Axiomatic Attribution for Deep Networks

Benjamin Benteke Longau

African Institute for Mathematical Sciences, AIMS-Senegal

August 20, 2021

Overview

Overview

Introduction

Motivation

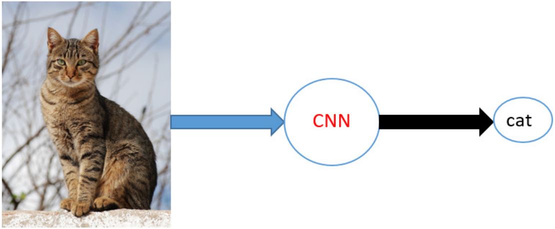


Figure 1: Image classification

Question: What pixels in this image are responsible for this classification?

Overview

Introduction

Research objectives

As research objectives, we have:

- Why Integrated gradient;
- Fundamental Axioms;
- Integrated gradient method;
- How to apply Integrated Gradient.

Overview

Fundamental Axioms

Definition

Axioms are a desirable characteristics that we want a methods to have, so we can trust that they will do a good job of attributing the right scores to the right input features. In this presentation we identify two axioms: *Sensitivity* and *Implementation Invariance*.

Overview

Fundamental Axioms

Axioms

- **Axiom of Sensitivity:** if two samples differ only by one feature and have different outputs by the NN, then the attribution of this feature should be non-null.

Fundamental Axioms

Axioms

- **Axiom of Sensitivity:** if two samples differ only by one feature and have different outputs by the NN, then the attribution of this feature should be non-null.
- **Axiom of Implementation Invariance:** When two neural networks compute the same mathematical function, regardless of how differently they are implemented, the attributions to all features should always be identical.

Overview

1. Introduction
   - Motivation
   - Research objectives
2. Fundamental Axioms
   - Definitions
   - Axioms
3. Integrated Gradient method
   - Concepts
4. Integrated Gradient method
   - Integrated Gradient methods
5. Simulation
   - Experiment and Results
6. Conclusion and perspectives

Concepts

Concepts

Let $F : \mathbb{R}^n \to [0, 1]$ be a function that represents a deep network, and an input $x = (x_1, ..., x_n) \in \mathbb{R}^n$.

- An *attribution* of the prediction at input x 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 $F(x)$.

Concepts

Concepts

Let $F : \mathbb{R}^n \to [0, 1]$ be a function that represents a deep network, and an input $x = (x_1, ..., x_n) \in \mathbb{R}^n$.

- An *attribution* of the prediction at input x 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 $F(x)$.
- A *baseline* is an informative input used at the starting point for calculating the input features importance.

Overview

Integrated Gradient methods

Integrated Gradient methods

$$IG_i(x) = (x_i - x'_i) \times \int_{\alpha=0}^{1} \frac{\partial F(x' + \alpha \times (x - x'))}{\partial x_i} d\alpha \tag{4.1}$$

- i, x and x' feature, input and baseline respectively;
- α interpolation constant.

Integrated Gradient methods

Integrated Gradient methods

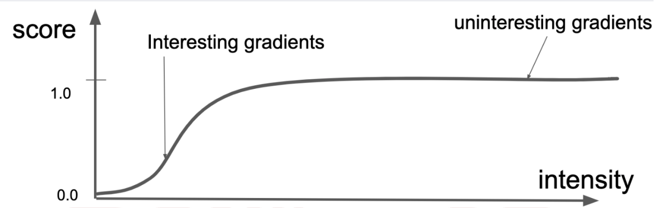


Figure 2: Integrated gradient illustration

Integrated Gradient methods

Integrated Gradient methods

$$IG_i(x) \approx (x_i - x_i') \times \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m} \times (x - x'))}{\partial x_i} \times \frac{k}{m} \qquad (4.2)$$

- k scaled feature perturbation constant;
- m the number of steps in the Riemann sum approximation of the integral.

Integrated Gradient methods

Integrated Gradient methods

$(x' + \alpha \times (x - x'))$ is the interpolated image. It is a path from the baseline to the input image. In this case IG is the integral in a straight line. The integrated gradient is just the average of the gradient of the output with respect to the inputs (series of interpolated images) and it gives the attributions of each feature.

Integrated Gradient methods steps

Integrated Gradient methods steps

1. consider a baseline image;
2. find the interpolated images by increasing intensity between the baseline and the original image;
3. compute the softmax scores of these interpolated images;
4. the region of interest are where the slope of score vs intensity graph does not remain stagnant. This is called gradients interesting gradients;
5. Gradients of the output with respect to these series of interpolated images.

Additional axioms

Additional axioms

- **Axiom of Completeness:**

$$\sum_{i=1}^{n} IG_i(x) = F(x) - F(x') \tag{4.3}$$

where $F(x') \approx 0$.

Additional axioms

Additional axioms

- **Axiom of Completeness:**

$$\sum_{i=1}^{n} IG_i(x) = F(x) - F(x') \tag{4.3}$$

where $F(x') \approx 0$.

- **Axiom of Linearity preservation:**

$$IG(\alpha \times i + \beta \times j) = \alpha \times IG(i) + \beta \times IG(j) \tag{4.4}$$

Additional axioms

Additional axioms

- **Axiom of Completeness:**

$$\sum_{i=1}^{n} IG_i(x) = F(x) - F(x') \tag{4.3}$$

where $F(x') \approx 0$.

- **Axiom of Linearity preservation:**

$$IG(\alpha \times i + \beta \times j) = \alpha \times IG(i) + \beta \times IG(j) \tag{4.4}$$

- **Axiom of Symmetry preservation:** Symmetric variables with identical values get equal attributions.

Overview

1. Introduction
   - Motivation
   - Research objectives
2. Fundamental Axioms
   - Definitions
   - Axioms
3. Integrated Gradient method
   - Concepts
4. Integrated Gradient method
   - Integrated Gradient methods
5. Simulation
   - Experiment and Results
6. Conclusion and perspectives

Experiments and results

Results and evaluation

Figure 3: Images classification

Experiments and results

Results and evaluation

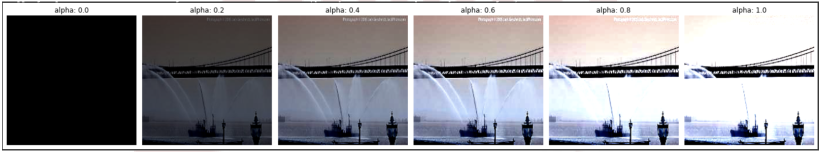


Figure 4: Interpolated images, where $m = 50$

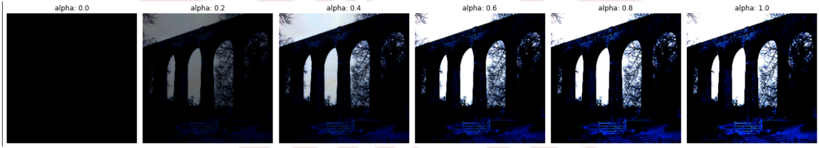
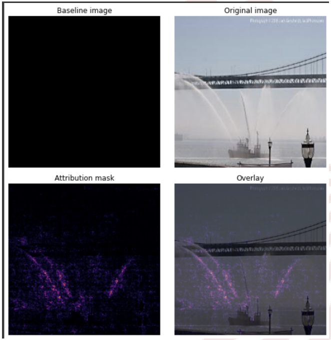


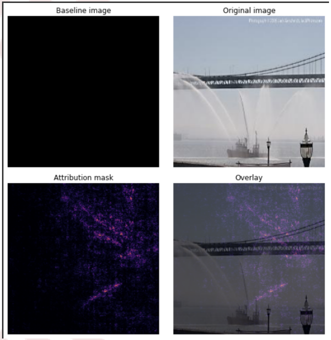
Figure 5: Interpolated images, where $m = 50$

Experiments and results

Results and evaluation



(a) Integrated gradient Result



(b) Gradient Result

Conclusion and Perspectives

In conclusion, we have been able to:

- understand how to apply integrated gradient for image classification.
- Integrated gradient is useful for model deployment.
- As limitation, it provides feature importances on individual sample, but not accross an entire dataset. And it does not explain feature interactions and combinations.

Further Reading

Sundararajan, Mukund and Taly, Ankur and Yan, Qiqi
Axiomatic attribution for deep networks
PMLR 2017.

Thank You for listening!