

---

# Explaining Time Series Predictions with Dynamic Masks

---

Jonathan Crabbé<sup>1</sup> Mihaela van der Schaar<sup>1 2 3</sup>

## Abstract

How can we explain the predictions of a machine learning model? When the data is structured as a multivariate time series, this question induces additional difficulties such as the necessity for the explanation to embody the time dependency and the large number of inputs. To address these challenges, we propose dynamic masks (Dynamask). This method produces instance-wise importance scores for each feature at each time step by fitting a perturbation mask to the input sequence. In order to incorporate the time dependency of the data, Dynamask studies the effects of dynamic perturbation operators. In order to tackle the large number of inputs, we propose a scheme to make the feature selection parsimonious (to select no more feature than necessary) and legible (a notion that we detail by making a parallel with information theory). With synthetic and real-world data, we demonstrate that the dynamic underpinning of Dynamask, together with its parsimony, offer a neat improvement in the identification of feature importance over time. The modularity of Dynamask makes it ideal as a plug-in to increase the transparency of a wide range of machine learning models in areas such as medicine and finance, where time series are abundant.

## 1. Introduction and context

What do we need to trust a machine learning model? If accuracy is necessary, it might not always be sufficient. With the application of machine learning to critical areas such as medicine, finance and the criminal justice system, the black-box nature of modern machine learning models has appeared as a major hindrance to their large scale deployment (Caruana et al., 2015; Lipton, 2016; Ching et al., 2018). With the necessity to address this problem, the field

---

<sup>1</sup>DAMTP, University of Cambridge, UK <sup>2</sup>University of California Los Angeles, USA <sup>3</sup>The Alan Turing Institute, UK. Correspondence to: Jonathan Crabbé <jc2133@cam.ac.uk>, Mihaela van der Schaar <mv472@cam.ac.uk>.

of explainable artificial intelligence (XAI) thrived (Barredo Arrieta et al., 2020; Das & Rad, 2020; Tjoa & Guan, 2020).

**Saliency Methods** Among the many possibilities to increase the transparency of a machine learning model, we focus here on *saliency methods*. The purpose of these methods is to highlight the features in an input that are relevant for a model to issue a prediction. We can distinguish them according to the way they interact with a model to produce importance scores.

*Gradient-based:* These methods use the gradient of the model’s prediction with respect to the features to produce importance scores. The premise is the following: if a feature is salient, we expect it to have a big impact on the model’s prediction when varied locally. This translates into a big gradient of the model’s output with respect to this feature. Popular gradient methods include Integrated Gradient (Sundararajan et al., 2017), DeepLIFT (Shrikumar et al., 2017) and GradSHAP (Lundberg et al., 2018).

*Perturbation-based:* These methods use the effect of a perturbation in the input on the model’s prediction to produce importance scores. The premise is similar to the one used in gradient based method. The key difference lies in the way in which the features are varied. If gradient based methods use local variation of the features (according to the gradient), perturbation-based method use the data itself to produce a variation. A first example is Feature Occlusion (Suresh et al., 2017) that replaces a group of features with a baseline. Another example is Feature Permutation that performs individual permutation of features within a batch.

*Attention-based:* For some models, the architecture allows to perform simple explainability tasks, such as determining feature saliency. A popular example, building on the success attention mechanisms (Vaswani et al., 2017), is the usage of attention layers to produce importance scores (Choi et al., 2016; Song et al., 2017; Xu et al., 2018; Kwon et al., 2018).

*Other:* There are some methods that don’t clearly fall in one of the above categories. A first example is SHAP (Lundberg & Lee, 2017), which attributes importance scores based on Shapley values. Another popular example is LIME in which the importance scores correspond to weights in a local linear model (Ribeiro et al., 2016a). Finally, some methods such as INVASE (Yoon et al., 2019a) or ASAC (Yoon et al., 2019b)

train a selector network to highlight important features.

**Time Series Saliency** Saliency methods were originally introduced in the context of image classification (Simonyan et al., 2013). Since then, most methods have focused on images and tabular data. Very little attention has been given to time series (Barredo Arrieta et al., 2020). A possible explanation for this is the increasing interest in *model agnostic methods* (Ribeiro et al., 2016b).

Model agnostic methods are designed to be used with a very wide range of models and data structures. In particular, nothing prevents us from using these methods for *Recurrent Neural Networks* (RNN) trained to handle multivariate time series. For instance, one could compute the Shapley values induced by this RNN for each input  $x_{t,i}$  describing a feature  $i$  at time  $t$ . In this configuration, all of these inputs are considered as individual features and the time ordering is forgotten. This approach creates a conceptual problem illustrated in Figure 1.

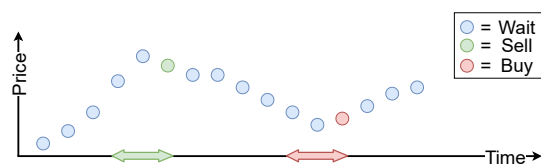


Figure 1. Context matters. This graph shows a fictional security price over time. A simplistic model recommends to sell just after a local maximum, to buy just after a local minimum and to wait otherwise. When recommending to sell or to buy, 3 consecutive time steps are required as a context.

In this example, the time dependency is crucial for the model. At each time step, the model provides a decision based on the two previous time steps. This oversimplified example illustrates that the time dependency induces a context, which might be crucial to understand the prediction of some models. In fact, recurrent models are inherently endowed with this notion of context as they rely on memory cells. This might explain the poor performances of model agnostic methods to identify salient features for RNNs (Ismail et al., 2020). These methods are *static* as the time dependency, and hence the context, is forgotten when treating all the time steps as separate features.

If we are ready to relax the model agnosticism, we can take a look at attention based methods. In methods such as RETAIN (Choi et al., 2016), attention weights are interpreted as importance scores. In particular, a high attention weight for a given input indicates that this input is salient for the model to issue a prediction. In this case, the architecture of the model allows us to provide importance scores without altering the dynamic nature of the data. However, it has

been shown recently that the attention weights of a model can significantly be changed without inducing any effect on the predictions (Jain & Wallace, 2019). This renders the parallel between attention weights and feature importance rather unclear. This discrepancy also appears in our experiments.

Another challenge induced by time series is the large number of inputs. The number of features is multiplied by the number of time steps that the model uses to issue a prediction. A saliency map can therefore become quickly overwhelming in this setting. To address these challenges, it is necessary to incorporate some treatment of *parsimony* and *legibility* in a time series saliency method. By *parsimony*, we mean that a saliency method should not select more inputs than necessary to explain a given prediction. For instance, a method that identifies 90% of the inputs as salient is not making the interpretation task easier. By *legibility*, we mean that the analysis of the importance scores by the user should be as simple as possible. Clearly, for long time sequences, analysing feature maps such as (c) and (d) on Figure 3 can quickly become daunting.

To address all of these challenges, we introduce Dynamask. It is a *perturbation-based* saliency method building on the concept of *masks* to produce post-hoc explanations for any time series model. Masks have been introduced in the context of image classification (Fong & Vedaldi, 2017; Fong et al., 2019). In this framework, a mask is fitted to each image. The mask highlights the regions of the image that are salient for the black-box classifier to issue its prediction. These masks are obtained by perturbing the pixels of the original image according to the surrounding pixels and study the impact of such perturbations on the black-box prediction. It has been suggested to extend the usage of masks beyond image classification (Phillips et al., 2018; Ho et al., 2019). However, to our knowledge, no available implementation and quantitative comparison with benchmarks exists in the context of multivariate time series. Moreover, few works in the literature explicitly mention explainability in a multivariate time series setting (Siddiqui et al., 2019; Tonekaboni et al., 2020). Both of these considerations motivate our proposition.

**Contributions** By building on the challenges that have been described, our work is, to our knowledge, the first saliency method to rigorously address the following questions in a time series setting.

- (1) *How to incorporate the context?* In our framework, this is naturally achieved by studying the effect of *dynamic* perturbations. Concretely, a perturbation is built for each feature at each time by using the value of this feature at adjacent times. This allows us to build meaningful perturbations that carry contexts such as the one illustrated in Figure 1.
- (2) *How to be parsimonious?* A great advantage of using

masks is that the notion of parsimony naturally translates into the *extremal* property. Conceptually, an *extremal mask* selects the minimal number of inputs allowing to reconstruct the black-box prediction with a given precision.

(3) *How to be legible?* To make our masks as simple as possible, we encourage them to be almost binary. Concretely, this means that we enforce a polarization between low and high saliency scores. Moreover, to make the notion of legibility quantitative, we propose a parallel with information theory<sup>1</sup>. This allows us to introduce two metrics : the mask information and the mask entropy. As illustrated in Figure 3, the entropy can be used to assess the legibility of a given mask. Moreover, these metrics can also be computed for other saliency methods, hence allowing insightful comparisons.

The paper is structured as follows. In Section 2, we outline the mathematical formalism related to our method. Then, we evaluate our method by comparing it with several benchmarks in Section 3 and conclude in Section 4.

## 2. Mathematical formulation

In this section, we formalize the problem of feature importance over time as well as the proposed solution. For the following, it is helpful to have the big picture in mind. Hence, we present the blueprint of Dynamask in Figure 2. The rest of this section makes this construction rigorous.

### 2.1. Preliminaries

Let  $\mathcal{X} \subset \mathbb{R}^{d_X}$  be an input (or feature) space and  $\mathcal{Y} \subset \mathbb{R}^{d_Y}$  be an output (or label) space, where  $d_X$  and  $d_Y$  are respectively the dimension of the input and the output space. For the sake of concision, we denote by  $[n_1 : n_2]$  the set of natural numbers between the natural numbers  $n_1$  and  $n_2$  with  $n_1 < n_2$ . We assume that the data is given in terms of time series  $(\mathbf{x}_t)_{t \in [1:T]}$ , where the inputs  $\mathbf{x}_t \in \mathcal{X}$  are indexed by a time parameter  $t \in [1 : T]$  with  $T \in \mathbb{N}^*$ . We consider the problem of predicting a sequence  $(\mathbf{y}_t)_{t \in [t_y:T]}$  of outputs  $\mathbf{y}_t \in \mathcal{Y}$  indexed by the same time parameter, but starting at a time  $t_y \geq 1$ , hence allowing to cover sequence to vector predictions ( $t_y = T$ ) as well as sequence to sequence predictions ( $t_y < T$ ). For the following, it is convenient to introduce a matrix notation for these time series. In this way,  $\mathbf{X} = (x_{t,i})_{(t,i) \in [1:T] \times [1:d_X]}$  denotes the matrix in  $\mathbb{R}^{T \times d_X}$  whose rows correspond to time steps and whose columns correspond to features. Similarly<sup>2</sup>, we denote  $\mathbf{Y} = (y_{t,i})_{(t,i) \in [t_y:T] \times [1:d_Y]}$  for the matrix in  $\mathbb{R}^{(T+1-t_y) \times d_Y}$ .

<sup>1</sup>Many connections exist between explainability and information theory, see for instance (Chen et al., 2018).

<sup>2</sup>In the following, we do not make a distinction between  $\mathbb{R}^{T \times d_X}$  and  $\mathcal{X}^T$  as these are isomorphic vector spaces. The same goes for  $\mathbb{R}^{(T+1-t_y) \times d_Y}$  and  $\mathcal{Y}^{T+1-t_y}$ .

Our task is to explain the prediction  $\mathbf{Y} = f(\mathbf{X})$  of a black box  $f$  that has been pre-trained for the aforementioned prediction task. In other words, our method aims to explain individual predictions of a given black box. More specifically, our purpose is to identify the parts of the input  $\mathbf{X}$  that are the most relevant for  $f$  to produce the prediction  $\mathbf{Y}$ . To do this, we use the concept of masks introduced in (Fong & Vedaldi, 2017; Fong et al., 2019), but in a different context. Here, we adapt the notion of mask to a dynamic setting.

**Definition 2.1** (Mask). A mask associated to an input sequence  $\mathbf{X} \in \mathbb{R}^{T \times d_X}$  and a black box  $f : \mathcal{X}^T \rightarrow \mathcal{Y}^{T+1-t_y}$  is a matrix  $\mathbf{M} = (m_{t,i}) \in [0, 1]^{T \times d_X}$  of the same dimension as the input sequence. The element  $m_{t,i}$  of this matrix represents the importance of feature  $i$  at time  $t$  for  $f$  to produce the prediction  $\mathbf{Y} = f(\mathbf{X})$ .

*Remark 2.1.* For a given coefficient in the mask matrix, a value close to 1 indicates that the feature is salient, while a value close to 0 indicates the opposite.

Now how can we obtain such a mask given a black box and an input sequence? To answer this question, we should let the mask act on the inputs and measure the effect it has on a black box prediction. As the name suggests, we would like this mask to hide irrelevant inputs contained in  $\mathbf{X}$ . To make this rigorous, it is useful to spend some time to think about perturbation operators.

### 2.2. Perturbation operators

The method that we are proposing in this paper is perturbation-based. Concretely, this means that a mask is used to build a *perturbation operator*. Examples of such operators are given in (Fong et al., 2019) in the context of image classification. Here, we propose a general definition and we explain how to take advantage of the dynamic nature of the data to build meaningful perturbations. By recalling that the mask coefficients indicate the saliency, we expect the perturbation to vanish for features  $x_{t,i}$  whose mask coefficient  $m_{t,i}$  is close to one. This motivates the following definition.

**Definition 2.2** (Perturbation operator). A perturbation operator associated to a mask  $\mathbf{M} \in [0, 1]^{T \times d_X}$  is a linear operator acting on the input sequence space  $\Pi_{\mathbf{M}} : \mathbb{R}^{T \times d_X} \rightarrow \mathbb{R}^{T \times d_X}$ . It needs to fulfil the two following assumptions for any given  $(t, i) \in [1 : T] \times [1 : d_X]$ :

1. The perturbation for  $x_{t,i}$  is dictated by  $m_{t,i}$  :

$$[\Pi_{\mathbf{M}}(\mathbf{X})]_{t,i} = \pi(\mathbf{X}, m_{t,i}; t, i),$$

where  $\pi$  is differentiable for  $m \in (0, 1)$  and continuous at  $m = 0, 1$ .

2. The action of the perturbation operator is trivial when the mask coefficient is set to one :

$$\pi(\mathbf{X}, 1; t, i) = x_{t,i}.$$

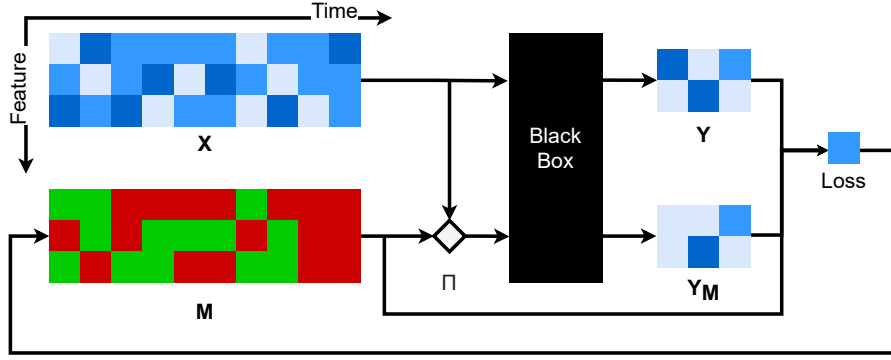


Figure 2. Diagram for Dynamask. An input matrix  $\mathbf{X}$ , extracted from a multivariate time series, is fed to a black-box to produce a prediction  $\mathbf{Y}$ . The objective is to give a saliency score for each component of  $\mathbf{X}$ . In Dynamask, these saliency scores are stored in a mask  $\mathbf{M}$  of the same shape as the input  $\mathbf{X}$ . To detect the salient information in the input  $\mathbf{X}$ , the mask produces a perturbed version of  $\mathbf{X}$  via a perturbation operator  $\Pi$ . This perturbed  $\mathbf{X}$  is fed to the black-box to produce a perturbed prediction  $\mathbf{Y}_M$ . The perturbed prediction is compared to the original prediction and the error is backpropagated to adapt the saliency scores contained in the mask.

Moreover, we say that the perturbation is *dynamic* if for any given feature at any given time, the perturbation is constructed with the values that this feature takes in neighbouring times. More formally, the perturbation is dynamic if there exist a couple  $(W_1, W_2) \in \mathbb{N} \times \mathbb{N} \setminus \{(0, 0)\}$  such that for all  $(t, i) \in [1 : T] \times [1 : d_X]$  we have:

$$\frac{\partial [\pi(\mathbf{X}, m_{t,i}; t, i)]}{\partial x_{t',i}} \neq 0$$

for all  $t' \in [t - W_1 : t + W_2] \cap [1 : T]$ .

*Remark 2.2.* The first assumption ensures that the perturbations are applied independently for all inputs and that our method is suitable for gradient-based optimization. The second assumption<sup>3</sup> translates the fact that the perturbation should have less effect on salient inputs.

*Remark 2.3.* The parameters  $W_1$  and  $W_2$  control how the perturbation depends on neighbouring times. A non-zero value for  $W_1$  indicates that the perturbation depends on past time steps. A non-zero value for  $W_2$  indicates that the perturbation depends on future time steps. For the perturbation to be dynamic, at least one of these parameters has to be different from zero.

This definition gives the freedom to design perturbation operators adapted to particular contexts. The method that we develop can be used for any such perturbation operator. In our case, we would like to build our masks by taking advantage the dynamic nature of the data into account. This is crucial if we want to capture local time variations of the features, such as a quick increase of the blood-pressure or

<sup>3</sup>Together with the continuity of  $\pi$  with respect to the mask coefficients.

an extremal security price from Figure 1. This is achieved by using dynamic perturbation operators. To illustrate, we provide three examples:

$$\begin{aligned} \pi^g(\mathbf{X}, m_{t,i}; t, i) &= \frac{\sum_{t'=1}^T x_{t',i} \cdot g_{\sigma(m_{t,i})}(t - t')}{\sum_{t'=1}^T g_{\sigma(m_{t,i})}(t - t')} \\ \pi^m(\mathbf{X}, m_{t,i}; t, i) &= m_{t,i} \cdot x_{t,i} + (1 - m_{t,i}) \cdot \mu_{t,i} \\ \pi^p(\mathbf{X}, m_{t,i}; t, i) &= m_{t,i} \cdot x_{t,i} + (1 - m_{t,i}) \cdot \mu_{t,i}^p, \end{aligned}$$

where  $\pi^g$  is a temporal Gaussian blur<sup>4</sup> with

$$g_{\sigma}(t) = \exp\left(-\frac{t^2}{2\sigma^2}\right); \sigma(m) = \sigma_{max} \cdot (1 - m).$$

Similarly,  $\pi^m$  can be interpreted as a fade to moving average perturbation with

$$\mu_{t,i} = \frac{1}{2W + 1} \sum_{t'=t-W}^{t+W} x_{t',i},$$

where  $W \in \mathbb{N}$  is the size of the moving window. Finally,  $\pi^p$  is similar to  $\pi^m$  with one difference: the former only uses past values of the features to compute the perturbation<sup>5</sup>:

$$\mu_{t,i}^p = \frac{1}{W + 1} \sum_{t'=t-W}^t x_{t',i}.$$

Note that the Hadamard product  $\mathbf{M} \odot \mathbf{X}$  used in (Ho et al., 2019) is a particular case of  $\pi^m$  with  $\mu_{t,i} = 0 =$  so that

<sup>4</sup>For this perturbation to be continuous, we assume that  $\pi^g(\mathbf{X}, 1; t, i) \equiv x_{t,i}$ .

<sup>5</sup>This is useful in a typical forecasting setting where the future values of the feature are unknown.



this perturbation is static. In the following, to stress that a mask is obtained by inspecting the effect of a dynamic perturbation operator, we shall refer to it as a dynamic mask (or Dynamask in short).

In terms of complexity, the computation of  $\pi^g$  requires  $\mathcal{O}(d_X \cdot T^2)$  operations<sup>6</sup> while  $\pi^m$  requires  $\mathcal{O}(d_X \cdot T \cdot W)$  operations<sup>7</sup>. When  $T$  is big<sup>8</sup>, it might therefore be more interesting to use  $\pi^m$  with  $W \ll T$  or a windowed version of  $\pi^g$ . With this analysis of perturbation operators, everything is ready to explain how dynamic masks are obtained.

### 2.3. Mask optimization

To design an objective function for our mask, it is helpful to keep in mind what an ideal mask does. From the previous subsection, it is clear that we should compare the black-box predictions for both the unperturbed and the perturbed input. Ideally, the mask will identify a subset of salient features contained in the input that explains the black-box prediction. Since this subset of features is salient, the mask should indicate to the perturbation operator to preserve it. More concretely, a first part of our objective function should keep the shift in the black-box prediction to be small. We call it the *error* part of our objective function. In practice, the expression for the error part depends on the task done by the black-box. In a regression context, we minimize the squared error between the unperturbed and the perturbed prediction:

$$\mathcal{L}_e(\mathbf{M}) = \sum_{t=t_y}^T \sum_{i=1}^{d_Y} \left( [(f \circ \Pi_{\mathbf{M}})(\mathbf{X})]_{t,i} - [f(\mathbf{X})]_{t,i} \right)^2.$$

Similarly, in a classification task, we minimize the cross-entropy between the predictions:

$$\mathcal{L}_e(\mathbf{M}) = - \sum_{t=t_y}^T \sum_{c=1}^{d_Y} [f(\mathbf{X})]_{t,c} \log [(f \circ \Pi_{\mathbf{M}})(\mathbf{X})]_{t,c}.$$

Now we have to make sure that the mask actually selects salient features and discards the others. By remembering that  $m_{t,i} = 0$  indicates that the feature  $x_{t,i}$  is irrelevant for the black-box prediction, this selection translates into imposing sparsity in the mask matrix  $\mathbf{M}$ . A first approach for this, used in (Fong & Vedaldi, 2017; Ho et al., 2019), is to add a  $l^1$  regularisation term on the coefficients of  $\mathbf{M}$ . However, it was noted in (Fong et al., 2019) that this produces mask that vary with the regularization coefficient  $\lambda$  in a way that renders comparisons between different  $\lambda$  difficult. To solve this issue, they introduce a new regularization term

<sup>6</sup>One must compute a perturbation for each  $d_X \cdot T$  component of the input and the sums in each perturbation have  $T$  terms

<sup>7</sup>One must compute a perturbation for each  $d_X \cdot T$  component of the input and each moving average involves  $2W + 1$  terms

<sup>8</sup>In our experiments, however, we never deal with  $T > 100$  so that both approaches are reasonable.

to impose sparsity:

$$\mathcal{L}_a(\mathbf{M}) = \|\text{vecsort}(\mathbf{M}) - \mathbf{r}_a\|^2,$$

where  $\|\cdot\|$  denotes the vector 2-norm,  $\text{vecsort}$  is a function that vectorizes  $\mathbf{M}$  and then sort the elements of the resulting vector in ascending order. The vector  $\mathbf{r}_a$  contains  $(1 - a) \cdot d_X \cdot T$  zeros followed by  $a \cdot d_X \cdot T$  ones, where  $a \in [0, 1]$ . In short, this regularization term encourages the mask to highlight a fraction  $a$  of the inputs. For this reason,  $a$  can also be referred to as the area of the mask. A first advantage of this approach is that the hyperparameter  $a$  can be modulated by the user to highlight a desired fraction of the input. For instance, one can start with a small value of  $a$  and slide it to higher values in order to see features gradually appearing in order of importance. Another advantage of this regularization term is that it encourages the mask to be binary, which makes the mask more legible as we will detail in the next section.

Finally, we might want to avoid quick variations in the saliency over time. This could either be a prior belief or a preference of the user with respect to the saliency map. If this is relevant, we can enforce the salient regions to be connected in time with the following loss that penalizes jumps of the saliency over time:

$$\mathcal{L}_c(\mathbf{M}) = \sum_{t=1}^{T-1} \sum_{i=1}^{d_X} |m_{t+1,i} - m_{t,i}|.$$

For a fixed fraction  $a$ , the mask optimization problem can therefore be written as

$$\mathbf{M}_a^* = \arg \min_{\mathbf{M} \in [0,1]^{T \times d_X}} \mathcal{L}_e(\mathbf{M}) + \lambda_a \cdot \mathcal{L}_a(\mathbf{M}) + \lambda_c \cdot \mathcal{L}_c(\mathbf{M}).$$

Note that, in this optimization problem, the fraction  $a$  is fixed. In some contexts, one might want to find the smallest fraction of input features that allows us to reproduce the black-box prediction with a given precision. Finding this minimal fraction corresponds to the following optimization problem:

$$a^* = \min \{a \in [0, 1] \mid \mathcal{L}_e(\mathbf{M}_a^*) < \varepsilon\},$$

where  $\varepsilon$  is the threshold that sets the acceptable precision. The resulting mask  $\mathbf{M}_{a^*}^*$  is then called *extremal mask*. The idea of an extremal mask is extremely interesting by itself: it explains the black-box prediction in terms of a minimal number of salient features. This is precisely the parsimony that we were referring to in the introduction.

Finally, it is worth mentioning that we have here presented a scheme where the mask preserves the features that minimize the error. There exists a variant of this scheme where the mask preserves the features that maximize the error. This other scheme, together with the detailed optimization algorithm used in our implementation, can be found in Section 2 of the supplementary materials.

## 2.4. Masks and information theory

Once the mask is obtained, it highlights the features that contain crucial information for the black-box to issue a prediction. This motivates a parallel between masks and information theory. To make this rigorous, we notice that the mask admits a natural interpretation in terms of information content. As aforementioned, a value close to 1 for  $m_{t,i}$  indicates that the input feature  $x_{t,i}$  carries information that is important for the black box  $f$  to predict an outcome. It is therefore natural to interpret a mask coefficient as a probability that the associated feature is salient for the black box to issue its prediction. It is tempting to use this analogy to build the counterpart of *Shannon information content* in order to measure the quantity of information contained in subsequences extracted from the time series. However, this analogy requires a closer analysis. We recall that the Shannon information content of an outcome decreases when the probability of this outcome increases (Shannon, 1948; MacKay, 2003; Cover & Thomas, 2005). In our framework, we would like to adapt this notion so that the information content increases with the mask coefficients (if  $m_{t,i}$  gets closer to one, this indicates that  $x_{t,i}$  carries more useful information for the black-box to issue its prediction). To solve this discrepancy, we have to use  $1 - m_{t,i}$  as the pseudo-probability appearing in our adapted notion of Shannon information content.

**Definition 2.3** (Mask information). The mask information associated to a mask  $\mathbf{M}$  and a subsequence  $(x_{t,i})_{(t,i) \in A}$  of the input  $\mathbf{X}$  with  $A \subseteq [1 : T] \times [1 : d_X]$  is

$$I_{\mathbf{M}}(A) = - \sum_{(t,i) \in A} \ln(1 - m_{t,i}).$$

*Remark 2.4.* Conceptually, the information content of a subsequence measures the quantity of useful information it contains for the black-box to issue a prediction. It allows us to associate a saliency score to a group of inputs according to the mask.

*Remark 2.5.* Note that, in theory, the mask information diverges when a mask coefficient is set to one. In practice, this can be avoided by imposing  $\mathbf{M} \in (0, 1)^{T \times d_X}$ .

As in traditional information theory, the information content is not entirely informative on its own. For instance, consider two subsequences indexed by, respectively,  $A, B$  with  $|A| = |B| = 10$ . We assume that the submask extracted from  $\mathbf{M}$  with  $A$  contains 3 coefficients  $m = 0.9$  and 7 coefficients  $m = 0$  so that the information content of  $A$  is  $I_{\mathbf{M}}(A) \approx 6.9$ . Now we consider that all the coefficient extracted from  $\mathbf{M}$  with  $B$  are equal to 0.5 so that  $I_{\mathbf{M}}(B) \approx 6.9$  and hence  $I_{\mathbf{M}}(A) \approx I_{\mathbf{M}}(B)$ . In this example,  $A$  clearly identifies 3 important features while  $B$  gives a mixed score for the 10 features. Intuitively, it is pretty clear that the information provided by  $A$  is sharper. Unfortunately,

the mask information by itself does not allow to distinguish these two subsequences. Hopefully, a natural distinction is given by the counterpart of Shannon entropy.

**Definition 2.4** (Mask entropy). The mask entropy associated to a mask  $\mathbf{M}$  and a subsequence  $(x_{t,i})_{(t,i) \in A}$  of the input  $\mathbf{X}$  with  $A \subseteq [1 : T] \times [1 : d_X]$  is

$$S_{\mathbf{M}}(A) = - \sum_{(t,i) \in A} m_{t,i} \ln m_{t,i} + (1 - m_{t,i}) \ln(1 - m_{t,i})$$

*Remark 2.6.* We stress that the mask entropy is *not* the Shannon entropy for the subsequence  $(x_{t,i})_{(t,i) \in A}$ . Indeed, the pseudo-probabilities that we consider are not the probabilities  $p(x_{t,i})$  for each feature  $x_{t,i}$  to occur. In particular, there is no reason to expect that the probability of each input decouple from the others so that it would be wrong to sum the individual contribution of each input separately<sup>9</sup> like we are doing here.

Clearly, it is desirable for our mask to provide explanations with low entropy. This stems from the fact that the entropy is maximized when mask coefficients  $m_{t,i}$  are close to 0.5. In this case, given our probabilistic interpretation, the mask coefficient is ambiguous as it does not really indicate whether the feature is salient. This is consistent with our previous example where  $S_{\mathbf{M}}(A) \approx 0.98$  while  $S_{\mathbf{M}}(B) \approx 6.93$  so that  $S_{\mathbf{M}}(A) \ll S_{\mathbf{M}}(B)$ . Since masks coefficients take various values in practice, masks with higher entropy appear less legible, as illustrated in Figure 3. Therefore, we use the entropy as a measure of the mask’s sharpness and legibility in our experiments. In particular, the entropy is minimized for perfectly binary masks  $M \in \{0, 1\}^{T \times d_X}$ , which are easy to read and contain no ambiguity. Consequently, the regularization term  $\mathcal{L}_a(\mathbf{M})$  that we used in Section 2.3 has the effect of reducing the entropy. Since our adapted notions of information and entropy rely on individual contributions from each feature, they come with natural properties.

**Proposition 2.1** (Metric properties). *For all labelling sets  $A, B \subseteq [1 : T] \times [1 : d_X]$ , the mask information and entropy enjoy the following properties:*

### 1. Positivity

$$I_{\mathbf{M}}(A) \geq 0 \quad S_{\mathbf{M}}(A) \geq 0$$

### 2. Additivity

$$\begin{aligned} I_{\mathbf{M}}(A \cup B) &= I_{\mathbf{M}}(A) + I_{\mathbf{M}}(B) - I_{\mathbf{M}}(A \cap B) \\ S_{\mathbf{M}}(A \cup B) &= S_{\mathbf{M}}(A) + S_{\mathbf{M}}(B) - S_{\mathbf{M}}(A \cap B) \end{aligned}$$

### 3. Monotonicity If $A \subset B$ :

$$I_{\mathbf{M}}(A) \leq I_{\mathbf{M}}(B) \quad S_{\mathbf{M}}(A) \leq S_{\mathbf{M}}(B).$$

<sup>9</sup> It is nonetheless possible to build a single mask coefficient for a group of inputs in order to imitate a joint distribution.

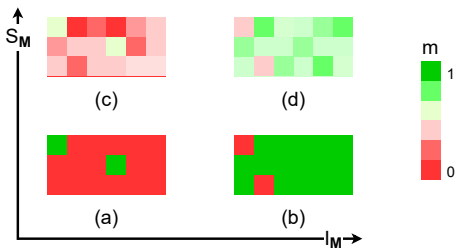


Figure 3. Mask information and entropy. For a given subsequence and a given mask, the information increases when more features are relevant ( $a \rightarrow b$  or  $c \rightarrow d$ ). The entropy increases when the sharpness of the saliency map decreases ( $a \rightarrow c$  or  $b \rightarrow d$ ). Masks with high entropy appear less legible, especially for long sequences.

*Proof.* See Section 1 of the supplementary material.  $\square$

Together, these properties guarantee that  $I_M$  and  $S_M$  define measures for the discrete  $\sigma$ -algebra  $\mathcal{P}([1 : T] \times [1 : d_X])$  on the set of input indexes  $[1 : T] \times [1 : d_X]$ . All of these metrics can be computed for any saliency method, hence allowing comparisons in our experiments. For more details, please refer to the Section 1 of the supplementary material.

### 3. Experiments

In this section<sup>10</sup>, we evaluate the quality of our dynamic masks. There are two big difficulties to keep in mind when evaluating the quality of a saliency method. The first one is that the true importance of feature is usually unknown with real-world data. The second one is that the performance of a saliency method in identifying relevant features depends on the black-box and its performances. In order to illustrate these challenges, we propose three different experiments in ascending order of difficulty. In the first experiment, a white box with known feature importance is used so that both difficulties are avoided. In the second experiment, a black-box trained on a dataset with known feature importance is used and hence only the second difficulty is encountered. In the third experiment, a black-box trained on a real-world clinical dataset is used and hence both difficulties are encountered. For each experiment, more details are given in Section 3 of the supplementary materials.

#### 3.1. Feature importance for a white-box regressor

**Experiment** In this experiment, we work with a trivial white-box regressor whose predictions only rely on a known

<sup>10</sup>Our implementation can be found at <https://github.com/JonathanCrabbe/Dynamask>.

subset  $A = A_T \times A_X \subset [1 : T] \times [1 : d_X]$  of salient inputs, where  $A_T$  and  $A_X$  respectively give the salient times and features:

$$[f(\mathbf{X})]_t = \begin{cases} \sum_{i \in A_X} (x_{t,i})^2 & \text{if } t \in A_T \\ 0 & \text{else.} \end{cases}$$

Note that, in this experiment,  $d_Y = 1$  so that we can omit the second index for  $f(\mathbf{X})$ . In our experiment, we consider two scenarios that are known to be challenging for state of the art saliency methods (Ismail et al., 2020). In the first one, the white-box depends on a small portion of salient features  $|A_X| \ll d_X$ . In the second one, the white-box depends on a small portion of salient times  $|A_T| \ll T$ . We fit a mask to this black-box by using the squared error loss together with the regularization. In our experiment:  $T = d_X = 50$ . The salient time and features are selected randomly, the input features are generated with an ARMA process. We repeat the experiment 10 times.

**Metrics** Since the true salient features are known unambiguously in this setup, we have a large set of metrics at hand to evaluate the performance of a saliency method. To measure the proportion of identified features that are indeed salient, we use the *area under the precision curve* (AUP, higher is better). To measure the portion of salient features that have indeed been identified, we use the *area under the recall curve* (AUR, higher is better). To measure how much information the saliency method predicts for the salient region, we use the mask information content ( $I_M(A)$ , higher is better). To measure the sharpness of the explanation in this region, we use the mask entropy ( $S_M(A)$ , lower is better).

**Benchmarks** The fact that we are dealing with a white-box regressor already disqualifies some methods such as FIT (Tonekaboni et al., 2020) or DeepLIFT. We compare our method with Feature Occlusion (FO), Feature Permutation (FP), Integrated Gradient (IG) and Shapley Value Sampling (SVS) (Castro et al., 2009). For fair comparison, we use these baselines to evaluate the importance of each feature at each given time. As explained in Section 2.4, a mask can be associated to each saliency method.

Table 1. Scores for the rare feature experiment.

	AUP	AUR	$I_M(A)$	$S_M(A)$
<b>MASK</b>	$0.99 \pm 0.01$	<b><math>0.58 \pm 0.03</math></b>	<b><math>252 \pm 69</math></b>	<b><math>0.7 \pm 0.7</math></b>
FO	<b><math>1.00 \pm 0.00</math></b>	$0.14 \pm 0.03$	$9 \pm 6$	$11.0 \pm 2.5$
FP	<b><math>1.00 \pm 0.00</math></b>	$0.16 \pm 0.04$	$13 \pm 7$	$12.6 \pm 3.3$
IG	$0.99 \pm 0.00$	$0.14 \pm 0.03$	$8 \pm 4$	$11.1 \pm 2.5$
SVS	<b><math>1.00 \pm 0.00</math></b>	$0.14 \pm 0.04$	$9 \pm 6$	$11.0 \pm 2.5$

**Discussion** The AUP is not useful to discriminate between the methods. Our method significantly outperforms all the other benchmarks for all other metric. In particular, we

Table 2. Scores for the rare time experiment.

	AUP	AUR	$I_M(A)$	$S_M(A)$
<b>MASK</b>	$0.99 \pm 0.01$	<b><math>0.68 \pm 0.04</math></b>	<b><math>1290 \pm 106</math></b>	<b><math>7.1 \pm 2.5</math></b>
FO	<b><math>1.00 \pm 0.00</math></b>	$0.14 \pm 0.04$	$49 \pm 14$	$48.3 \pm 6.5$
FP	<b><math>1.00 \pm 0.00</math></b>	$0.16 \pm 0.03$	$53 \pm 8$	$54.7 \pm 5.8$
IG	$0.99 \pm 0.00$	$0.14 \pm 0.04$	$38 \pm 12$	$48.7 \pm 6.7$
SVS	<b><math>1.00 \pm 0.00</math></b>	$0.14 \pm 0.04$	$49 \pm 14$	$48.3 \pm 6.5$

notice that, for both experiments, our method identifies a significantly higher portion of features that are truly salient (higher AUR). As claimed in Section 2.3, we observe that our optimization technique is indeed efficient to produce explanations with low mask entropy.

### 3.2. Feature importance for a black-box classifier

**Experiment** We reproduce the state experiment from (Tonekaboni et al., 2020) but in a more challenging setting. In this case, the data is generated according to a 2-state hidden Markov model (HMM) whose state at time  $t \in [1 : T]$  ( $T = 200$ ) is denoted  $s_t \in \{0, 1\}$ . At each time, the input feature vector has three components ( $d_X = 3$ ) and is generated according to the current state via  $\mathbf{x}_t \sim \mathcal{N}(\boldsymbol{\mu}_{s_t}, \boldsymbol{\Sigma}_{s_t})$ . To each of these input vectors is associated a binary label  $y_t \in \{0, 1\}$ . This binary label is conditioned by one of the three component of the feature vector, based on the state:

$$p_t = \begin{cases} (1 + \exp[-x_{2,t}])^{-1} & \text{if } s_t = 0 \\ (1 + \exp[-x_{3,t}])^{-1} & \text{if } s_t = 1 \end{cases}.$$

The label is then emitted via a Bernoulli distribution  $y_t \sim \text{Ber}(p_t)$ . The experiment proposed in (Tonekaboni et al., 2020) focuses on identifying the salient feature at each time where a state transition occurs. However, it is clear from the above discussion that there is exactly one salient feature at any given time. More precisely, the set of salient indexes is  $A = \{(t, 2 + s_t) \mid t \in [1 : T]\}$ . We generate 1000 such time series, 800 of them are used to train a RNN black-box classifier  $f$  with one hidden layer made of 200 hidden GRU cells<sup>11</sup>. We then fit an extremal mask to the black-box by minimizing the cross entropy error for each test time series. We repeat the experiment 5 times.

**Metrics** We use the same metrics as before.

**Benchmarks** In addition to the previous benchmarks, we use Augmented Feature Occlusion (AFO), FIT, RETAIN (RT), DeepLIFT (DL), LIME and GradSHAP (GS).

**Discussion** Our method outperforms the other benchmarks

<sup>11</sup>Implementation details of the original experiment can be found at [https://github.com/sanatonek/time\\_series\\_explainability](https://github.com/sanatonek/time_series_explainability)

Table 3. Scores for the state experiment.

	AUP	AUR	$I_M(A) \div 10^5$	$S_M(A) \div 10^4$
<b>MASK</b>	<b><math>0.88 \pm 0.01</math></b>	$0.70 \pm 0.00$	<b><math>2.24 \pm 0.01</math></b>	<b><math>0.04 \pm 0.00</math></b>
FO	$0.63 \pm 0.01$	$0.45 \pm 0.01$	$0.21 \pm 0.00$	$1.79 \pm 0.00$
AFO	$0.63 \pm 0.01$	$0.42 \pm 0.01$	$0.19 \pm 0.00$	$1.76 \pm 0.00$
IG	$0.56 \pm 0.00$	<b><math>0.78 \pm 0.00</math></b>	$0.05 \pm 0.00$	$1.39 \pm 0.00$
GS	$0.49 \pm 0.00$	$0.62 \pm 0.00$	$0.33 \pm 0.00$	$1.73 \pm 0.00$
LIME	$0.49 \pm 0.01$	$0.50 \pm 0.01$	$0.04 \pm 0.00$	$1.11 \pm 0.00$
DL	$0.57 \pm 0.01$	$0.20 \pm 0.00$	$0.09 \pm 0.00$	$1.18 \pm 0.00$
RT	$0.42 \pm 0.03$	$0.51 \pm 0.01$	$0.03 \pm 0.00$	$1.75 \pm 0.00$
FIT	$0.44 \pm 0.01$	$0.60 \pm 0.02$	$0.47 \pm 0.02$	$1.57 \pm 0.00$

for 3 of the 4 metrics. The AUR suggests that IG identifies more salient features. However, the significantly lower AUP for IG suggests that it identifies too many features as salient. We found that IG identifies 87% of the inputs as salient<sup>12</sup> versus 32% for our method. From the above discussion, it is clear that only 1/3 of the inputs are really salient, hence Dynamask offers more parsimony. We use two additional metrics to offer further comparisons between the methods: the area under the receiver operating characteristic (AUROC) and the area under the precision-recall curve (AUPRC). We reach the conclusion that Dynamask outperforms the other saliency methods with AUROC =  $0.93 \pm 0.00$  and AUPRC =  $0.85 \pm 0.00$ . The second best method is again Integrated Gradient with AUROC =  $0.91 \pm 0.00$  and AUPRC =  $0.79 \pm 0.00$ . All the other methods have AUROC  $\leq 0.85$  and AUPRC  $\leq 0.70$ .

### 3.3. Feature importance on clinical data

**Experiment** We reproduce the MIMIC mortality experiment from (Tonekaboni et al., 2020). We fit a RNN black-box with 200 hidden GRU cells to predict the mortality of a patient based on 48 hours ( $T = 48$ ) of patient features ( $d_X = 31$ ). This is a binary classification problem for which the RNN estimates the probability of each class. For each patient, we fit a mask with area  $a$  to identify the most important observations  $x_{t,i}$ . Since the ground-truth feature saliency is unknown, we must find an alternative way to assess the quality of this selection. If these observations are indeed salient, we expect them to have a big impact on the black-box’s prediction if they are replaced. For this reason, we replace the most important observations by the time average value for each associated feature  $x_{t,i} \mapsto \tilde{x}_{t,i} = \frac{1}{T} \sum_{t=1}^T x_{t,i}$ . We then compare the prediction for the original input  $f(\mathbf{X})$  with the prediction for the input where the most important observations have been replaced  $f(\tilde{\mathbf{X}})$ . A big shift in the prediction indicates that these observations are salient. We repeat this experiment 3 times for various values of  $a$ .

**Dataset** We use the MIMIC-III dataset (Johnson et al.,

<sup>12</sup>In this case, we consider a feature as salient if its normalized importance score (or mask coefficient) is above 0.5.



2016), that contains the health record of 40, 000 ICU de-identified patients at the Beth Israel Deaconess Medical Center. The selected data and its preprocessing is the same as the one done by (Tonekaboni et al., 2020).

**Metrics** To estimate the importance of the shift induced by replacing the most important observations, we use the *cross-entropy* between  $f(\mathbf{X})$  and  $f(\tilde{\mathbf{X}})$  (CE, higher is better). To evaluate the number of patient whose prediction has been flipped, we compute the *accuracy* of  $f(\tilde{\mathbf{X}})$  with respect to the initial predictions  $f(\mathbf{X})$  (ACC, lower is better).

**Benchmarks** We use the same benchmarks as before.

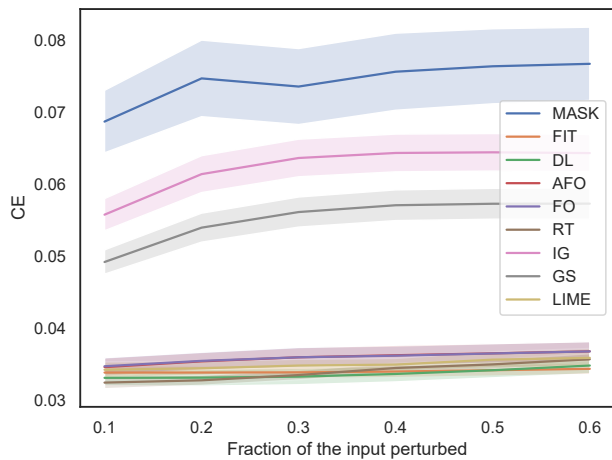


Figure 4. CE as a function of  $a$  for the MIMIC experiment.

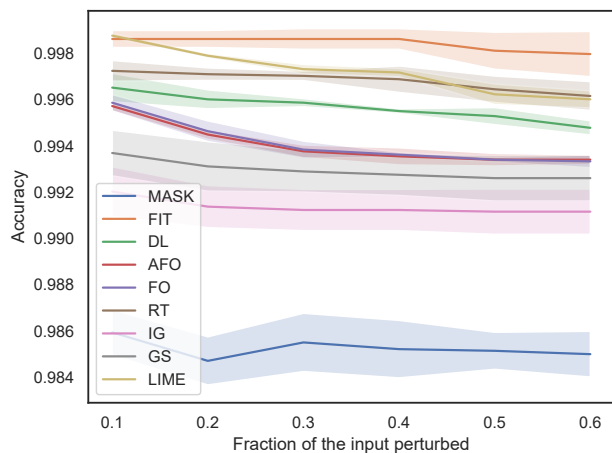


Figure 5. ACC as a function of  $a$  for the MIMIC experiment.

**Discussion** The results are shown in Figure 4 & 5. The ob-

servations selected by Dynamask have the most significant impact when replaced. The high accuracy suggests that the perturbation we use (replacing the most important observations by their time average) rarely produces a counterfactual input.

#### 4. Conclusion

In this paper, we introduced Dynamask, a saliency method specifically designed for multivariate time series. These masks are endowed with an insightful information theoretic interpretation and offer a neat improvement in terms of performance. Dynamask has immediate applications in medicine and finance, where black-box predictions require more transparency.

For future works, it would be interesting to design more sophisticated consistency tests for saliency methods in a dynamic setting, like the ones that exist in image classification (Adebayo et al., 2018). This could be used to study the advantages or disadvantages of our method in more details. Another interesting avenue would be to investigate what the dynamic setting can offer to provide richer explanations with some treatment of causality (Moraffah et al., 2020).

#### Acknowledgments

The authors are grateful to Ioana Bica, James Jordon, Yao Zhang and the 4 anonymous ICML reviewers for their useful comments on an earlier version of the manuscript. Jonathan Crabbé would like to acknowledge Bilyana Tomova for many insightful discussions and her constant support. Jonathan Crabbé is funded by Aviva. Mihaela van der Schaar is supported by the Office of Naval Research (ONR), NSF 1722516.

## References

- Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., and Kim, B. Sanity Checks for Saliency Maps. *Advances in Neural Information Processing Systems*, 2018-December:9505–9515, oct 2018.
- Alvarez-Melis, D. and Jaakkola, T. S. On the Robustness of Interpretability Methods. *arXiv*, jun 2018.
- Baehrens, D., Harmeling, S., Kawanabe, M., Hansen Khansen, K., and Edward Rasmussen, C. How to Explain Individual Classification Decisions. *Journal of Machine Learning Research*, 11(61):1803–1831, 2010. ISSN 1533-7928.
- Barredo Arrieta, A., Díaz-Rodríguez, N., Del Ser, J., Benetot, A., Tabik, S., Barbado, A., Garcia, S., Gil-Lopez, S., Molina, D., Benjamins, R., Chatila, R., and Herrera, F. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information Fusion*, 58(December 2019):82–115, 2020. ISSN 15662535. doi: 10.1016/j.inffus.2019.12.012.
- Caruana, R., Lou, Y., Gehrke, J., Koch, P., Sturm, M., and Elhadad, N. Intelligible models for healthcare: Predicting pneumonia risk and hospital 30-day readmission. In *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, volume 2015-Augus, pp. 1721–1730, New York, NY, USA, aug 2015. Association for Computing Machinery. ISBN 9781450336642. doi: 10.1145/2783258.2788613.
- Castro, J., Gómez, D., and Tejada, J. Polynomial calculation of the Shapley value based on sampling. *Computers and Operations Research*, 36(5):1726–1730, may 2009. ISSN 03050548. doi: 10.1016/j.cor.2008.04.004.
- Chen, J., Song, L., Wainwright, M. J., and Jordan, M. I. Learning to Explain: An Information-Theoretic Perspective on Model Interpretation. *35th International Conference on Machine Learning, ICML 2018*, 2:1386–1418, feb 2018.
- Ching, T., Himmelstein, D. S., Beaulieu-Jones, B. K., Kalinin, A. A., Do, B. T., Way, G. P., Ferrero, E., Agapow, P.-M., Zietz, M., Hoffman, M. M., Xie, W., Rosen, G. L., Lengerich, B. J., Israeli, J., Lanchantin, J., Woloszynek, S., Carpenter, A. E., Shrikumar, A., Xu, J., Cofer, E. M., Lavender, C. A., Turaga, S. C., Alexandari, A. M., Lu, Z., Harris, D. J., DeCaprio, D., Qi, Y., Kundaje, A., Peng, Y., Wiley, L. K., Segler, M. H. S., Boca, S. M., Swamidass, S. J., Huang, A., Gitter, A., and Greene, C. S. Opportunities and obstacles for deep learning in biology and medicine. *Journal of The Royal Society Interface*, 15(141):20170387, apr 2018. ISSN 1742-5689. doi: 10.1098/rsif.2017.0387.
- Choi, E., Bahadori, M. T., Kulas, J. A., Schuetz, A., Stewart, W. F., and Sun, J. RETAIN: An Interpretable Predictive Model for Healthcare using Reverse Time Attention Mechanism. *Advances in Neural Information Processing Systems*, pp. 3512–3520, 2016. ISSN 10495258.
- Cover, T. M. and Thomas, J. A. *Elements of Information Theory*. Wiley, 2005. ISBN 9780471241959. doi: 10.1002/047174882X.
- Das, A. and Rad, P. Opportunities and Challenges in Explainable Artificial Intelligence (XAI): A Survey. *arXiv*, jun 2020.
- Fong, R., Patrick, M., and Vedaldi, A. Understanding deep networks via extremal perturbations and smooth masks. *Proceedings of the IEEE International Conference on Computer Vision*, 2019-October:2950–2958, 2019. ISSN 15505499. doi: 10.1109/ICCV.2019.00304.
- Fong, R. C. and Vedaldi, A. Interpretable Explanations of Black Boxes by Meaningful Perturbation. *Proceedings of the IEEE International Conference on Computer Vision*, 2017-October:3449–3457, 2017. ISSN 15505499. doi: 10.1109/ICCV.2017.371.
- Gimenez, J. R. and Zou, J. Discovering Conditionally Salient Features with Statistical Guarantees. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:4140–4152, may 2019.
- Guo, T., Lin, T., and Antulov-Fantulin, N. Exploring Interpretable LSTM Neural Networks over Multi-Variable Data. *36th International Conference on Machine Learning, ICML 2019*, 2019-June:4424–4440, may 2019.
- Ho, L. V., Aczon, M. D., Ledbetter, D., and Wetzel, R. Interpreting a Recurrent Neural Network’s Predictions of ICU Mortality Risk. *Journal of Biomedical Informatics*, pp. 103672, may 2019. doi: 10.1016/j.jbi.2021.103672.
- Ismail, A. A., Gunady, M., Pessoa, L., Bravo, H. C., and Feizi, S. Input-Cell Attention Reduces Vanishing Saliency of Recurrent Neural Networks. *Advances in Neural Information Processing Systems*, 32, oct 2019.
- Ismail, A. A., Gunady, M., Bravo, H. C., and Feizi, S. Benchmarking Deep Learning Interpretability in Time Series Predictions. *Advances in Neural Information Processing Systems*, 2020.
- Jain, S. and Wallace, B. C. Attention is not explanation. In *NAACL-HLT*, 2019.
- John, G. H., Kohavi, R., and Pflieger, K. Irrelevant Features and the Subset Selection Problem. In *Machine Learning Proceedings 1994*, pp. 121–129. Elsevier, jan 1994. doi: 10.1016/b978-1-55860-335-6.50023-4.

- Johnson, A. E., Pollard, T. J., Shen, L., Lehman, L. W. H., Feng, M., Ghassemi, M., Moody, B., Szolovits, P., Anthony Celi, L., and Mark, R. G. MIMIC-III, a freely accessible critical care database. *Scientific Data*, 3(1):1–9, may 2016. ISSN 20524463. doi: 10.1038/sdata.2016.35.
- Karpathy, A., Johnson, J., and Fei-fei, L. Visualizing and Understanding Recurrent Networks. In *ICLR*, 2016.
- Kindermans, P.-J., Hooker, S., Adebayo, J., Alber, M., Schütt, K. T., Dähne, S., Erhan, D., and Kim, B. The (Un)reliability of saliency methods. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 11700 LNCS:267–280, nov 2017.
- Kwon, B. C., Choi, M.-J., Kim, J. T., Choi, E., Kim, Y. B., Kwon, S., Sun, J., and Choo, J. RetainVis: Visual Analytics with Interpretable and Interactive Recurrent Neural Networks on Electronic Medical Records. *IEEE Transactions on Visualization and Computer Graphics*, 25(1): 299–309, may 2018. doi: 10.1109/TVCG.2018.2865027.
- Lipton, Z. C. The Mythos of Model Interpretability. *Communications of the ACM*, 61(10):35–43, jun 2016.
- Lundberg, S. and Lee, S.-I. A Unified Approach to Interpreting Model Predictions. *Advances in Neural Information Processing Systems*, pp. 4766–4775, 2017.
- Lundberg, S. M., Nair, B., Vavilala, M. S., Horibe, M., Eisses, M. J., Adams, T., Liston, D. E., Low, D. K. W., Newman, S. F., Kim, J., and Lee, S. I. Explainable machine-learning predictions for the prevention of hypoxaemia during surgery. *Nature Biomedical Engineering*, 2(10):749–760, oct 2018. ISSN 2157846X. doi: 10.1038/s41551-018-0304-0.
- MacKay, D. J. C. *Information Theory, Inference and Learning Algorithms*, volume 13. Press, Cambridge University, 2003. ISBN 0521642981.
- Moraffah, R., Karami, M., Guo, R., Raglin, A., and Liu, H. Causal Interpretability for Machine Learning - Problems, Methods and Evaluation. *ACM SIGKDD Explorations Newsletter*, 22(1):18–33, 2020. ISSN 1931-0145. doi: 10.1145/3400051.3400058.
- Phillips, L., Goh, G., and Hodas, N. Explanatory Masks for Neural Network Interpretability. In *IJCAI Workshop on Explainable Artificial Intelligence (XAI)*, pp. 1–4, 2018.
- Ribeiro, M. T., Singh, S., and Guestrin, C. "why should I trust you?": Explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016*, pp. 1135–1144, 2016a.
- Ribeiro, M. T., Singh, S., and Guestrin, C. Model-agnostic interpretability of machine learning. *ArXiv*, abs/1606.05386, 2016b.
- Shannon, C. E. A Mathematical Theory of Communication. *Bell System Technical Journal*, 27(4):623–656, 1948. ISSN 00058580. doi: 10.1002/j.1538-7305.1948.tb00917.x.
- Shrikumar, A., Greenside, P., and Kundaje, A. Learning Important Features Through Propagating Activation Differences. *34th International Conference on Machine Learning, ICML 2017*, 7:4844–4866, apr 2017.
- Siddiqui, S. A., Mercier, D., Munir, M., Dengel, A., and Ahmed, S. TSViz: Demystification of Deep Learning Models for Time-Series Analysis. *IEEE Access*, 7:67027–67040, feb 2019. ISSN 21693536. doi: 10.1109/ACCESS.2019.2912823.
- Simonyan, K., Vedaldi, A., and Zisserman, A. Deep Inside Convolutional Networks: Visualising Image Classification Models and Saliency Maps. *2nd International Conference on Learning Representations, ICLR 2014 - Workshop Track Proceedings*, dec 2013.
- Song, H., Rajan, D., Thiagarajan, J. J., and Spanias, A. Attend and Diagnose: Clinical Time Series Analysis using Attention Models. *32nd AAAI Conference on Artificial Intelligence, AAAI 2018*, pp. 4091–4098, nov 2017.
- Sundararajan, M., Taly, A., and Yan, Q. Axiomatic Attribution for Deep Networks. *34th International Conference on Machine Learning, ICML 2017*, 7:5109–5118, mar 2017.
- Suresh, H., Hunt, N., Johnson, A., Celi, L. A., Szolovits, P., and Ghassemi, M. Clinical Intervention Prediction and Understanding using Deep Networks. *arXiv*, 2017.
- Tjoa, E. and Guan, C. A Survey on Explainable Artificial Intelligence (XAI): Toward Medical XAI. *IEEE Transactions on Neural Networks and Learning Systems*, pp. 1–21, 2020. ISSN 2162-237X. doi: 10.1109/tnnls.2020.3027314.
- Tonekaboni, S., Joshi, S., Campbell, K., Duvenaud, D., and Goldenberg, A. What went wrong and when? Instance-wise Feature Importance for Time-series Models. In *Advances in Neural Information Processing Systems*, 2020.
- Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*, volume 2017-December, pp. 5999–6009. Neural information processing systems foundation, jun 2017.

- Xu, Y., Biswal, S., Deshpande, S. R., Maher, K. O., and Sun, J. RAIM: Recurrent Attentive and Intensive Model of Multimodal Patient Monitoring Data. *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 18:2565–2573, jul 2018.
- Yang, Y., Tresp, V., Wunderle, M., and Fasching, P. A. Explaining therapy predictions with layer-wise relevance propagation in neural networks. In *Proceedings - 2018 IEEE International Conference on Healthcare Informatics, ICHI 2018*, pp. 152–162. Institute of Electrical and Electronics Engineers Inc., jul 2018. ISBN 9781538653777. doi: 10.1109/ICHI.2018.00025.
- Yoon, J., Jordon, J., and Van Der Schaar, M. INVASE: Instance-wise Variable Selection using Neural Networks. In *ICLR*, 2019a.
- Yoon, J., Jordon, J., and van der Schaar, M. ASAC: Active Sensing using Actor-Critic models. *Machine Learning for Healthcare Conference*, pp. 1–18, jun 2019b.
- Zhang, Q. and Zhu, S.-C. Visual Interpretability for Deep Learning: a Survey. *Frontiers of Information Technology and Electronic Engineering*, 19(1):27–39, feb 2018.