Keep the Faith: Faithful Explanations in Convolutional Neural Networks for Case-Based Reasoning

Tom Nuno Wolf[1, 2, 3], **Fabian Bongratz**[1, 2, 3], **Anne-Marie Rickmann**[1, 2], **Sebastian Pölsterl**[2], **Christian Wachinger**[1, 2, 3]

[1]Department of Radiology, Technical University Munich, Munich, Germany
[2]Lab for Artificial Intelligence in Medical Imaging, Ludwig-Maximilians-University, Munich, Germany
[3]Munich Center for Machine Learning (MCML)
tom_nuno.wolf@tum.de

Abstract

Explaining predictions of black-box neural networks is crucial when applied to decision-critical tasks. Thus, attribution maps are commonly used to identify important image regions, despite prior work showing that humans prefer explanations based on similar examples. To this end, ProtoPNet learns a set of class-representative feature vectors (prototypes) for case-based reasoning. During inference, similarities of latent features to prototypes are linearly classified to form predictions and attribution maps are provided to explain the similarity. In this work, we evaluate whether architectures for case-based reasoning fulfill established axioms required for faithful explanations using the example of ProtoPNet. We show that such architectures allow the extraction of faithful explanations. However, we prove that the attribution maps used to explain the similarities violate the axioms. We propose a new procedure to extract explanations for trained ProtoPNets, named ProtoPFaith. Conceptually, these explanations are Shapley values, calculated on the similarity scores of each prototype. They allow to faithfully answer which prototypes are present in an unseen image and quantify each pixel's contribution to that presence, thereby complying with all axioms. The theoretical violations of ProtoPNet manifest in our experiments on three datasets (CUB-200-2011, Stanford Dogs, RSNA) and five architectures (ConvNet, ResNet, ResNet50, WideResNet50, ResNeXt50). Our experiments show a qualitative difference between the explanations given by ProtoPNet and ProtoPFaith. Additionally, we quantify the explanations with the Area Over the Perturbation Curve, on which ProtoPFaith outperforms ProtoPNet on all experiments by a factor $>10^3$.

Introduction

With continued progress in deep learning, AI models become ubiquitous in daily life. For many tasks, they are able to outperform humans (Shin et al. 2023). At the same time, it is difficult for humans to understand the decision-making of such complex black-box models with millions of parameters (Adadi and Berrada 2018; Arrieta et al. 2020). This fundamental issue is particularly relevant in decision-critical areas, such as medicine, finance, or justice. To increase transparency, methods have been developed that try to locally approximate the decision-making of neural networks in a *post-hoc* fashion (Vale, El-Sharif, and Ali 2022). However, such explanations can vary dramatically between explanation techniques and models (Rudin 2019): For example, pixel-wise attribution maps aim at highlighting the contribution of each pixel to the output logit, which is realized in Simonyan, Vedaldi, and Zisserman (2013) via propagation of gradients from the output logit to the input pixels. Such approaches are directly influenced by the model parameters. Alternatively, classifying image latents with a k-nearest-neighbor (kNN) algorithm allows showing the k most similar images as explanations. Although the theoretic motivation of pixel-wise attributions is to mimic the visual cortex, prior studies (Jeyakumar et al. 2020; Kim et al. 2023; Nguyen, Kim, and Nguyen 2021) have concluded that example-based explanations are easiest to understand for humans. While case-based reasoning, as in kNN, answers *what* contributed to a prediction, it does not allow to explain *how* the model transformed the image to arrive at its prediction. In other words, the explanations give insight into the decision-making of the model but lack to give insight into the high-dimensional non-linear function that a neural network represents. To compare and analyze explanation techniques from a theoretical point of view, several axioms have been introduced previously (Lundberg and Lee 2017; Sundararajan, Taly, and Yan 2017). In this work, we refer to an explanation method that satisfies all axioms as *faithful*.

ProtoPNet (Chen et al. 2019) is a prominent implementation of case-based reasoning and has been widely adopted, for image classification (Chen et al. 2019; Donnelly, Barnett, and Chen 2022) and decision-critical tasks, like automated diagnosis from chest radiography (Kim et al. 2021) and dementia-diagnosis (Wolf, Pölsterl, and Wachinger 2023). It linearly classifies similarities between trainable class-representative feature vectors (*prototypes*) and latent features of unseen images. Importantly, its inherently interpretable architecture allows the extraction of explanations on a pixel-level. The model's decision-making is explained by visualizing image crops that each prototype represents next to pixel-level explanations of the unseen image, as seen in Fig. 1, leading to the type of explanations that humans prefer (Jeyakumar et al. 2020; Kim et al. 2023; Rudin 2019). Therefore, its pixel-level explanations can, supposedly, be incorporated to answer the question *how* it came to the con-

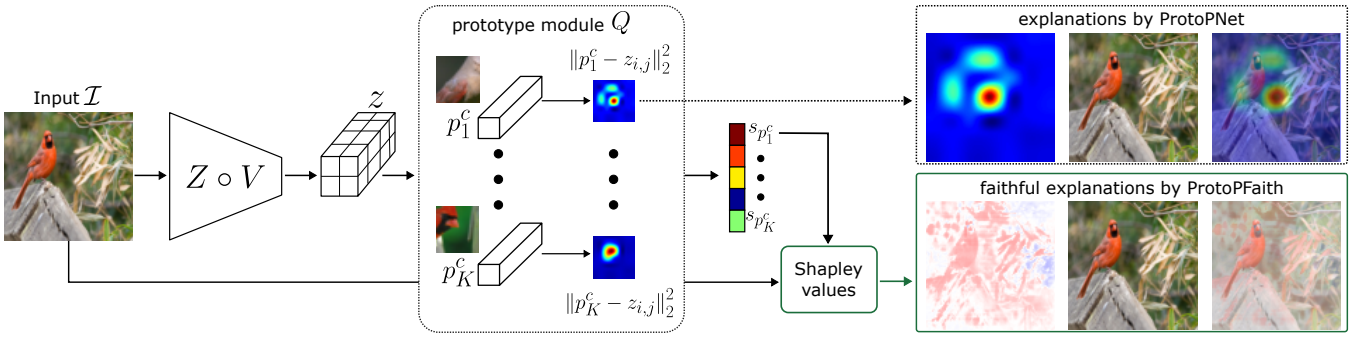

Figure 1: In ProtoPNet (Chen et al. 2019), an image $\mathcal{I}$ is classified by feeding a vector of distance scores $s_{p_k^c}$ through a linear layer (omitted for clarity). Each distance represents the minimum distance between a trainable prototype p_k^c to each spatial latent feature vector in z. An explanation in ProtoPNet is the up-scaled distance map of a prototype $\|p_k^c - z_{i,j}\|_2^2$ overlayed onto the input image. In this work, we demonstrate that these explanations do not adhere to established explainability axioms and, thus, cannot be considered faithful representations of the underlying processes. We propose to resolve this shortcoming of ProtoPNet with faithful explanations based on decomposing distance scores in terms of Shapley values w.r.t. the input image.

clusion that the unseen image belongs to a certain class.

In this work, we evaluate whether the explanations given by ProtoPNet are perfectly faithful. Our theoretical examination shows that case-based reasoning, as implemented by ProtoPNet, i.e., classification based on similarities, allows for faithful explanations. In contrast, the explanations on the image level assume spatial dependency of latent feature maps and the image space, which breaks in the general case of convolutional neural network (CNN) backbones. Therefore, we propose to exploit the architectural properties of ProtoPNet to extract attribution maps that faithfully show the contribution of each pixel to the similarities: We transform a trained ProtoPNet into a lightweight probabilistic model (Gast and Roth 2018) and extract Shapley values with DASP (Ancona, Oztireli, and Gross 2019). Additionally, we demonstrate that the theoretical violations of ProtoPNet arise in real-world applications. We qualitatively show differences in explanations between ProtoPNet and our proposed procedure, named ProtoPFaith, on three established datasets (CUB-200-2011 (Wah et al. 2011), Stanford Dogs (Aditya et al. 2011), RSNA (Shih et al. 2019)) and on five CNN backbones (ConvNet, ResNet, ResNet50, Wide-ResNet50, ResNeXt50). Additionally, we quantify the difference with the Area Over the Perturbation Curve (AOPC) (Tomsett et al. 2020).

Our key contributions are as follows:

- We prove that the explanations generated by ProtoPNet-like architectures are not faithful to the decision-making of the model.
- Instead, we propose to leverage the architecture of case-based reasoning implemented by ProtoPNet to extract explanations based on Shapley values.
- ProtoPFaith allows efficient extraction of explanations for high-dimensional image inputs and requires conversion of layers into probabilistic layers, for which we derive closed-form solutions for mean and variance if they are not readily available in the literature (i.e., squared L2-norm, ReLU1).
- We empirically demonstrate that there are substantial differences, both qualitative and quantitative, between the explanations extracted with ProtoPNet and ProtoPFaith.
- Explanations of ProtoPFaith outperform ProtoPNet on the AOPC score by a factor $>10^3$ for all experiments.

Axiomatic Evaluation of Explanations for Case-Based Reasoning

First, we evaluate explanations found in the literature with respect to the axioms required for faithful explanations. Then, we investigate the architecture of case-based reasoning with the example of ProtoPNet and show why some axioms are violated.

Related Work: Pixel-Level Explanation Methods for CNNs. Typically, explanations ought to represent each input feature's effect on the prediction. Lundberg and Lee (2017) and Sundararajan, Taly, and Yan (2017) independently proposed a set of useful axioms that a deep learning attribution method should fulfill, which can be summarized as:

- **Sensitivity:** If there is a change in the value of a feature and the prediction, the attribution of that feature should not be zero.
- **Implementation Invariance:** Attributions for two models whose predictions are identical for all inputs should be identical.
- **Completeness:** Attributions of two inputs should add up to the difference in the model output.
- **Dummy:** If the prediction of a model is independent of a feature, its attribution should always be zero.
- **Linearity:** If a model f is a linear combination of other models ($f = af_1 + bf_2$), the attributions of f should follow the same linear combination.
- **Symmetry-Preserving:** If the prediction of a model is identical when replacing two input features with one another, the attribution of both features should be equal.

Gradient-based methods are characterized by a backward pass of the neural network used to calculate the gradient of a prediction with respect to input features, which can be used to create feature attribution maps (Simonyan, Vedaldi, and Zisserman 2013). However, as demonstrated by Sundararajan, Taly, and Yan (2017), gradient-based feature attribution methods violate either the **Sensitivity** axiom due to vanishing gradients in the ReLU activation, concerning methods like Baehrens et al. (2010), Simonyan, Vedaldi, and Zisserman (2013), Springenberg et al. (2014), Zeiler and Fergus (2014), or **Implementation Invariance** if the gradient is computed in a discrete fashion, as in Binder et al. (2016) and Shrikumar et al. (2016). Moreover, Shah, Jain, and Netrapalli (2021) showed that gradient-based methods tend to suffer from feature leakage, i.e., the contribution of unrelated features to the prediction. *Activation-based methods* (Chattopadhay et al. 2018; Desai and Ramaswamy 2020; Fu et al. 2020; Jiang et al. 2021; Selvaraju et al. 2017; Wang et al. 2020; Zhou et al. 2016), on the other hand, leverage neuron activations in (usually deep/final) convolutional layers to assess the importance of input regions for the network output. Even though being widely established in practice, these methods typically violate **Sensitivity** and/or **Completeness** (Fu et al. 2020), which renders their applicability to decision-critical tasks questionable.

Different from gradient- and activation-based methods, *perturbation-based methods* mask or alter an input image feature and calculate the difference to the output of the original input. Existing work proposed to occlude (Zeiler and Fergus 2014), marginalize with a sliding window (Zintgraf et al. 2017), randomly perturb (Petsiuk, Das, and Saenko 2018), or occlude parts of an input image with perturbation space-exploration (Fel et al. 2022). In Dabkowski and Gal (2017), Fong and Vedaldi (2017), and Ribeiro, Singh, and Guestrin (2016), the black-box predictor is approximated by an interpretable model locally and super-pixel explanations summarized to the global input.

Another perturbation-based approach is based on *Shapley values* (Shapley 1953), which originate from cooperative game theory; each feature is treated as a player in a game and contributes to the final prediction. Removing a player i from all possible coalitions $S \subseteq P$ of a set of players P , i.e., marginalizing a player, yields its contribution ψ_i to the set function $\hat{f} : P \to \mathbb{R}$:

$$\psi_i = \sum_{S \subseteq P \setminus \{i\}} \frac{|S|!(|P| - |S| - 1)!}{|P|!} \left(\hat{f}(S \cup \{i\}) - \hat{f}(S) \right).$$

In contrast to other attribution methods, Shapley values satisfy *all* axioms and, thus, provide faithful explanations (Covert, Lundberg, and Lee 2020; Lundberg and Lee 2017; Sundararajan, Taly, and Yan 2017; Zheng et al. 2022). Specifically, they satisfy **Completeness**, as the sum of contributions of all players equals the prediction:

$$\sum_i \psi_i = \hat{f}(P) - \hat{f}(\emptyset). \tag{1}$$

When aiming to compute Shapley values for an input image $\mathcal{I}$ of a DNN, $\hat{f}(S)$ signifies the output of the DNN when all pixels not in S are replaced by a baseline value. Approximating Shapley values has gained a lot of attention (Ancona, Oztireli, and Gross 2019; Lundberg and Lee 2017; Shrikumar, Greenside, and Kundaje 2017; Štrumbelj and Kononenko 2014; Sundararajan, Taly, and Yan 2017; Wang et al. 2022), as their exact calculation grows exponentially with the number of input features, with DASP (Ancona, Oztireli, and Gross 2019) outperforming all other approximation methods in terms of approximation error.

Preliminary: Interpretability with ProtoPNet-like Architectures. ProtoPNet, outlined in Fig. 1, implements case-based reasoning as a function $f(\mathcal{I}) = (F \circ Q \circ Z \circ V)(\mathcal{I})$ mapping an input image $\mathcal{I} \in \mathbb{R}^{H \times W \times \tilde{C}}$ to a set of output logits. More precisely, a CNN encoder $V : \mathbb{R}^{H \times W \times \tilde{C}} \to \mathbb{R}^{H' \times W' \times C'}$, extract features and is typically pre-trained on the desired task (H, W, $\tilde{C}$ the spatial height, width, and channel dimension of the input image; H', W', C' the latent feature map height, width, and channel dimensions). The feature extractor $Z : \mathbb{R}^{H' \times W' \times C'} \to \mathbb{R}^{H' \times W' \times L}$ maps to a latent feature map z, which matches the channel-size L of prototypes, and consists of two 1×1 convolutions and non-linearities. The prototype module $Q : \mathbb{R}^{H' \times W' \times L} \to \mathbb{R}^{K \cdot C}$ extracts a distance vector s for K prototypes per class C. It consists of the minimum distances $s_{p_k^c}$ of the squared L2-norm of each prototype to all spatial latent feature vectors $z_{i,j}$ in z:

$$\begin{aligned} s_{p_k^c}(\mathcal{I}) &= \min_{i=1,\dots,H',j=1,\dots,W'} \|p_k^c - z_{i,j}\|_2^2 \\ &= \min_{i=1,\dots,H',j=1,\dots,W'} \sum_{l=1}^{L} \left(p_{k,l}^c - z_{i,j,l}\right)^2, \end{aligned} \tag{2}$$

with $p_k^c \in \mathbb{R}^{1 \times 1 \times L}$ and $z = (Z \circ V)(\mathcal{I}) \in \mathbb{R}^{H' \times W' \times L}$. Lastly, the classification layer $F : \mathbb{R}^{K \cdot C} \to \mathbb{R}^C$ is implemented with a single linear layer.

Importantly, prototypes p_k^c are trainable parameters (vectors) of the network. After a certain number of iterations, each prototype is replaced by the closest (squared L2-distance) latent feature vector $z_{i,j}$ extracted from all samples of the training set that are of class c. Thus, a prototype p_k^c represents exactly one class-representative latent feature vector $z_{i,j}$ from a training image. An unseen image is classified by feeding the distance vector s (see Fig. 1) to the classification layer F.

The *distance map* $\|p_k^c - z_{i,j}\|_2^2$ is utilized to extract pixel-level explanations. First, the maximum distance of the distance map is selected, which is globally defined for all possible inputs if the last layer of the feature extractor Z is a bounded non-linearity. The maximum distance is subtracted by the values of the distance map[1]. Then, the flipped distance map is up-scaled to the input image size and overlaid with the input image, forming an *attribution map*. A prototype is visualized by extracting the image crop around the 95%-percentile region of the corresponding attribution map.

[1]Originally, Chen et al. (2019) used a log-activation instead, which we discuss in detail in the next section.

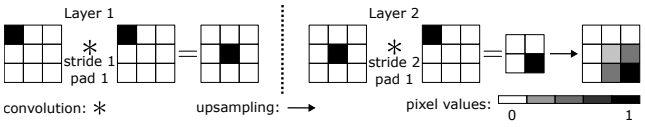

Figure 2: An up-sampled latent feature map itself cannot serve as a spatial indicator of pixel attribution in the general case.

Theoretic Evaluation of the Faithfulness of ProtoPNet Explanations. The decision-making of ProtoPNet is modeled as a linear function F, which maps distances $s_{p_k^c}$ to class logits. A linear mapping satisfies all introduced axioms (Štrumbelj and Kononenko 2014). Therefore, the model prediction, i.e., the decision-making of the model, is faithful with respect to the distance vector s. However, ProtoPNet adds a log-activation to the prototype module, i.e., $Q'(s_{p_k^c}(\mathcal{I})) = \log((s_{p_k^c}(\mathcal{I}) + 1)/(s_{p_k^c}(\mathcal{I}) + \epsilon_{\to 0}))$. This introduces non-linearity between the distance maps (which are used to yield attribution maps) and the classification. Therefore, the **Linearity** axiom is not preserved.

As explained in the previous section, attribution maps are the result of up-scaling distance maps from the latent space to the image space. This process implies a spatial relationship between distances in latent space and image space, as the distances of prototypes to latent feature vectors are calculated over the spatial dimensions in the latent space. In established CNN architectures used in ProtoPNet (VGG (Simonyan and Zisserman 2015), ResNet (He et al. 2016), DenseNet (Huang et al. 2017)), the locality of latent features is, however, lost after a few layers, i.e., the receptive field of each latent feature is the whole input image. We now prove by counterexample that there is no general spatial dependency between latent feature maps and the input image space:

Proof. Suppose a CNN consists of two convolutional layers (one input and output channel with linear activation) with kernel weights $\theta_1 \in \{0,1\}^{3\times3\times1}$ and $\theta_2 \in \{0,1\}^{3\times3\times1}$, as depicted in Fig. 2. Suppose the first layer has a stride and padding of one, and the second layer has a stride of two and padding of one. Both kernel weights are identical, with zeros everywhere beside the top left value, which is one. Feeding an exemplary input of size 3×3, consisting of zeros with the exception of the top left feature set to one, to this network yields an output of size 2×2, which consists of zeros and the bottom right feature activated at one. Upscaling this latent feature map to the input dimension, as done in ProtoPNet, creates an attribution map that suggests the bottom right part of the input to be relevant. However, the only feature that activated the bottom right latent feature is the top left feature in the input space. Thus, there is no spatial relation between the features of the input space and the latent space. □

As a result, features that do not affect the latent features, and should therefore be treated as dummies, have a high attribution. Therefore, the **Dummy** axiom is not fulfilled.

Additionally, ProtoPNet explanations violate the axiom **Completeness**, as each distance $s_{p_k^c}$ is exactly the distance of *one* latent feature vector of the latent feature map, i.e., $\min_{i=0,\dots,H',j=0,\dots,W'} \|p_k^c - z_{i,j}\|_2^2$. Therefore, the sum of pixel-level attributions does not equal the distance. For the same reason, the attribution maps violate the **Dummy** axiom. Lastly, thresholding the attribution map with the 95%-percentile violates **Sensitivity**: As the percentile is chosen heuristically, it suppresses the attributions of some input features that have non-zero attribution. In contrast, we will show that ProtoPFaith calculates attribution maps with respect to the minimum distance $s_{p_k^c}$ faithfully, which manifests in our experiments.

In summary, the decision-making of ProtoPNet requires a linear activation to be faithful, while the explanations provided on an image-level break the overall faithfulness of the model, as they break **Completeness**, **Dummy**, and **Sensitivity**. Next, we introduce ProtoPFaith to extract faithful explanations from ProtoPNet, rendering it feasible for decision-critical tasks.

Methods

As described in the previous section, the explanations given by ProtoPNet are only faithful if **Linearity** of the classification w.r.t. minimum distances $s_{p_k^c}$ can be restored, and attribution maps are faithful to changes in the minimum distances themselves. We showed that the attribution maps are not faithful. In contrast, Shapley values are guaranteed to satisfy *all* required axioms, albeit their exact calculation requires exponentially many model evaluations. To this end, DASP (Ancona, Oztireli, and Gross 2019) has shown to approximate true Shapley values with only few model evaluations, rendering it feasible for high-dimensional image inputs while outperforming competing methods in terms of approximation error. We show below how to convert ProtoPNet with this framework. Our proposed method to extract explanations, named ProtoPFaith, is an important step towards a transparent ProtoPNet, allowing to faithfully explain the case-based decision-making referenced by Chen et al. (2019) as "*this* looks like *that*". The source code is available at https://github.com/ai-med/KeepTheFaith.

Leveraging Faithful Explanations in ProtoPNet. We propose to restore **Linearity** of the classification of the distance vector s by dropping the log-activation introduced by ProtoPNet (note that the minimum of the squared L2-norm is zero). Instead of showing up-scaled log-activations of distance maps and the related weight of the classification layer, we need to calculate a contribution score Ψ_k of a similarity to the log-probability:

$$
\begin{aligned}
&\log(P(y=c \mid \mathcal{I})) \\
&= \log\left(\frac{\exp\sum_{k=1}^{K} -a_{c,k} s_{p_k^c}(\mathcal{I})}{\sum_{\hat{c}=1}^{C}\exp\sum_{k=1}^{K} -a_{\hat{c},k} s_{p_k^{\hat{c}}}(\mathcal{I})}\right) \\
&= \log\left(\frac{\exp\sum_{k=1}^{K} -a_{c,k} s_{p_k^c}(\mathcal{I})}{R}\right) \\
&= \sum_{k=1}^{K} -a_{c,k} s_{p_k^c}(\mathcal{I}) - \log R
\end{aligned}
$$

$$= \sum_{k=1}^{K} \Psi_k, \text{ with } \Psi_k = -a_{c,k} s_{p_k^c}(\mathcal{I}) - \frac{\log R}{K},$$

where $a_{c,k}$ are the coefficients of the linear layer F and y is the target label of the image $\mathcal{I}$. Note that we use the contribution score to compare the contribution of prototypes of the desired class, i.e. high contribution scores contribute more than low contribution scores.

In ProtoPNet, we are interested in the portion of an image that is representative of a prototype. Therefore, we propose to utilize the implementation for case-based reasoning introduced by ProtoPNet to extract Shapley values via the minimum distance $s_{p_k^c}$. After network training, precisely one image exists in the training set from which a prototype arises. As a result, we can visualize a prototype faithfully by calculating the Shapley values w.r.t. the distance between this training image and the prototype (distance is 0 for this image). The resulting explanations highlight the pixels that are compressed into a prototype.

Likewise, we calculate the Shapley values of pixels w.r.t. the squared L2-distance of a prototype and visualize them for unseen test images. As the vector of minimum distances s is fed through the final linear layer only, we can use the **Linearity** axiom to transform similarity-based Shapley values into attribution w.r.t. the model prediction. This would be the weighted sum of all attribution maps. To allow quantitative comparison between pixel attributions of different models and prototypes, we opt for a bounded non-linearity before calculating the distances. Furthermore, a bounded non-linearity, such as ReLU1 (Eq. 4), helps for faster model convergence and allows us to derive the first and second-order moments analytically, which is required for DASP.

Approximation of Shapley Values with DASP. Recently, advances in uncertainty propagation for DNNs (Gast and Roth 2018) were applied to approximate Shapely values in DASP (Ancona, Oztireli, and Gross 2019) with a few network evaluations. This is desirable when the number of input features is high, enabling efficient approximation of Shapley values of a whole input image.

DASP calculates the contribution $\psi_{i,d}$ of the i-th feature to a coalition S_d of fixed size d by modeling each S_d as an aleatoric uncertainty, which is propagated through a probabilistic network to yield an expected value $\mathbb{E}_d[\psi_{i,d}]$. Then, the expectation of a Shapley value is estimated as:

$$\mathbb{E}[\psi_i] = \frac{1}{|P|} \sum_{d=0}^{|P|-1} \mathbb{E}_d[\psi_{i,d}]. \tag{3}$$

If a DNN can be converted into a probabilistic model, each $\mathbb{E}_d[\psi_{i,d}]$ can be calculated with a single forward pass. To this end, closed-form solutions for mean and variance need to be derived. While they were summarized for standard DNN layers in Ancona, Oztireli, and Gross (2019), they are not yet readily available for the prototype module Q and the ReLU1 non-linearity. Hence, we derive them in the following. Notably, the converted probabilistic model does not need to be trained, as the trained weights of a ProtoPNet are re-used.

Derivation of Lightweight Probabilistic Layers for ProtoPNet. We analytically derive expectation $\mathbb{E}$ and variance $\mathbb{V}$ of a layer input mean μ and variance σ^2 for ReLU1 for $\mathbb{E}(\mu,\sigma) = \int g(x)\frac{1}{\sigma}\phi(\frac{x-\mu}{\sigma})\,dx$ and variance $\mathbb{V}(\mu,\sigma) = \int (g(x)-\mathbb{E}(\mu,\sigma))^2\frac{1}{\sigma}\phi(\frac{x-\mu}{\sigma})\,dx$, with ϕ the standard Gaussian probability density function $\phi(x) = \frac{1}{\sqrt{2\pi}}e^{-\frac{x^2}{2}}$, the corresponding cumulative distribution function $\Phi(x) = \int_{-\infty}^{x}\phi(t)\,dt$ (Frey and Hinton 1999), and $g(x) = \text{ReLU1}(x)$ defined as:

$$\text{ReLU1}(x) = \begin{cases} 0, & \text{if x} < 0, \\ x, & \text{if } 0 \le \text{x} \le 1, \\ 1, & \text{if x} > 1. \end{cases} \tag{4}$$

The expectation and variance are (full derivation in Sec. A.1[2], which can easily be extended to any bound other than 1):

$$\begin{aligned} \mathbb{E}(\mu,\sigma) =: \bar{\mu} &= \sigma\left(\phi\left(-\frac{\mu}{\sigma}\right) - \phi\left(\frac{1-\mu}{\sigma}\right)\right) \\ &+ \mu\left(\Phi\left(\frac{1-\mu}{\sigma}\right) - \Phi\left(-\frac{\mu}{\sigma}\right)\right) + 1 - \Phi\left(\frac{1-\mu}{\sigma}\right) \end{aligned} \tag{5}$$

$$\begin{aligned} \mathbb{V}(\mu,\sigma) = \; & \left(\mu^2 - 2\mu\bar{\mu} + \sigma^2 + 2\bar{\mu} - 1\right)\Phi\left(\frac{1-\mu}{\sigma}\right) \\ & - \left(\mu^2 - 2\mu\bar{\mu} + \sigma^2\right)\Phi\left(-\frac{\mu}{\sigma}\right) \\ & - \left(\mu\sigma - 2\bar{\mu}\sigma + \sigma\right)\phi\left(\frac{1-\mu}{\sigma}\right) \\ & + \left(\mu\sigma - 2\bar{\mu}\sigma\right)\phi\left(-\frac{\mu}{\sigma}\right) \\ & + \bar{\mu}^2 - 2\bar{\mu} + 1 \end{aligned} \tag{6}$$

Further, we derive expectation and variance of $s_{p_k^c}$ (see Eq. 2) as follows: As seen in Ancona, Oztireli, and Gross (2019), we assume independence of the input Gaussian signals $z_{i,j,l}$. Thus, we can represent each latent feature vector $z_{i,j}$ as a multivariate Gaussian with diagonal covariance matrix $\mathbf{\Sigma}_{i,j} \in \mathbb{R}^{L\times L}$ consisting of the individual means $\mu_{i,j,l}$ and variances $\sigma^2_{i,j,l}$, i.e., $z_{i,j} \sim \mathcal{N}(\boldsymbol{\mu}_{i,j}, \mathbf{\Sigma}_{i,j})$, $\boldsymbol{\mu}_{i,j} \in \mathbb{R}^L$. We can then introduce a new multivariate random variable $Y_{i,j}$:

$$Y_{i,j} = z_{i,j} - p_k^c \quad \rightarrow \quad Y_{i,j} \sim \mathcal{N}(\boldsymbol{\mu}_{i,j} - p_k^c, \mathbf{\Sigma}_{i,j}).$$

Now, we calculate the squared L2-norm in Eq. 2 as $Z_{i,j} = \sum_{l=1}^{L} Y^2_{i,j,l} = Y^T_{i,j}Y_{i,j}$, which allows calculating mean and variance via quadratic forms of random variables (Baldessari 1967):

$$\mathbb{E}[Z_{i,j}] = \mathbf{trace}(\mathbf{\Sigma}_{i,j}) + Y^T_{i,j}Y_{i,j} = \tilde{\mu}_{i,j} \tag{7}$$

$$\mathbb{V}[Z_{i,j}] = 2\,\mathbf{trace}(\mathbf{\Sigma}^2_{i,j}) + 4\tilde{\mu}^T_{i,j}\mathbf{\Sigma}_{i,j}\tilde{\mu}_{i,j} = \tilde{\sigma}_{i,j} \tag{8}$$

$$Z_{i,j} \sim \mathcal{N}(\tilde{\mu}_{i,j}, \tilde{\sigma}^2_{i,j}). \tag{9}$$

Finally, we extract the minimum distance with max-pooling over all negated $Z_{i,j}, i = 1,\ldots,H', j = 1,\ldots,W'$, for which mean and variance are given in Ancona, Oztireli, and Gross (2019).

[2] available at https://arxiv.org/abs/2312.09783.

Experiments and Results

We evaluate ProtoPFaith on datasets similar to the ones used by ProtoPNet and its adaptations (Chen et al. 2019; Donnelly, Barnett, and Chen 2022; Kim et al. 2021): CUB-200-2011 (Wah et al. 2011), Stanford Dogs (Aditya et al. 2011), and a subset of RSNA[3] (Shih et al. 2019) consisting of pneumonia and healthy samples only. Tab. 1 reports the classification performance of individual experiments, Sec. A.2 lists training details, and Sec. A.3 and Sec. A.4 present additional results.

Qualitative Evaluation. Fig. 3 visualizes explanations of image predictions from the test sets. Each predicted image was classified correctly by the model. Therefore, we visualize the attributions of the prototypes of that class and the corresponding contribution scores Ψ_k, which allows us to evaluate which prototype contributed most to the prediction (greater Ψ_k).

The model learned duplicate prototypes when trained on CUB-200-2011 and Stanford Dogs. As seen in almost all explanations of prototypes, attribution maps of ProtoPNet appear focused on a small location of an image. Its prototype activations for the test image are less sparse, except for CUB-200-2011, in which some attributions comprise background, while the prototype appeared to capture the animal only. There are no duplicate prototypes for the pneumonia class on the RSNA dataset. However, almost all prototype explanations of ProtoPNet are located in the background of their corresponding image. In contrast, the prototypes found by ProtoPFaith comprise the body and head for most prototypes of birds. For dogs, the attributions of prototypes are not as apparent, but always focus on the hair of the animal around the face. Additionally, the explanations given by ProtoPFaith contain large portions of the background. While the log-activations of ProtoPNet are not helpful for pneumonia on the RSNA dataset, explanations of ProtoPFaith capture parts of the lung, with healthy parts indicating a dissimilarity to the prototype. However, all prototypes in RSNA encode general anatomy like the heart and spine. Attributions of the test image are similar for most prototypes. Additional results are presented in Sec. A.3.

Quantitative Evaluation. We evaluate the explanations given by ProtoPNet and extracted with ProtoPFaith based on the squared L2-distance $s_{p_k^c}$. For each prototype of a model, we select the input image that this prototype originates from (note that per definition of the training procedure, the squared L2-distance $s_{p_k^c}$ between each prototype and its input image equals 0). We iteratively remove the most relevant features (according to the attribution map of this prototype), as proposed in Tomsett et al. (2020) as the Area Over the Perturbation Curve (AOPC):

$$\text{AOPC}(Q \circ Z \circ V) = \frac{1}{C+K+T-1} \sum_{c=1}^{C} \sum_{k=1}^{K} \sum_{t=1}^{T} s_{p_k^c}(\mathcal{I}^{(0)}_{\xi(p_k^c)}) - s_{p_k^c}(\mathcal{I}^{(t)}_{\xi(p_k^c)}), \tag{10}$$

where $\xi(p_k^c)$ denotes a mapping from prototypes to the corresponding index of an image in the training set, and $^{(t)}$ indicates the t most important features removed. It is expected that removing the most important features first leads to a faster decrease in the AOPC (negative values) if explanations are more meaningful. Tab. 1 demonstrates that removing features according to ProtoPFaith yields a decrease in AOPC that differs in several orders of magnitude from removing features according to ProtoPNet. Thus, the explanations given by ProtoPFaith are more accurate and discriminative than the original explanations given by ProtoPNet.

Discussion

As seen in Fig. 3, the explanations of ProtoPNet and ProtoPFaith are very different. While the explanations of ProtoPNet appear sparse, the close approximation of Shapley values given by ProtoPFaith indicates that prototypes learned by the network indeed comprise most of the input domain. The introduction of granularity on a pixel-level prohibits extraction of crops of the input image that are believed to represent a prototype by ProtoPNet. While the proposed explanations are harder to interpret, they are faithful to how the model transformed the input into a prototype and $what$ lead to the classification. For medical applications like pneumonia detection, granularity can be beneficial: When the actual disease is behind ribs, attributions of ProtoPFaith do not comprise the ribs, as seen in Fig. 3. The background explanations found by ProtoPNet would not allow a clinician to learn anything about the model's decision-making. Enlightened by theoretical derivation and the real-world explanations of ProtoPFaith, we can infer that the features encoded in the prototype are indeed just mapped to an arbitrary position in latent space. During training of ResNet on RSNA (see Fig. A1), the prototypes seemed to collapse for each class. This model achieved a test BAcc of 79.8% nevertheless. This is possible because the loss introduced by Chen et al. (2019) does not enforce prototypes of a single class to be distant from one another. With a typical black-box, identifying such unexpected issues would be impossible. Finally, human-understandable explanations, as given by ProtoPNet, are typically designed to fulfill a desired property like sparsity or locality. However, opting for transparency as in ProtoPFaith involves a trade-off, which likely applies to any explainability method. Hence, future work needs to address the question of how to accomplish explanations that are both human-understandable *and* faithful.

Limitations ProtoPFaith is currently only applied to single-label image classification but can be extended to multi-label classification as done by XProtoNet (Kim et al. 2021), in which prototypes only contribute to the prediction of their own class. This allows the calculation of Shapley values w.r.t. the output logit, yielding attribution maps

[3] available at https://www.rsna.org/education/ai-resources-and-training/ai-image-challenge/RSNA-Pneumonia-Detection-Challenge-2018

Dataset	Model	Validation BAcc ↑	Test BAcc ↑	$\text{AOPC}_{\text{ProtoPNet}}$ ↓	$\text{AOPC}_{\text{ProtoPFaith}}$ ↓
CUB-200-2011	ResNeXt50	72.8 ± 0.5	72.0 ± 0.6	-0.000446	-3.721184
Stanford Dogs	ResNeXt50	84.1 ± 0.3	83.4 ± 0.6	-0.001767	-1.850909
RSNA	ConvNet	74.0 ± 10.1	72.6 ± 10.8	-0.001360	-8.816450

Table 1: Balanced Accuracy (BAcc) and AOPC scores for models visualized in Fig. 3. See Tab. A1 and Tab. A2 for full results.

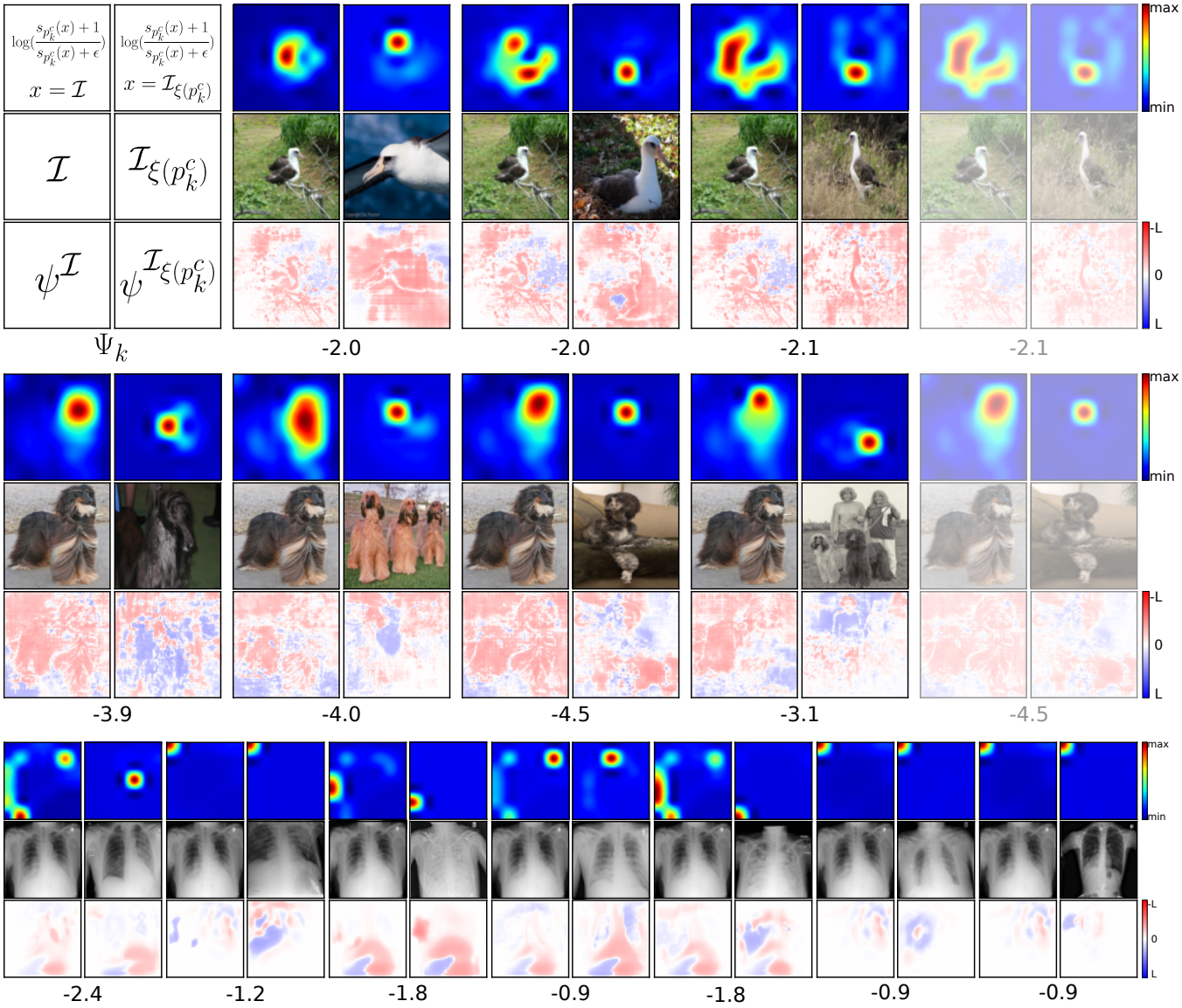

Figure 3: Explanations of the forward pass of: (top) a bird with ResNeXt trained on CUB-200-2011; (middle) a dog with ResNeXt trained on Stanford Dogs; (bottom) ConvNet trained for pneumonia detection on RSNA. We explain the layout of each explanation in the top left with formal notation (left to right, top to bottom): (1) explanation of ProtoPNet for the occurrence of a prototype within the test image; (2) explanation of ProtoPNet for the activation of an image, from which a prototype was extracted; (3) test image; (4) training image, from which the prototype was extracted; (5) explanation of ProtoPFaith for the occurrence of a prototype within the test image; (6) explanation of ProtoPFaith about the image, from which the prototype was extracted. We grayed out duplicate prototypes, which are identical feature vectors. Ψ_k denotes the prototype contribution towards log-probability. See Sec. A.4 for more visual results.

in image space for each output logit. Using this attribution map, we would be able to infer the attribution map of each prototype via using the **Linearity** property reversely, which would speed up the computation of attribution maps (see Sec. A.2).

Conclusion

We elucidated the conceptual and theoretical problems of explanations for case-based reasoning with ProtoPNet. While this work focuses on the improvement of the transparency of ProtoPNet, it can easily be applied to similar case-based reasoning architectures such as ProtoTrees (Nauta, Van Bree, and Seifert 2021) or XProtoNet (Kim et al. 2021). The results highlighted that explanations given by ProtoPNet are merely well-defined rather than giving insight into how the model transformed an image. We showed that the theoretical flaws can be overcome by estimating Shapley values of the similarity score with minor adaptations of the architecture. Quantitatively, the explanations of ProtoPFaith outperformed the explanations of ProtoPNet on the AOPC by a factor $>10^3$ in all experiments.

Acknowledgements

This research was partially funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation). The authors gratefully acknowledge the computational and data resources provided by the Leibniz Supercomputing Centre (www.lrz.de).

References

Adadi, A.; and Berrada, M. 2018. Peeking inside the black-box: a survey on explainable artificial intelligence (XAI). *IEEE access*, 6: 52138–52160.

Aditya, K.; Nityananda, J.; Bangpeng, Y.; and Li, F.-F. 2011. Novel dataset for fine-grained image categorization. In *Proceedings of the First Workshop on Fine-Grained Visual Categorization, IEEE Conference on Computer Vision and Pattern Recognition. IEEE, Springs, USA*.

Ancona, M.; Oztireli, C.; and Gross, M. 2019. Explaining deep neural networks with a polynomial time algorithm for shapley value approximation. In *International Conference on Machine Learning*, 272–281. PMLR.

Arrieta, A. B.; Díaz-Rodríguez, N.; Del Ser, J.; Bennetot, A.; Tabik, S.; Barbado, A.; García, S.; Gil-López, S.; Molina, D.; Benjamins, R.; et al. 2020. Explainable Artificial Intelligence (XAI): Concepts, taxonomies, opportunities and challenges toward responsible AI. *Information fusion*, 58: 82–115.

Baehrens, D.; Schroeter, T.; Harmeling, S.; Kawanabe, M.; Hansen, K.; and Müller, K.-R. 2010. How to Explain Individual Classification Decisions. *J. Mach. Learn. Res.*, 11: 1803–1831.

Baldessari, B. 1967. The Distribution of a Quadratic Form of Normal Random Variables. *The Annals of Mathematical Statistics*, 38(6): 1700–1704.

Binder, A.; Montavon, G.; Lapuschkin, S.; Müller, K.-R.; and Samek, W. 2016. Layer-Wise Relevance Propagation for Neural Networks with Local Renormalization Layers. In *International Conference on Artificial Neural Networks*.

Chattopadhay, A.; Sarkar, A.; Howlader, P.; and Balasubramanian, V. N. 2018. Grad-cam++: Generalized gradient-based visual explanations for deep convolutional networks. In *2018 IEEE winter conference on applications of computer vision (WACV)*, 839–847. IEEE.

Chen, C.; Li, O.; Tao, D.; et al. 2019. This Looks Like That: Deep Learning for Interpretable Image Recognition. In *NeurIPS*, volume 32.

Covert, I.; Lundberg, S. M.; and Lee, S.-I. 2020. Understanding global feature contributions with additive importance measures. *Advances in Neural Information Processing Systems*, 33: 17212–17223.

Dabkowski, P.; and Gal, Y. 2017. Real Time Image Saliency for Black Box Classifiers. In Guyon, I.; Luxburg, U. V.; Bengio, S.; Wallach, H.; Fergus, R.; Vishwanathan, S.; and Garnett, R., eds., *Advances in Neural Information Processing Systems*, volume 30. Curran Associates, Inc.

Desai, S.; and Ramaswamy, H. G. 2020. Ablation-CAM: Visual Explanations for Deep Convolutional Network via Gradient-free Localization. In *2020 IEEE Winter Conference on Applications of Computer Vision (WACV)*, 972–980.

Donnelly, J.; Barnett, A. J.; and Chen, C. 2022. Deformable ProtoPNet: An Interpretable Image Classifier Using Deformable Prototypes. In *CVPR*, 10265–10275.

Fel, T.; Ducoffe, M.; Vigouroux, D.; Cadène, R.; Capelle, M.; Nicodème, C.; and Serre, T. 2022. Don't Lie to Me! Robust and Efficient Explainability with Verified Perturbation Analysis. *arXiv preprint arXiv:2202.07728*.

Fong, R. C.; and Vedaldi, A. 2017. Interpretable explanations of black boxes by meaningful perturbation. In *Proceedings of the IEEE international conference on computer vision*, 3429–3437.

Frey, B. J.; and Hinton, G. E. 1999. Variational learning in nonlinear Gaussian belief networks. *Neural Computation*, 11(1): 193–213.

Fu, R.; Hu, Q.; Dong, X.; Guo, Y.; Gao, Y.; and Li, B. 2020. Axiom-based Grad-CAM: Towards Accurate Visualization and Explanation of CNNs. In *31st British Machine Vision Conference 2020, BMVC 2020, Virtual Event, UK, September 7-10, 2020*. BMVA Press.

Gast, J.; and Roth, S. 2018. Lightweight probabilistic deep networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 3369–3378.

He, K.; Zhang, X.; Ren, S.; and Sun, J. 2016. Deep Residual Learning for Image Recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*.

Huang, G.; Liu, Z.; van der Maaten, L.; and Weinberger, K. Q. 2017. Densely Connected Convolutional Networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*.

Jeyakumar, J. V.; Noor, J.; Cheng, Y.-H.; Garcia, L.; and Srivastava, M. 2020. How can i explain this to you? an empirical study of deep neural network explanation methods. *Advances in Neural Information Processing Systems*, 33: 4211–4222.

Jiang, P.-T.; Zhang, C.-B.; Hou, Q.; Cheng, M.-M.; and Wei, Y. 2021. LayerCAM: Exploring Hierarchical Class Activation Maps For Localization. *IEEE Transactions on Image Processing*.

Kim, E.; Kim, S.; Seo, M.; and Yoon, S. 2021. XProtoNet: Diagnosis in Chest Radiography With Global and Local Explanations. In *CVPR*, 15719–15728.

Kim, S. S.; Watkins, E. A.; Russakovsky, O.; Fong, R.; and Monroy-Hernández, A. 2023. " Help Me Help the AI": Understanding How Explainability Can Support Human-AI Interaction. In *Proceedings of the 2023 CHI Conference on Human Factors in Computing Systems*, 1–17.

Lundberg, S. M.; and Lee, S.-I. 2017. A unified approach to interpreting model predictions. *Advances in neural information processing systems*, 30.

Nauta, M.; Van Bree, R.; and Seifert, C. 2021. Neural prototype trees for interpretable fine-grained image recognition. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 14933–14943.

Nguyen, G.; Kim, D.; and Nguyen, A. 2021. The effectiveness of feature attribution methods and its correlation with automatic evaluation scores. *Advances in Neural Information Processing Systems*, 34: 26422–26436.

Petsiuk, V.; Das, A.; and Saenko, K. 2018. Rise: Randomized input sampling for explanation of black-box models. *arXiv preprint arXiv:1806.07421*.

Ribeiro, M. T.; Singh, S.; and Guestrin, C. 2016. " 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*, 1135–1144.

Rudin, C. 2019. Stop explaining black box machine learning models for high stakes decisions and use interpretable models instead. *Nat Mach Intell*, 1(5): 206–215.

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

Shah, H.; Jain, P.; and Netrapalli, P. 2021. Do input gradients highlight discriminative features? *Advances in Neural Information Processing Systems*, 34: 2046–2059.

Shapley, L. S. 1953. *A Value for n-Person Games*, 307–318. Princeton: Princeton University Press.

Shih, G.; Wu, C. C.; Halabi, S. S.; Kohli, M. D.; Prevedello, L. M.; Cook, T. S.; Sharma, A.; Amorosa, J. K.; Arteaga, V.; Galperin-Aizenberg, M.; et al. 2019. Augmenting the national institutes of health chest radiograph dataset with expert annotations of possible pneumonia. *Radiology: Artificial Intelligence*, 1(1): e180041.

Shin, M.; Kim, J.; van Opheusden, B.; and Griffiths, T. L. 2023. Superhuman artificial intelligence can improve human decision-making by increasing novelty. *Proceedings of the National Academy of Sciences*, 120(12): e2214840120.

Shrikumar, A.; Greenside, P.; and Kundaje, A. 2017. Learning important features through propagating activation differences. In *International conference on machine learning*, 3145–3153. PMLR.

Shrikumar, A.; Greenside, P.; Shcherbina, A.; and Kundaje, A. 2016. Not just a black box: Learning important features through propagating activation differences. *arXiv preprint arXiv:1605.01713*.

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

Simonyan, K.; and Zisserman, A. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In Bengio, Y.; and LeCun, Y., eds., *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*.

Springenberg, J. T.; Dosovitskiy, A.; Brox, T.; and Riedmiller, M. 2014. Striving for simplicity: The all convolutional net. *arXiv preprint arXiv:1412.6806*.

Štrumbelj, E.; and Kononenko, I. 2014. Explaining prediction models and individual predictions with feature contributions. *Knowledge and information systems*, 41: 647–665.

Sundararajan, M.; Taly, A.; and Yan, Q. 2017. Axiomatic attribution for deep networks. In *International conference on machine learning*, 3319–3328. PMLR.

Tomsett, R.; Harborne, D.; Chakraborty, S.; Gurram, P.; and Preece, A. 2020. Sanity checks for saliency metrics. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, 6021–6029.

Vale, D.; El-Sharif, A.; and Ali, M. 2022. Explainable artificial intelligence (XAI) post-hoc explainability methods: Risks and limitations in non-discrimination law. *AI and Ethics*, 1–12.

Wah, C.; Branson, S.; Welinder, P.; Perona, P.; and Belongie, S. 2011. The caltech-ucsd birds-200-2011 dataset.

Wang, G.; Chuang, Y.-N.; Du, M.; Yang, F.; Zhou, Q.; Tripathi, P.; Cai, X.; and Hu, X. 2022. Accelerating Shapley Explanation via Contributive Cooperator Selection. In *International Conference on Machine Learning*, 22576–22590. PMLR.

Wang, H.; Wang, Z.; Du, M.; Yang, F.; Zhang, Z.; Ding, S.; Mardziel, P.; and Hu, X. 2020. Score-CAM: Score-weighted visual explanations for convolutional neural networks. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 24–25.

Wolf, T. N.; Pölsterl, S.; and Wachinger, C. 2023. Don't PANIC: Prototypical Additive Neural Network for Interpretable Classification of Alzheimer's Disease. In *International Conference on Information Processing in Medical Imaging*, 82–94. Springer.

Zeiler, M. D.; and Fergus, R. 2014. Visualizing and understanding convolutional networks. In *Computer Vision–ECCV 2014: 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part I 13*, 818–833. Springer.

Zheng, Q.; Wang, Z.; Zhou, J.; and Lu, J. 2022. Shap-CAM: Visual Explanations for Convolutional Neural Networks Based on Shapley Value. In *Lecture Notes in Computer Science*, 459–474. Springer Nature Switzerland.

Zhou, B.; Khosla, A.; Lapedriza, A.; Oliva, A.; and Torralba, A. 2016. Learning Deep Features for Discriminative Localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE.

Zintgraf, L. M.; Cohen, T. S.; Adel, T.; and Welling, M. 2017. Visualizing Deep Neural Network Decisions: Prediction Difference Analysis. In *International Conference on Learning Representations*.