

Essay on Axiomatic Attribution for Deep Network

Explainable Machine Learning

Jaykumar Bhagiya¹

1 INTRODUCTION

Although deep neural networks perform well on unseen data, users often wonder how these models achieve such remarkable accuracy. To address this curiosity, various techniques have been developed for interpreting machine learning models. Among these methods, feature attribution techniques stand out by assigning importance scores to each feature for a specific input. This allows users to understand which features significantly influence the network's prediction. Additionally, saliency methods reveal which parts of an image the model focuses on when making decisions. Though these categories can overlap—attribution methods can be visualized similarly to saliency methods, and saliency methods can assign importance scores to individual pixels—each offers unique insights into the model's behavior.

One particularly notable attribution method is Integrated Gradients (IG), introduced in the paper "Axiomatic Attribution for Deep Networks." Integrated Gradients identify the features crucial for a network's prediction on a given data point. This method adheres to two fundamental axioms: sensitivity and implementation invariance. Sensitivity ensures that if a feature's value changes, the importance score assigned to that feature should change accordingly. Implementation invariance guarantees that the attribution method's results remain consistent regardless of the specific implementation details of the network. Many previous attribution methods fail to satisfy these axioms, making IG a robust and reliable approach.

In practice, Integrated Gradients assign an importance score $\phi_i(f, x)$ to the i feature of the input x , reflecting how much that feature contributes to or detracts from the network's output $f(x)$. This is done by comparing the model's output for

AFFILIATION¹ Saarland University

CORRESPONDENCE jabh00002@stud.uni-saarland.de

VERSION July 19, 2024

the input x against a baseline—a reference input often chosen to be a neutral or zero value. The method calculates the importance of each feature by integrating the gradients of the model’s output with respect to the input features along a path from the baseline to the target input. A large positive or negative score indicates that the feature significantly influences the network’s prediction, while a score close to zero suggests minimal impact. This approach provides a clear, quantifiable measure of feature importance, adhering to the principles of sensitivity and implementation invariance to ensure accurate and interpretable results.

2 BACKGROUND

While Integrated Gradients (IG) provides a robust solution for feature attribution, it is important to recognize that many earlier methods fall short of these standards. Techniques such as DeepLift [3, 4], Layer-wise Relevance Propagation (LRP) [2], Deconvolutional Networks [8], and Guided Backpropagation [6] fail to satisfy one or both of the essential axioms: sensitivity and implementation invariance.

One simple method for attributing features involves examining the gradients of model outputs with respect to input features. The gradient-based approach calculates the importance score by taking the product of the gradient and feature values, making it an intuitive starting point for attribution [1, 5]. However, gradients often break the sensitivity axiom, highlighting irrelevant features due to their inability to account for zero gradients in certain cases. For example, consider a simple two-layer neural network with a sigmoid activation function, $f(x) = \sigma(w \cdot x + b)$, where $\sigma(z) = \frac{1}{1+e^{-z}}$. Let $w = 1$ and $b = -10$. If the baseline input is $x = 0$ and the input of interest is $x = 15$, the function value changes significantly from $\sigma(-10) \approx 0$ to $\sigma(5) \approx 1$. Despite this substantial change, the gradient at $x = 15$ is zero because $\sigma'(5) = \sigma(5) \cdot (1 - \sigma(5)) \approx 0$. Hence, the gradient method assigns an attribution of 0 to x , even though the function’s output differs greatly from the baseline. This violation of sensitivity occurs because the sigmoid function flattens in its saturation region, resulting in zero gradients despite significant output changes.

Methods such as Deconvolutional Networks and Guided Backpropagation also suffer from sensitivity issues. These techniques backpropagate through ReLU activations only when they are active, ignoring the contribution of features with zero gradients despite their non-zero impact at the baseline. This limitation causes these methods to overlook significant features when gradients are zero at the input but non-zero at the baseline, leading to incomplete attributions.

Techniques like DeepLift and LRP attempt to address sensitivity by using a baseline and computing "discrete gradients," avoiding issues associated with flat regions in gradient calculations. Despite these efforts, they still violate the implementation invariance axiom. While gradients inherently satisfy implementation invariance due to the chain rule, discrete gradients used in methods like DeepLift and LRP do not maintain this property. The chain rule for gradients, $\frac{\partial f}{\partial g} = \frac{\partial f}{\partial h} \cdot \frac{\partial h}{\partial g}$, breaks down for discrete gradients as $\frac{f(x_1) - f(x_0)}{g(x_1) - g(x_0)} \neq \frac{f(x_1) - f(x_0)}{h(x_1) - h(x_0)}$, leading to inconsistent attributions across different implementations.

3 UNDERSTANDING INTEGRATED GRADIENTS (IG)

To understand Integrated Gradients (IG), imagine a path from a baseline input x' to the actual input x . Integrated gradients accumulate gradients along this path to determine the contribution of each feature to the model's output. This method is mathematically formalized as:

$$\phi_i^{IG}(f, x, x') = \overbrace{(x_i - x'_i)}^{\text{Difference from baseline}} \times \underbrace{\int_{\alpha=0}^1 \overbrace{\frac{\delta f(x' + \alpha(x - x'))}{\delta x_i}}^{\text{accumulate local gradients}} d\alpha}_{\text{From baseline to input}} \quad (1)$$

where f is the model function, x is the input, x' is the baseline, and $\frac{\partial f}{\partial x_i}$ represents the gradient of f with respect to the i -th feature.

This method computes importance scores by integrating the gradients of the model's output with respect to the input features, over the path from the baseline to the input. The baseline typically represents an input where all features are absent or neutral. By integrating gradients along the path, we can capture how each feature contributes to the model's prediction as it transitions from the baseline to the actual input.

Gradients indicate the direction and rate of the steepest increase in the model's output. However, directly using gradients for feature importance has limitations due to saturation. Saturation occurs when the model's output becomes insensitive to changes in input features, causing gradients to diminish even if those features are significant. For instance, small shifts in pixel values in an image might not change the model's perception, leading to flattened gradients¹.

Integrated gradients address this issue by considering the entire path from the baseline to the input. By integrating over this path, the method accounts for

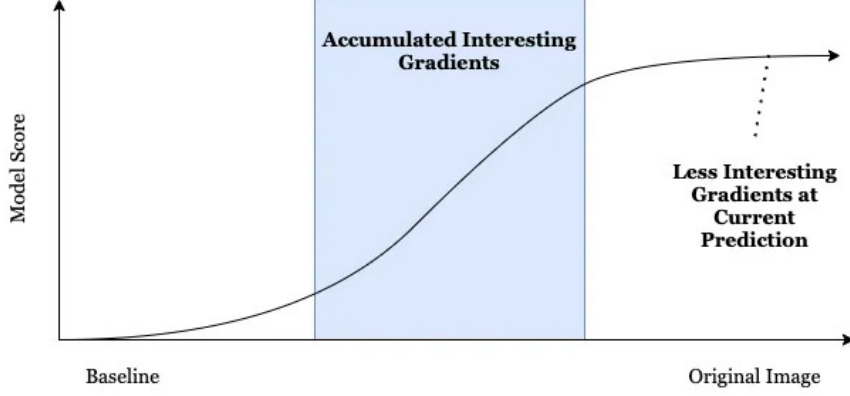


Figure 1: A graph depicting the network outputs for inputs of the form $x' + \alpha(x - x')$. Observe that the output for the correct class becomes saturated at small values of α . By $\alpha = 1$, the network’s output changes very little.

the total contribution of each feature, mitigating the impact of local saturation. The formula for integrated gradients can be broken down into three parts: the interpolation between baseline and input, the gradients at each interpolated point, and the accumulation of these gradients.

$$\int_{\alpha'=0}^{\alpha} \underbrace{(x_i - x'_i) \times \frac{\delta f(\overbrace{x' + \alpha'(x - x')}^{(1): \text{interpolated point}})}{\delta x_i}}_{(2): \text{gradients at interpolation}} d\alpha' = \overbrace{\phi_i^{IG}(f, x, x'; \alpha)}^{(3): \text{accumulated gradients till } \alpha} \quad (2)$$

See the figure 3 in Appendix A for visualization of equation 2.

Completeness is a fundamental property that sets integrated gradients apart from other feature attribution methods. It ensures that the sum of the importance scores for all features accurately reflects the total change in the model’s output when transitioning from a baseline to an actual input. This means that the combined contributions of all features fully account for the difference between the model’s predictions at the baseline and at the input. Formally, completeness is expressed as:

$$\sum_i \phi_i^{IG}(f, x, x') = f(x) - f(x') \quad (3)$$

This property is crucial because it confirms that no portion of the output difference is overlooked, and every feature's contribution is properly represented.

In practice, the integral is approximated using a discrete sum with k points between 0 and 1. This practical computation of integrated gradients can be expressed as:

$$\phi_i^{IG}(f, x, x') \approx (x_i - x'_i) \times \frac{1}{k} \sum_{j=1}^k \frac{\partial f(x' + \frac{j}{k}(x - x'))}{\partial x_i} \quad (4)$$

If we only chose one point to approximate the integral, it would likely be insufficient. Would 10 points be adequate? 100? While 1,000 points might seem sufficient intuitively, it's not guaranteed. To ensure accuracy, as suggested in the original paper, we can use the completeness axiom as a sanity check for convergence: run integrated gradients with k points, measure $|\sum_i \phi_i^{IG}(f, x, x') - (f(x) - f(x'))|$, and if the discrepancy is significant, increase k and re-run the calculations.

4 CHOICES OF BASELINE

The concept of a baseline in integrated gradients is pivotal as it represents the feature values when they are considered "absent" or "missing." However, selecting an appropriate baseline can significantly influence the interpretation of feature importance.

A common practice is to use a baseline of all zeros (i.e., $x' = \mathbf{0}$). However, this approach can lead to misleading results, especially when zero values have specific interpretations. For instance, in a healthcare context, using zero as a baseline for features like blood sugar levels can be problematic. A blood sugar level of zero, which is clinically significant and dangerous, may not appropriately represent the concept of "missingness." In such cases, using zero could result in incorrect attributions, as the difference term $x_i - x'_i$ becomes zero, leading to zero importance for a crucial feature.

The baseline choice is equally critical in image domains. For instance, using a black image as a baseline might fail to highlight black pixels' importance even when they form part of the object of interest. See the figure 2. This issue arises because the method is insensitive to the specific color values of the baseline. As acknowledged by [7], the baseline's role is to highlight missing features. However,

selecting a baseline that does not align with the typical feature values in the dataset can result in distorted attributions.

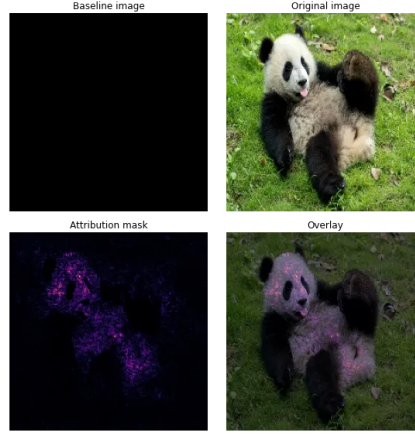


Figure 2: IG fail to emphasize black pixels in the panda image, which are crucial for detection, when a black image is used as the baseline.

Consider another scenario where a dataset contains images with a watermark present in 99% of the samples. If a constant baseline is used, the watermark may receive arbitrary attributions across different classifications. This is due to the fact that each instance’s attributions are computed relative to a baseline that does not account for the watermark’s prevalence. In this case, using a distribution mode baseline, which reflects the most common watermark, might yield more stable and meaningful attributions.

5 CONCLUSION

In summary, Integrated Gradients (IG) represents a significant advancement in the realm of feature attribution, addressing key limitations found in earlier methods. By adhering to the principles of sensitivity and implementation invariance, IG offers a more reliable means of understanding how individual features contribute to a model’s predictions. Its ability to integrate gradients along a path from a baseline to the actual input ensures that feature importance is assessed in a way that captures both subtle and substantial changes in model output. Despite its strengths, the choice of baseline remains a crucial factor in the accuracy of IG’s attributions, as it can influence the interpretation of feature importance. Therefore, while Integrated Gradients provides a robust framework for model

interpretability, careful consideration of baseline selection is essential to fully leverage its capabilities and obtain meaningful insights from machine learning models.

REFERENCES

- [1] David Baehrens, Michael Schroeder, Stephen L Smith, Hans Kern, Chloé Kermorvant, and Jean-Christophe Chappelier. Explainable artificial intelligence (xai). *arXiv preprint arXiv:1004.0421*, 2010.
- [2] Alexander Binder, Greti Montavon, Sebastian Bach, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. Layer-wise relevance propagation for deep neural networks. *arXiv preprint arXiv:1606.08265*, 2016.
- [3] Avanti Shrikumar, Peter Greenside, and Anshul Kundaje. Not just a black box: Learning important features through interpretable machine learning. *arXiv preprint arXiv:1605.01713*, 2016.
- [4] Avanti Shrikumar, Peter Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. *arXiv preprint arXiv:1705.07874*, 2017.
- [5] Karen Simonyan, Andrea Vedaldi, and Andrew Zisserman. Deep inside convolutional networks: Visualising image classification models and saliency maps. In *International Conference on Learning Representations (ICLR)*, 2014.
- [6] Jost Tobias Springenberg, Alexey Dosovitskiy, and Thomas Brox. Striving for simplicity: The all convolutional net. In *International Conference on Learning Representations (ICLR)*, 2015.
- [7] M. Sundararajan, A. Taly, and Q. Yan. Axiomatic attribution for deep networks. *ICML*, 2018. A note about: Local explanation methods for deep neural networks lack sensitivity to parameter values.
- [8] Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. *European Conference on Computer Vision (ECCV)*, 2014.

A APPENDIX

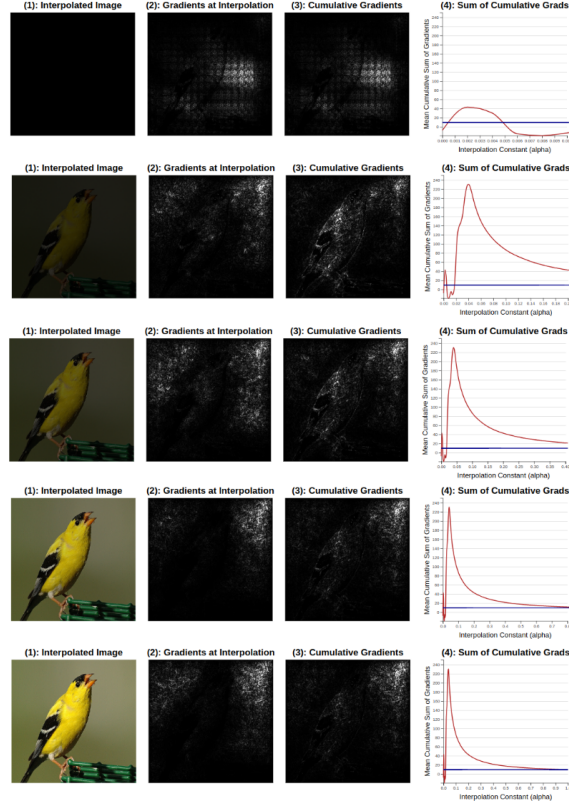


Figure 3: The line chart illustrates integrated gradients: the red line represents the integral computed from (3) in equation 2, while the blue line shows $f(x) - f(x')$. Observe that high-magnitude gradients are concentrated at smaller values of α .