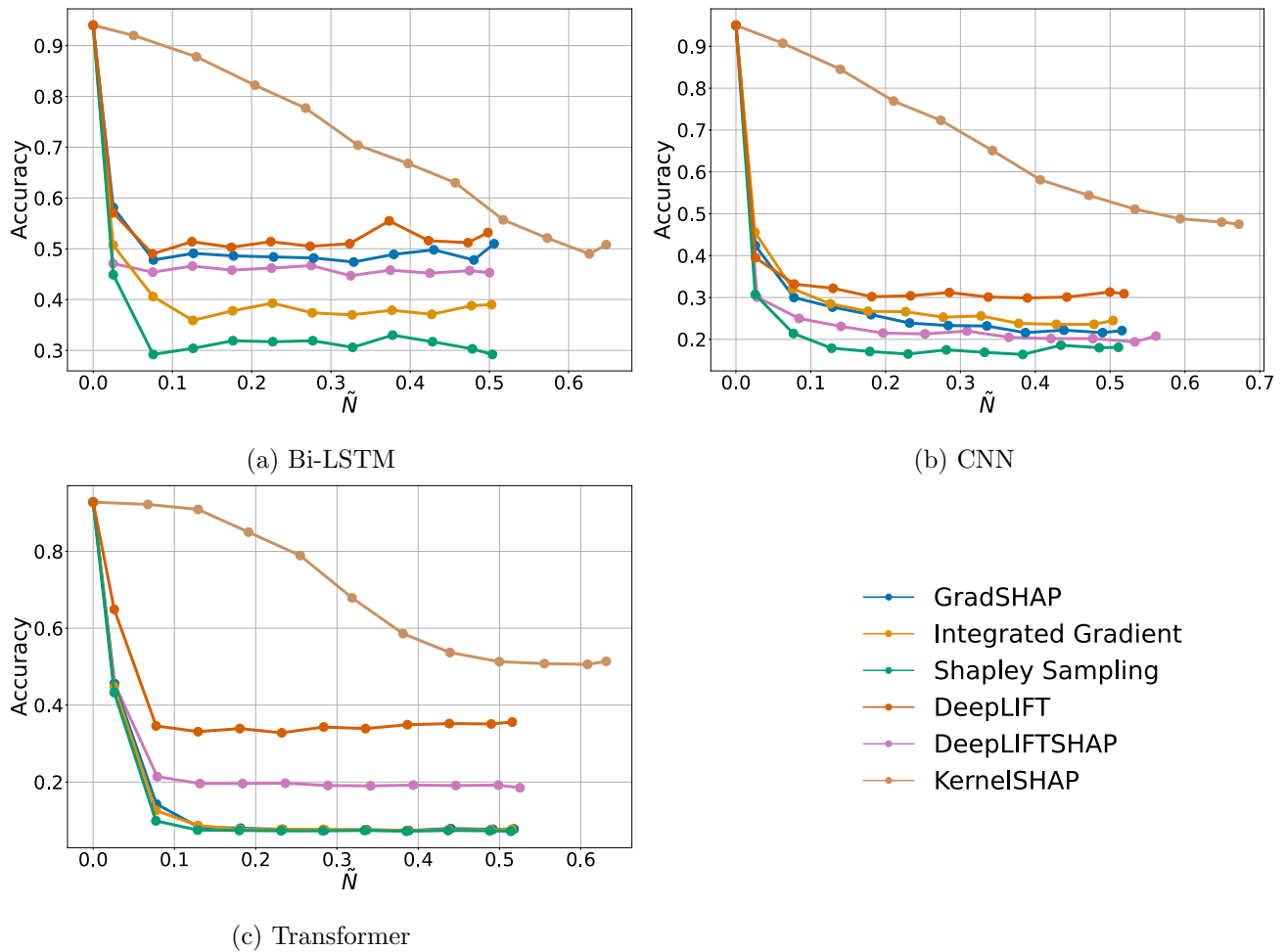
Extended Data Fig. 3 | \tilde{S} as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} . Each subfigure represents one of the six interpretability methods considered for a Bi-LSTM trained on the ECG dataset.



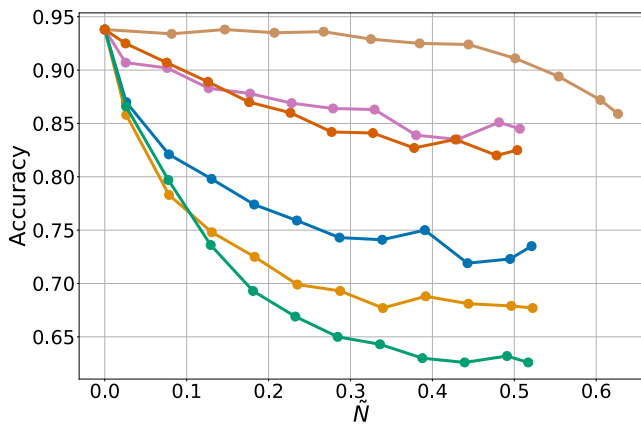
Extended Data Fig. 4 | \tilde{S} as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} . Each subfigure represents one of the six interpretability methods considered for a CNN trained on the ECG dataset.



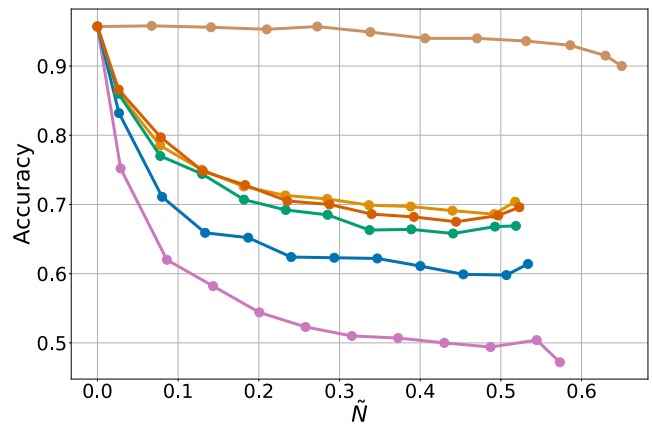
Extended Data Fig. 5 | \tilde{S} as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} . Each subfigure represents one of the six interpretability methods considered for a Transformer trained on the ECG dataset.



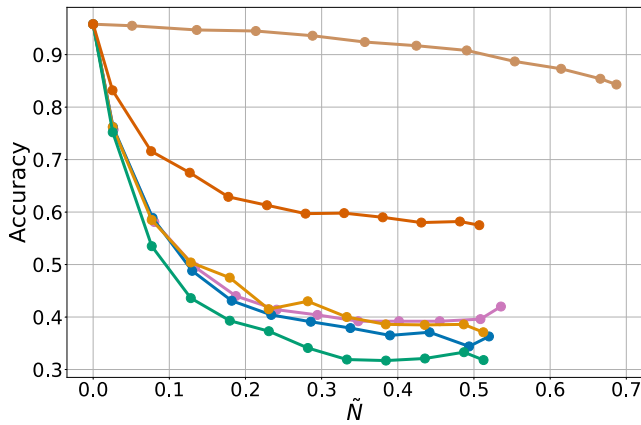
Extended Data Fig. 6 | Change in accuracy as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} for the six interpretability methods considered using the synthetic dataset. Results depicted for (a) Bi-LSTM, (b) CNN and (c) Transformer.



(a) Bi-LSTM



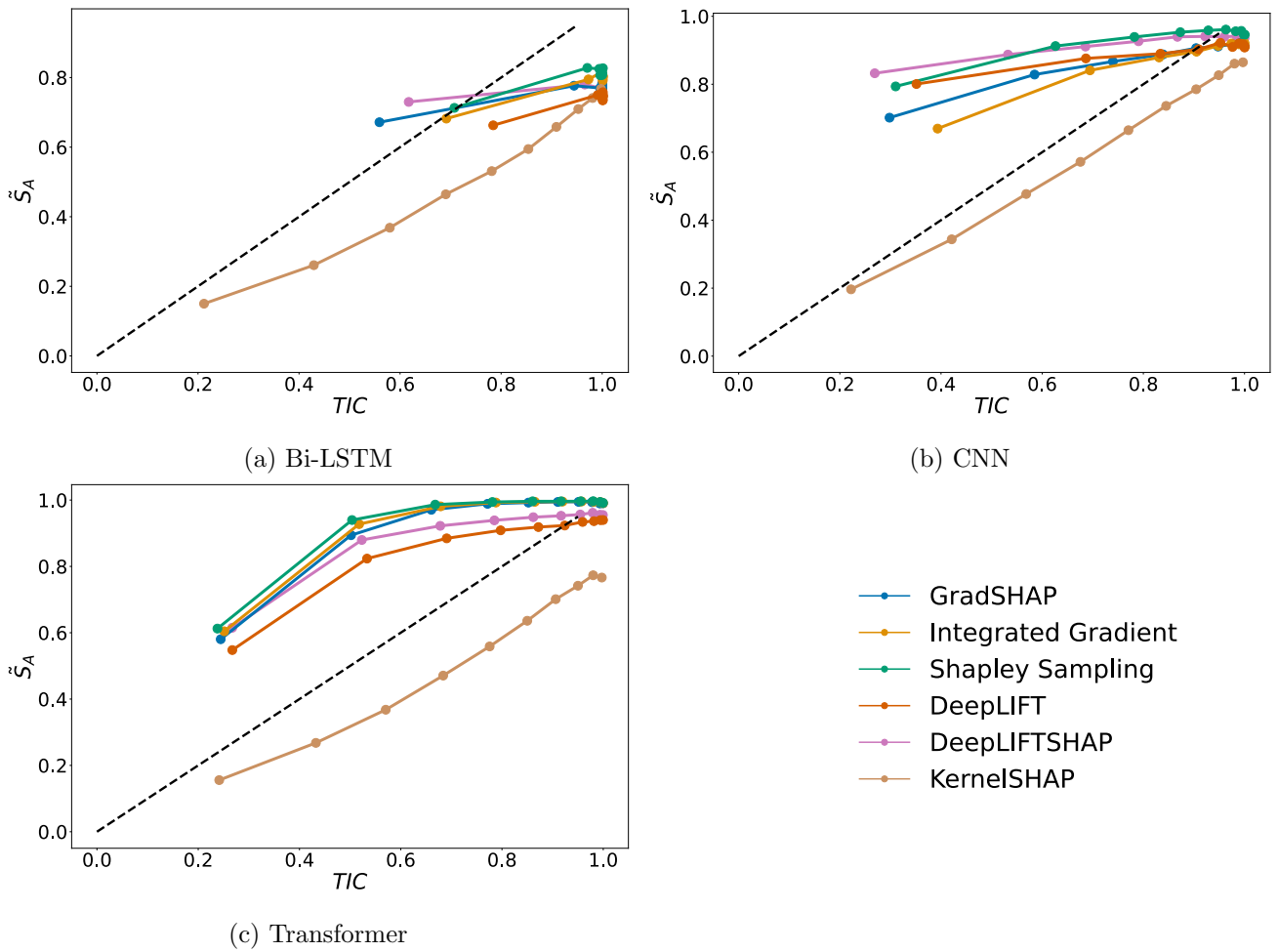
(b) CNN



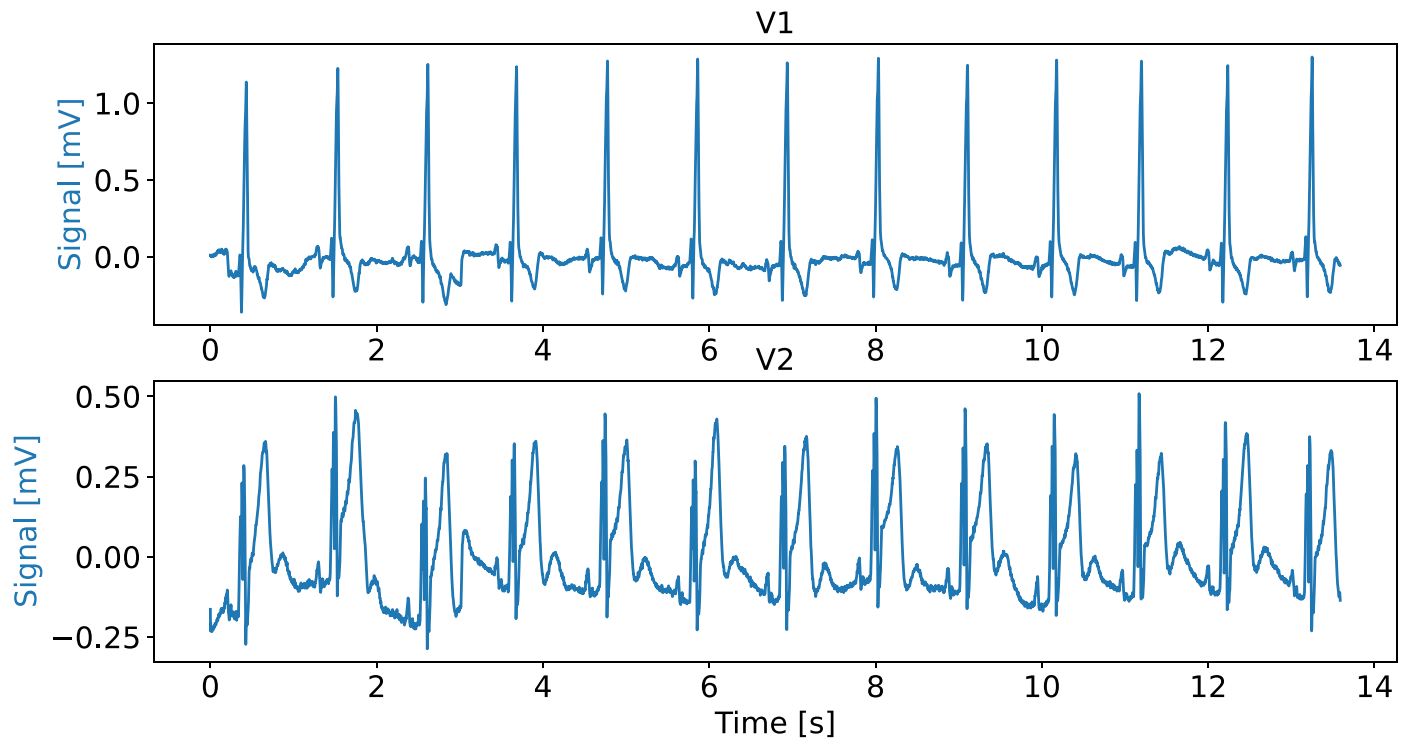
(c) Transformer

- GradSHAP
- Integrated Gradient
- Shapley Sampling
- DeepLIFT
- DeepLIFTSHAP
- KernelSHAP

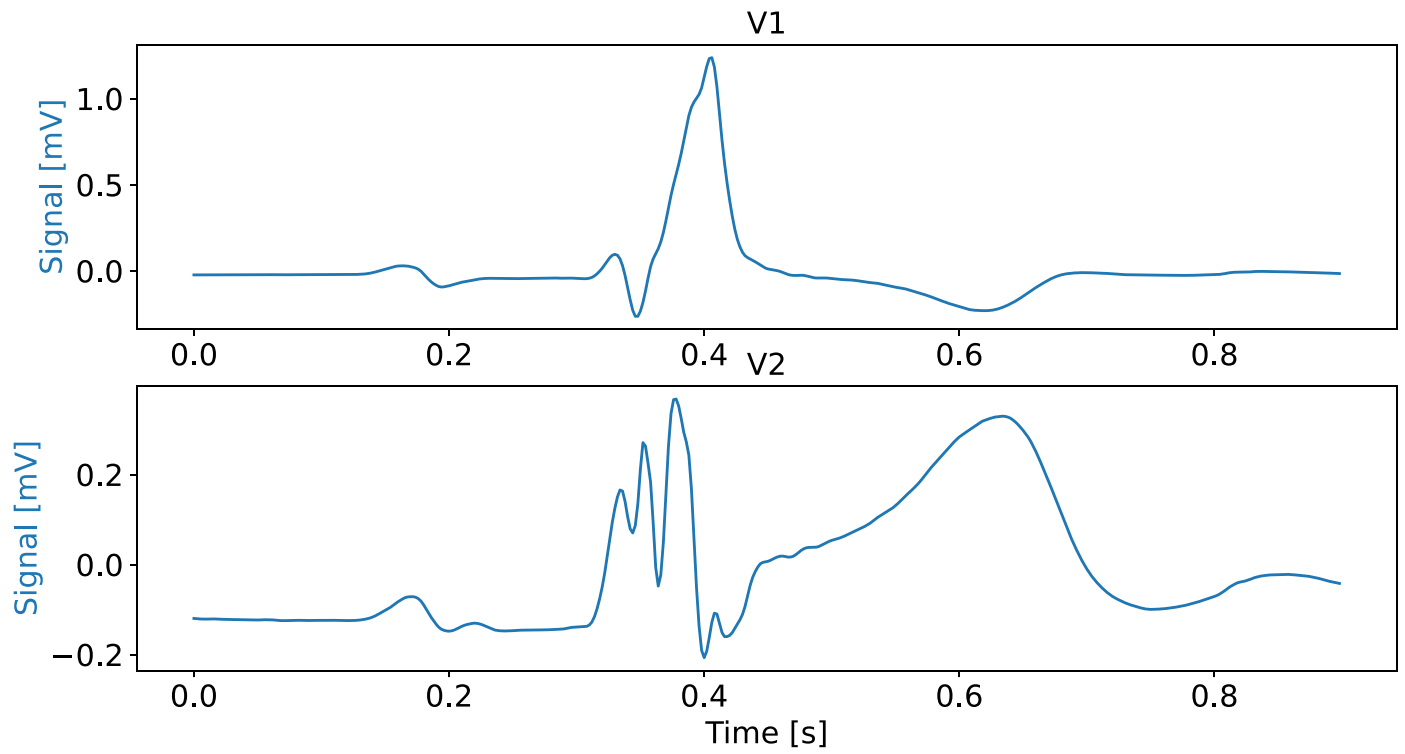
Extended Data Fig. 7 | Change in accuracy as a function of the ratio of points removed with respect to the total number of time steps in the sample, \tilde{N} for the six interpretability methods considered using the ECG dataset. Results depicted for (a) Bi-LSTM, (b) CNN and (c) Transformer.



Extended Data Fig. 8 | \hat{S}_A as a function of the TIC index for the six interpretability methods considered using the synthetic dataset. Results depicted for (a) Bi-LSTM, (b) CNN and (c) Transformer.



Extended Data Fig. 9 | Raw ECG signal for two selected leads from a given sample.



Extended Data Fig. 10 | Processed ECG signal for two selected leads from a given sample.

Reporting Summary

Nature Portfolio wishes to improve the reproducibility of the work that we publish. This form provides structure for consistency and transparency in reporting. For further information on Nature Portfolio policies, see our [Editorial Policies](#) and the [Editorial Policy Checklist](#).

Statistics

For all statistical analyses, confirm that the following items are present in the figure legend, table legend, main text, or Methods section.

n/a | Confirmed

- | | | |
|-------------------------------------|-------------------------------------|--|
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | The exact sample size (n) for each experimental group/condition, given as a discrete number and unit of measurement |
| <input type="checkbox"/> | <input checked="" type="checkbox"/> | A statement on whether measurements were taken from distinct samples or whether the same sample was measured repeatedly |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | The statistical test(s) used AND whether they are one- or two-sided <i>Only common tests should be described solely by name; describe more complex techniques in the Methods section.</i> |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | A description of all covariates tested |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | A description of any assumptions or corrections, such as tests of normality and adjustment for multiple comparisons |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | A full description of the statistical parameters including central tendency (e.g. means) or other basic estimates (e.g. regression coefficient) AND variation (e.g. standard deviation) or associated estimates of uncertainty (e.g. confidence intervals) |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | For null hypothesis testing, the test statistic (e.g. F , t , r) with confidence intervals, effect sizes, degrees of freedom and P value noted <i>Give P values as exact values whenever suitable.</i> |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | For Bayesian analysis, information on the choice of priors and Markov chain Monte Carlo settings |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | For hierarchical and complex designs, identification of the appropriate level for tests and full reporting of outcomes |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> | Estimates of effect sizes (e.g. Cohen's d , Pearson's r), indicating how they were calculated |

Our web collection on [statistics for biologists](#) contains articles on many of the points above.

Software and code

Policy information about [availability of computer code](#)

Data collection

Custom Python code published on GitHub (<https://github.com/hturbe/InterpretTime>) was used to create the synthetic dataset. The specific version of the code used to generate the results is also archived in Zenodo (<https://doi.org/10.5281/zenodo.7560836>). Python 3.8.16 was used and the requirements are all listed on GitHub in pyproject.toml.

Data analysis

Custom Python code published on GitHub (<https://github.com/hturbe/InterpretTime>) was used to analyse the results. The specific version of the code used to generate the results is also archived in Zenodo (<https://doi.org/10.5281/zenodo.7560836>). Python 3.8.16 was used and the requirements are all listed on GitHub in pyproject.toml.

For manuscripts utilizing custom algorithms or software that are central to the research but not yet described in published literature, software must be made available to editors and reviewers. We strongly encourage code deposition in a community repository (e.g. GitHub). See the Nature Portfolio [guidelines for submitting code & software](#) for further information.

Data

Policy information about [availability of data](#)

All manuscripts must include a [data availability statement](#). This statement should provide the following information, where applicable:

- Accession codes, unique identifiers, or web links for publicly available datasets
- A description of any restrictions on data availability
- For clinical datasets or third party data, please ensure that the statement adheres to our [policy](#)

All datasets and trained models used in this paper have been made available on Zenodo (<https://zenodo.org/record/7534770#.Y8lkkXbMI2w>). The ECG dataset is based on the public dataset released as part of The PhysioNet/Computing in Cardiology Challenge 2020 available under the following DOI: <https://doi.org/10.13026/f4ab-0814>. The Ford A dataset comes from the UEA & UCR Time Series Classification Repository. The synthetic dataset used as part of this study can be generated using the code shared on github: <https://github.com/hturbe/InterpretTime>.

Human research participants

Policy information about [studies involving human research participants and Sex and Gender in Research](#).

Reporting on sex and gender

The sex of the participants included in the ECG dataset was never considered in the analysis of our results. Indeed the presented research focussed on evaluating post-hoc interpretability methods for models trained on time series. The sex of the patient was not used as an input to the model.

Population characteristics

A full description of the population characteristics for the dataset used for the ECG classification task can be found in the publication associated with the dataset: Alday, E. A. P., Gu, A., Shah, A. J., Robichaux, C., Wong, A. K. I., Liu, C., ... & Reyna, M. A. (2020). Classification of 12-lead ecgs: the physionet/computing in cardiology challenge 2020. *Physiological measurement*, 41(12), 124003.

Recruitment

NA

Ethics oversight

NA

Note that full information on the approval of the study protocol must also be provided in the manuscript.

Field-specific reporting

Please select the one below that is the best fit for your research. If you are not sure, read the appropriate sections before making your selection.

- Life sciences Behavioural & social sciences Ecological, evolutionary & environmental sciences

For a reference copy of the document with all sections, see [nature.com/documents/nr-reporting-summary-flat.pdf](https://www.nature.com/documents/nr-reporting-summary-flat.pdf)

Life sciences study design

All studies must disclose on these points even when the disclosure is negative.

Sample size

Samples sizes for the synthetic dataset were tuned to achieve near 100% accuracy on the training set for the corresponding classification tasks. The number of dataset included in this study was determined to reflect a range of time series classification tasks with a synthetic dataset aiming to replicate a general classification task on time series.

Data exclusions

No data were excluded from the analyses.

Replication

The analysis was replicated across three datasets. For each datasets, the analysis was performed independently across 1000 samples randomly sampled from the test set.

Randomization

All datasets were randomly split into train/validation/test. For datasets with a published split, the latter was used as part of our analysis.

Blinding

Blinding was not relevant to our study as samples were randomly allocated to the different sets used for the analysis and no human judgement was involved to generate the results. In addition, all metrics are tested on randomly corrupted signals with no observational bias.

Reporting for specific materials, systems and methods

We require information from authors about some types of materials, experimental systems and methods used in many studies. Here, indicate whether each material, system or method listed is relevant to your study. If you are not sure if a list item applies to your research, read the appropriate section before selecting a response.

Materials & experimental systems

- | n/a | Included in the study |
|-------------------------------------|--|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Antibodies |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Eukaryotic cell lines |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Palaeontology and archaeology |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Animals and other organisms |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Clinical data |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Dual use research of concern |

Methods

- | n/a | Included in the study |
|-------------------------------------|---|
| <input checked="" type="checkbox"/> | <input type="checkbox"/> ChIP-seq |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> Flow cytometry |
| <input checked="" type="checkbox"/> | <input type="checkbox"/> MRI-based neuroimaging |