

Axiomatic Attribution for Deep Networks

Definition 1. 假设神经网络可以有一个函数表达： $F: R^n \rightarrow [0,1]$ (class score)

输入： $x = (x_1, x_2, \dots, x_n) \in R^n$ ；在参考值 x' 的基础上，某个输入 x 的特定预测值的

attribution map： $A_F(x, x') = (a_1, \dots, a_n) \in R^n$ ，其中 a_i 代表 x_i 对 $F(x)$ 的贡献量

value






Two Fundamental Axioms

Sensitivity(a) :

当改变某个特征的输入值或参考值时，NN产生了不同的预测结果，那么应该给予这个特征非0的attribution *saturation*

Implementation Invariance :

若两个神经网络输入输出完全相等，则称这两个神经网络功能性等同（functional equivalent），即使网络中的运算不同；两个功能性等同的网络attribution也应相同 *chain rule*



Integrated Gradients

Deep Network $F : R^n \rightarrow [0,1]$

Input $x = (x_1, x_2, \dots, x_n) \in R^n$

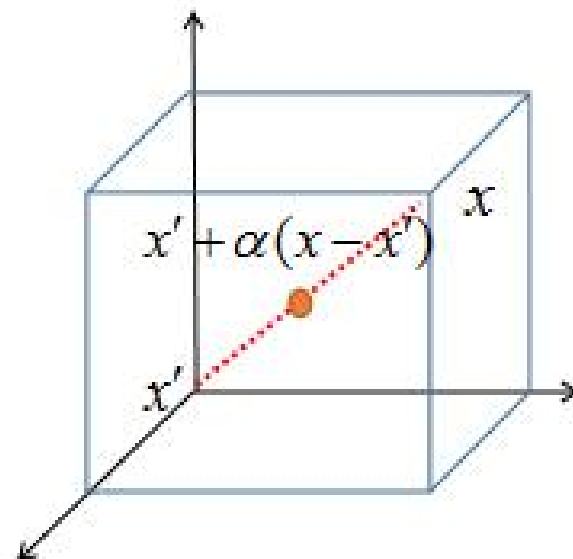
Baseline Input $x' \in R^n$ (img : black img ; text : zero embedding vector)

Conditions : x' 与 x 在一条直线上 (R^n)

计算这条直线上某个特征维度的所有点的梯度，再积分

$$IntegratedGrads_i(x) ::= (x_i - x'_i) \times \int_{\alpha=0}^1 \frac{\partial F(x' + \alpha \times (x - x'))}{\partial (x'_i + \alpha \times (x_i - x'_i))} d\alpha \quad \alpha \in [0,1]$$

Integrated Gradients



计算的是 x' 到 x 路径上的平均梯度

Computing integrated gradients. 积分近似

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

Original image



Top label and score

Top label: reflex camera

Score: 0.993755

Integrated gradients



Gradients at image



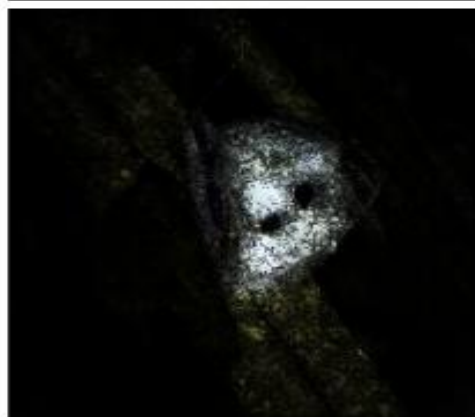
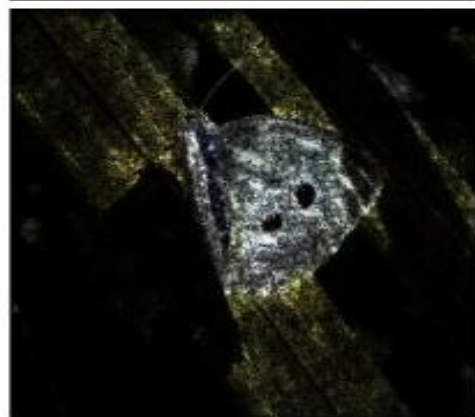
Top label: viaduct

Score: 0.999994



Top label: cabbage butterfly

Score: 0.996838



谢谢