Tutorials

Interpretable Deep Learning: Towards Understanding & Explaining DNNs

Part 3: Validating Explanations

Wojciech Samek, Grégoire Montavon, Klaus-Robert Müller

From ML Successes to Applications

Deep Net outperforms humans in image classification

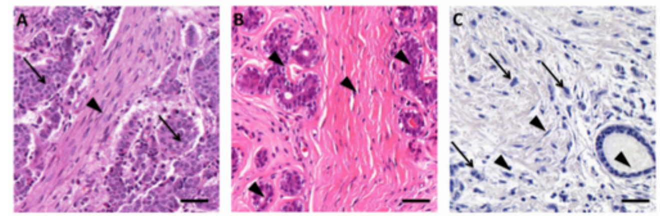
IMAGENET

AlphaGo beats Go human champ

Visual Reasoning

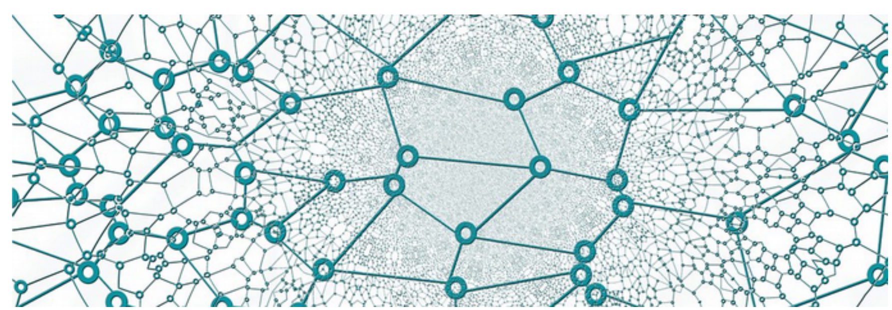
What size is the cylinder that is left of the brown metal thing that is left of the big sphere?

Medical Diagnosis

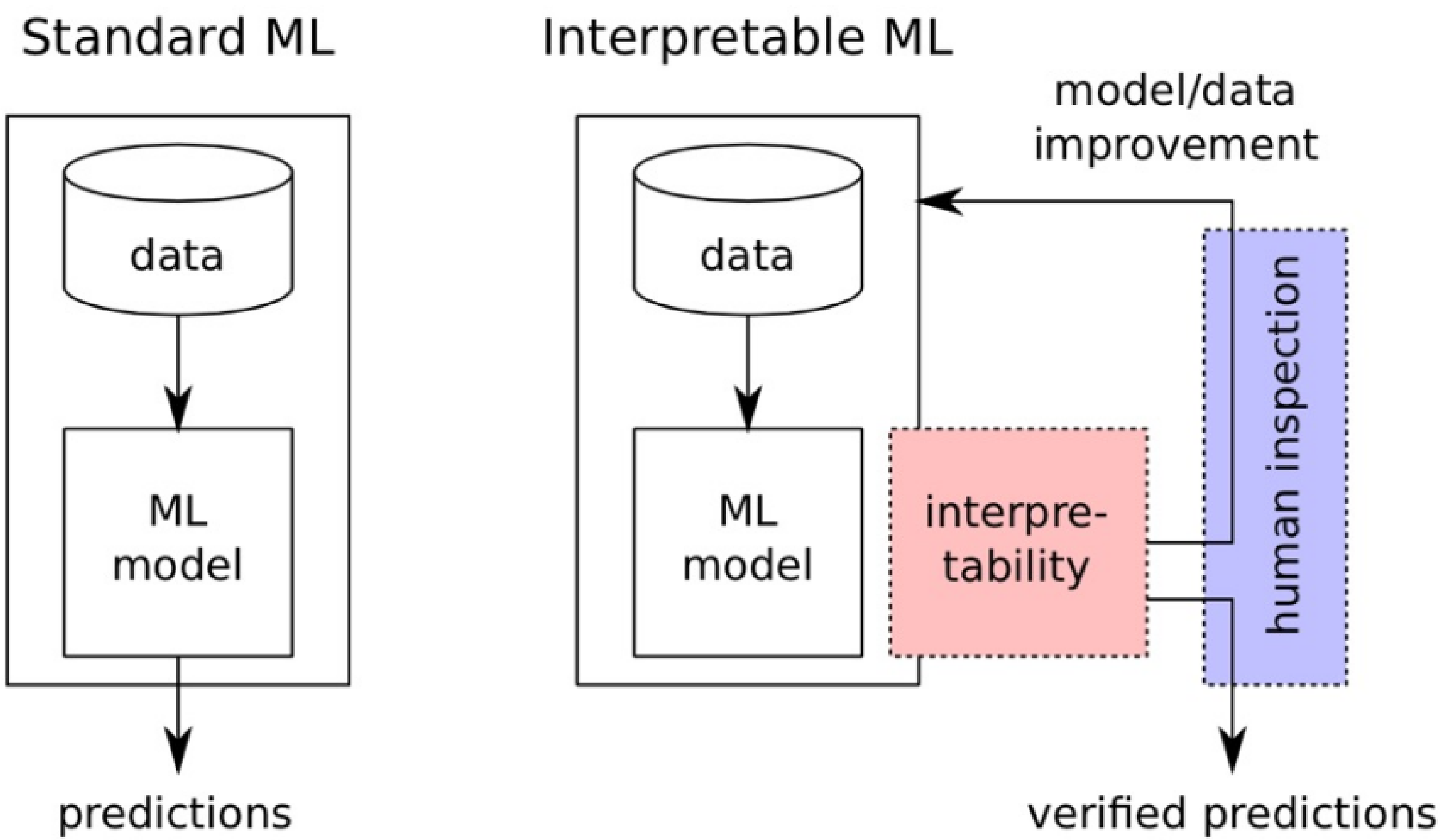


Autonomous Driving

Networks (smart grids, etc.)



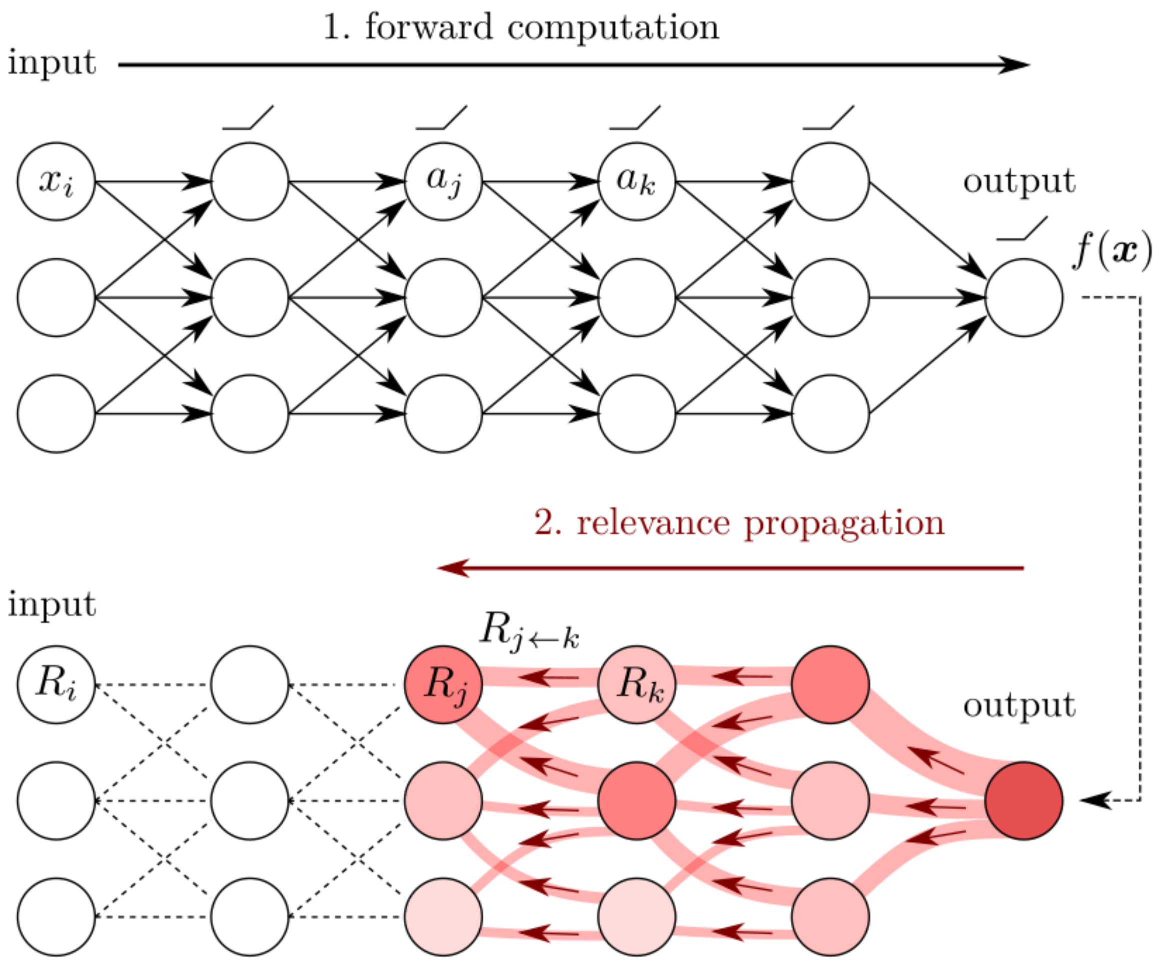
Making ML Models Interpretable

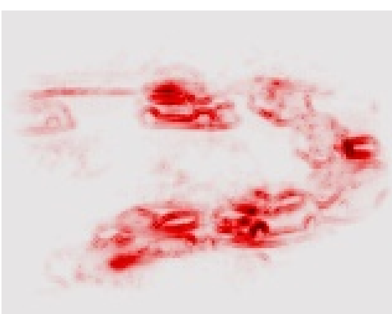


Generalization error

Generalization error + human experience

Layer-Wise Relevance Propagation (LRP) [Bach'15]

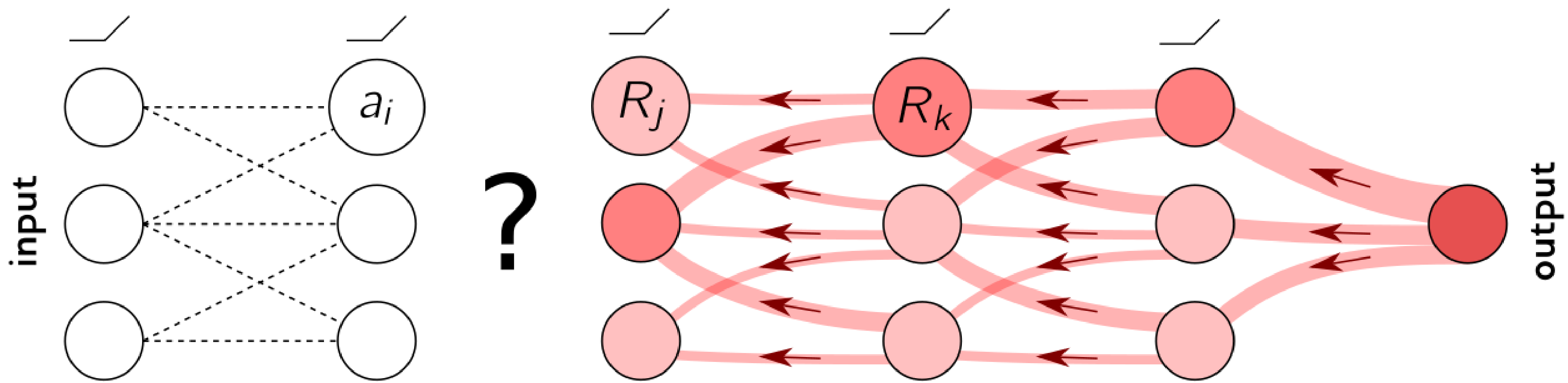




$$R_j = \sum_k \frac{a_j w_{jk}^+}{\sum_j a_j w_{jk}^+} R_k$$

Deep Taylor Decomposition [Montavon'17]

Question: Suppose that we have propagated the relevance until a given layer. How should it be propagated one layer further?



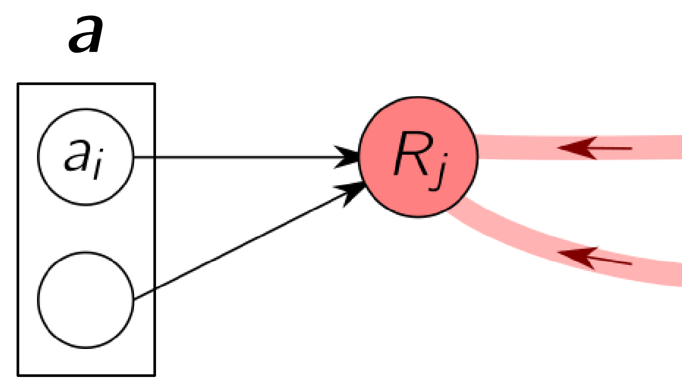
$$R_j = \sum_k \frac{a_j w_{jk}^+}{\sum_j a_j w_{jk}^+} R_k$$

Idea: By performing a Taylor expansion of the relevance.

Deep Taylor Decomposition

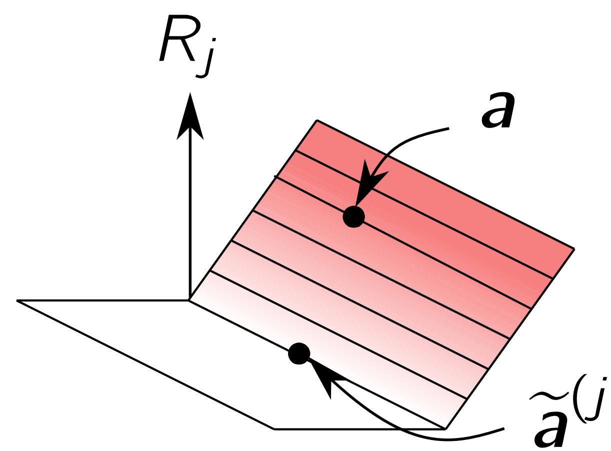
Relevance neuron:

$$R_j(\boldsymbol{a}) = \max(0, \textstyle\sum_i a_i w_{ij} + b_j) \cdot c_j$$



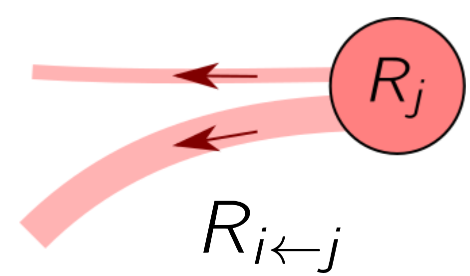
Taylor expansion:

$$R_j(\boldsymbol{a}) = \sum_i \underbrace{\frac{\partial R_j}{\partial a_i}\bigg|_{\widetilde{\boldsymbol{a}}^{(j)}} \cdot (a_i - \widetilde{a}_i^{(j)})}$$



Redistribution:

$$R_{i \leftarrow j} = \frac{(a_i - \widetilde{a}_i^{(j)})w_{ij}}{\sum_i (a_i - \widetilde{a}_i^{(j)})w_{ij}} R_j$$



Revisiting the DTD Root Point

$$R_{i\leftarrow j} = \frac{(a_i - \widetilde{a}_i^{(j)})w_{ij}}{\sum_i (a_i - \widetilde{a}_i^{(j)})w_{ij}} R_j$$

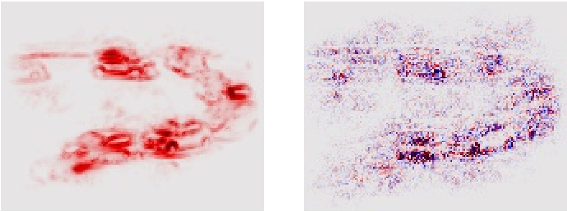
(Deep Taylor generic)

Choice of root point		$\widetilde{\boldsymbol{a}}^{(j)} \in \mathcal{D}$
1. nearest root	$\widetilde{\boldsymbol{a}}^{(j)} = \boldsymbol{a} - t \cdot \boldsymbol{w}_j$	
2. rescaled excitations	$\widetilde{\boldsymbol{a}}^{(j)} = \boldsymbol{a} - t \cdot \boldsymbol{a} \odot \mathbf{1}_{\boldsymbol{w}_j \succ 0}$	✔
3. generalized	$\widetilde{\boldsymbol{a}}^{(j)} = \boldsymbol{a} - t \cdot \boldsymbol{a} \odot (\mathbf{1} - \gamma)_{\boldsymbol{w}_j \succ 0}$	✔

Generalized rule
$(0 \leq \gamma \leq 1)$

$$R_{i\leftarrow j} = \frac{a_i(w_{ij}^+ + \gamma w_{ij}^-)}{\sum_i a_i(w_{ij}^+ + \gamma w_{ij}^-)} R_j$$

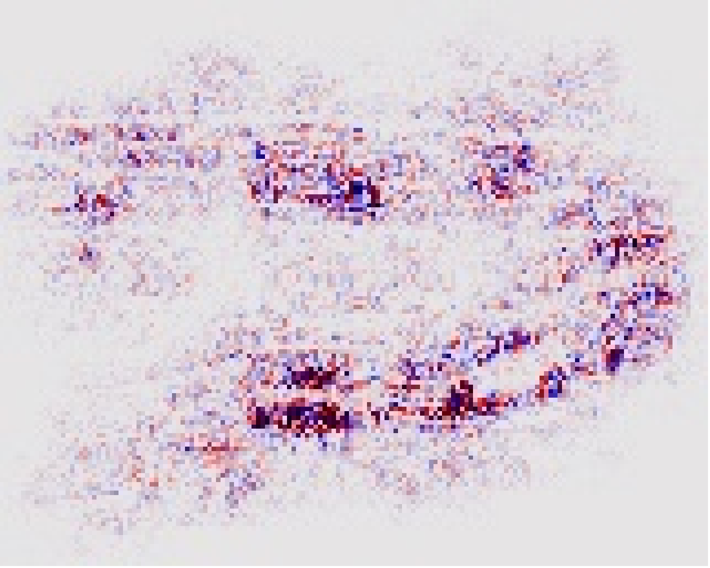
$\gamma = 0$ $\gamma = 1$



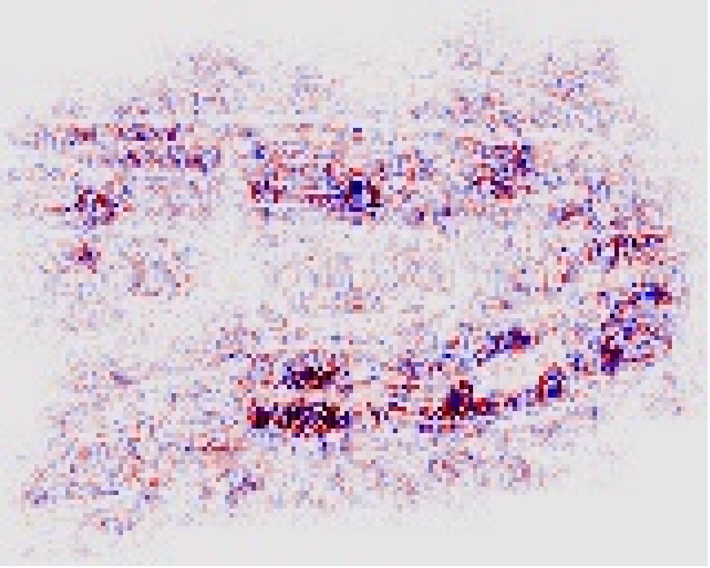
The Special Case "γ = 1.0"

Find the difference...

$\gamma = 1.0$



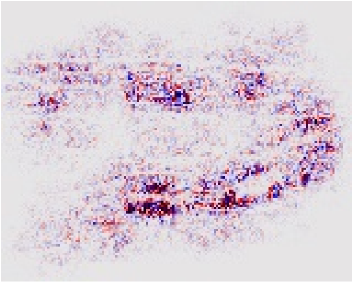
gradient $\times$ input



Question: Is there a connection between the two methods?

The Special Case "$\gamma = 1.0$"

$$R_i = \sum_j \frac{a_i(w_{ij}^+ + \gamma w_{ij}^-)}{\sum_i a_i(w_{ij}^+ + \gamma w_{ij}^-)} R_j$$



$\gamma = 1.0$

$$R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$$

which can also be rewritten as:

$$a_i \delta_i = a_i \sum_j w_{ij} \frac{(\sum_i a_i w_{ij} + b_j)^+}{\sum_i a_i w_{ij}} \delta_j$$

For networks with bias zero, the procedure becomes equivalent to grad x input [see also Shrikumar'17]

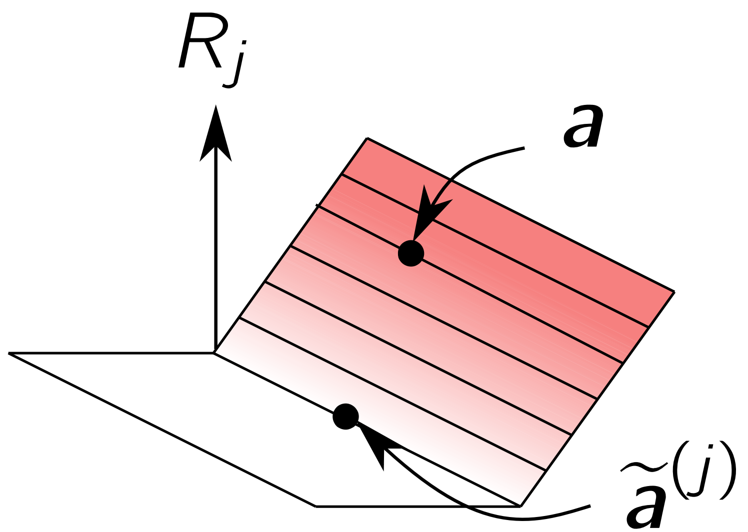
[Shrikumar'17] Not Just a Black Box: Learning Important Features Through Propagating Activation Differences, ArXiv

(LRP-$\alpha_1\beta_0$)
$\gamma = 0.0$ $\longleftrightarrow$ $\gamma = 1.0$
(grad $\times$ input)

Question: How to select the optimal parameter "γ" ?

Explanation Selection

$$\widetilde{\boldsymbol{a}}^{(j)} = \boldsymbol{a} - t \cdot \boldsymbol{a} \odot (\mathbf{1} - \gamma)_{\boldsymbol{w}_j \succ 0}$$



Idea: Choose γ such that:

$$\| \boldsymbol{a} - \widetilde{\boldsymbol{a}}^{(j)} \| \text{ is small.}$$
$$\| \widetilde{\boldsymbol{a}}^{(j)} - \widetilde{\boldsymbol{a}}^{(j')} \| \text{ is small.}$$

Problem: How to weight these two objectives?

Explanation Selection

More direct approach: Try all parameters, and select the one producing the best explanations.

$$R_{i \leftarrow j} = \frac{a_i (w_{ij}^{+} + \gamma w_{ij}^{-})}{\sum_i a_i (w_{ij}^{+} + \gamma w_{ij}^{-})} R_j$$

(LRP-$\alpha_1\beta_0$) $\gamma = 0.0$ | $\gamma = 0.3$ | $\gamma = 0.6$ | $\gamma = 0.9$ | (grad $\times$ input) $\gamma = 1.0$

Question: How to assess explanation quality?

Evaluating Explanations

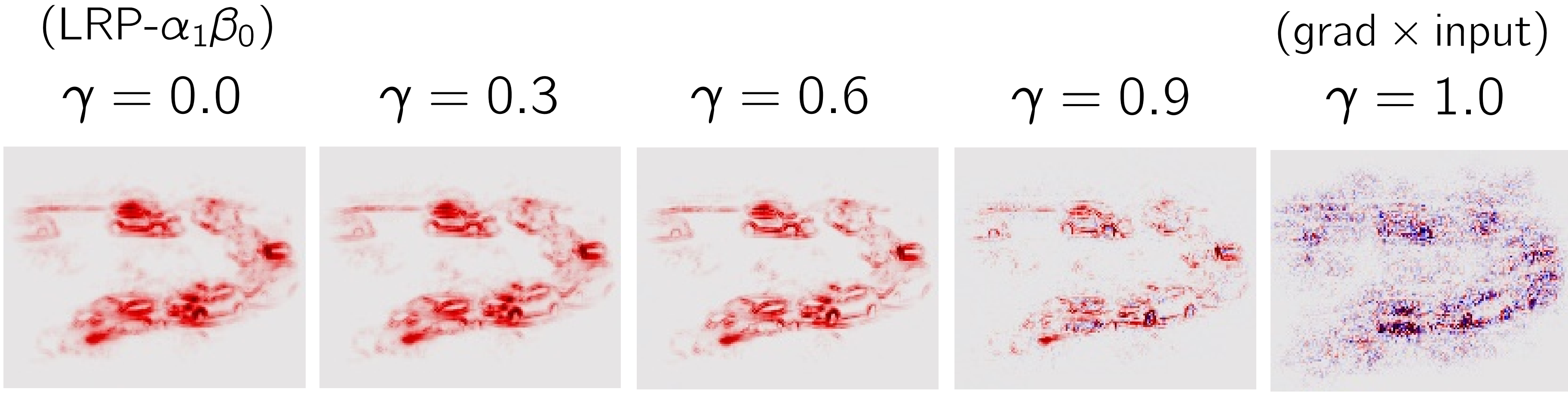
(LRP-$\alpha_1\beta_0$)				(grad $\times$ input)
$\gamma = 0.0$	$\gamma = 0.3$	$\gamma = 0.6$	$\gamma = 0.9$	$\gamma = 1.0$

Human assessment

- Aesthetic properties
- Usability of the explanation
  (e.g. to understand the classifier).

→ **Requires an experimental study.**

Evaluating Explanations



Idea: Testing if explanations satisfy certain axioms/properties.

Examples:

- Explanation must be self-consistent (e.g. *conservation of evidence*)
- Explanation must be consistent in input domain (e.g. *continuity*)
- Explanation must be consistent in the space of models (e.g. *implementation invariance*)

Example 1: Conservation

$$\left(\sum_i R_i = f(\boldsymbol{x}) \right) \wedge \left(\sum_i |R_i| < A \cdot |f(\boldsymbol{x})| \right)$$

Simple example:

$f(x_1, x_2) = 10$

Possible explanations:

$(R_1, R_2) = (1, 2)$ ✗

$(R_1, R_2) = (3, 7)$ ✓

$(R_1, R_2) = (-995, 1005)$ ✗

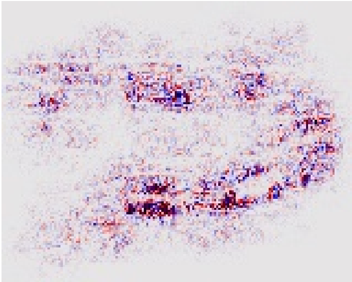
$(R_1, R_2) = (-1, 11)$ ✓

Example 1: Conservation

$$\left(\sum_i R_i = f(\boldsymbol{x})\right) \wedge \left(\sum_i |R_i| < A \cdot |f(\boldsymbol{x})|\right)$$

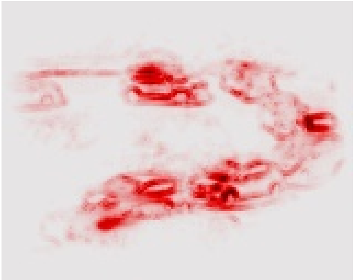
$\gamma = 1.0$

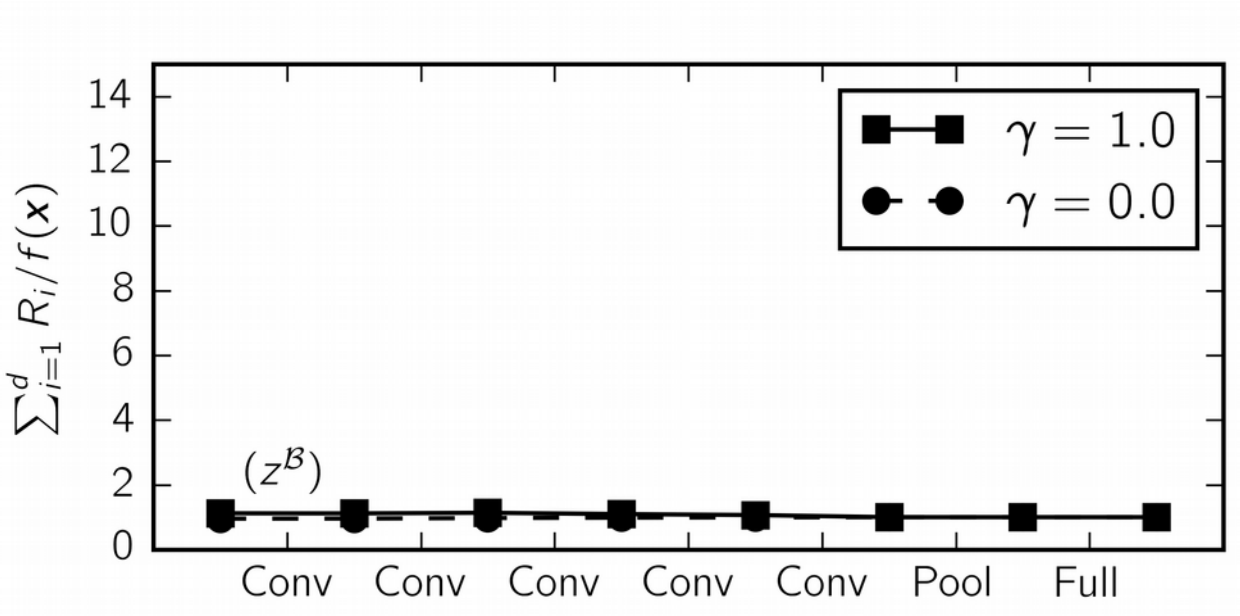
(grad $\times$ input)



$\gamma = 0.0$

(LRP-$\alpha_1\beta_0$)

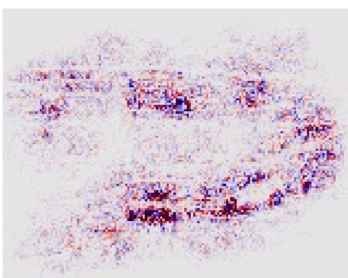




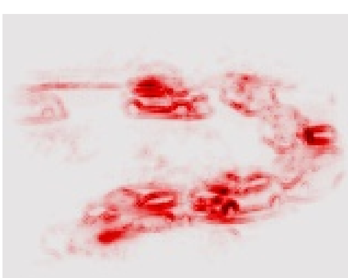
Example 1: Conservation

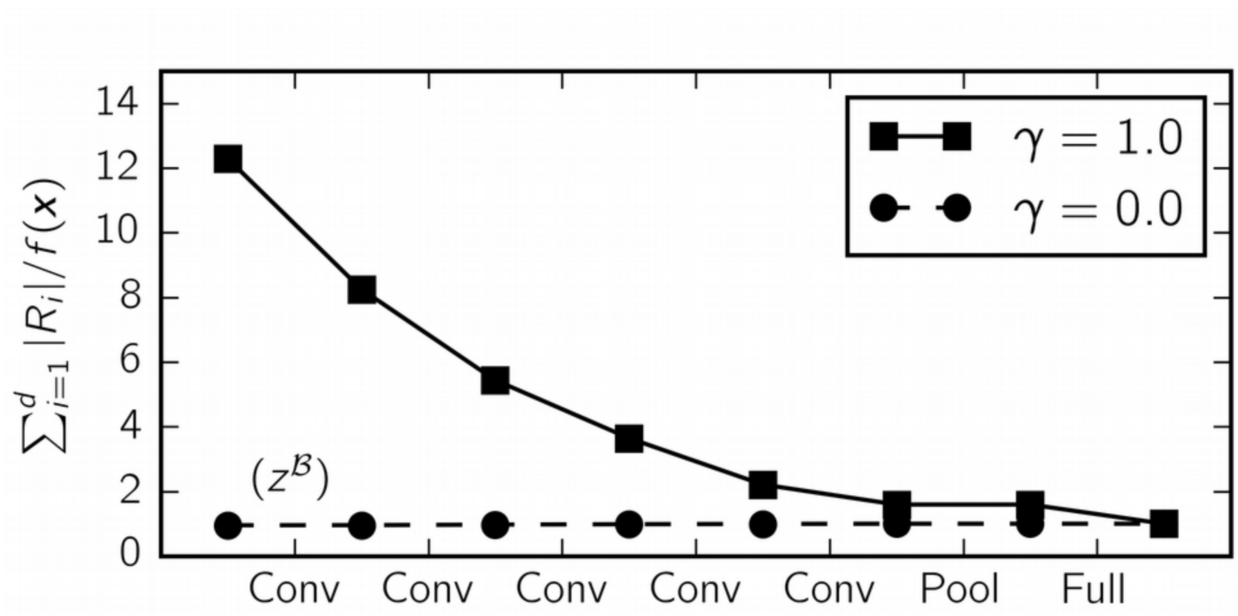
$$\left(\sum_i R_i = f(\boldsymbol{x})\right) \wedge \left(\sum_i |R_i| < A \cdot |f(\boldsymbol{x})|\right)$$

$\gamma = 1.0$
(grad $\times$ input)



$\gamma = 0.0$
(LRP-$\alpha_1\beta_0$)





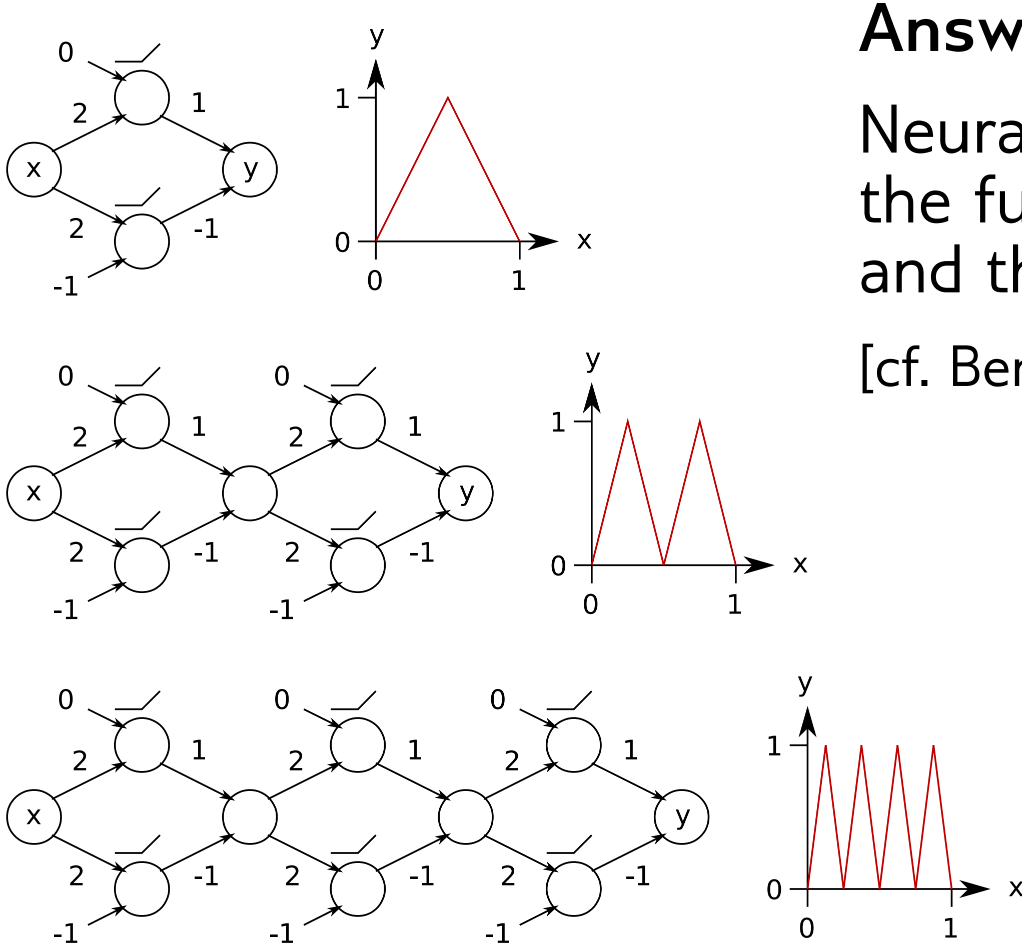
Why Grad x Input Scores Explode?

Conservation: $\left(\sum_i R_i = f(\boldsymbol{x})\right) \wedge \left(\sum_i |R_i| < A \cdot |f(\boldsymbol{x})|\right)$

depth 1

depth 2

depth 3



Answer:

Neural network depth causes the function to become steep and the gradient very large.

[cf. Bengio'94, Montufar'14]

[Bengio'94] Learning long-term dependencies with gradient descent is difficult. IEEE Trans. Neural Networks

[Montufar'14] On the Number of Linear Regions of DNNs. NIPS 2014.

Why Grad x Input Scores Explode?

Conservation: $\left(\sum_i R_i = f(\boldsymbol{x})\right) \wedge \left(\sum_i |R_i| < A \cdot |f(\boldsymbol{x})|\right)$

This can also be seen from the formulas:

$\gamma = 0.0$ (LRP-$\alpha_1\beta_0$)

$$R_i = \sum_j \frac{a_i w_{ij}^+}{\sum_i a_i w_{ij}^+} R_j$$

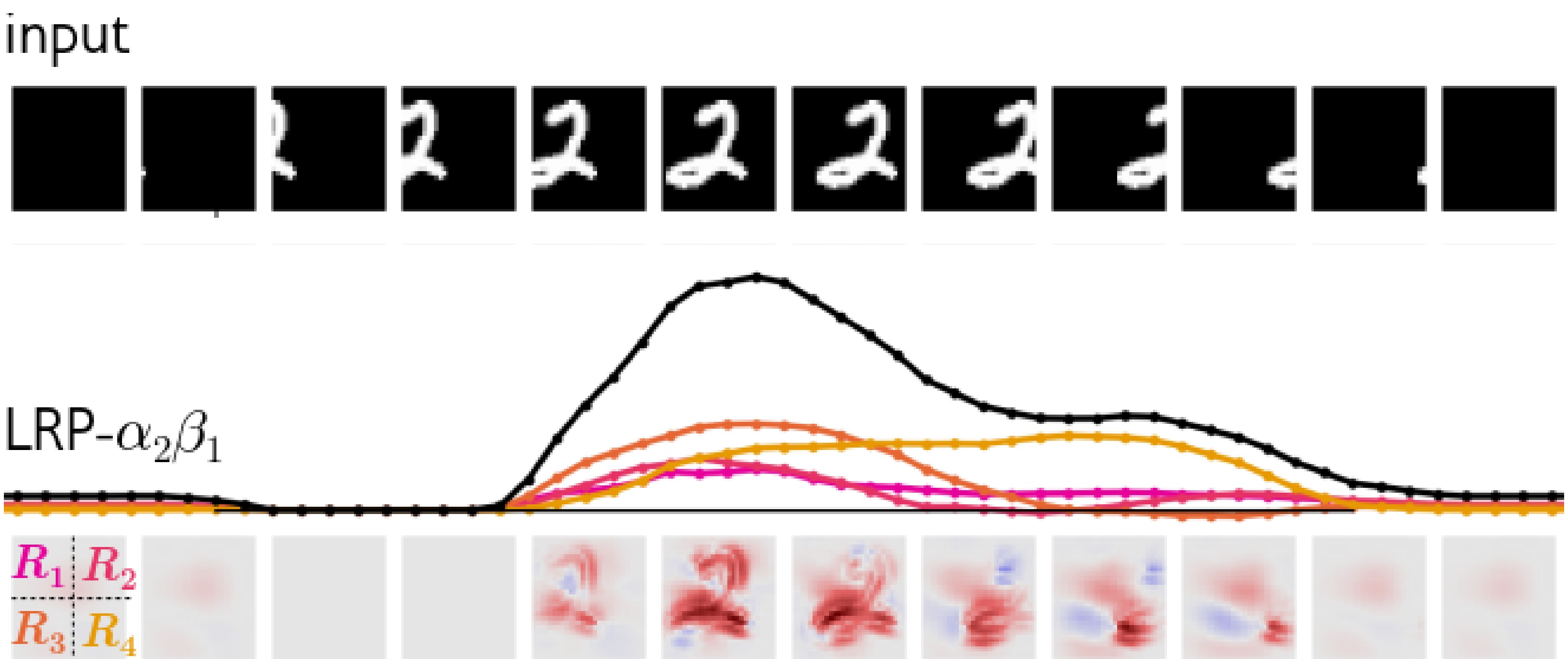
$\gamma = 1.0$ (grad $\times$ input)

$$R_i = \sum_j \frac{a_i w_{ij}}{\sum_i a_i w_{ij}} R_j$$

division by zero

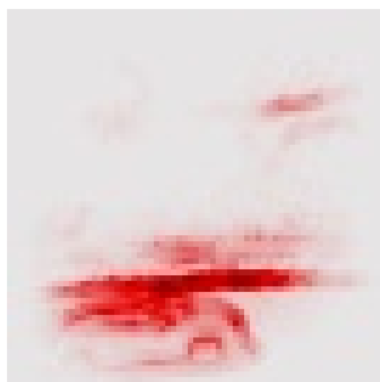
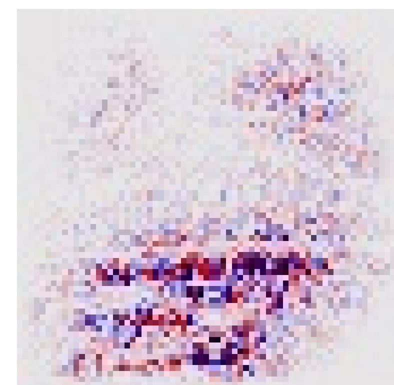
Example 2: Continuity [Montavon'18]

$$\left(\boldsymbol{x} \approx \boldsymbol{x}'\right) \wedge \left(f(\boldsymbol{x}) \approx f(\boldsymbol{x}')\right) \Rightarrow \boldsymbol{R}(\boldsymbol{x}) \approx \boldsymbol{R}(\boldsymbol{x}')$$



Explanation scores must be continuous in input domain.

Continuity Demo

video input	(LRP-$\alpha_1\beta_0$) $\gamma = 0.0$	$\gamma = 0.7$	(grad $\times$ input) $\gamma = 1.0$
			
	✓	✓	✗

Animations available at:
http://www.heatmapping.org/evaluating

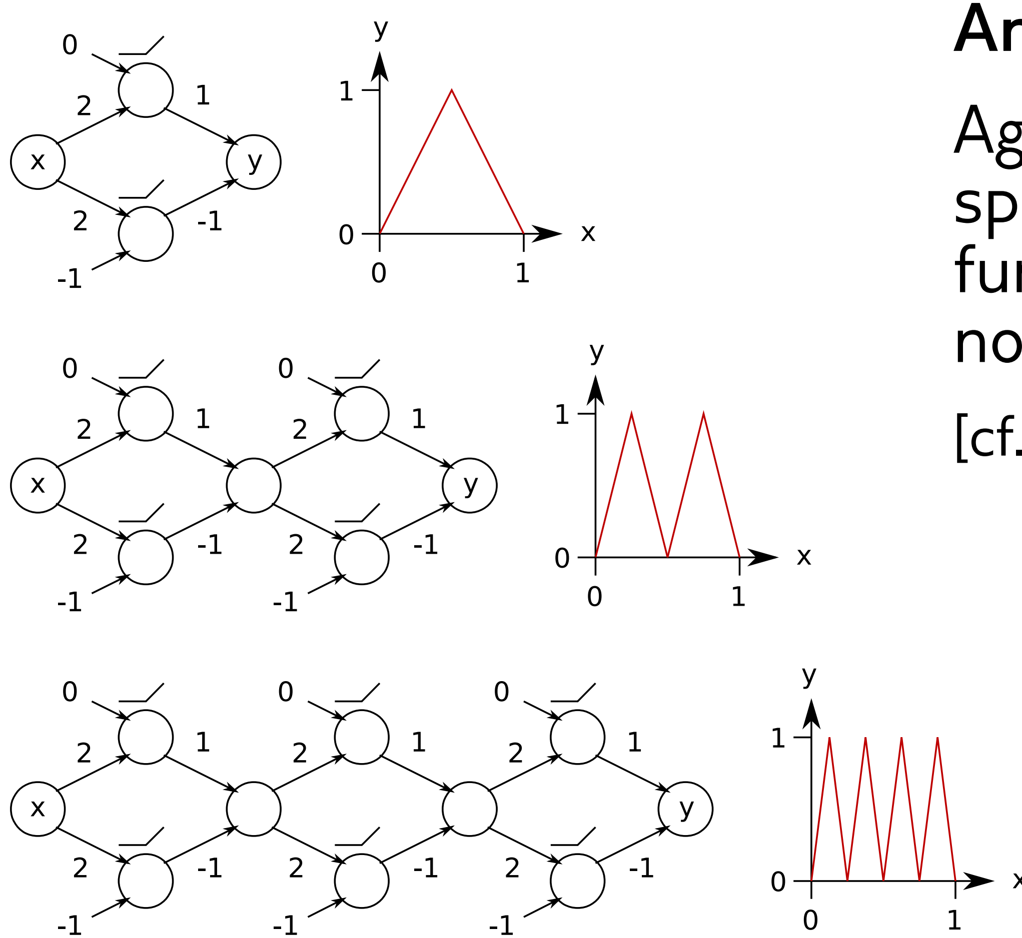
Why is Grad x Input Discontinuous ?

Continuity: $\big(\boldsymbol{x} \approx \boldsymbol{x}'\big) \wedge \big(f(\boldsymbol{x}) \approx f(\boldsymbol{x}')\big) \Rightarrow \boldsymbol{R}(\boldsymbol{x}) \approx \boldsymbol{R}(\boldsymbol{x}')$

depth 1

depth 2

depth 3



Answer:

Again, because of depth, specifically, because the function becomes highly non-smooth.

[cf. Montufar'14, Balduzzi'17]

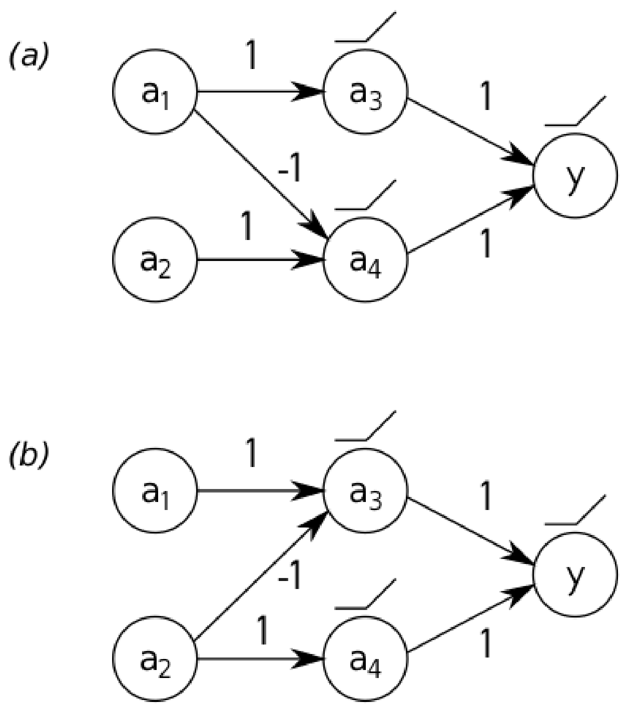
[Montufar'14] On the Number of Linear Regions of DNNs. NIPS 2014.

[Balduzzi'17] The Shattered Gradients Problem: If resnets are the answer [...] ICML 2017

Example 3: Impl. Invariance [Sundararajan'17]

Implementation Invariance: $f_\theta = f_\Phi \Rightarrow \boldsymbol{R}(\boldsymbol{x}; f_\theta) = \boldsymbol{R}(\boldsymbol{x}; f_\phi)$

Example: two networks implementing the maximum function:



LRP-$\alpha_1\beta_0$

Counter-example for:

$$a_1 = a_2 + \varepsilon$$

Network (a):

a_1 receives all

Network (b):

a_2 receives all

✗

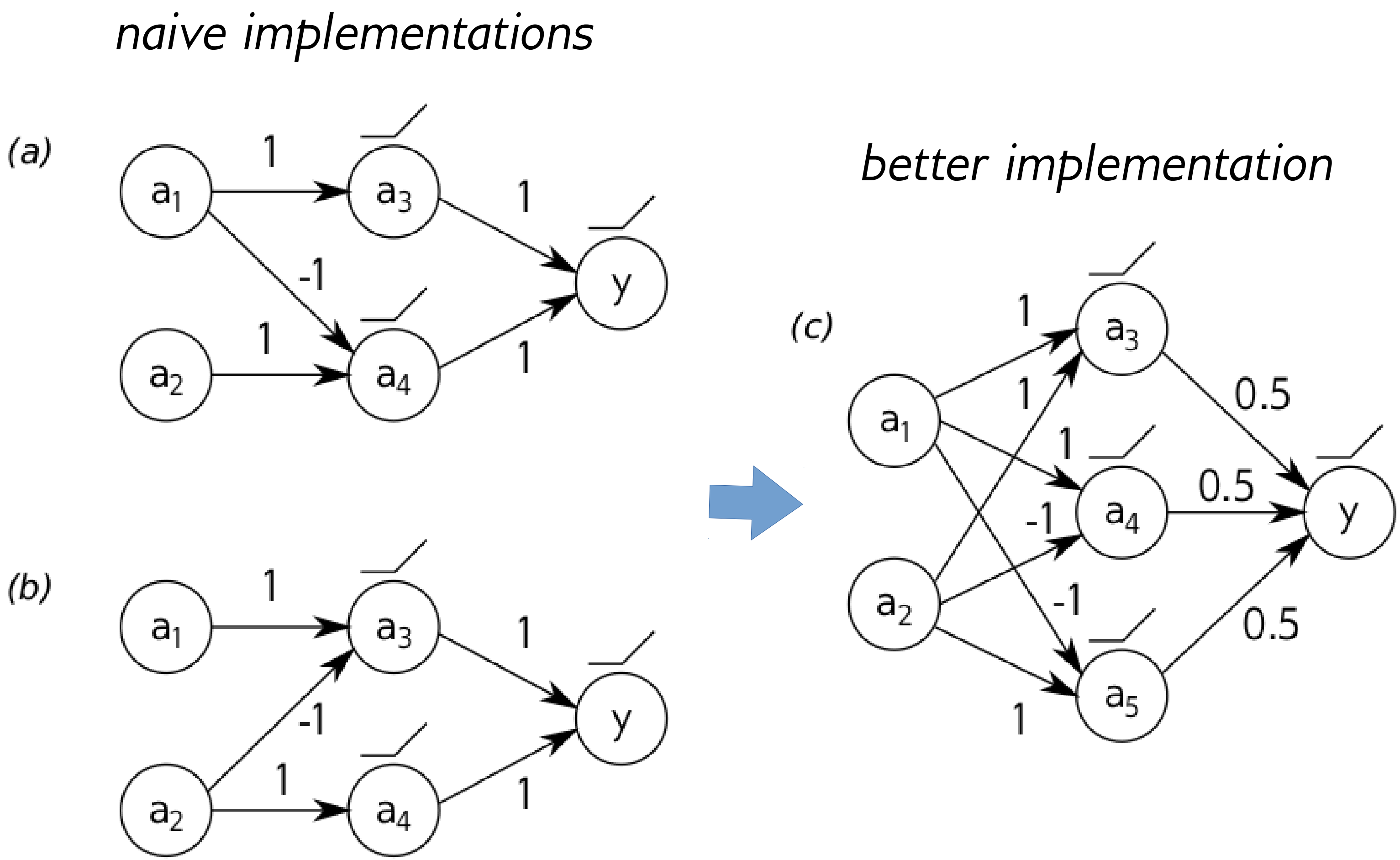
grad $\times$ input

Gradient is implementation invariant, therefore explanation too.

✓

[Sundararajan'17] M Sundararajan, A Taly, Q Yan: Axiomatic Attribution for Deep Networks. ICML 2017

Implementation Matters for LRP

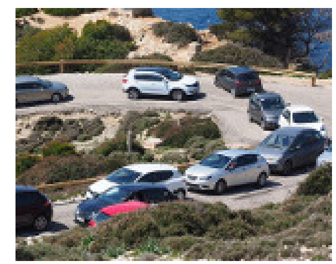


A Blind Spot in Explanation Selection

Consider the simple explanation technique:

$$R_i(\boldsymbol{x}) = \frac{1}{d} \cdot f(\boldsymbol{x})$$

redistributing uniformly on pixels.



It is:

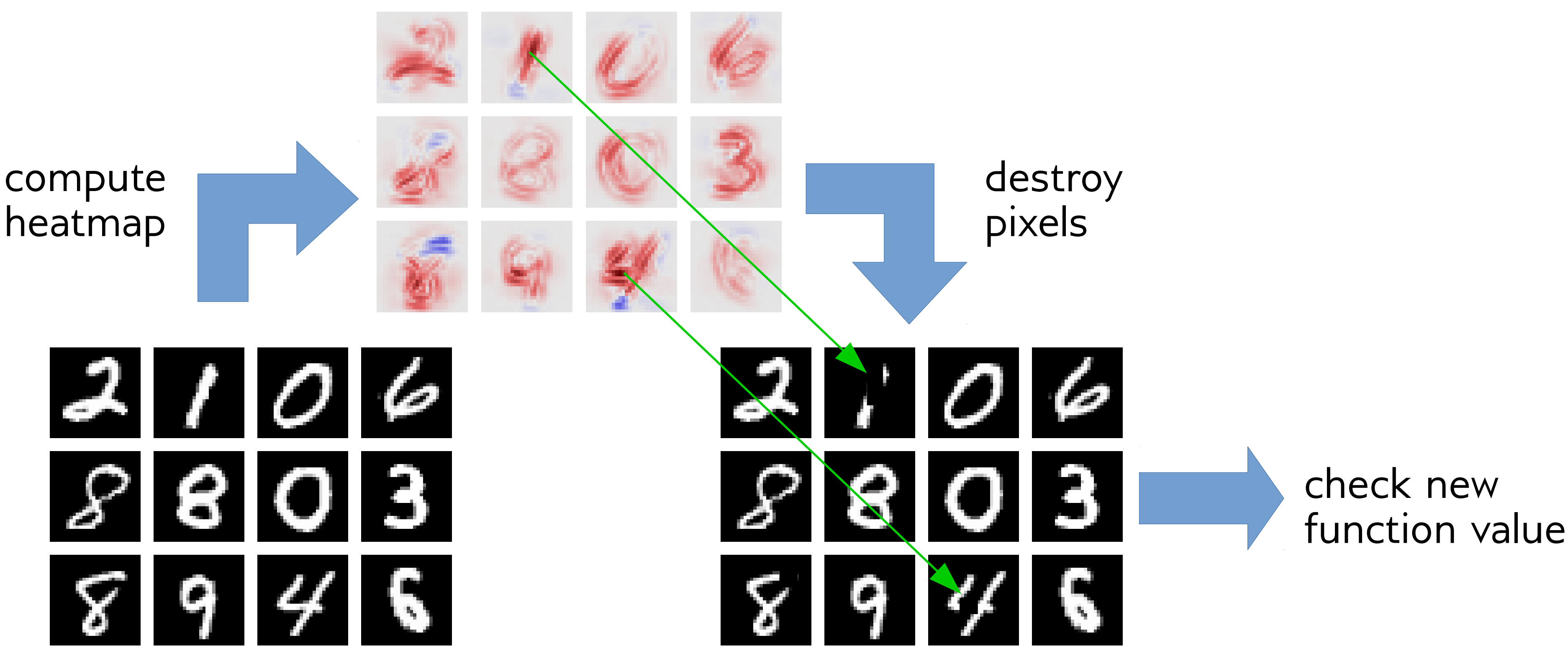
- conservative
- continuous
- implementation invariant

<u>but</u> it is also completely uninformative.

→ Need to verify *selectivity* (i.e. the explanation should discriminate between relevant and irrelevant variables.)

Pixel-Flipping [Bach'15, Samek'17]

Idea: Test that removing input variables with high assigned relevance makes the function value drop quickly.

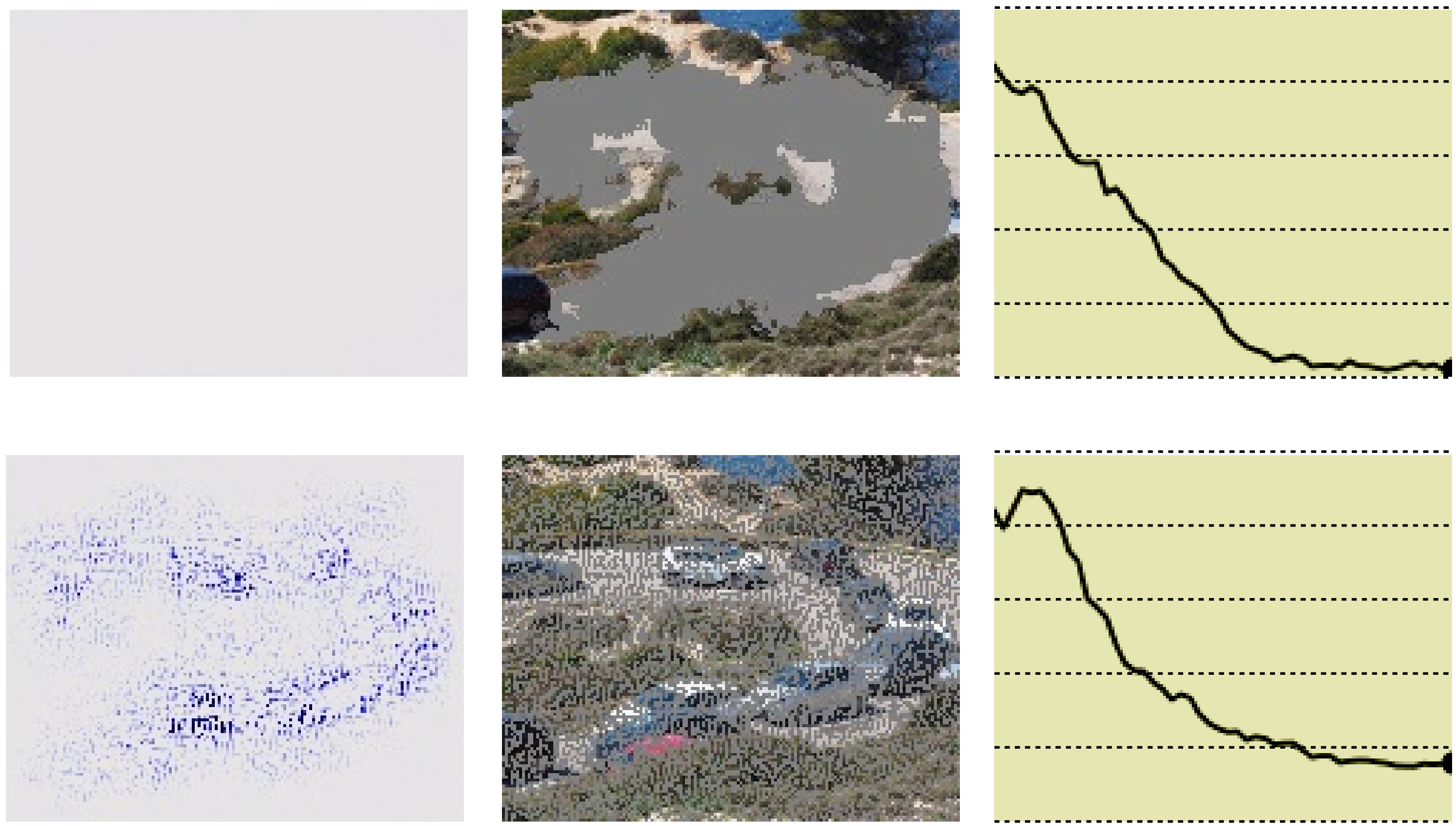


Pixel-Flipping Demo

explanation | input | $f(\boldsymbol{x})$

$\gamma = 0.0$
(LRP-$\alpha_1\beta_0$)

$\gamma = 1.0$
(grad $\times$ input)



Animations available at:
http://www.heatmapping.org/evaluating

Conclusion for Part 3

1. Most explanation methods have hyperparameters. As there is no ground-truth explanations available, standard model selection techniques do not apply.

2. The problem of explanation selection can be addressed axiomatically (e.g. conservation, continuity, implementation invariance).

3. Axioms may not suffice in selecting a good explanation. It is also important to design experiments that test the explanation against the model (e.g. pixel-flipping).