Integrated gradients

Axiomatic Attribution for Deep Networks

Mukund Sundararajan, Ankur Taly, Qiqi Yan

Ronen Rojas

01.11.2021

About the paper

- Cited 1868 (Google Scholar)
- 8 sections
  1. Definition and motivation
  2. 2 axioms
  3. Integrated gradients (IG)
  4. IG uniqueness
  5. IG implementation
  6. IG applications
  7. Related work
  8. Conclusions

- Function (deep network) $F : \mathbb{R}^n \rightarrow [0,\ 1]$
- Attribution x relative to a baseline x'
- $A_F(x, x') = (a_1, ..., a_n) \in R^n$
- Choice of baseline is important:
  - Black Image
  - Zero vector embedding

Section 2 - Axioms

Summary

- 2 fundemuntals Axioms
  1. *Sensitivity*
  2. *Implementation Invariance*
- Method mentioned that break it.
  - Deconvolutional networks (DeConvNets) - break *Sensitivity*
  - Guided back-propagation - break *Sensitivity*
  - DeepLift - break *Implementation Invariance*
  - Layer-wise relevance (LRP) - break *Implementation Invariance*

Axiom 1 - *Sensitivity*

An attribution method satisfies Sensitivity(a) 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

- Gradients are natural analog of the model coefficients
- Starting point for attribution method
- Break Sensitivity - DeConvNets, Guided back-propagation

Simple example

$$f(x) = 1 - \text{ReLU}(1 - x)$$
$$x = 2,\ x' = 0$$
$$f(0) = 0,\ f(2) = 1$$
$$f'(2) = 0$$

- DeepLift and LRP tackle the Sensitivity - baseline.
- "Discrete gradients" instead of (instantaneous) gradients

Axiom 2 - 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

Example for functionally equivalent:

- Network architecture has more degrees freedom to represent a function
- Training process can converge to either sets of weights
- Attributions should be the same

Section 2 - Implementation Invariance

- Gradients are invariant to implementation
- f output, g input.
- $\frac{\partial f}{\partial g} = \frac{\partial f}{\partial h} \cdot \frac{\partial h}{\partial g}$, h is implementation dependent (detail)
- LRP and DeepLift uses discrete gradients
- $\frac{f(x_1)-f(x_0)}{g(x_1)-g(x_0)} \neq \frac{f(x_1)-f(x_0)}{h(x_1)-h(x_0)} \cdot \frac{h(x_1)-h(x_0)}{g(x_1)-g(x_0)}$

Section 3 - Integrated Gradients

In a nutshell

$$\text{IG} = \overbrace{\{LRP, DeepLift\}}^{Sensitivity} + \{\text{Implementation Invariance}\}$$

Formal Defintion

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

Defintion

If F is differentiable almost everywhere then:
$\sum_{i=1}^{n} IG_i(x) = F(x) - F(x')$

- Sanity check the attribution.
- Fundamental theorem of calculus for path integrals.
- Completeness $\Rightarrow$ Sensitivity.
- IG - satisfy Implementation Invariance.

Section 4 - Uniqueness

therein lies the rub

- Every empirical evaluation technique didn't work
- Couldn't delineate data error, a misbehaving model, and a misbehaving attribution method
- This explains the axiomatic approach
- IG isn't unique but somewhat canonical

Setup

$\gamma = (\gamma_1, ..., \gamma_n) : [0, 1] \mapsto \mathbb{R}^n$
$\gamma(0) = x', \gamma(1) = x$

Definition

$$\text{PathIntegratedGrads}_i^{\gamma}(x) ::= \int_{\alpha=0}^{1} \frac{\partial F(\gamma(\alpha))}{\partial \gamma_i(\alpha)} \cdot \frac{\partial \gamma_i(\alpha)}{\partial \alpha} d\alpha$$

- Satisfy Implementation Invariance and Completeness.

Sensitivity(b)

If the function implemented by the deep network does not depend (mathematically) on some variable, then the attribution to that variable is always zero

Proposition

Path methods are the only attribution methods that always satisfy Implementation Invariance, Sensitivity(b), Linearity, and Completeness.

- Integrated gradients correspond to a cost-sharing method called Aumann-Shapley

Section 4 - Uniqueness

Symmetry-Preserving

Two input variables are symmetric w.r.t. a function if swapping them does not change the function.

Symmetry-Preserving attribution

An attribution method is symmetry preserving, if for all inputsthat have identical values for symmetric variables and baselines that have identical values for symmetric variables, the symmetric variables receive identical attributions.

Proposition

Integrated gradients is the unique path method that is symmetry-preserving.

Section 5 - Implementation

- Near zero predication on baseline
- Baseline convey a complete absence of signal.
- $\text{IntegratedGrad}_i^{approx}(x) ::= (x_i - x_i') \cdot \sum_{k=1}^{m} \frac{\partial F(x' + \frac{k}{m}(x - x'))}{\partial x_i} \cdot \frac{1}{m}$
- Use sanity check to tune m

Section 6 - Applications

1. Object Recognition Network - Vision
2. Diabetic Retinopathy Prediction - Vision
3. Question Classification -NLP
4. Neural Machine Translation -NLP
5. Chemistry Models

Future Reads

1. *Improving performance of deep learning models with axiomatic attribution priors and expected gradients* 2020- Attribution priors with expected gradients attributions.
2. *Attribution in Scale and Space* 2020 - Blur Integrated Gradients.
3. *Explaining Explanations: Axiomatic Feature Interactions for Deep Networks* 2021 - Integrated Hessians , pairwise feature interactions
4. *Visualizing Deep Networks by Optimizing with Integrated Gradients* 2019 - I-GOS, which optimizes for a heatmap.
5. *Robust Attribution Regularization* 2019 - Training objectives in classic robust optimization models to achieve robust IG attributions
6. *Understanding Integrated Gradients with SmoothTaylor for Deep Neural Network Attribution* 2021 - SmoothTaylor = Integrated Gradients + SmoothGrad