Layerwise Relevance Propagation

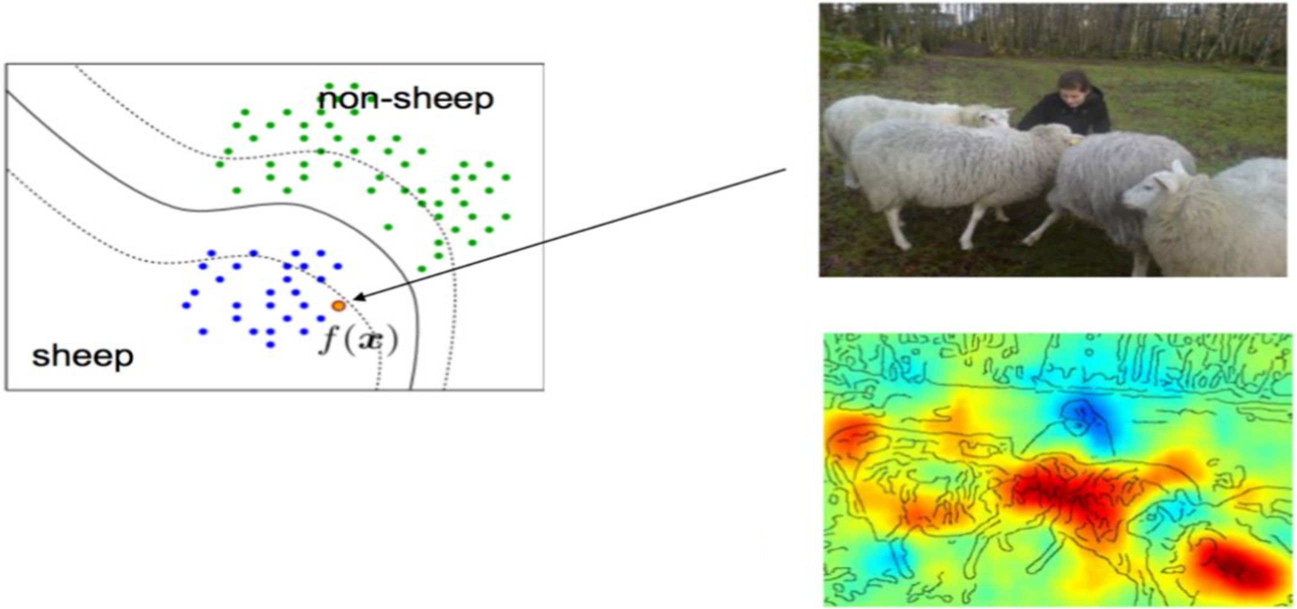
For Neural Nets with Local Renormalization Layer

Structure of the talk:

- What? Explainability
- Why?
    - Previous methods - Sensitivity.
    - Issues
    - New Idea?
- How? - Layer Wise Relevance Propagation.
    - Intuition
    - Setup
    - Hurdle - Non Linearity
    - Solution - Taylor Decomposition
        - Deep Taylor Decomposition
- A special case:
    - LRP for Layer Normalization.
    - Experiments for LRP with Layer Normalization.

The WHAT: Explainability

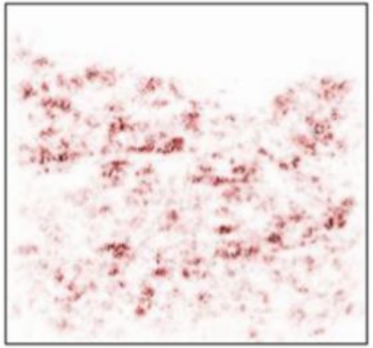
"Why is a given image classified as a sheep?"



Previous Approach: Sensitivity

Sensitivity analysis:

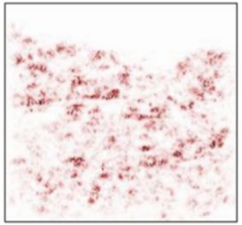
$$R_i = \left(\frac{\partial f}{\partial x_i}\right)^2$$



Previous Approach: Sensitivity, cont.

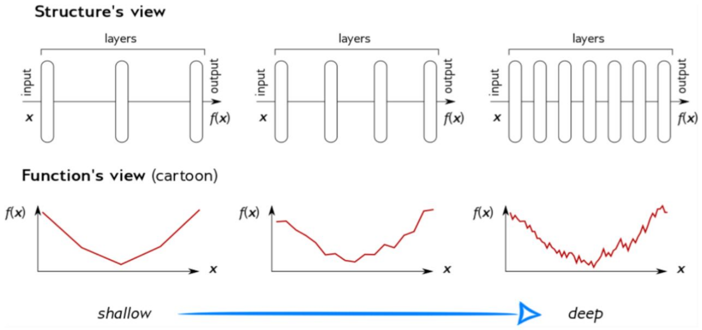
Sensitivity analysis:

$$R_i = \left(\frac{\partial f}{\partial x_i}\right)^2$$



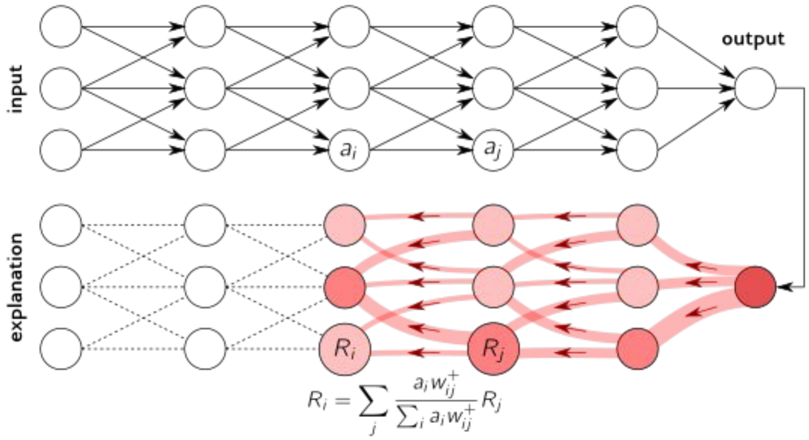
Problem: sensitivity analysis does not highlight cars

Input gradient (on which sensitivity analysis is based), becomes increasingly highly varying and unreliable with neural network depth.



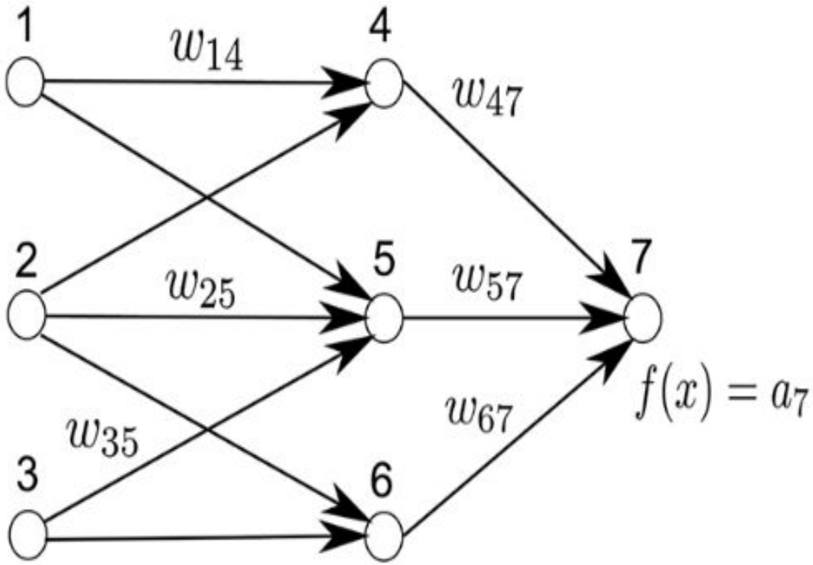
New Idea:

Propagate from backwards?



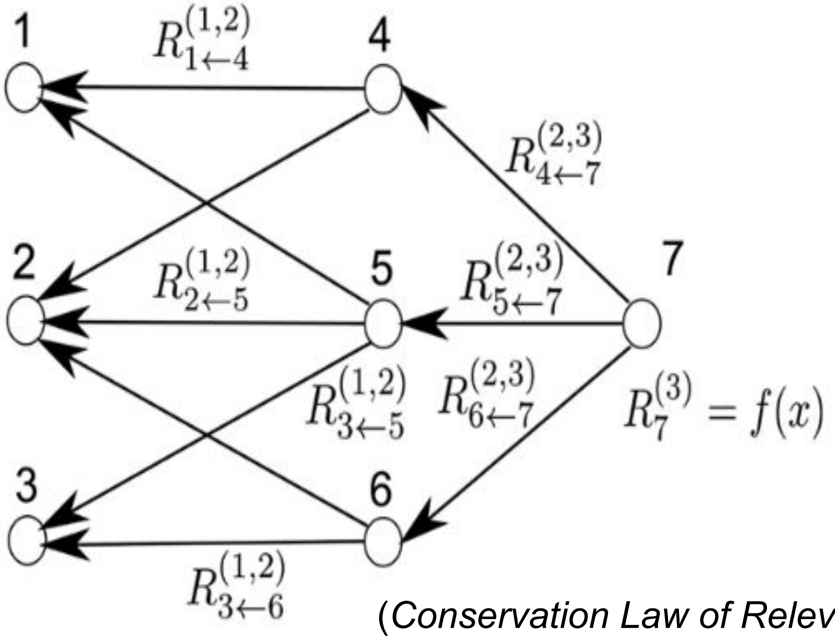
Cue: LRP
Layer Wise Relevance Propagation

Forward Training



$$R_7^{(3)} = R_4^{(2)} + R_5^{(2)} + R_6^{(2)}$$

Relevance Backprop

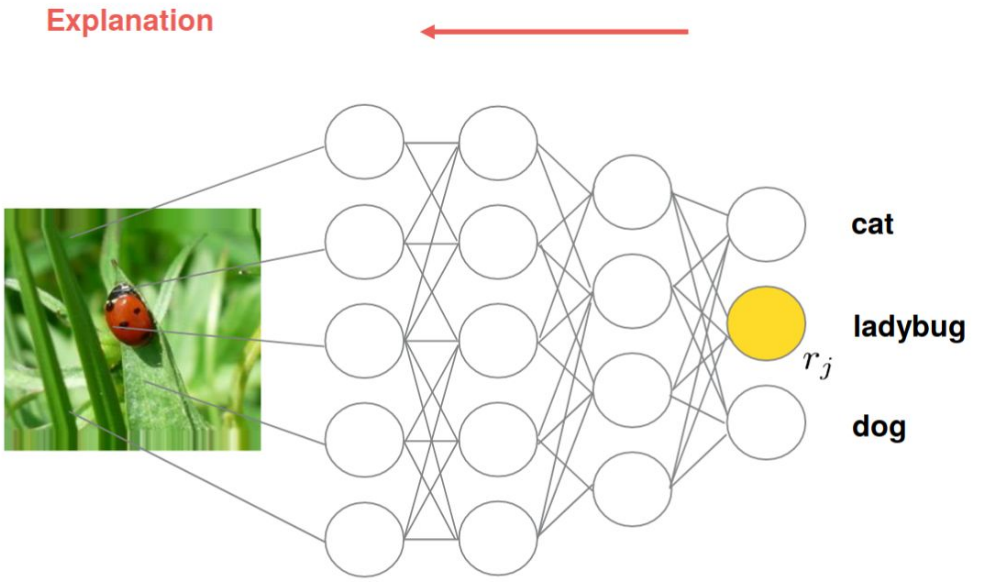


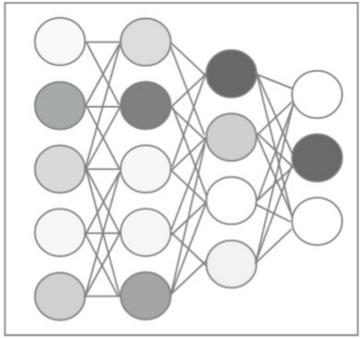
(*Conservation Law of Relevance*)

$$R_4^{(2)} + R_5^{(2)} + R_6^{(2)} = R_1^{(1)} + R_2^{(1)} + R_3^{(1)}$$

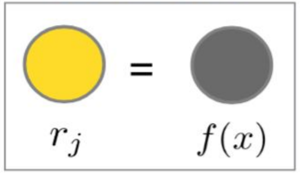
where $R^{(L, L+1)}_{i \leftarrow j}$ = Message sent to neuron *i* at layer (*L)* by neuron *j* at layer *(L+1)*

Explaining Neural Network Predictions

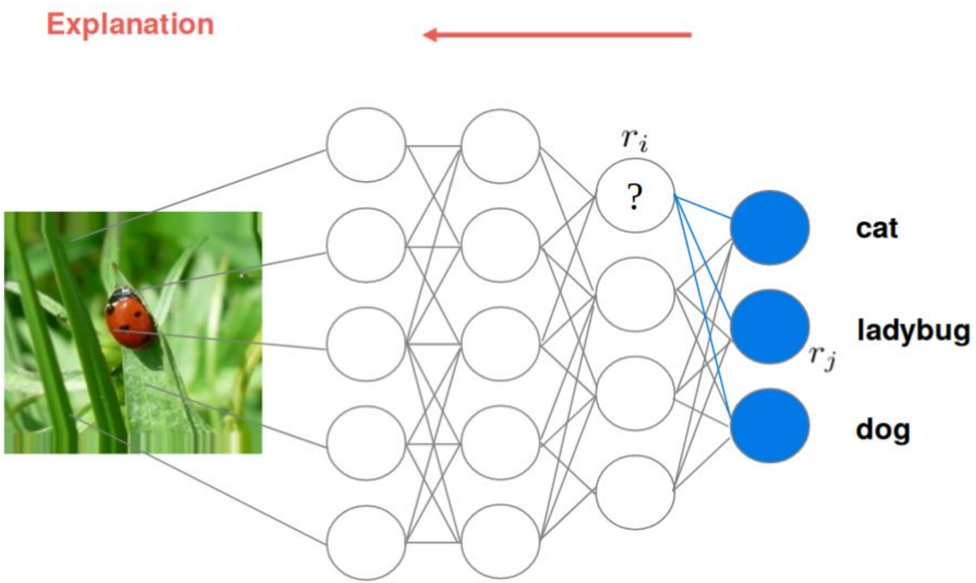


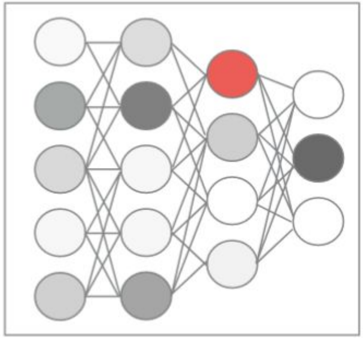


Initialization



Explaining Neural Network Predictions



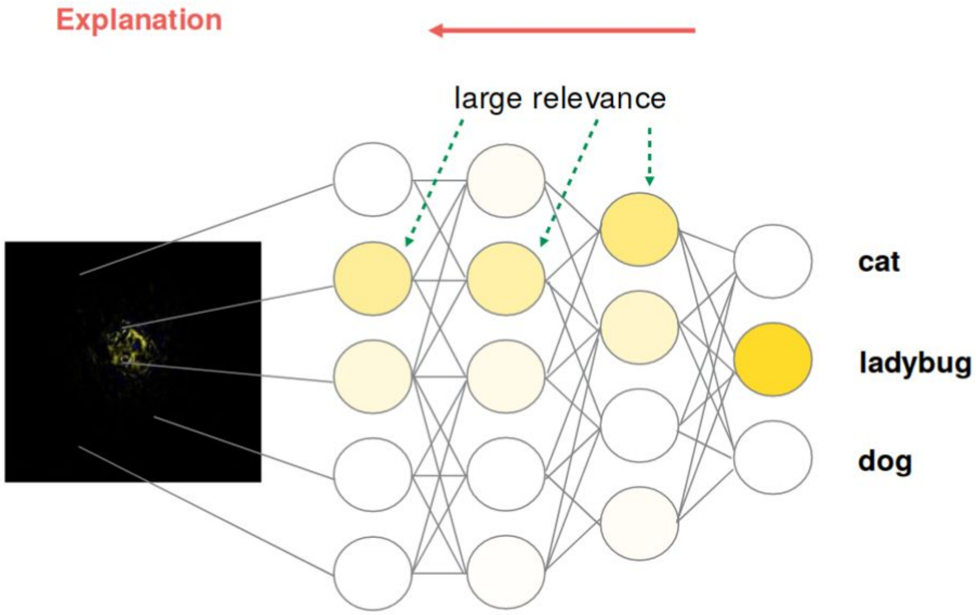


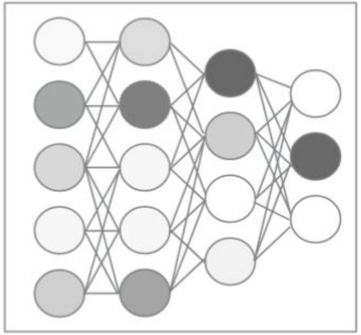
$$r_i = x_i \sum_j \frac{w_{ij} r_j}{\sum_i x_i w_{ij}} = x_i c_i$$

Theoretical interpretation
Deep Taylor Decomposition

r_i depends on the activations **and** the weights

Explaining Neural Network Predictions





Relevance Conservation Property

$$\sum_p r_p = \ldots = \sum_i r_i = \sum_j r_j = \ldots = f(x)$$

General Framework of LRP for neural nets

The general framework of LRP is to find $\mathbf{v_{ij}}$ which intuitively make sense

$$R_{i\leftarrow j}^{(l,l+1)} = v_{ij} R_j^{(l+1)} \quad \text{with} \quad \textstyle\sum_i v_{ij} = 1$$

1. Linear network: $\mathbf{v_{ij}}$ can be directly proportional to the activation.
2. Non linear function like **ReLU** and **tanh :**
   Due to monotonicity of ReLU and **tanh** pre-activation inputs (weight * inputs) still makes sense

$$R_{i\leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j + \epsilon \cdot \mathrm{sign}(z_j)} R_j^{(l+1)}$$

$$R_{i\leftarrow j}^{(l,l+1)} = \left((1+\beta)\frac{z_{ij}^{+}}{z_j^{+}} - \beta \frac{z_{ij}^{-}}{z_j^{-}} \right) R_j^{(l+1)}$$

Deep Taylor Decomposition / LRP

$$R_j = \sum_k \left(\alpha \frac{a_j w_{jk}^+}{\sum_j a_j w_{jk}^+} - \beta \frac{a_j w_{jk}^-}{\sum_j a_j w_{jk}^-} \right) R_k$$

intuition

[Bach'15]

analysis

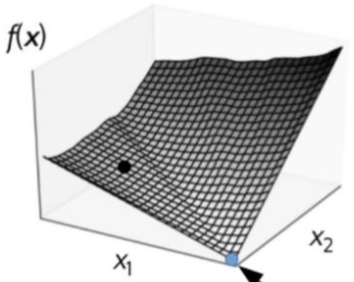
[Montavon'17]

Relevance should be redistributed to the lower-layer neurons $(a_j)_j$ in proportion to their excitatory effect on a_k. "Counter-relevance" should be redistributed to the lower-layer neurons $(a_j)_j$ in proportion to their inhibitory effect on a_k.

For the specific case $\alpha = 1$, the whole LRP procedure can be seen as a *deep Taylor decomposition* of the neural network function.

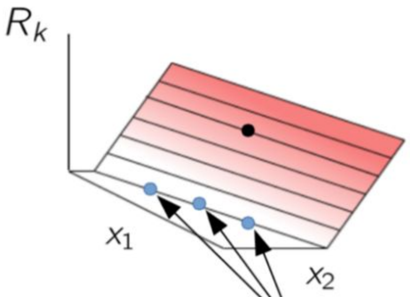
Deep Taylor Decomposition / LRP

Simple Taylor



$$R_i = \frac{\partial f}{\partial x_i} \cdot (x_i - 0)$$

Deep Taylor

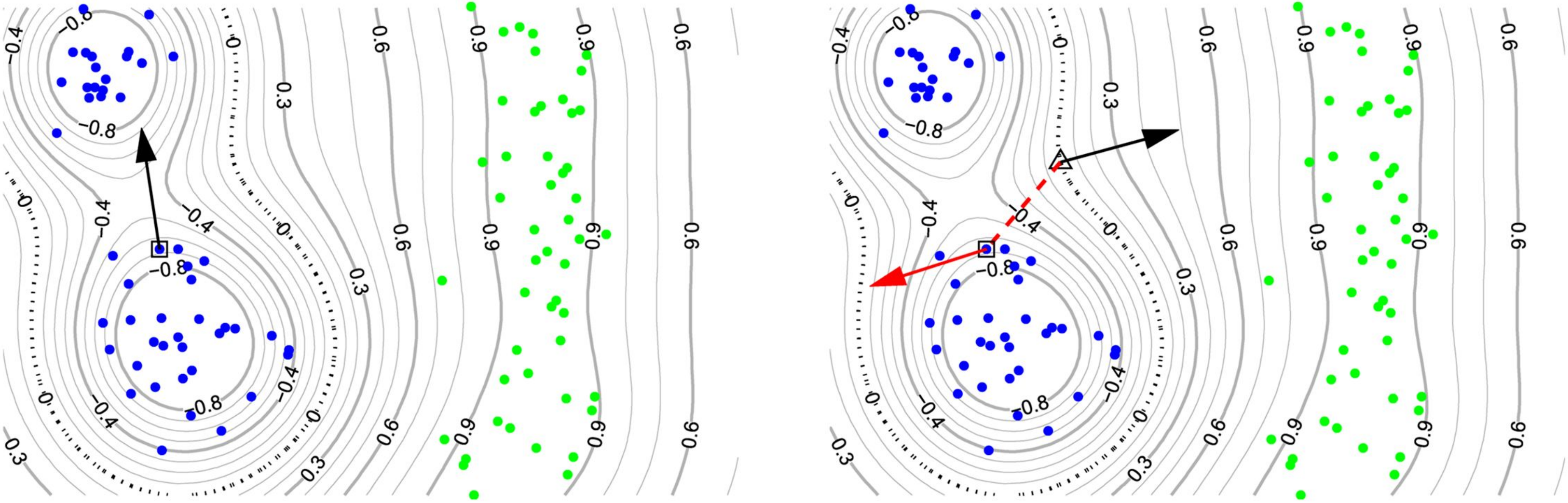


$$R_i = \frac{\partial R_k}{\partial a_j} \cdot (a_j - \widetilde{a}_j)$$

Taylor Expansion for like ReLU and tanh- Root Points

$$f(x) \approx f(x_0) + Df(x_0)[x - x_0]$$
$$= f(x_0) + \sum_{d=1}^{V} \frac{\partial f}{\partial x_{(d)}}(x_0)(x_{(d)} - x_{0(d)})$$

$$f(x) \approx \sum_{d=1}^{V} \frac{\partial f}{\partial x_{(d)}}(x_0)(x_{(d)} - x_{0(d)}) \quad \text{such that } f(x_0) = 0$$



Local Renormalization Layers in Neural Nets

Until Now: A Taylor-based approach was used in for decomposing ReLU neurons by exploiting their local linearity.

This Paper:The paper considers how to deal with a special class of non-linear neurons known for local renormalization i.e. calculate v_ij (distributing relevance)

$$R_{i \leftarrow j}^{(l,l+1)} = v_{ij} R_j^{(l+1)} \quad \text{with} \quad \sum_i v_{ij} = 1$$

Where the non linearity is renormalization and described by

$$y_k(x_1, \ldots, x_n) = \frac{x_k}{\left(1 + b \sum_{i=1}^{n} x_i^2\right)^c}$$

Non Linearity to Linearity - Taylor Series

For a nonlinear function y_k:

$$\frac{\partial y_k}{\partial x_j} = \frac{\delta_{kj}}{\left(1 + b\sum_{i=1}^{n} x_i^2\right)^c} - 2bc\frac{x_k x_j}{\left(1 + b\sum_{i=1}^{n} x_i^2\right)^{c+1}}$$

Choice of root point ?

- Z = (0 ,0 ,0 ……. , x_k , …….0) - No off diagonal contributions

Therefore we choose this

- Z = (x_1, x_2 , ……… …x_n)

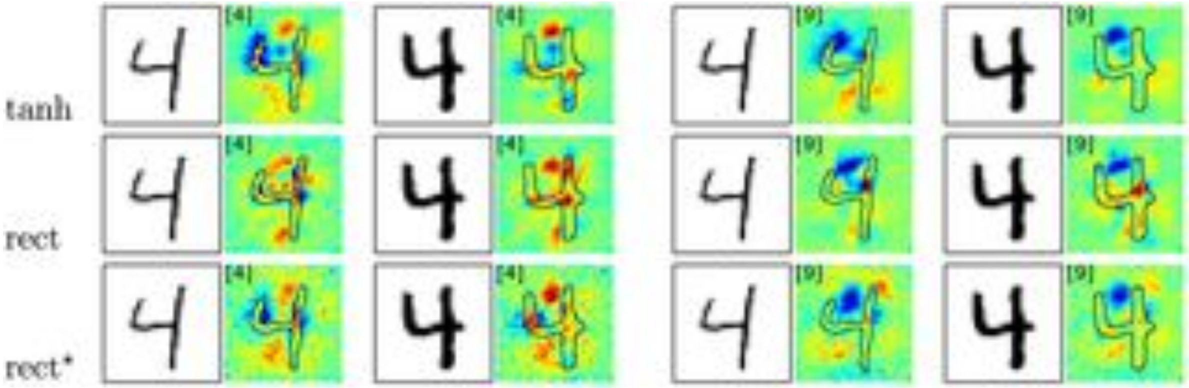
Taylor series for Batch Renormalization

$$y_k(z_1) \approx \frac{x_k}{(1+bx_k^2)^c} - 2bc \sum_{j:j\neq k} \frac{x_k x_j^2}{(1+b\sum_{i=1}^{n} x_i^2)^{c+1}}$$

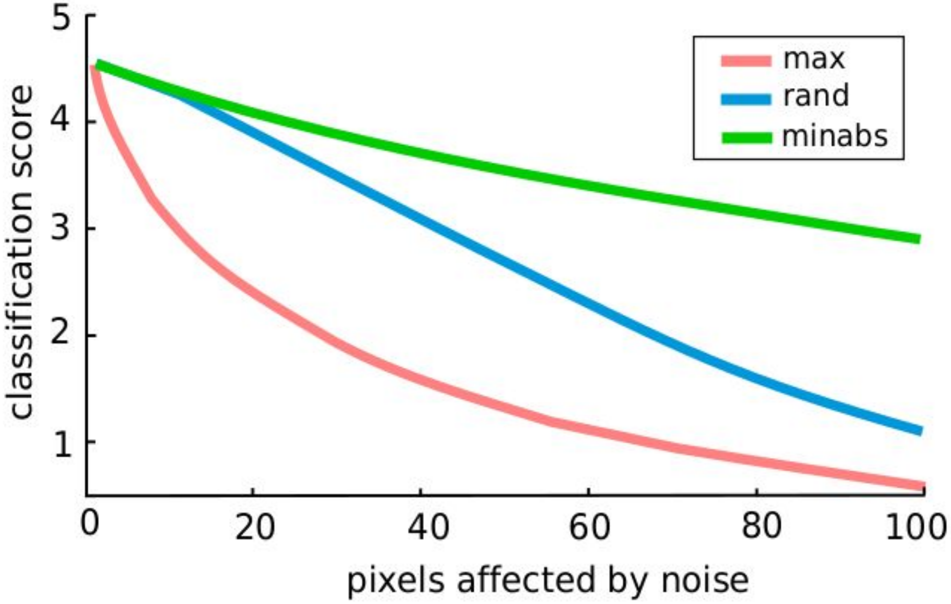
Qualitative Checks on Approximation:

1. The sign of relevance is preserved for x_k
2. For suppressing neurons, their relevance can be flipped with the $\mathbf{x_j^2}$ term
3. In the limit of constant normalization, the identity is recovered

Experiment: MNIST



Experiment : Pixel Flipping



Observation

- Randomly assigning most relevant pixels leads to highest rate of decay
- Randomly assigning least relevant pixels lead to lowest rate of decay

Conclusion: Meaningful pixel-wise decomposition

Experiment : Non Linear vs Linear Neurons

rule for basic layers	rule for normalization layers	AUC score
eq. 4,5, $\epsilon = 0.01$	identity	37.10
eq. 4,5, $\epsilon = 0.01$	first-order Taylor	35.47
eq. 4,6, $\beta = 1$	identity	56.13
eq. 4,6, $\beta = 1$	first-order Taylor	53.82

$$R_{i \leftarrow j}^{(l,l+1)} = \frac{z_{ij}}{z_j + \epsilon \cdot \mathrm{sign}(z_j)} R_j^{(l+1)}$$

$$R_{i \leftarrow j}^{(l,l+1)} = \left((1+\beta)\frac{z_{ij}^{+}}{z_j^{+}} - \beta \frac{z_{ij}^{-}}{z_j^{-}}\right) R_j^{(l+1)}$$

Observations

- Lower AUC represents lower accuracy after perturbing highest relevant pixels first implying lower AUC is better
- In both realizations of messages, the non linear renormalization using first order Taylor series performs better

Experiment : Taylor vs No Taylor

dataset	methods	$\Delta_{\epsilon=1}^{\epsilon=0.01}$	$\Delta_{\epsilon=0.01}^{\epsilon=100}$	$\Delta_{\epsilon=1}^{\beta=1}$	$\Delta_{\beta=1}^{\beta=0}$
Imagenet	identity	-21.29	2.75	-42.61	-49.07
	Taylor	-12.29	-41.75	-34.44	-50.76
MIT Places	identity	-20.19	12.91	-14.55	-49.37
	Taylor	-11.65	-22.55	-8.82	-48.7

dataset	methods	$\epsilon = 1$	$\epsilon = 0.01$	$\epsilon = 100$	$\beta = 1$	$\beta = 0$
Imagenet	$\mathrm{AUC}_{\mathrm{Taylor}} - \mathrm{AUC}_{\mathrm{identity}}$	-35.84	-26.84	8.47	0.29	1.98
MIT Places	$\mathrm{AUC}_{\mathrm{Taylor}} - \mathrm{AUC}_{\mathrm{identity}}$	-33.13	-24.59	5.34	-0.39	-1.06

Observations

- Taylor approximations for local renormalization have the best results searching over a range of parameters for both Imagenet and MIT Places

Resources

1. Initial Paper: https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0130140
2. Deep Taylor Decomposition: http://www.heatmapping.org/deeptaylor/
3. http://www.heatmapping.org/ (Knowledge Hub LRP)
4. Demo: https://lrpserver.hhi.fraunhofer.de/image-classification