Visualizing the Impact of Integrated Gradients Method

Diwakar Singh, Nadine Fahed, Yu Otsuki

1 Project Summary

The wild success of Deep Neural Network (DNN) models in a variety of domains has created considerable excitement in the machine learning community. Although DNNs are being increasingly adopted in real-world contexts, explaining their behavior has often been difficult. *Explainability* is crucial for a variety of reasons, and various notions of what constitutes an explanation have been proposed. One specific type of explanation is referred to as *attribution*. Attributing the prediction of a deep network to its input features can offer great insight into its behavior. Furthermore, attribution methods are crucial to help the user improve their understanding of networks, debug networks, extract rules from networks, and engage better with networks. Nevertheless, attribution methods are difficult to assess empirically. Therefore, an axiomatic framework is used to overcome this problem. Consequently, a new method named *Integrated Gradients (IG)* based on [1] is introduced and is shown to satisfy the two important axioms for attribution methods: *Sensitivity* and *Implementation Invariance*. The IG method has become a popular interpretability technique due to its broad applicability to any differentiable model (e.g. images, text, structured data), ease of implementation, theoretical justifications, and computational efficiency relative to alternative approaches that allow it to scale to large networks and feature spaces such as images.

This project studied the IG attribution method and identified the fundamental axioms that it satisfies. Another crucial goal this project achieved was to understand the design of this novel IG method and successfully implement it on an image task using the trained network [2] and dataset. For this project, we used feature attribution in an image classification network. Since this network is differentiable, we could successfully apply the integrated gradients method. The results from the implementation were analyzed and compared with gradients at the image, an older attribution method, to demonstrate the superiority of this axiomatic attribution method. Furthermore, an important use of the attribution method is debugging model performance. To highlight this capability, the IG method was used to understand an important limitation of convolutional neural networks like Inception V1 - CNNs that they are not naturally rotationally or scale-invariant. For this purpose, the IG was applied to a wrongfully predicted zoomed-in image to get a deeper feature-level insight into why the model made an error. Finally, a case study was performed to visualize the effect of an important hyperparameter to the IG attribution method: the baseline. This project investigated the effect of using different baselines, namely black, white, uniform, blurred, Gaussian, and maximum distance, on the IG approach to determine the sensitivity of this method to the input baseline hyperparameter.

2 Detailed Project Description

2.1 Introduction and Motivation

In recent years, *Deep Learning* has emerged as an overwhelmingly successful field, and it currently is one of the main foundations behind disruptive technologies such as *self-driving cars* or *virtual assistants*. However, in spite of the fact that functions approximated by Deep Learning can achieve high accuracy, they are regarded as black-boxes. That is, in general, we humans do not understand what reasons drive the relationships between their input and output variables. This tension between interpretability and accuracy is of increasing importance as applications of Deep Learning continue to expand across fields such as Health Care where, arguably, the reasoning behind facts are just as important, if not more, than facts themselves.

In recent years, several approaches have been developed with the goal of tackling this problem. *Attribution Methods* are one of them. The goal of this project is to understand and implement a novel attribution method called "Integrated Gradients (IG)". Attributing the prediction of a deep network to its input features can offer great insight into its behavior. Therefore, it is important to formally define attribution.

Definition: Suppose we have a function $F : \mathbb{R}^n \to [0,1]$ that represents a deep network, and an input $x = (x_1, ..., x_n) \in \mathbb{R}^n$. An attribution of the prediction at x input relative to a baseline input x' is a vector $A_F(x, x') = (a_1, ..., a_n) \in \mathbb{R}^n$ where a_i is the contribution of x_i to the prediction $F(x)$.

The attribution method outputs which features are important to a neural network for the prediction of a particular data point. In this project, we focus on an image classification network where an attribution method could tell which pixels of the image are responsible for a certain label classified by the network. Hence, the attribution method helps us improve our understanding of networks, debug networks, extract rules from networks, and engage better with networks.

Nevertheless, attribution methods are difficult to assess empirically. It is often challenging to identify whether the fault comes from the attribution method or the model itself. Therefore, an axiomatic framework is used to overcome this problem and develop the IG method. The IG method has a strong theoretical justification, can be easily implemented, and has been successfully applied on a variety of networks. Our goal in this project is to evaluate the performance of the IG in an image classification network.

2.2 Integrated Gradients Method

Integrated gradients are defined as the path integral of the gradients along the straight line path from the baseline x' to the input x. Here $\frac{\partial F(x)}{\partial x_i}$ is the gradient of $F(x)$ along the i^{th} dimension.

$$IntegratedGrads_i(x) = (x_i - x_i') \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha(x - x'))}{\partial x_i} d\alpha$$

Here, we introduce two fundamental axioms for attribution methods: *Sensitivity(a)* and *Implementation Invariance*.

Axiom: Sensitivity
An attribution method satisfies *Sensitivity* if for every input and baseline that differ in one feature but have different predictions then the differing feature should be given a non-zero attribution.

Axiom: Implementation Invariance
Two networks are functionally equivalent if their outputs are equal for all inputs, despite having very different implementations. Attribution methods should satisfy *Implementation Invariance*, i.e., the attributions are always identical for two functionally equivalent networks.

Most of the previous attribution methods do not satisfy one of these two axioms. On one hand, if an attribution method fails to satisfy the *Sensitivity(a)* axiom, gradients will tend to focus on unimportant features. On the other hand if an attribution method does not satisfy the *Implementation Invariance* axiom, then the attributions will not necessarily be the same for networks that are functionally equivalent. The IG satisfy the two fundamental axioms. In this sense, using the the axiomatic IG method can rule out faults from the attribution method itself.

2.3 Implementation of Integrated Gradients

Computing Integrated Gradients: In practice, computing a definite integral is not always numerically possible and can be computationally costly. Therefore, for the implementation of IG, the continuous integral is approximated using Riemann sums. The gradients are simply summed at points occurring at sufficiently small intervals along the straight line path from the baseline x' to the input x.

$$IntegratedGrads_i^{approx}(x) = (x_i - x_i') \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m}(x - x'))}{\partial x_i} \frac{1}{m}$$

Here m is the number of steps in the Riemann approximation of the integral. This approximation can be made by computing the gradient in a for loop which is straightforward and efficient for most of the deep networks. $(x_i - x_i')$ is a term for the difference from the baseline. This is necessary to scale the IG and keep them in terms of the original image.

A feature's gradient will vary in magnitude over the interpolated images between the baseline and input. It is important to choose a method to best approximate the area of difference between the baseline and input in the feature space. To implement IG, we care about approximation accuracy and convergence. For this purpose, the Trapezoidal Riemann Sum with $m = 1100$ steps was identified to approximate feature importance within 5% error and have relatively fast convergence. Full implementation using Tensorflow APIs can be found here.

2.4 Experiments

We focus on image classification as a task, as it will allow us to visually plot integrated gradients attributions, and compare them with our intuition about which pixels we think should be important. We use the Inception V4 architecture, a convolutional neural network designed for the ImageNet dataset, in which the task is to determine which class an image belongs to out of 1000 classes.

We use the integrated gradients method to study pixel importance in predictions made by this network. The gradients are computed for the output of the highest-scoring class with respect to pixel of the input image. The integrated gradient also requires choosing a hyperparameter known as the *baseline input.*

Comparison between IG and gradients: IG can be visualized by aggregating them along the color channel and scaling the pixels in the actual image by them. Figure 1 shows visualizations for a bunch of images using blur baseline. For comparison, it also presents the corresponding visualization obtained from the product of the image with the gradients at the actual image. We can see that the gradients at the image barely highlights the important features of the image while the IG attributions are better at reflecting distinctive features.

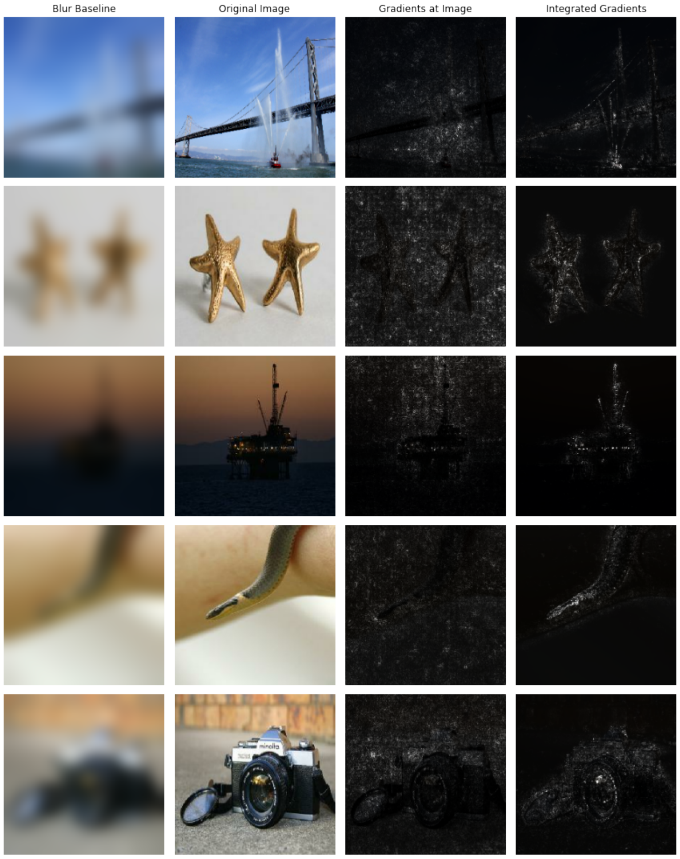

Figure 1: **Comparison of integrated gradients and gradients at the image** Left-to-right: baseline input, original input image, visualization of gradients image and visualization of IG.

Baseline Selection: Implementing the IG attribution method requires the selection of an important hyperparameter: the baseline. In most deep networks, there exists in the input space a natural baseline where the prediction is neutral. It is recommended that the baseline impart a complete absence of a signal in the feature space in addition to having a zero score. Consequently, the attributions can be interpreted as a function of the input only and not the baseline. Sturmfels et al [3] shows that the pixel attributions are different between two baseline images, which indicates that the selection of baseline is crucial. However, the effect of these different baselines are not explored in the original paper [1].

We investigated the impact of the use of different baselines on the IG approach to determine the sensitivity of this method to the input baseline hyperparameter. Figure 2 shows the IG feature attributions with different baseline images: constant black, white, blur, unifrom noise, Gaussian noise, and maximum distance. Clearly, the IG feature attributions highlight different features depending on the baseline images, which indicates the importance of the baseline image selection.

Debugging Model Performance: The IG feature attributions are well suited for counterfactual reasoning to gain insight into model's performance and limitations. This involves comparing feature attributions for images of the same class that receive different predictions. When combined with model performance metrics and dataset statistics, the IG feature attributions give greater insight into model errors during debugging to understand which features contributed to the incorrect prediction when compared to feature attributions on correct predictions.

For the illustration of debugging, Figure 3 shows two input images for the "Yellow Labrador Retriever" image, one is original and the other is zoomed in. Zooming in on the Labrador Retriever image causes Inception V1 to incorrectly predict a different dog breed, a Saluki. By comparing the IG attributions on the incorrect and correct predictions, as shown in Figure 4, one can see the IG attributions on the zoomed image still focus on the legs but they are now much further apart and the midsection is proportionally narrower. This example serves to highlight an important limitation of convolutional neural networks like Inception V1, i.e. CNNs are not naturally scale invariant. Comparing two example attributions, i.e., one incorrect prediction vs one known correct prediction, gives a deeper feature-level insight into why the model made an error to take corrective action.

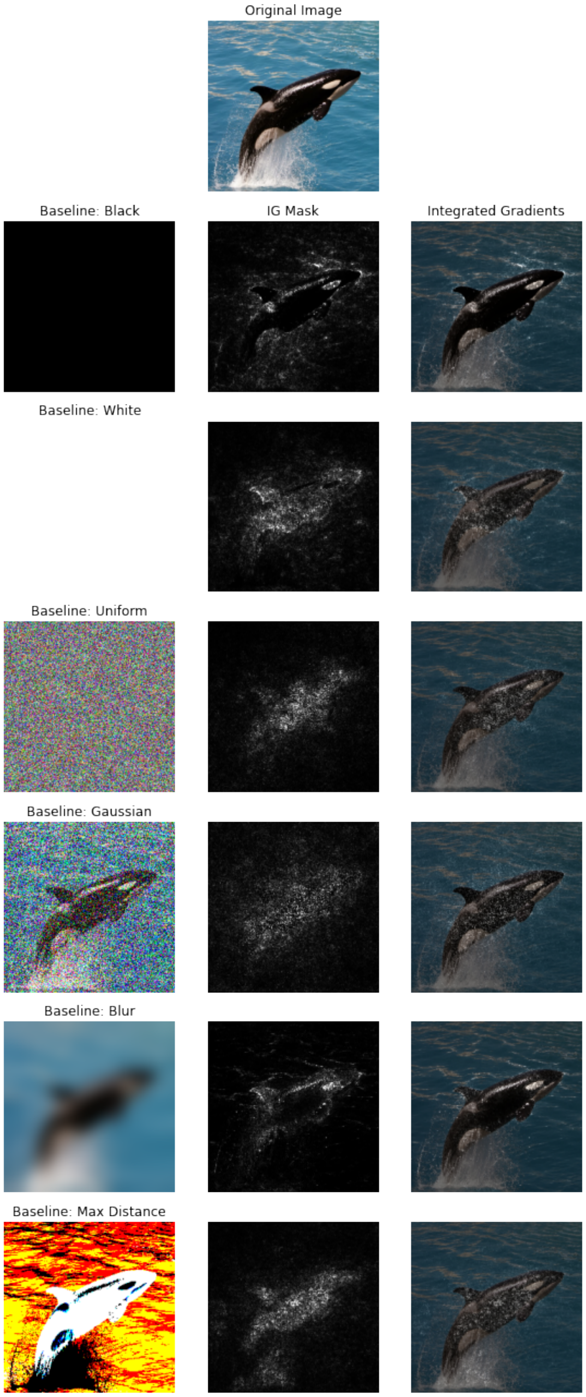

Figure 2: IG pixel attributions for different baseline input

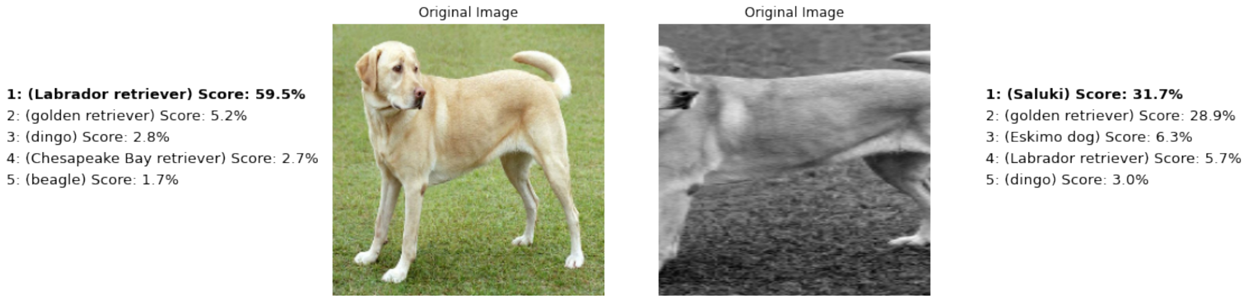
1: (Labrador retriever) Score: 59.5%
2: (golden retriever) Score: 5.2%
3: (dingo) Score: 2.8%
4: (Chesapeake Bay retriever) Score: 2.7%
5: (beagle) Score: 1.7%
Original Image
Original Image
1: (Saluki) Score: 31.7%
2: (golden retriever) Score: 28.9%
3: (Eskimo dog) Score: 6.3%
4: (Labrador retriever) Score: 5.7%
5: (dingo) Score: 3.0%

Figure 3: Yellow Labrador image and the zoomed in image along with the top-5 prediction score and label

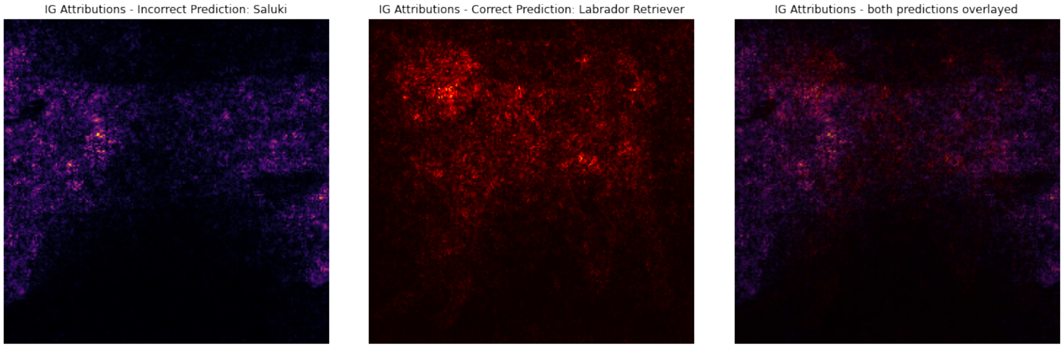

Figure 4: IG pixel attributions for the original and zoomed in image of Yellow Labrador

2.5 Conclusions

The novel IG attribution method was successfully implemented and tested on an image classification trained network. Consequently the results were analyzed and compared with an older attribution method, the gradients at the image, to demonstrate the superiority of the axiomatic IG method. Furthermore, the IG approach was used for counterfactual reasoning to gain a deeper feature-level insight into model errors during debugging. Finally, the effect of the use of different baselines, an important hyperparameter to the IG, was investigated to determine the sensitivity of this method to the input baseline. Additional results can be found in our github repository.

References

[1] M. Sundararajan, A. Taly, and Q. Yan, “Axiomatic attribution for deep networks,” *arXiv preprint arXiv:1703.01365*, 2017.

[2] P. K. Mudrakarta, A. Taly, M. Sundararajan, and K. Dhamdhere, “Did the model understand the question?,” *arXiv preprint arXiv:1805.05492*, 2018.

[3] P. Sturmfels, S. Lundberg, and S.-I. Lee, “Visualizing the impact of feature attribution baselines,” *Distill*, vol. 5, no. 1, p. e22, 2020.