XtrAIn: Training-Guided Occlusion for Feature Attribution

Thodoris Lymperopoulos Ioannis Kakogeorgiou Denia Kanellopoulou
NCSR Demokritos, Athens, Greece

Abstract

Occlusion-based attribution methods provide an intuitive way to estimate feature importance by perturbing input features and measuring the resulting change in model output. However, their reliability is strongly affected by how feature removal is implemented: externally selected baselines can introduce bias, out-of-distribution samples, and unstable explanations, while in nonlinear models the occlusion of a set of features can also alter the contribution of non-occluded features. We refer to this effect as attribution shift, as the attribution scores of the non-occluded features drift from their initial values. To challenge these major issues that render explanations unstable, we introduce XtrAIn, a training-guided attribution method that transfers the occlusion operation from the input space to the parameter space. Instead of replacing input values with handcrafted baselines, XtrAIn follows the model's training trajectory and measures how feature-associated parameter updates affect the output logits. We further introduce Xstep, a lightweight approximation for reducing computational cost, and XtrAIn+, a target-focused variant that emphasizes updates aligned with the target class. Experiments on controlled image datasets and PAM50 breast-cancer subtype classification show that the proposed methods produce cleaner and more interpretable attribution patterns than standard attribution baselines. Overall, XtrAIn provides a training-aware perspective on feature attribution and offers a useful diagnostic tool for studying how feature-level evidence is formed during training.

Keywords

Explainable AI, Attribution Methods, Occlusion Techniques, Model Training and Interpretability

1 Introduction

As AI systems are increasingly used in high-impact decision-making settings, understanding the factors that drive their predictions has become a central concern in modern machine learning. Explainable AI (XAI) [9, 41, 62] has emerged as a primary vehicle for this effort, with attribution methods [6, 51] forming a widely adopted paradigm for quantifying the contribution of individual input features to a model's output. Among them, occlusion-based approaches [17, 36, 45, 64] occupy a foundational position — operating by systematically removing subsets of input features and measuring the resulting change in model output. The magnitude of this change is taken as a proxy for the contribution of the occluded features to the model's decision. This perturbation logic is closely related to causal reasoning [12, 21], making occlusion a persistent component of explainability techniques for increasingly complex architectures [14].

Nevertheless, occlusion-based methods remain sensitive to how feature removal is implemented, especially through the choice of baseline values. In their attempt to simulate such removal, different techniques for value selection may introduce biases [48], as demonstrated by several studies [28, 39, 58]. Additionally, the shape and size of the occluded regions are not standardized, and different parametrizations can lead to different attribution maps [7]. Another important issue is out-of-distribution (OoD) behavior, where an incautious selection of baseline values can move samples to regions outside the original distribution [23, 29, 30, 33, 34, 39]. This introduces safety concerns for sensitive applications [50]. Researchers have addressed this either by introducing corrupted data during training [8, 26, 30] or by using generative models for more natural inpainting [2, 4, 11]. While these approaches may alleviate OoD issues, they do not fully remove the dependence on the chosen feature-removal rule. In this regard, some techniques apply criteria for a theoretically grounded information removal [32, 47, 54], yet defining a generally reliable and objective removal strategy remains challenging. Similar sensitivities also affect feature-removal-based evaluation metrics, whose outcomes may depend on the chosen perturbation strategy [22, 43, 48].

Beyond baseline selection, we highlight an additional limitation of input-space occlusion: because feature removal is implemented through an *externally imposed* input intervention, the attribution assigned to an occluded feature subset may also reflect changes in the effect of the remaining, non-occluded features. Let $I_1 \subset I$ denote an occluded subset of input features, and let $I_2 \subseteq I \setminus I_1$ denote the subset of the remaining non-occluded features. Occlusion estimates the contribution of I_1 by comparing the original model response $f(I)$ with the response obtained after removing or replacing I_1, denoted $f(I \setminus I_1)$. This implicitly assumes that the effect of the non-occluded features remains constant across the two evaluations and is therefore neglected. However, in nonlinear models, the intervention on I_1 can also change how the remaining features I_2 interact with the model, as illustrated in Fig. 1. Therefore, the differences in model's response does not necessarily isolate the effect of I_1 alone — as popular occlusion-based methods assume — but may include may include interaction-dependent effects from the remaining features. We refer to this phenomenon as *attribution shift*: the attribution assigned to I_1 may include effects from feature dependencies, induced by the disturbed behavior of I_2. In this regard, attribution scores resulting from occlusion-based techniques are indeed interaction dependent (despite the hypothesis for interaction-blind feature attributions [60]), but in a perilous way that perturbs feature estimation scores and complicates the disentanglement of the intended feature-removal effect.

Attribution shift stems from the input-space occlusion mechanism itself and is not fully resolved by changing only the baseline-filling strategy. This motivates us moving the perturbation operation beyond the input space, towards transferring the occlusion operation to the parameter space: instead of relying on externally imposed input values, *XtrAIn* uses perturbation values naturally induced by the model's own training updates. In this work, we instantiate this principle through the feature-associated weights of fully connected neural networks (FCNNs). By perturbing these

Corresponding author: thodoris.lymperopoulos@gmail.com

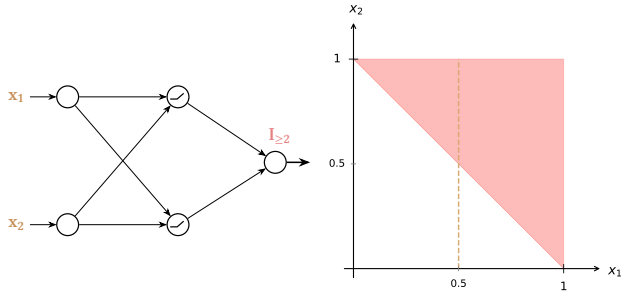

Figure 1: *Illustration of attribution shift in input-space occlusion.* In the simple FCNN on the left, all weights are set to one and all biases are zero. The two hidden ReLU neurons therefore receive the same input, $x_1 + x_2$. For $x_1 = x_2 = 0.5$, both hidden neurons output 1, so the input to the final threshold neuron is $1 + 1 = 2$ and the output is activated. However, a small decrease $\epsilon > 0$ in x_2 changes both hidden activations to values below 1, making the final input smaller than 2 and deactivating the output. This abrupt change in model's response should spark large shifts in feature attributions — since these scores are related to the model's output. This illustrates how input-space occlusion can produce interaction-dependent attribution shifts.

weights between consecutive model states, we measure the induced change in the model logits and aggregate these parameter-space occlusions over training to obtain a final attribution score that reflects how each feature-associated update contributes to the model's output.

Our work makes the following contributions:

- We analyse how *attribution shift* arises in input-space occlusion for nonlinear models, showing that occluding one feature can also alter the contribution of other, non-occluded features.
- We introduce $XtrAIn$, a new occlusion-based attribution method that transfers the occlusion operation from the input space to feature-associated parameter updates. We further introduce $Xstep$, a lightweight approximation for compute-intensive settings, and $XtrAIn^{+}$, a target-focused variant designed to emphasize updates that strengthen the target class.
- We propose *CleanScore*, a diagnostic metric for evaluating attribution cleanness in settings where signal and background regions are known.
- We evaluate the proposed method on controlled image datasets and a PAM50 breast-cancer subtype classification task, showing that parameter-space occlusion produces cleaner, safer and more interpretable explanations than standard attribution baselines.

2 Related Work

Training Dynamics. Many ML disciplines study training dynamics, with the most notable being the framework of *Developmental Interpretability*, which examines the learning phenomena of deep models as they form over time. This has led to in-depth investigations of a model's complex behaviors, such as grokking [46] and emergent capabilities [52, 57], while offering insights into its interpretability, as with the detection of personality traits [13]. However, this line of research focuses primarily on interpreting model behavior, rather than estimating feature importance. Conversely, another approach that addresses this theoretical question is that of *Feature Selection* [25] which focuses on training statistics, feature importance and data influence. A related subfield, namely *Embedded Methods* [49] operates by introducing an artificial layer between the input and the first hidden layer to collect different statistics throughout the training process—most commonly weights and gradients [10]—which are then used to infer feature importance using different techniques applied to the collected gradient or weight profiles. Nonetheless, the relationship between these statistics and the model's overall decision-making process remains ambiguous.

Occlusion-based Techniques. These highly popular attribution methods estimate feature importance by systematically altering parts of the input representation and measuring the corresponding drop in the model's output score. The work of [15] comprises the most detailed study of occlusion-based methods, grouping them all under the same framework and providing a foundation for theoretical comparison. Despite their strengths and successes, critical limitations such as added bias [48] and out-of-distribution data [29, 30] persist.

Beyond Classical Occlusion. A series of techniques have attempted to alleviate the theoretical limitations of Occlusion methods for feature removal. Statistical measures being applied to the whole data distribution, such as PDP [18], FMI [16] and ALE [3] offer insights into feature importance, while marginalizing the effect of complementary features. However, they are designed to provide global explanations. Authors in [53] apply perturbations in the transformed Fourier space, yet with unknown effects for the problems at hand. A different line of work [61] computes a similarity score among the inner representations of augmented images and their occluded counterparts, yet the limitations of feature removal persist. Recently, authors in [37] developed a novel occlusion-based approach using weight perturbations, estimating feature importance as a distance score between the untrained and trained model versions. Inspired by this direction, we build upon it while significantly strengthening its foundations.

Our proposed method advances beyond classical occlusion rules using weight perturbations. We exploit mechanisms within training for feature independence and transfer the field of occlusion rules to the parameter space, where the problems of Added Bias and OoD are not expressed, while attribution shift is carefully eliminated.

3 Methodology

In general, attribution methods assign importance scores using rules derived from specific model properties. However, the choice of which properties should define *importance* is not uniquely specified, since importance itself lacks a precise and universally accepted definition [27, 35]. As a result, every attribution method implicitly relies on assumptions that connect the model's mathematical behavior to a human-interpretable notion of feature contribution. In

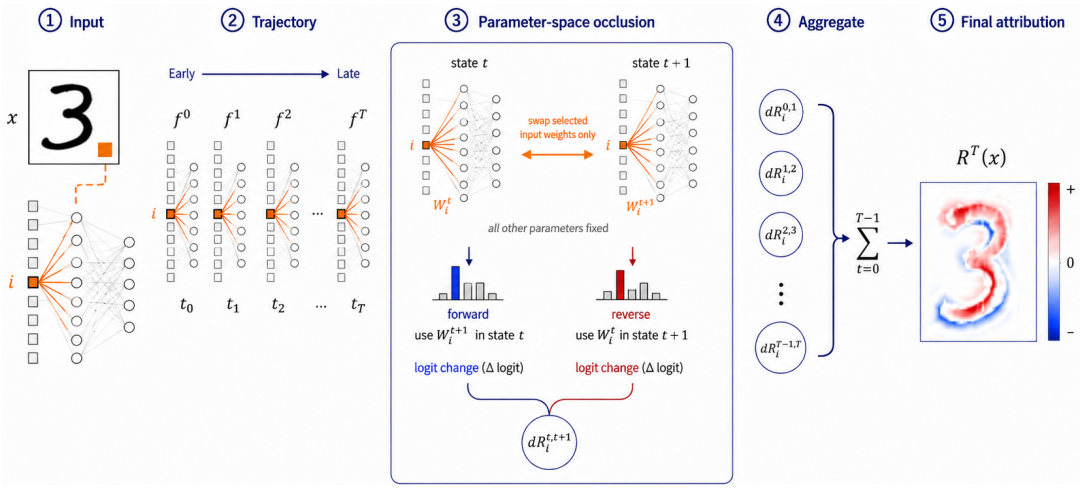

Figure 2: *Overview of XtrAIn.* Given an input sample x, the method follows the training trajectory of the model and tracks the weights W_i^t associated with each input feature i. For two consecutive model states, t and $t+1$, *XtrAIn* performs parameter-space occlusion by replacing only the feature-associated weights while keeping all other parameters fixed. The resulting forward and reverse logit changes are combined into a step-wise attribution update $dR_i^{t,t+1}$, and these updates are accumulated over training to obtain the final attribution map $R^T(x)$.

this work, we make these assumptions explicit and use them to motivate the design of our training-guided attribution rule.

Assumption 1. *Attribution scores are relative values whose interpretation depends primarily on their sign, ranking, and relative magnitude across features.*

Mathematical Notation. We assume an arbitrary FCNN $f = f^L \circ f^{L-1} \circ \cdots \circ f^0$ receiving input $x \in \mathcal{X}$, with $\mathcal{X}$ being the input space. Given an arbitrary intermediate layer l with n_l neurons, weights $W^l \in \mathbb{R}^{n_{l-1} \times n_l}$ and bias $b^l \in \mathbb{R}^{n_l}$, the input x_j^l and output z_j^l of a neuron j in this layer are calculated as:

$$x_j^l = \sum_{i \in [l-1]} z_i^{l-1} \cdot w_{ij}^l + b_j^l, \tag{1}$$

$$z_j^l = \sigma(x_j^l), \tag{2}$$

where $[l]$ represents the neurons of layer l and σ is a nonlinear activation function (usually ReLU).

3.1 Foundations

Conceptually, model parameters govern the interactions among neurons within the network. In this regard, attribution methods are expected to depend on these values; otherwise, the resulting scores would be model-independent, thereby weakening their role as model explanations. This implies that the changes in parameter values result in the alteration of the attribution scores assigned to a given input [1]. Therefore, during training, where parameters change across optimization steps, the evolution of the model can be associated with a sequence of attribution scores $\mathbf{R}^t \in \mathbb{R}^{n_0}$, with $t \in \{1, \ldots, T\}$ representing the training step and T the total number of steps.

Assumption 2. *In the general case, an update of the model parameters alters the attribution scores for a given input.*

Building on Assumption 2, we develop a methodology for feature-importance estimation that departs from the standard *static* view, where attribution scores are extracted only from the final trained model. Instead, we propose a *training-guided* attribution score, obtained by aggregating attribution changes across consecutive model states. This shifts the core question from '*how to explain a trained model*' to '*how to explain a model update*'. We posit that this presents a more manageable problem to solve, as it allows for the exploitation of meaningful, tractable data drawn from the training process.

To tackle this challenge, we introduce a step-wise attribution rule that estimates the added attribution value $d\mathcal{R}^t$ induced by each model update. It will form the updating mechanism of a recursive function for the estimation of feature attribution. This perspective requires that the model update process remains consistent across training, so that attribution changes can be compared and accumulated over consecutive optimization steps.

Assumption 3. *The model's operational principles and updating mechanism remain consistent throughout training. Accordingly, the attribution-updating mechanism must remain invariant across training steps.*

3.2 Update Rule & XtrAIn

To develop an update rule for the estimation of $d\mathcal{R}^t$, we reflect upon the model's overarching goal, that is the *meaningful update of its parameters* towards loss minimization. This is ultimately expressed at the model logits.

Assumption 4. *The change in model logits represents the effect of parameters' update.*

Assumption 4 fosters the estimation of the effect for each feature's parameters update directly at the output logits (in special

case, a zero score is assigned to parameter updates that cause no effect in model output). Therefore, for each input feature i and target class $\odot$, we define the update-level attribution score as:

$$d\mathcal{R}_i^{t,t+1}(x;\odot) = \mathcal{I}^{\odot\top} \cdot \left(df_i^{t,t+1}(x) + d\tilde{f}_i^{t,t+1}(x) \right), \tag{3}$$

where $df_i^{t,t+1}(x), d\tilde{f}_i^{t,t+1}(x) \in \mathbb{R}^{n_L}$ are logit-change vectors defined as:

$$df_i^{t,t+1}(x) = f^t_{W_i^t \to W_i^{t+1}}(x) - f^t(x), \tag{4}$$

$$d\tilde{f}_i^{t,t+1}(x) = f^{t+1}(x) - f^{t+1}_{W_i^{t+1} \to W_i^t}(x). \tag{5}$$

Here, f^t and f^{t+1} denote the model before and after a training update at time t, respectively, while W_i^t denotes the set of weights directly connected to input feature i at this time, i.e.,

$$W_i^t = \{w_{ij}^t \mid j \in \{1, \ldots, n_1\}\}.$$

The notation $f^t_{W_i^t \to W_i^{t+1}}$ denotes the model at state t where only the feature-associated weights W_i^t are replaced by their updated values W_i^{t+1}, while all other parameters remain fixed. Similarly, $f^{t+1}_{W_i^{t+1} \to W_i^t}$ denotes the reverse replacement in the updated model.

The sign vector $\mathcal{I}^{\odot} \in \{-1, 1\}^{n_L}$ maps logit changes to rewards:

$$\mathcal{I}_c^{\odot} = \begin{cases} 1, & \text{if } c = \odot, \\ -1, & \text{if } c \neq \odot. \end{cases} \tag{6}$$

Thus, $d\mathcal{R}_i^{t,t+1}(x;\odot)$ is a scalar attribution update for feature i, computed with respect to a fixed target class $\odot$ by projecting the logit-change vector onto the corresponding target-class sign vector. Collecting these scalar scores over all input features yields the step-wise attribution vector $d\mathcal{R}^{t,t+1}(x;\odot) \in \mathbb{R}^{n_0}$. For simplicity, we omit the explicit target argument in the following and use $\mathcal{R}^t(x)$ and $d\mathcal{R}^{t,t+1}(x)$ to denote their target-conditioned counterparts.

The two logit-change vectors in Eq. (3), defined in Eqs. (4) and (5), correspond to forward and reverse parameter-space occlusions between consecutive model states. The first term measures the effect of replacing the feature-associated weights at state t with their updated values from state $t+1$, while the second term measures the complementary effect of reverting the same weights in the updated model. Considering both directions makes the update rule symmetric with respect to consecutive model states and supports the inverse property discussed in Section 3.3.2.

Regarding the sign vector $\mathcal{I}^{\odot}$, its role is to map each logit update to a reward signal. Its design stems from the model's training principle, namely, the promotion of the target neuron and suppression of non-target neurons.

Assumption 5. *A logit update is positively rewarded if it represents:*

- *an increase, in case of the target neuron,*
- *a decrease, in case of the non-target neurons.*

The opposite cases are considered negative.

We examine whether the proposed update rule produces structured attribution patterns during training. As shown in Fig. 3, the intermediate update scores reveal feature-associated structures that evolve across optimization steps. Early updates tend to highlight stable input-aligned patterns, while later updates include more class-competitive structures, reflecting how the model progressively adjusts its decision boundaries.

Considering all intermediate updates, we introduce a training-guided attribution method, named $XtrAIn$, which accumulates the attribution contributions induced by consecutive model updates. Given the step-wise attribution vector $d\mathcal{R}^{t,t+1}(x)$, $XtrAIn$ updates the attribution score recursively as:

$$\mathcal{R}^{t+1}(x) = \mathcal{R}^t(x) + d\mathcal{R}^{t,t+1}(x). \tag{7}$$

The recursion is initialized with $\mathcal{R}^0(x) = 0$, treating the randomly initialized model as a neutral attribution baseline, since it has not yet learned input-output associations from the data.

Since the exact version of $XtrAIn$ requires evaluating attribution contributions across all consecutive training states, its computational cost can become prohibitive for large models or long training trajectories. To address this limitation, we introduce $Xstep$, a lightweight approximation that applies the same update rule only between selected model states, thereby skipping intermediate updates. As shown in Section 4, this approximation preserves the main attribution patterns while substantially reducing the computational burden, making it suitable for time- and resource-constrained settings. The total disregard of all intermediate model states results in an enhanced version of methods introduced in [37].

Together, $XtrAIn$ and $Xstep$ provide a training-guided approach to feature-importance estimation, where attribution is derived from the model's own parameter updates rather than from externally chosen input baselines. This avoids hand-crafted baseline choices and reduces the risk of introducing artificial or out-of-distribution perturbations.

3.3 Theoretical Considerations

3.3.1 Rationale. To identify a mechanism for feature independence we reflect upon a parameter's update: In every step of the training process, it receives a gradient signal that points toward loss minimization. Its updated value will be defined by the term:

$$\Delta w_{ij} = -\eta * \frac{\partial \mathcal{L}}{\partial w_{ij}}. \tag{8}$$

Crucially, during the calculation of the gradient signal, all contributing parameters, which in FCNNs include the weights and biases of subsequent layers, are frozen, with their effect being captured within this signal through the chain rule. In this regard, this update is:

- *meaningful*: it is applied for loss minimization,
- *independent*: all other model parameters are considered stationary.

The gradient signal carries the parameter's particular *responsibility* towards loss optimization. Hence, we assume that its contribution can be theoretically traced according to Assumption 4 by an occlusion rule among the model and a replica, in which the value w_{ij} is altered to w'_{ij}.

Shifting our analysis from parameter to neuronal contributions, a higher level of abstraction is required, since the loss signal is only meaningful for the model's parameters, and neurons are the intermediate steps. In this process, the gradient of the loss with

(a) Patterns at the first training epoch **(b) Patterns at the fifth training epoch**

Figure 3: *Intermediate update-level attribution patterns for two samples across training epochs.* The resulting scores of the update rule are displayed at different steps of a given training epoch (as indicated by their titles), uncovering a diverse set of attribution patterns within the FCNN's training process. In the early training state of the model, heatmaps indicate higher stability in pattern formation, while in later stages, more intricate patterns emerge.

respect to the input features is calculated as:

$$\frac{\partial \mathcal{L}}{\partial i} = \sum_{j \in [n_1]} w_{ij} * \frac{\partial \mathcal{L}}{\partial j} \tag{9}$$

where i, j represent neurons of the input and first hidden layer respectively. However, the values $\frac{\partial \mathcal{L}}{\partial j}$ are shared among all input features i; a feature's individual traits are only introduced by weights w_{ij}. Hence, we can conceptually reduce a neuron to its attached parameters (for the input neurons, these being their weights W_i). These appear equivalent in Eq. (9), enabling their grouping and reduction under one occlusion rule, estimating feature attribution by generalizing an individual parameter's occlusion rule to a group of carefully selected parameters.

Assumption 6. *A neuron's effect should be estimated according to the occlusion rule, when grouping and considering the changes in its attached parameters.*

Having established the rationale behind the design of the terms df, df', we now proceed to the explanation of their combined effect.

3.3.2 Criteria and Theoretical Aspects. Researchers in XAI [5, 19, 36, 56, 59] have widely adopted foundational criteria to assess their methodologies, in response to the scarcity of theoretically sound evaluation metrics. These criteria are considered as necessary conditions any attribution method should satisfy. The methodology of *XtrAIn* introduces a new field for criteria development, with an objective to guide the method's design and validate its effectiveness.

Criterion 1. ***(Inverse Property)*** *Let t, $t+1$, $t+2$ be three consecutive time-steps and f^t, f^{t+1}, f^{t+2} be the corresponding models. If step $t+1$ is the reverse step of t, meaning that $f^{t+2} = f^t$, then, it should hold that $\mathcal{R}^{t+2}(x) = \mathcal{R}^t(x),\ \forall x \in \mathbb{R}^{n_0}$.*

Adherence to the Inverse Property yields robustness against attribution distortion in case of an immediate reverse in the model's state.

Theorem 3.1. *XtrAIn satisfies the Inverse Property.*

The theorem's proof can be found in Section A.1. This criterion secures information erasure in the case of error correction. While many operations could combine terms df and df' while satisfying the Inverse Property, the addition operation was selected for its simplicity.

3.4 Disentangling Target & Non-Target Updates

The update of model parameters is a synchronous process, simultaneously reinforcing target neurons and suppressing non-target ones. Different attribution methods have attempted to disentangle output neuron effects [24, 31, 55], yet we argue that they are limited when attempting to retroactively separate class-specific signals within a model whose internal representations have already converged into a deeply entangled, non-linear synthesis of all classes. In this section, we demonstrate *XtrAIn*'s capacity to disentangle the effects of output neurons, focusing on target neurons to further improve interpretability and faithfulness to the model.

Indeed, the loss value and parameter update can be decomposed into *target* and *non-target* terms as:

$$\mathcal{L} = \mathcal{L}_{target} + \mathcal{L}_{non\_target} \tag{10}$$

$$\Delta w_{ij} = \Delta w_{target} + \Delta w_{non-target}, \tag{11}$$

which in the case of Cross Entropy Loss becomes:

$$\mathcal{L} = \underbrace{-o_t}_{target} \underbrace{+ log(\sum_{k=1}^{[L]} e^k)}_{non-target} \tag{12}$$

A detailed mathematical proof can be found in Section A.2.

In this regard, *XtrAIn* has the capacity to exploit the analysis of the loss into the two complementary terms according to Eq. (10), and apply the update rule to $\mathcal{L}_{target}$. A further step in this analysis potentially removes any negative updates that decreased the target-class logit for a sample x, thus prioritizing patterns used explicitly for the emergence of the target neuron. Thus, the updates within the calculation of $\mathcal{R}_T^+$ that lead to a decrease of the target neuron activation are filtered out. This leads to the calculation of positive target attribution, defined as $XtrAIn_T^+$.

However, due to the simultaneous application of both target and non-target effects in parameter updates, any attempt to isolate and measure the individual effects of each change is inherently constrained; the result is a theoretical construct, pointing to the aggregated *tendency* of the model towards strengthening the target class. Despite this, these scores remain insightful for validating the model's expression of causes and enhancing trust in them.

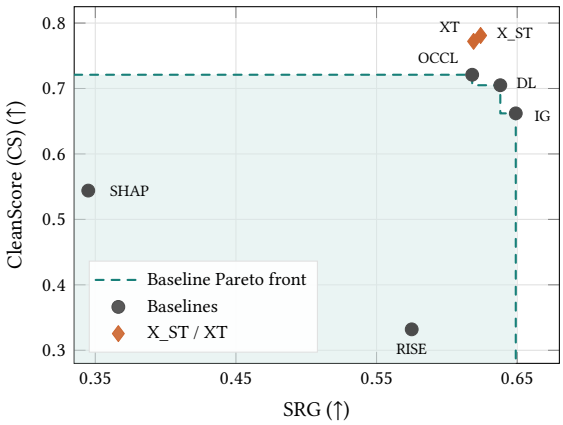

Figure 4: Baseline-frontier Pareto analysis of SRG and CleanScore on TMNIST. Higher is better for both metrics.

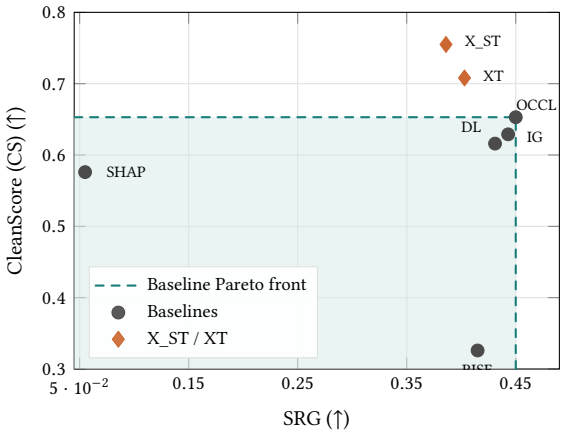

Figure 5: Baseline-frontier Pareto analysis of SRG and CleanScore on TMNIST_L. Higher is better for both metrics.

3.5 CleanScore

We introduce *CleanScore*, a metric designed to detect noise in attribution maps, which in case of occlusion-based approaches, can partially capture biases resulting from attribution shift. On simple datasets with pixel-level attributions, this bias appears mainly as small stochastic perturbations rather than systematic mean shifts. These perturbations inflate attribution variance within both the signal region and background, providing a proxy for method-induced noise.

We exploit the structure of synthetic datasets commonly used to evaluate FCNNs, in which the image signal is confined to a central region C, with $\mathcal{B} = C^c$ denoting the background, where by construction the model has no informative signal to read. This partition in C and $\mathcal{B}$ is achieved by thresholding on the mean value of the aggregated training data. Thus, given a signed attribution map $h \in \mathbb{R}^{H \times W}$ normalized to [-1,1], we define:

Table 1: Quantitative analysis on the proposed metric, SRG ∠ CS for the Typeface MNIST (TMNIST) and Typeface MNIST Lines (TMNIST_L) datasets. *XtrAIn* (XT) and *Xstep* (X_ST) outperform other occlusion-based methods in this score by a margin. Best results are highlighted in bold.

Dataset	OCCL SRG ∠ CS (↑)	DL SRG ∠ CS (↑)	SHAP SRG ∠ CS (↑)	RISE SRG ∠ CS (↑)	IG SRG ∠ CS (↑)	X_ST SRG ∠ CS (↑)	XT SRG ∠ CS (↑)
TMNIST	0.950	0.951	0.644	0.664	0.927	**1.0**	0.99
TMNIST_L	0.793	0.752	0.579	0.528	0.770	**0.850**	0.815

$$\text{CleanScore}(h) = \underbrace{\frac{\sum_{i\in C}|h_i|}{\sum_i |h_i|}}_{\text{concentration}} \cdot \underbrace{\frac{1}{1+\hat{\sigma}_C^+}}_{\text{positive uniformity}} \cdot \underbrace{\frac{1}{1+\hat{\sigma}_{\mathcal{B}}}}_{\text{background silence}} \tag{13}$$

where $\hat{\sigma}_C^+$ and $\hat{\sigma}_{\mathcal{B}}$ are the standard deviations of positive attributions within C and all attributions within $\mathcal{B}$, respectively.

The *concentration* term penalizes methods that diffuse attribution mass into the background, where, by construction, no informative signal exists. The *positive uniformity* and *background silence* terms use intra-region standard deviation as a comparative proxy for noise. The underlying intuition is that observed variance decomposes as $\sigma^2_{obs} \approx \sigma^2_{true} + \sigma^2_{noise}$ where σ_{true} reflects genuine attribution structure and σ_{noise} reflects occlusion-induced perturbations. Under the assumption that faithful methods broadly agree on which pixels carry signal — defensible in the simple-dataset regime, where informative pixels are well-defined — differences in σ_{obs} across methods are attributable to differences in σ_{noise}. Therefore, CleanScore is intended as a comparative measure rather than an absolute one: it ranks methods by their relative noise levels rather than certifying any as bias-free.

Negative attributions within C are excluded from the uniformity term for simplicity, as the positive term is sufficient. All three factors lie in $(0, 1]$, and their product gives a scalar in $(0, 1]$, with higher values indicating cleaner attributions.

4 Evaluation

This section describes the experimental setup used to evaluate *XtrAIn* and its variants. We conduct experiments on a standard FCNN architecture with ReLU activation for the intermediate layer and softmax for the output layer. The models have four layers with gradually reduced neurons and are trained until convergence (typically achieving test accuracy above 90% within 10 epochs) on an NVIDIA GeForce RTX 4060 GPU.

We select standard synthetic datasets for explainability of simple architectures while also satisfying the constraints of the CleanScore metric: Typeface MNIST (TMNIST) [63] with digits rendered from Google Fonts, Typeface MNIST Lines (TMNIST_L) with random added lines on the TMNIST dataset, and AffineMNIST (AMNIST) [38] with affine transformations applied to MNIST. The latter is selected as a theoretical basis for interpretability of complex datasets where FCNNs cannot converge (achieving at best 25% accuracy after 20 epochs) — thus excluded from quantitative evaluation. All images have size 28×28 with values normalized to [0.5, 1] to ensure positive input values across all pixels.

We compare *XtrAIn* (*XT*) and *Xstep* (*X_ST*) against common occlusion methods: Shapley values (SHAP) [36], Integrated Gradients (*IG*) [59] (as it can be expressed as an occlusion-based method [20]), DeepLift (*DL*) [55] and Randomized Input Sampling (*RISE*) [45]. The parameterization of these methods follows the standard setup of the Captum library. For SHAP, we use a uniform baseline fill, as it yielded the best results.

We use SRG [7] as the evaluation metric to compare attribution methods, selected for its robustness and stability in AUC score calculation. This score is computed as:

$$SRG(x) = LIF(x) - MIF(x), \tag{14}$$

where LIF and MIF compute the AUC score of the deletion curve starting from the least and most important features respectively — according to the ranking produced by an attribution method.

However, SRG is itself computed via occlusion, and so inherits the same OoD and artifact introduction problems that affect occlusion-based attribution methods. This creates a structural overlap between what these methods optimize for — the change in model output under feature masking — and what the metric measures. As a result, methods built on occlusion are expected to score well on SRG largely by construction, independent of whether they identify features the model actually relies on.

We therefore do not consider the competitive, but not leading performance on SRG as a failure mode for our method. To break this circularity, we pair SRG with CleanScore, forming an orthogonal measure of interpretability accuracy and bias. We apply it to 100 test samples and report the norm of the two scores in Table 1. The resulting Baseline-frontier Pareto plots are shown in Figs. 4 and 5. Additional attribution visualizations for popular occlusion-based methods are shown in Fig. 7, together with *XtrAIn* and its variants.

For the real-world dataset, we selected the Prediction Analysis of Microarray 50 (PAM50) dataset [42, 44], a 50-gene signature assay used to classify breast cancer based on its intrinsic molecular subtype. We followed the same process as in [40] to prepare the dataset for binary classification with basal-like or luminal-like classes. In this setting, the model converges within four epochs, reaching perfect accuracy. However, we observe a spurious strategy: for the first two epochs, it learns only the luminal-like class while ignoring the basal class, and only later starts learning the second class. We use this behavior to test whether the methods detect dynamics. The top 10 aggregated features per class are shown in Fig. 6.

5 Discussion

A visual inspection of the attribution maps in Fig. 7 shows that *XtrAIn* produces clean and interpretable explanations. Patterns within the central area correspond to the inputs at hand, while the predominantly non-activated background indicates the static, non-contributing nature of the corresponding neurons.

Different occlusion-based methods produce broadly similar patterns. However, *XtrAIn* robustly generalizes these patterns: while other techniques produce artifacts across image regions *XtrAIn*

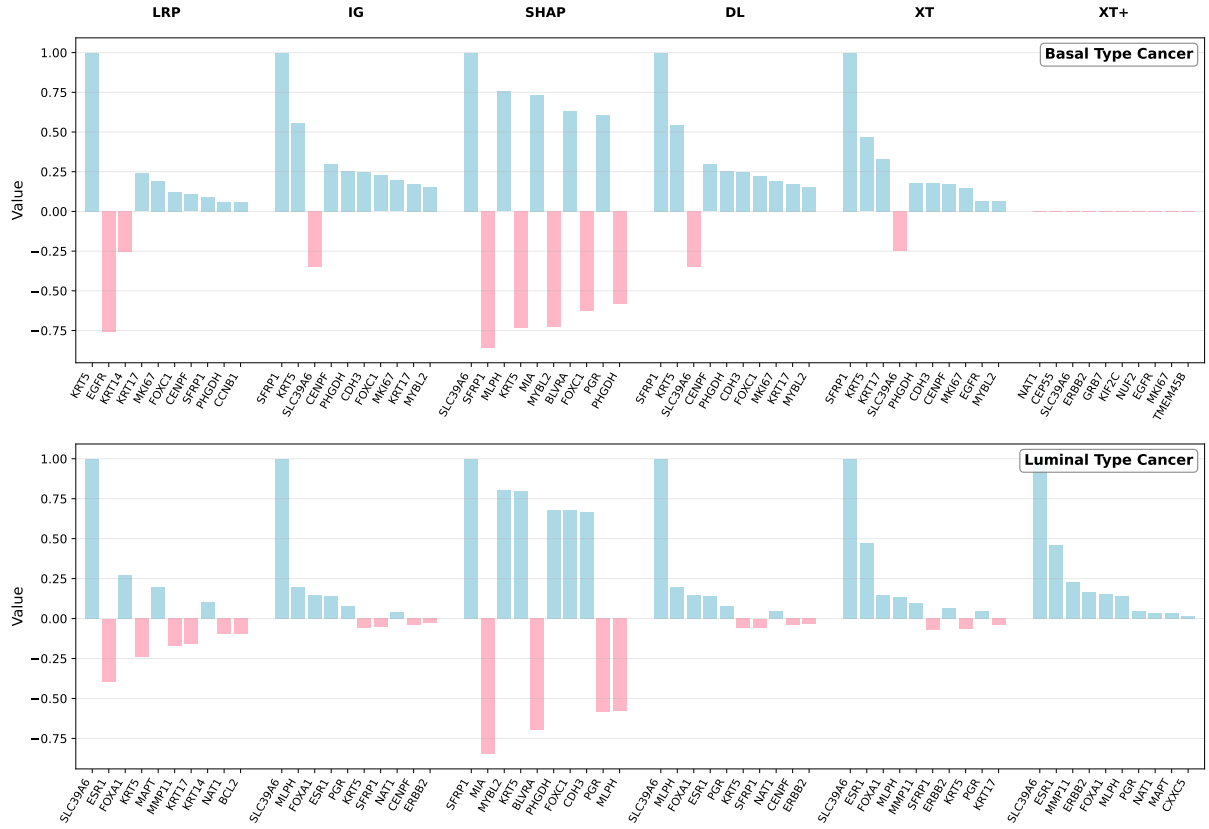

Figure 6: Bar plots for the top 10 normalized features of the aggregated attribution scores for each class, for the methods SHAP, LRP, IG, DL, $XtrAIn$ (XT) and $XtrAIn^+$ (XT+). In a controlled setting, we interrupt training at the second epoch, when only the Luminal Type class has been learned; $XtrAIn^+$ then successfully assigns zero importance to all features for this class.

silences the background and generates clean patterns with gradual fill in the central region. These observations support our claims about feature dependence in standard occlusion-based methods and feature independence for $XtrAIn$.

Furthermore, a comparison between $XtrAIn$ and its variant $Xstep$ reveals strong similarities in the results, providing evidence for a sufficient approximation for relatively simple models and datasets. Additionally, the consistency across samples of each class suggests these methods capture dominant, *class-specific patterns* — which prevail over the more intricate patterns discovered during training (Fig. 3). In contrast, $XtrAIn^+$ reveals explanations driven by *input-specific patterns* that better fit the image signals.

We provide quantitative validation in Table 1, where $XtrAIn$ and $Xstep$ outperform baseline occlusion-based methods according to our proposed metric. Encouraged by these results, we conducted a deeper analysis by applying our methods to Affine MNIST, a dataset without spatial dependencies. While input-based occlusion methods only detect traces of the input signal, our variants confirm the model's clean signal identification. The results suggest the model cannot identify dependencies between patterns and spatial location—a known limitation of FCNNs.

Applying our framework to the PAM50 breast cancer dataset reveals similarities with other attribution methods. Most methods, including $XtrAIn$ identify common features for classification to some extent. Despite this, they all provide evidence for features positively rewarding the Basal Type Cancer class for two epochs of training. On the other hand, $XtrAIn^+$ is only triggered by positive target class updates, thus resulting in a blank feature map. This application demonstrates its capacity in fostering safety in critical applications and system diagnostics.

Overall, $XtrAIn$ and its variants provide a powerful diagnostic lens for model explainability, revealing clean, interpretable patterns while uncovering intricate learning strategies within the model.

6 Conclusion

In this work, we presented a novel methodology for estimating attribution scores, operating by chaining evolution, occlusion through weight perturbation and importance. We introduced $XtrAIn$ and its light variant $Xstep$, and demonstrated their complementary effectiveness in explaining predictions and revealing previously unknown insights into the model's capabilities and classification strategies, offering deeper insight into its decision-making process.

This methodology establishes a new, unexplored path for transparent and granular model interpretation. By extending the usage of update mechanisms to more complex model architectures or functional aspects, its explanatory potential grows substantially.

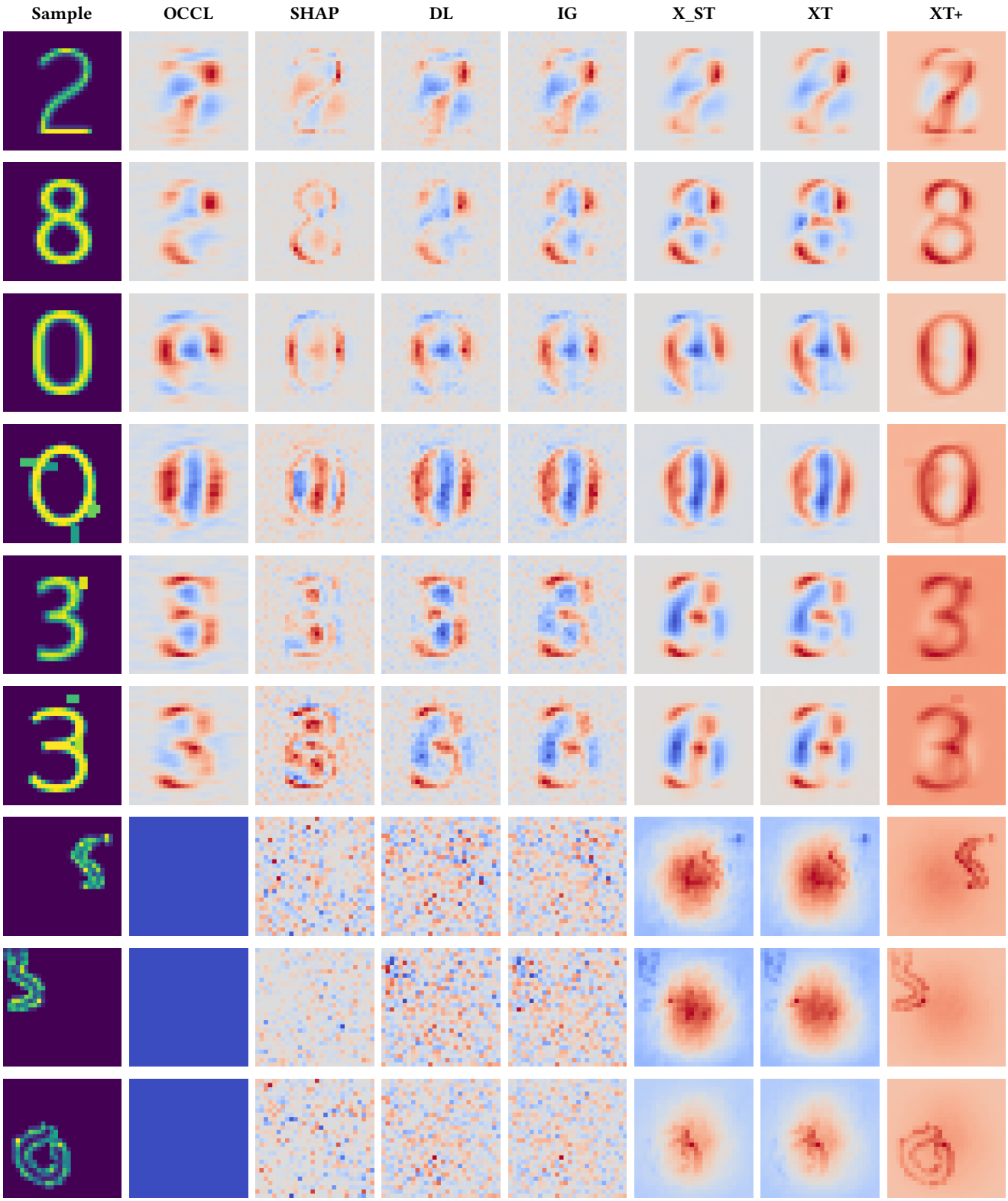

Figure 7: *Attribution maps produced by different methods on TMNIST, TMNIST_L, and AMNIST.* $XtrAIn$ **(XT),** $Xstep$ **(X_ST) and** $XtrAIn^{+}$ **(XT+) produce visually cleaner explanations in these examples, featuring a suppressed background and smooth foreground patterns.**

References

[1] Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. 2020. Sanity Checks for Saliency Maps. arXiv:1810.03292 [cs.CV] https://arxiv.org/abs/1810.03292

[2] Chirag Agarwal and Anh Nguyen. 2020. Explaining Image Classifiers by Removing Input Features Using Generative Models. In *Computer Vision – ACCV 2020: 15th Asian Conference on Computer Vision, Kyoto, Japan, November 30 – December 4, 2020, Revised Selected Papers, Part VI* (Kyoto, Japan). Springer-Verlag, Berlin, Heidelberg, 101–118. doi:10.1007/978-3-030-69544-6_7

[3] Daniel W. Apley and Jingyu Zhu. 2019. Visualizing the Effects of Predictor Variables in Black Box Supervised Learning Models. arXiv:1612.08468 [stat.ME] https://arxiv.org/abs/1612.08468

[4] Maximilian Augustin, Yannic Neuhaus, and Matthias Hein. 2024. DiG-IN: Diffusion Guidance for Investigating Networks – Uncovering Classifier Differences Neuron Visualisations and Visual Counterfactual Explanations. arXiv:2311.17833 [cs.CV] https://arxiv.org/abs/2311.17833

[5] Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. 2015. On Pixel-Wise Explanations for Non-Linear Classifier Decisions by Layer-Wise Relevance Propagation. *PLOS ONE* 10 (07 2015), 1–46. doi:10.1371/journal.pone.0130140

[6] Deepshikha Bhati, MD Amiruzzaman, Ye Zhao, Angela Guercio, and Tram Le. 2025. A Survey of Post-Hoc XAI Methods From a Visualization Perspective: Challenges and Opportunities. *IEEE Access* 13 (2025), 120785–120806. doi:10.1109/ACCESS.2025.3581136

[7] Stefan Bluecher, Johanna Vielhaben, and Nils Strodthoff. 2024. Decoupling Pixel Flipping and Occlusion Strategy for Consistent XAI Benchmarks. *Transactions on Machine Learning Research* (2024). https://openreview.net/forum?id=bIiLXdtUVM

[8] Lennart Brocki and Neo Christopher Chung. 2023. Feature perturbation augmentation for reliable evaluation of importance estimators in neural networks. *Pattern Recognition Letters* 176 (2023), 131–139. doi:10.1016/j.patrec.2023.10.012

[9] Nadia Burkart and Marco F. Huber. 2021. A Survey on the Explainability of Supervised Machine Learning. *J. Artif. Int. Res.* 70 (May 2021), 245–317. doi:10.1613/jair.1.12228

[10] Ho Chan and Eduardo Veas. 2024. Importance Estimate of Features via analysis of their Weight and Gradient profile. (04 2024). doi:10.21203/rs.3.rs-4217886/v1

[11] Chun-Hao Chang, Elliot Creager, Anna Goldenberg, and David Duvenaud. 2019. Explaining Image Classifiers by Counterfactual Generation. arXiv:1807.08024 [cs.CV] https://arxiv.org/abs/1807.08024

[12] Aditya Chattopadhyay, Piyushi Manupriya, Anirban Sarkar, and Vineeth N Balasubramanian. 2019. Neural Network Attributions: A Causal Perspective. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 981–990. https://proceedings.mlr.press/v97/chattopadhyay19a.html

[13] Runjin Chen, Andy Arditi, Henry Sleight, Owain Evans, and Jack Lindsey. 2025. Persona Vectors: Monitoring and Controlling Character Traits in Language Models. arXiv:2507.21509 [cs.CL] https://arxiv.org/abs/2507.21509

[14] Ian Covert, Chanwoo Kim, and Su-In Lee. 2023. Learning to Estimate Shapley Values with Vision Transformers. arXiv:2206.05282 [cs.CV] https://arxiv.org/abs/2206.05282

[15] Ian Covert, Scott Lundberg, and Su-In Lee. 2021. Explaining by Removing: A Unified Framework for Model Explanation. *Journal of Machine Learning Research* 22, 209 (2021), 1–90. http://jmlr.org/papers/v22/20-1316.html

[16] Aaron Fisher, Cynthia Rudin, and Francesca Dominici. 2019. All Models are Wrong, but Many are Useful: Learning a Variable's Importance by Studying an Entire Class of Prediction Models Simultaneously. *Journal of Machine Learning Research* 20, 177 (2019), 1–81. http://jmlr.org/papers/v20/18-760.html

[17] Ruth C. Fong and Andrea Vedaldi. 2017. Interpretable Explanations of Black Boxes by Meaningful Perturbation. In *2017 IEEE International Conference on Computer Vision (ICCV)*. Association for Computing Machinery, 3449–3457. doi:10.1109/ICCV.2017.371

[18] Jerome H. Friedman. 2001. Greedy function approximation: A gradient boosting machine. *The Annals of Statistics* 29, 5 (2001), 1189 – 1232. doi:10.1214/aos/1013203451

[19] Ruigang Fu, Qingyong Hu, Xiaohu Dong, Yulan Guo, Yinghui Gao, and Biao Li. 2020. Axiom-based Grad-CAM: Towards Accurate Visualization and Explanation of CNNs. arXiv:2008.02312 [cs.CV] https://arxiv.org/abs/2008.02312

[20] Atticus Geiger, Duligur Ibeling, Amir Zur, Maheep Chaudhary, Sonakshi Chauhan, Jing Huang, Aryaman Arora, Zhengxuan Wu, Noah Goodman, Christopher Potts, and Thomas Icard. 2025. Causal Abstraction: A Theoretical Foundation for Mechanistic Interpretability. arXiv:2301.04709 [cs.AI] https://arxiv.org/abs/2301.04709

[21] Atticus Geiger, Hanson Lu, Thomas Icard, and Christopher Potts. 2021. Causal Abstractions of Neural Networks. In *Advances in Neural Information Processing Systems*, M. Ranzato, A. Beygelzimer, Y. Dauphin, P.S. Liang, and J. Wortman Vaughan (Eds.), Vol. 34. Curran Associates, Inc., 9574–9586. https://proceedings.neurips.cc/paper_files/paper/2021/file/4f5c422f4d49a5a807eda27434231040-Paper.pdf

[22] Arne Gevaert, Axel-Jan Rousseau, Thijs Becker, Dirk Valkenborg, Tijl De Bie, and Yvan Saeys. 2024. Evaluating feature attribution methods in the image domain. *Machine Learning* 113, 9 (01 Sep 2024), 6019–6064. doi:10.1007/s10994-024-06550-x

[23] Tristan Gomez, Thomas Fréour, and Harold Mouchère. 2022. Metrics for saliency map evaluation of deep learning explanation methods. In *Pattern Recognition and Artificial Intelligence: Third International Conference, ICPRAI 2022, Paris, France, June 1–3, 2022, Proceedings, Part I* (Paris, France). Springer-Verlag, Berlin, Heidelberg, 84–95. doi:10.1007/978-3-031-09037-0_8

[24] Jindong Gu, Yinchong Yang, and Volker Tresp. 2019. Understanding Individual Decisions of CNNs via Contrastive Backpropagation. arXiv:1812.02100 [cs.CV] https://arxiv.org/abs/1812.02100

[25] Isabelle Guyon and André Elisseeff. 2003. An introduction to variable and feature selection. *J. Mach. Learn. Res.* 3, null (March 2003), 1157–1182.

[26] Peter Hase, Harry Xie, and Mohit Bansal. 2021. The out-of-distribution problem in explainability and search methods for feature importance explanations. In *Proceedings of the 35th International Conference on Neural Information Processing Systems (NIPS '21)*. Curran Associates Inc., Red Hook, NY, USA, Article 279, 17 pages.

[27] Stefan Haufe, Rick Wilming, Benedict Clark, Rustam Zhumagambetov, Ahcène Boubekki, Jörg Martin, and Danny Panknin. 2026. Explainable AI needs formalization. *npj Artificial Intelligence* 2, 1 (April 2026), 42. doi:10.1038/s44387-026-00095-1

[28] Johannes Haug, Stefan Zurn, Peter El-Jiz, and Gjergji Kasneci. 2021. On Baselines for Local Feature Attributions. *ArXiv* abs/2101.00905 (2021). https://api.semanticscholar.org/CorpusID:230435957

[29] Giles Hooker, Lucas Mentch, and Siyu Zhou. 2021. Unrestricted permutation forces extrapolation: variable importance requires at least one more model, or there is no free variable importance. *Statistics and Computing* 31, 6 (Nov. 2021), 16 pages. doi:10.1007/s11222-021-10057-z

[30] Sara Hooker, Dumitru Erhan, Pieter-Jan Kindermans, and Been Kim. 2019. *A benchmark for interpretability methods in deep neural networks*. Curran Associates Inc., Red Hook, NY, USA.

[31] Brian Kenji Iwana, Ryohei Kuroki, and Seiichi Uchida. 2019. Explaining Convolutional Neural Networks using Softmax Gradient Layer-wise Relevance Propagation. arXiv:1908.04351 [cs.CV] https://arxiv.org/abs/1908.04351

[32] Cosimo Izzo, Aldo Lipani, Ramin Okhrati, and Francesca Medda. 2021. A Baseline for Shapley Values in MLPs: from Missingness to Neutrality. In *ESANN 2021 proceedings (ESANN 2021)*. Ciaco - i6doc.com, 605–610. doi:10.14428/esann/2021.es2021-18

[33] Saachi Jain, Hadi Salman, Eric Wong, Pengchuan Zhang, Vibhav Vineet, Sai Vemprala, and Aleksander Madry. 2022. Missingness Bias in Model Debugging. arXiv:2204.08945 [cs.CV] https://arxiv.org/abs/2204.08945

[34] I. Elizabeth Kumar, Suresh Venkatasubramanian, Carlos Scheidegger, and Sorelle Friedler. 2020. Problems with Shapley-value-based explanations as feature importance measures. In *Proceedings of the 37th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 119)*, Hal Daumé III and Aarti Singh (Eds.). PMLR, 5491–5500. https://proceedings.mlr.press/v119/kumar20e.html

[35] Zachary C. Lipton. 2018. The mythos of model interpretability. *Commun. ACM* 61, 10 (Sept. 2018), 36–43. doi:10.1145/3233231

[36] Scott M. Lundberg and Su-In Lee. 2017. A unified approach to interpreting model predictions. In *Proceedings of the 31st International Conference on Neural Information Processing Systems* (Long Beach, California, USA) *(NIPS'17)*. Curran Associates Inc., Red Hook, NY, USA, 4768–4777.

[37] Thodoris Lymperopoulos and Denia Kanellopoulou. 2026. From Weight Perturbation to Feature Attribution for Explaining Fully Connected Neural Networks. arXiv:2605.15328 [cs.LG] https://arxiv.org/abs/2605.15328

[38] K Scott Mader. 2013. affNIST (Affine MNIST). Department of Computer Science, University of toronto. https://www.cs.toronto.edu/~tijmen/affNIST/

[39] Antonios Mamalakis, Elizabeth A. Barnes, and Imme Ebert-Uphoff. 2023. Carefully Choose the Baseline: Lessons Learned from Applying XAI Attribution Methods for Regression Tasks in Geoscience. *Artificial Intelligence for the Earth Systems* 2, 1 (2023), e220058. doi:10.1175/AIES-D-22-0058.1

[40] Jacqueline Michelle Metsch and Anne-Christin Hauschild. 2025. BenchXAI: Comprehensive benchmarking of post-hoc explainable AI methods on multi-modal biomedical data. *Computers in Biology and Medicine* 191 (2025), 110124. doi:10.1016/j.compbiomed.2025.110124

[41] Fuseini Mumuni and Alhassan G. Mumuni. 2025. Explainable artificial intelligence (XAI): from inherent explainability to large language models. *ArXiv* abs/2501.09967 (2025). https://api.semanticscholar.org/CorpusID:275606857

[42] National Cancer Institute. n.d.. Genomic Data Commons Data Portal. https://portal.gdc.cancer.gov/. Accessed: 2025-11-10.

[43] Meike Nauta, Jan Trienes, Shreyasi Pathak, Elisa Nguyen, Michelle Peters, Yasmin Schmitt, Jörg Schlötterer, Maurice van Keulen, and Christin Seifert. 2023. From Anecdotal Evidence to Quantitative Evaluation Methods: A Systematic Review

on Evaluating Explainable AI. *Comput. Surveys* 55, 13s (July 2023), 1–42. doi:10.1145/3583558

[44] Joel S. Parker, Michael Mullins, Maggie C.U. Cheang, Samuel Leung, David Voduc, Tammi Vickery, Sherri Davies, Christiane Fauron, Xiaping He, Zhiyuan Hu, John F. Quackenbush, Inge J. Stijleman, Juan Palazzo, J.S. Marron, Andrew B. Nobel, Elaine Mardis, Torsten O. Nielsen, Matthew J. Ellis, Charles M. Perou, and Philip S. Bernard. 2009. Supervised Risk Predictor of Breast Cancer Based on Intrinsic Subtypes. *Journal of Clinical Oncology* 27, 8 (2009), 1160–1167. arXiv:https://ascopubs.org/doi/pdf/10.1200/JCO.2008.18.1370 doi:10.1200/JCO.2008.18.1370 PMID: 19204204.

[45] Vitali Petsiuk, Abir Das, and Kate Saenko. 2018. RISE: Randomized Input Sampling for Explanation of Black-box Models. arXiv:1806.07421 [cs.CV] https://arxiv.org/abs/1806.07421

[46] Alethea Power, Yuri Burda, Harri Edwards, Igor Babuschkin, and Vedant Misra. 2022. Grokking: Generalization Beyond Overfitting on Small Algorithmic Datasets. arXiv:2201.02177 [cs.LG] https://arxiv.org/abs/2201.02177

[47] J. Ren, Zhanpeng Zhou, Qirui Chen, and Quanshi Zhang. 2023. Can We Faithfully Represent Absence States to Compute Shapley Values on a DNN?. In *International Conference on Learning Representations*. https://api.semanticscholar.org/CorpusID:259298245

[48] Yao Rong, Tobias Leemann, Vadim Borisov, Gjergji Kasneci, and Enkelejda Kasneci. 2022. A Consistent and Efficient Evaluation Strategy for Attribution Methods. In *Proceedings of the 39th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 162)*, Kamalika Chaudhuri, Stefanie Jegelka, Le Song, Csaba Szepesvari, Gang Niu, and Sivan Sabato (Eds.). PMLR, 18770–18795. https://proceedings.mlr.press/v162/rong22a.html

[49] Shota Saito, Shinichi Shirakawa, and Youhei Akimoto. 2018. Embedded feature selection using probabilistic model-based optimization. In *Proceedings of the Genetic and Evolutionary Computation Conference Companion* (Kyoto, Japan) *(GECCO '18)*. Association for Computing Machinery, New York, NY, USA, 1922–1925. doi:10.1145/3205651.3208227

[50] Mohammadreza Salehi, Hossein Mirzaei, Dan Hendrycks, Yixuan Li, Mohammad Hossein Rohban, and Mohammad Sabokrou. 2022. A Unified Survey on Anomaly, Novelty, Open-Set, and Out of-Distribution Detection: Solutions and Future Challenges. *Transactions on Machine Learning Research* (2022). https://openreview.net/forum?id=aRtjVZvbpK

[51] Wojciech Samek, Grégoire Montavon, Sebastian Lapuschkin, Christopher J. Anders, and Klaus-Robert Müller. 2021. Explaining Deep Neural Networks and Beyond: A Review of Methods and Applications. *Proc. IEEE* 109, 3 (2021), 247–278. doi:10.1109/JPROC.2021.3060483

[52] Rylan Schaeffer, Brando Miranda, and Sanmi Koyejo. 2023. Are Emergent Abilities of Large Language Models a Mirage? arXiv:2304.15004 [cs.AI] https://arxiv.org/abs/2304.15004

[53] Fabian Schmeisser, Adriano Lucieri, Andreas Dengel, and Sheraz Ahmed. 2026. Spectral Occlusion - Attribution Beyond Spatial Relevance Heatmaps. In *Explainable Artificial Intelligence*, Riccardo Guidotti, Ute Schmid, and Luca Longo (Eds.). Springer Nature Switzerland, Cham, 159–183.

[54] Rui Shi, Tianxing Li, and Yasushi Yamaguchi. 2022. Output-targeted baseline for neuron attribution calculation. *Image Vision Comput.* 124, C (Aug. 2022), 13 pages. doi:10.1016/j.imavis.2022.104516

[55] Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. 2019. Learning Important Features Through Propagating Activation Differences. arXiv:1704.02685 [cs.CV] https://arxiv.org/abs/1704.02685

[56] Suraj Srinivas and Francois Fleuret. 2019. Full-Gradient Representation for Neural Network Visualization. arXiv:1905.00780 [cs.LG] https://arxiv.org/abs/1905.00780

[57] Jacob Steinhardt. 2023. Emergent Deception and Emergent Optimization. https://bounded-regret.ghost.io/emergent-deception-optimization/

[58] Pascal Sturmfels, Scott Lundberg, and Su-In Lee. 2020. Visualizing the Impact of Feature Attribution Baselines. *Distill* (2020). doi:10.23915/distill.00022 https://distill.pub/2020/attribution-baselines.

[59] Mukund Sundararajan, Ankur Taly, and Qiqi Yan. 2017. Axiomatic Attribution for Deep Networks. arXiv:1703.01365 [cs.LG] https://arxiv.org/abs/1703.01365

[60] Michael Tsang, Sirisha Rambhatla, and Yan Liu. 2020. How does This Interaction Affect Me? Interpretable Attribution for Feature Interactions. In *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (Eds.), Vol. 33. Curran Associates, Inc., 6147–6159. https://proceedings.neurips.cc/paper_files/paper/2020/file/443dec3062d0286986e21dc0631734c9-Paper.pdf

[61] Pedro Valois, Koichiro Niinuma, and Kazuhiro Fukui. 2024. Occlusion Sensitivity Analysis with Augmentation Subspace Perturbation in Deep Feature Space. 4817–4826. doi:10.1109/WACV57701.2024.00476

[62] Giulia Vilone and Luca Longo. 2020. Explainable Artificial Intelligence: a Systematic Review. arXiv:2006.00093 [cs.AI] https://arxiv.org/abs/2006.00093

[63] Saurabh Vyawahare. 2024. TMNIST (Typeface MNIST). Kaggle. https://www.kaggle.com/datasets/saurabhvyawahare/tmnist-typeface-mnist

[64] Matthew D. Zeiler and Rob Fergus. 2014. Visualizing and Understanding Convolutional Networks. In *Computer Vision – ECCV 2014*, David Fleet, Tomas Pajdla, Bernt Schiele, and Tinne Tuytelaars (Eds.). Springer International Publishing, Cham, 818–833.

A Proofs

A.1 Inverse Property of Attribution

Proof. First, we express $\mathcal{R}^{t+2}$ in terms of $\mathcal{R}^t$, $d\mathcal{R}^{t,t+1}$ and $d\mathcal{R}^{t+1,t+2}$. Given an input $x \in \mathbb{R}^{n_0}$ and feature $i \in \{1, \ldots, n_0\}$ it holds that:

$$\mathcal{R}_i^{t+2}(x) = \mathcal{R}_i^t(x) + \underbrace{(d\mathcal{R}_i^{t,t+1}(x) + d\mathcal{R}_i^{t+1,t+2}(x))}_{d\mathcal{R}_i^{t,t+2}}. \tag{15}$$

The proof of the theorem reduces to showing that $d\mathcal{R}_i^{t,t+2} = 0$. We proceed by decomposing $d\mathcal{R}^{t,t+2}$ into its constituent terms:

$$d\mathcal{R}^{t,t+2} = \mathcal{I}^T \cdot \underbrace{\underbrace{(f^t_{W_i^t \to W_i^{t+1}}(x) - f^t(x))}_{A} + \underbrace{(f^{t_1}(x) - f^{t+1}_{W_i^{t+1} \to W_i^t}(x))}_{B}}_{d\mathcal{R}^t}$$

$$+ \underbrace{\underbrace{(f^{t+1}_{W_i^{t+1} \to W_i^{t+2}}(x) - f^{t+1}(x))}_{C} + \underbrace{(f^{t+2}(x) - f^{t+2}_{W_i^{t+2} \to W_i^{t+1}}(x))}_{D}}_{d\mathcal{R}^{t+1}} \tag{16}$$

In this expression, $\mathcal{I}^T$ can be excluded. The terms B and C cancel, since $W_i^t = W_i^{t+2}$. Similarly, terms A and D, cancel due to the identity $f^{t+2} = f^t$ and the condition $f^{t+2}_{W_i^{t+2} \to W_i^{t+1}} = f^t_{W_i^t \to W_i^{t+1}}$. Consequently,

$$d\mathcal{R}_i^{t,t+2} = 0, \tag{17}$$

as required. □

A.2 Loss Disentanglement

Cross-Entropy loss is widely regarded as the loss function of choice for classification tasks, owing to its robustness. For a multi-class classification problem, the loss is defined as:

$$\mathcal{L}_{CE} = -\sum_{m=1}^{N}\sum_{c=1}^{[n_L]} y_{m,c}\log(\hat{y}_{m,c}). \tag{18}$$

Here m denotes a sample drawn from N total samples, while $y_{m,c}$ and $\hat{y}_{m,c}$ represent the one-hot encoded label and predicted probability, respectively, for sample m belonging to class to class c. The predicted probabilities are obtained by applying the softmax function to the logits of the output neurons:

$$\hat{y}_{m,c} = \frac{e^{z_{m,c}}}{\sum_{c=1}^{[n_L]} e^{z_{m,c}}}. \tag{19}$$

For this selection of loss function (simplified as $\mathcal{L}$), given a sample m it holds that:

$$\mathcal{L}_m = \log\left(\frac{e^{z_{m,c}}}{\sum_{c=1}^{[n_L]} e^{z_{m,c}}}\right) \tag{20}$$

$$= z_{m,t} - \log\left(\sum_{c=1}^{[n_L]} e^{z_{m,c}}\right) \tag{21}$$

To this point, we observe that the computation in Eq. (21) is skewed by the presence of the term $e^{z_{m,t}}$, which corresponds to the target neuron. This bias is inherent, as no algebraic identity exists to separate the logarithm of a sum of terms. Nevertheless, the induced bias is negligible, enabling the derivation of an effective approximation, leading to:

$$\mathcal{L} \simeq \mathcal{L}_{target} + \mathcal{L}_{non-target}. \tag{22}$$

Deeper into this analysis, the update step requires the calculation of the loss gradient with respect to weight parameters. By exploiting Eq. (22) and the linearity in gradients, this $\frac{\partial \mathcal{L}}{\partial w_{ij}}$ can be expressed as the sum of disentangled gradients:

$$\frac{\partial \mathcal{L}}{\partial w_{ij}} = \frac{\partial \mathcal{L}_{target}}{\partial w_{ij}} + \frac{\mathcal{L}_{non-target}}{\partial w_{ij}} \tag{23}$$

Therefore, by considering the standard procedure of parameter update of a FCNN, the total weight update is:

$$\Delta w_{ij} = -\eta * \frac{\partial \mathcal{L}}{\partial w_{ij}} \tag{24}$$

$$= -\eta * \left[\frac{\partial(\mathcal{L}_{target})}{\partial w_{ij}} + \frac{\partial(\mathcal{L}_{non-target})}{\partial w_{ij}}\right] \tag{25}$$

$$= \Delta w_{target} + \Delta w_{non-target} \tag{26}$$

Thus, the change Δw_{ij} can indeed be approximately disentangled into a component from the target neuron's loss (Δw_{target}) and a component from the aggregation of non-target neurons' loss ($\Delta w_{non-target}$). The transition from a single sample to an entire minibatch only requires the summation of Eqs. (21) and (26) over all m samples.