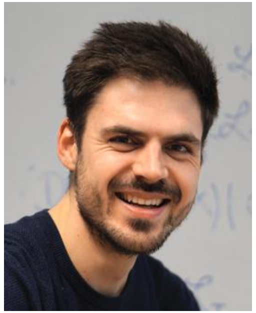
Freie Universität Berlin

When Explanations Lie:

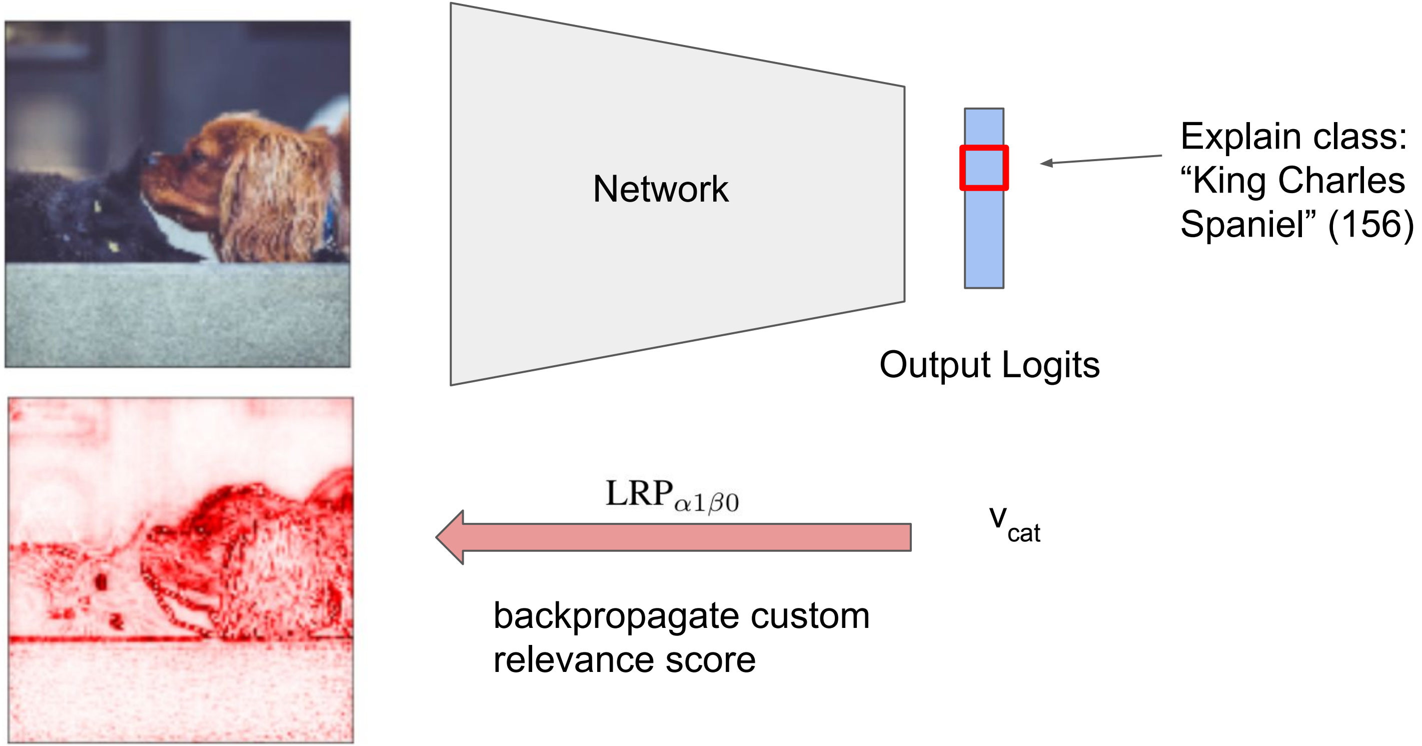
Why Many Modified BP Attributions Fail



Leon Sixt, Maximilian Granz, Tim Landgraf

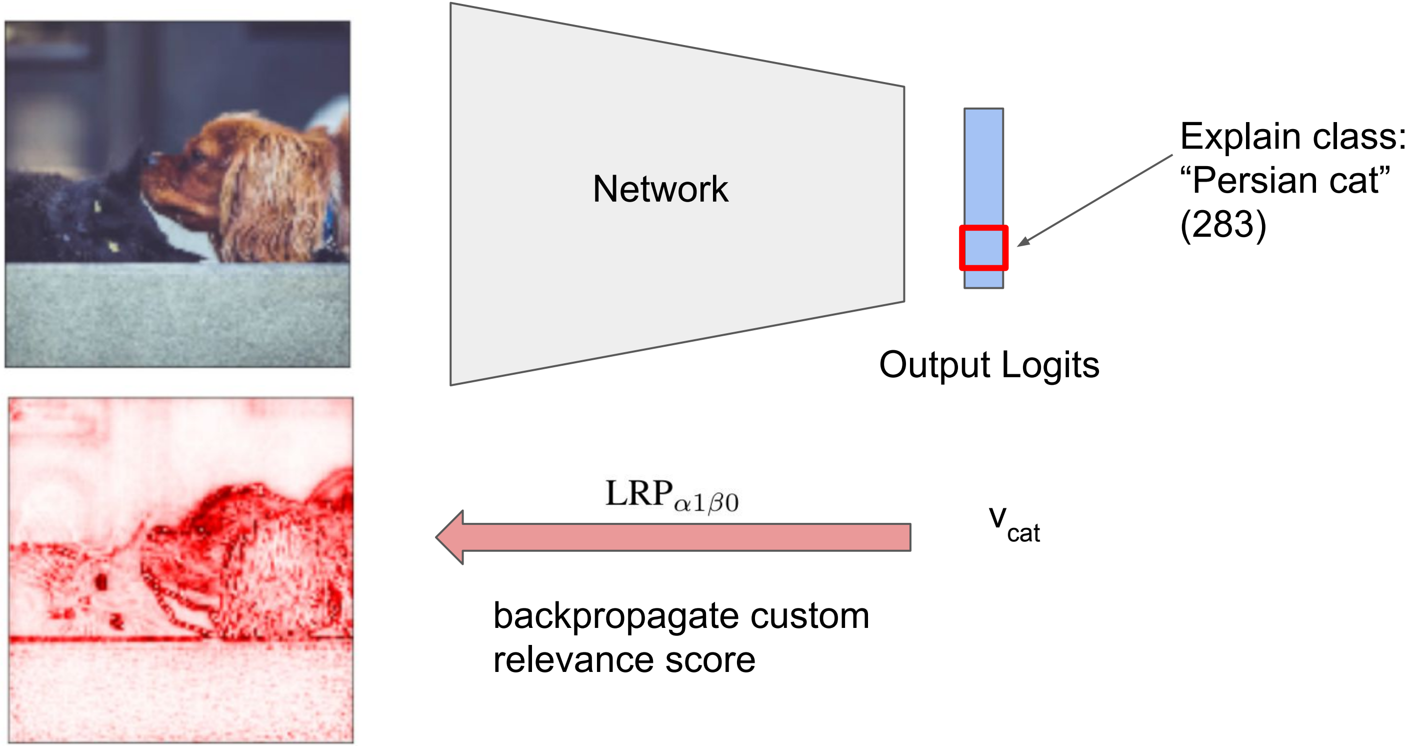
ICML
International Conference On Machine Learning

Attribution Method: $\mathrm{LRP}_{\alpha 1 \beta 0}$



Saliency map indicates ‘important’ areas

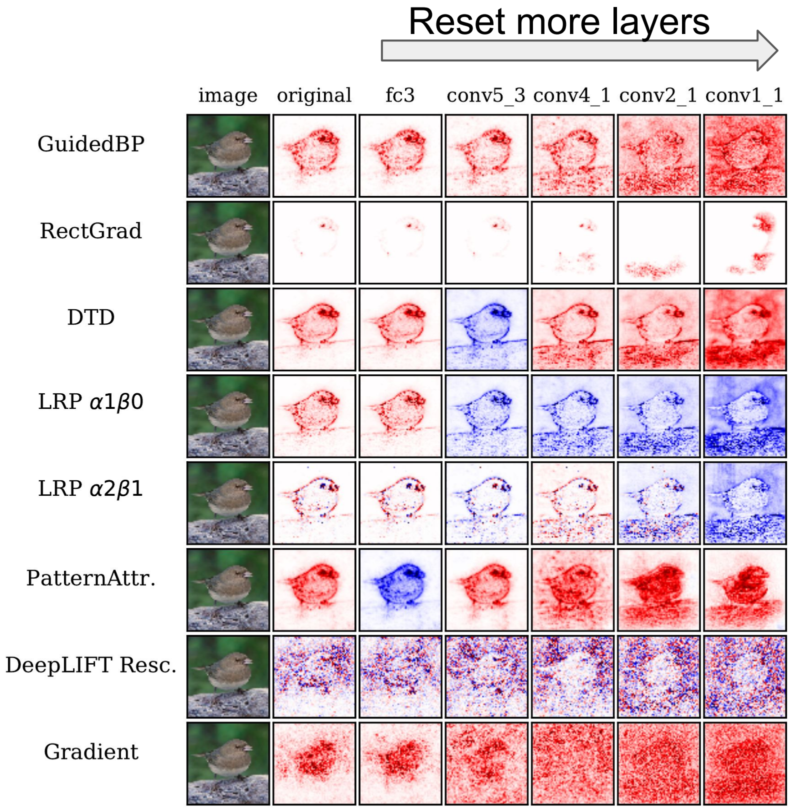
Attribution Method: $\mathrm{LRP}_{\alpha 1 \beta 0}$



Does the saliency map indicate ‘important’ areas?

Sanity Check (Adebayo et al., 2018)

- Reset network parameter to initialization
- Saliency maps should change!
- Many modified BP methods fail:
  - PatternAttribution *(Kindermans et al, 2017)*
  - Deep Taylor Decomposition *(Montavon et al., 2017)*
  - LRP-αβ *(Bach et al., 2015)*
  - RectGrad *(Kim et al., 2019)*
  - Deconv *(Zeiler & Fergus, 2014)*
  - ExcitationBP *(Zhang et al., 2018)*
  - GuidedBP* *(Springenberg et al., 2014)*



VGG-16

*already found by (Adebayo et al., 2018; Nie et al., 2018)

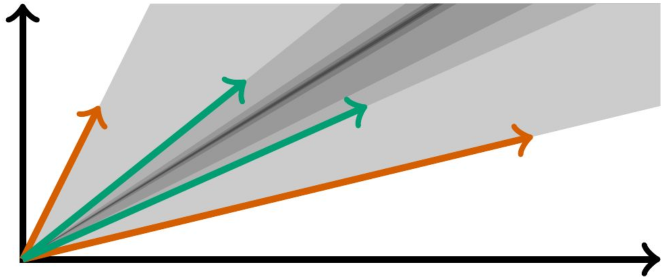
Short summary

Main Finding:

- Many modified BP methods ignore deeper layers!
- Important to know if you can trust the explanations!

In the talk:

- Intuition: Why later layers are ignored?
- Can we measure this behaviour?



z^+-Rule

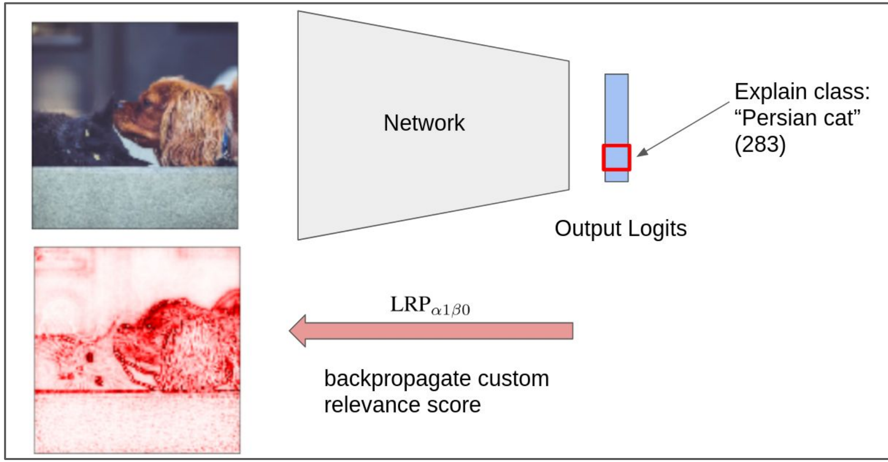
Backpropagates a custom relevance score.

Used by:

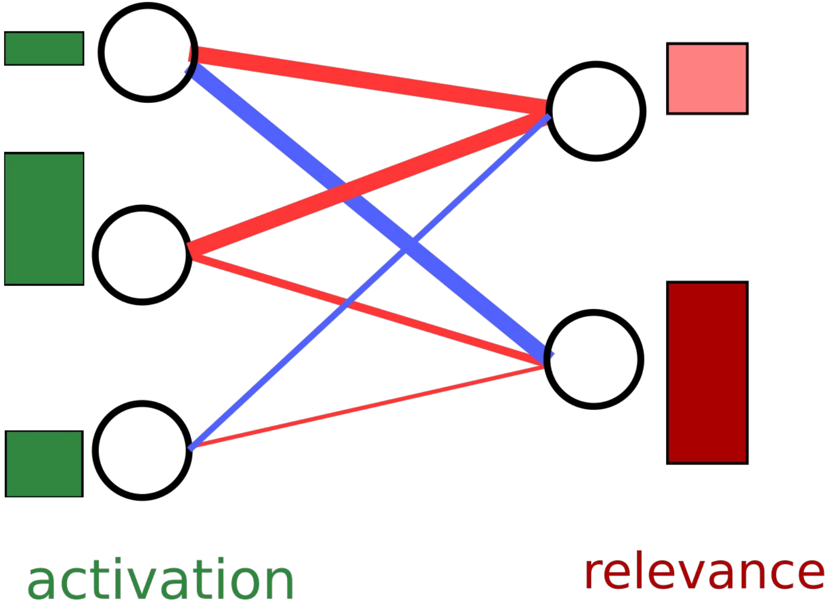
- Deep Taylor Decomposition
- LRP-α1β0
- ExcitationBP (equivalent to LRP-α1β0)

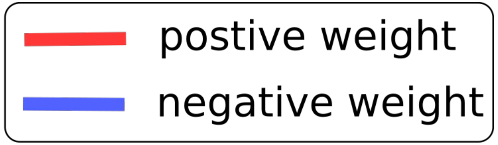
Next Steps:

1. How does the z^+-rule work for a layer?
2. What happens for multiple layers?



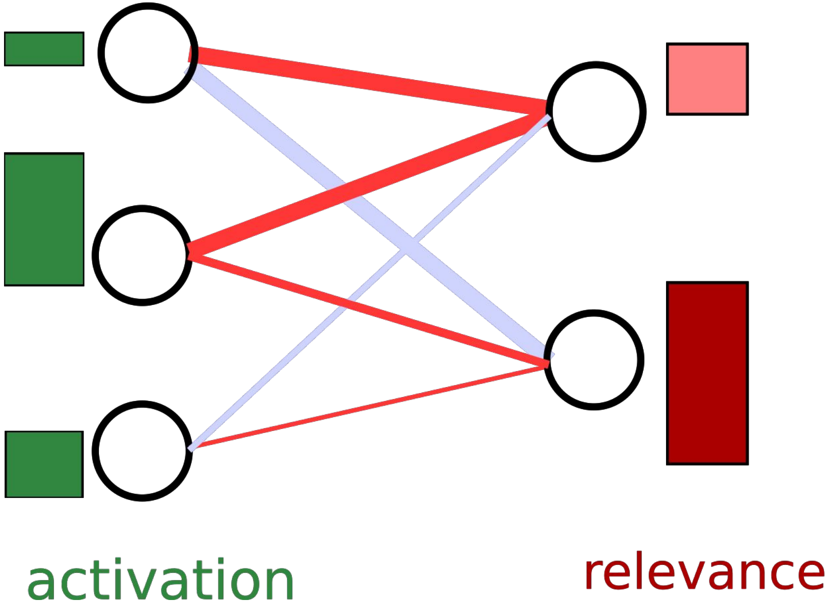
z^+-Rule: A single layer

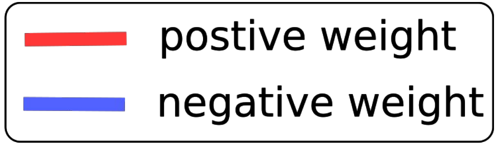




$$\frac{\text{positive weight} \times \text{activation}}{\text{normalization}}$$

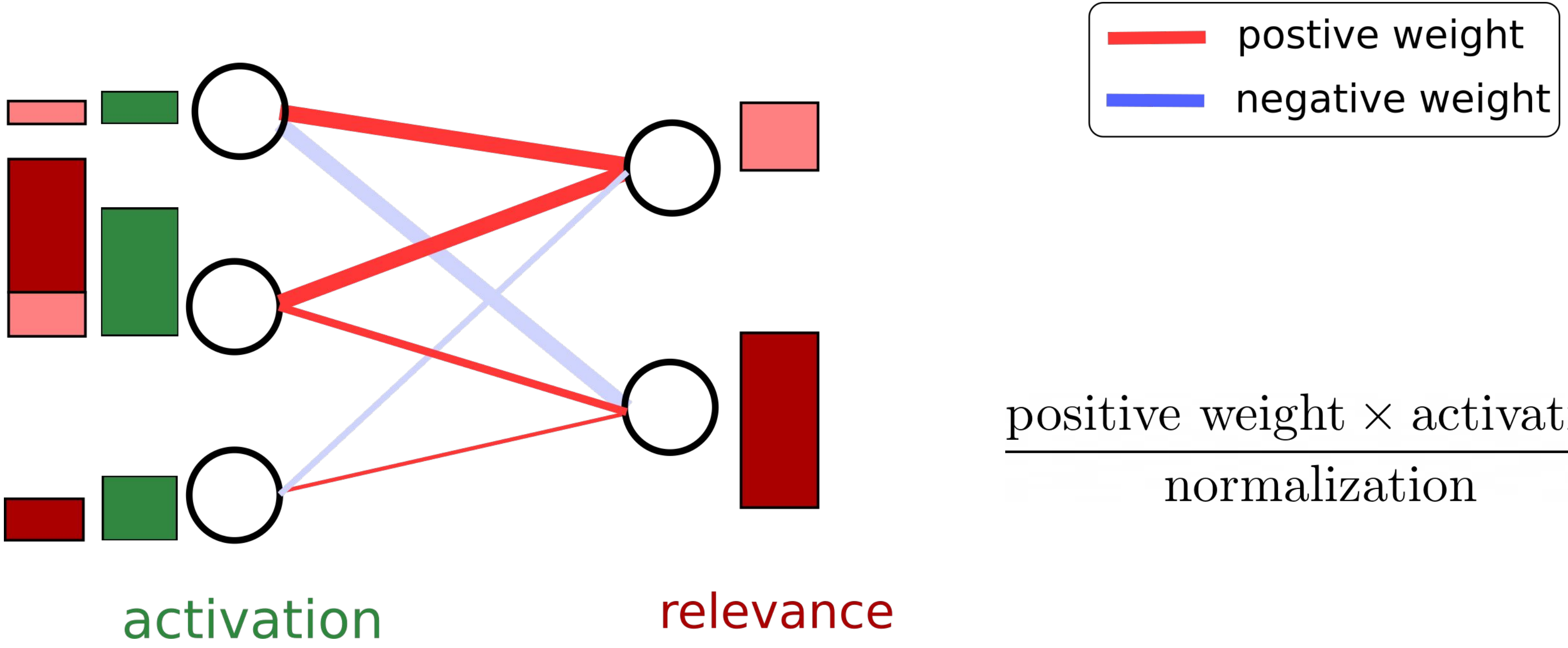
z^+-Rule: A single layer





$$\frac{\text{positive weight} \times \text{activation}}{\text{normalization}}$$

z^+-Rule: A single layer



$$\frac{\text{positive weight} \times \text{activation}}{\text{normalization}}$$

z^+-Rule: Matrix

$$\frac{\text{weight} \times \text{activation}}{\text{normalization}}$$

Weight strength

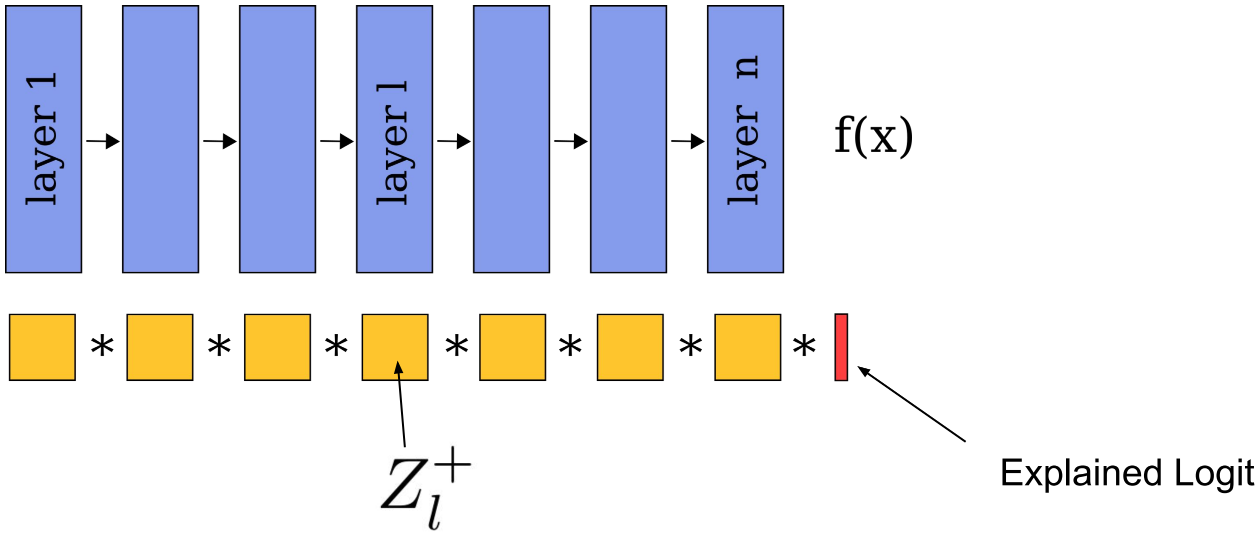
Activation at layer l

$$Z_l^{+^T} = \left(\frac{[w_{ij}\boldsymbol{h}_{l_{[j]}}]^+}{\sum_k [w_{ik}\boldsymbol{h}_{l_{[k]}}]^+} \right)_{[ij]}$$

Normalize! The sum of relevance should remains equal

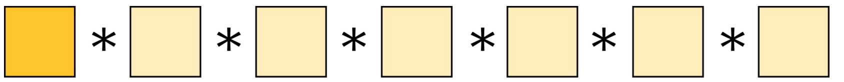
z^+-Rule: Matrix Chain

Per Layer, we obtain a Z_l^+ **matrix**

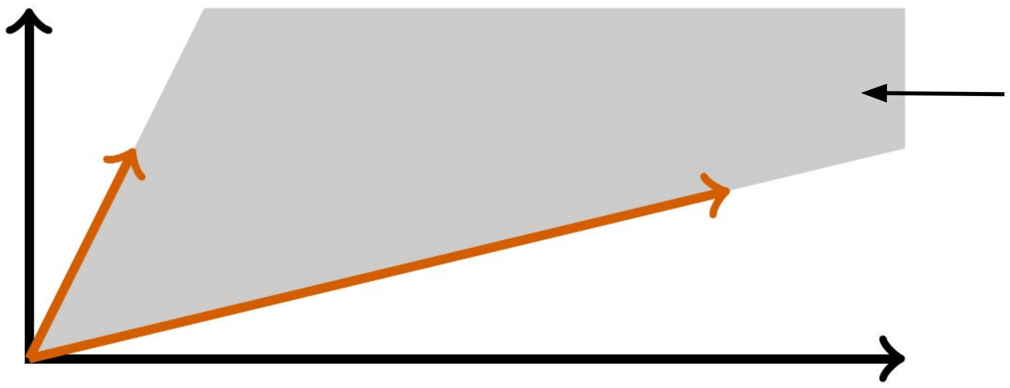


The matrix chain can be multiplied from left to right!

Geometric Intuition



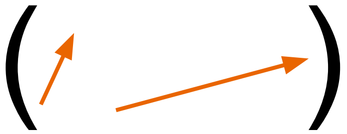
1st Layer



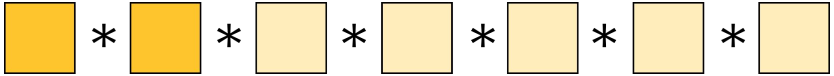
$$\lambda_1\mathbf{a}_1 + \lambda_2\mathbf{a}_2$$

$$\lambda_1, \lambda_2 \geq 0$$

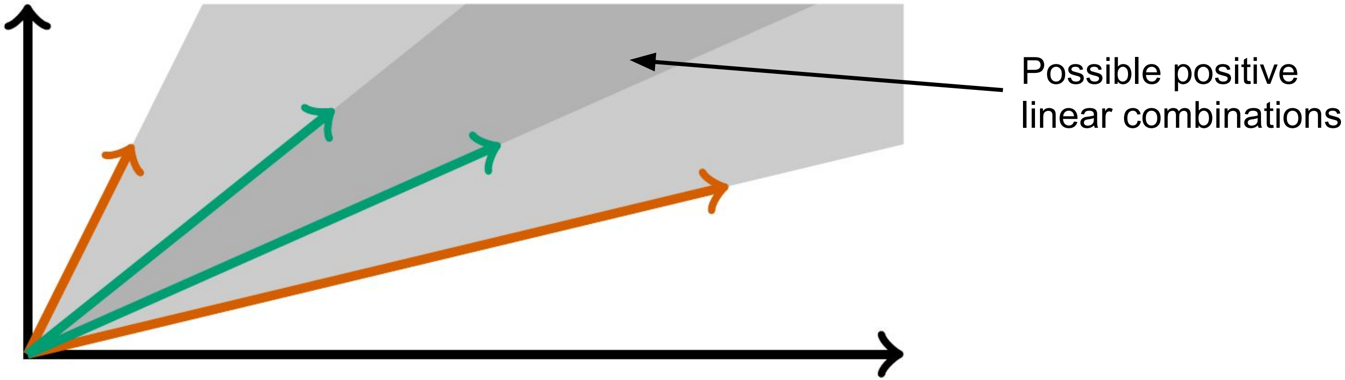
$$Z^+ = \begin{pmatrix}\mathbf{a}_1 & \mathbf{a}_2\end{pmatrix} =$$



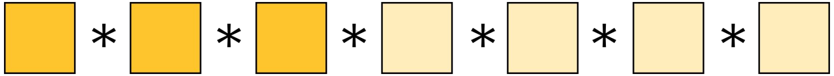
Geometric Intuition



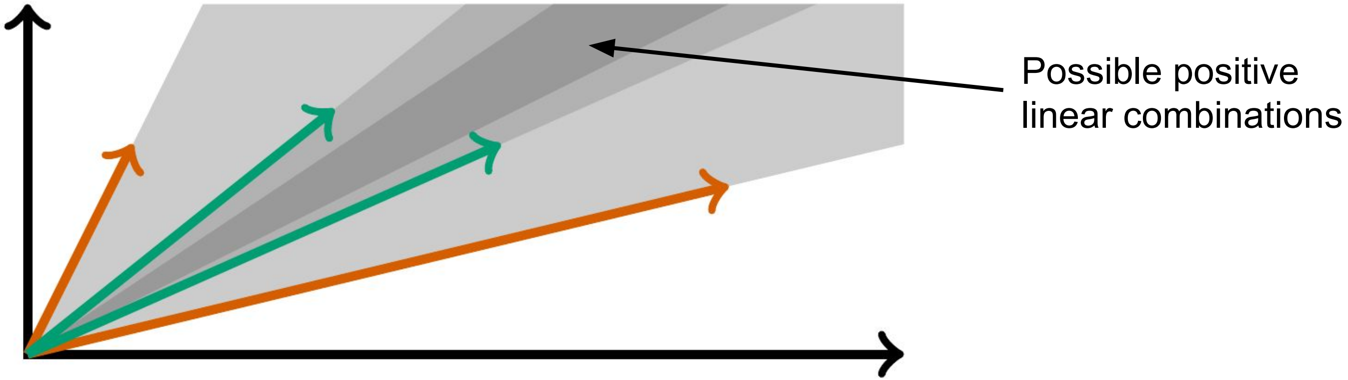
2nd Layer



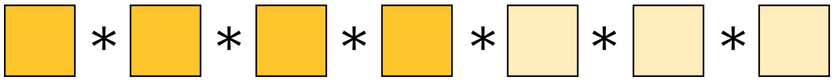
Geometric Intuition



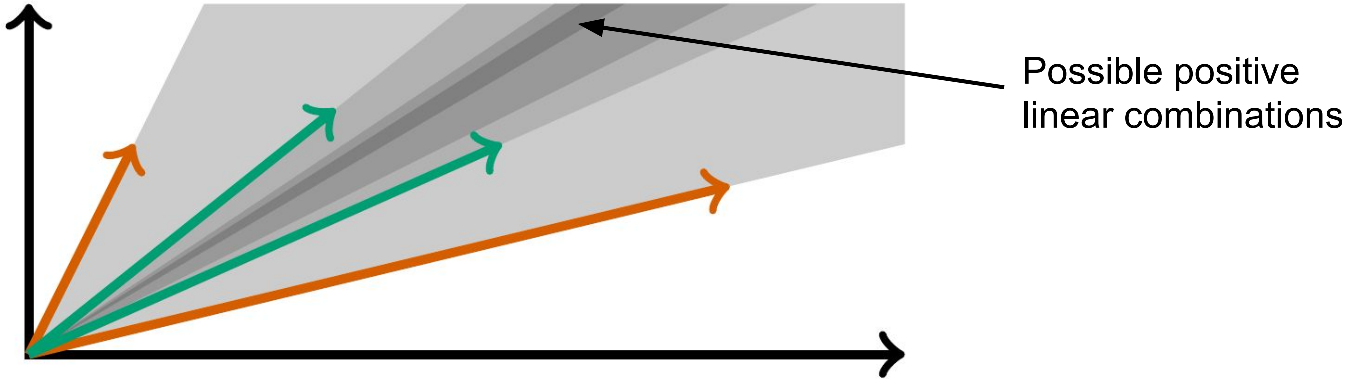
3rd Layer



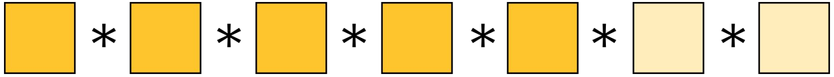
Geometric Intuition



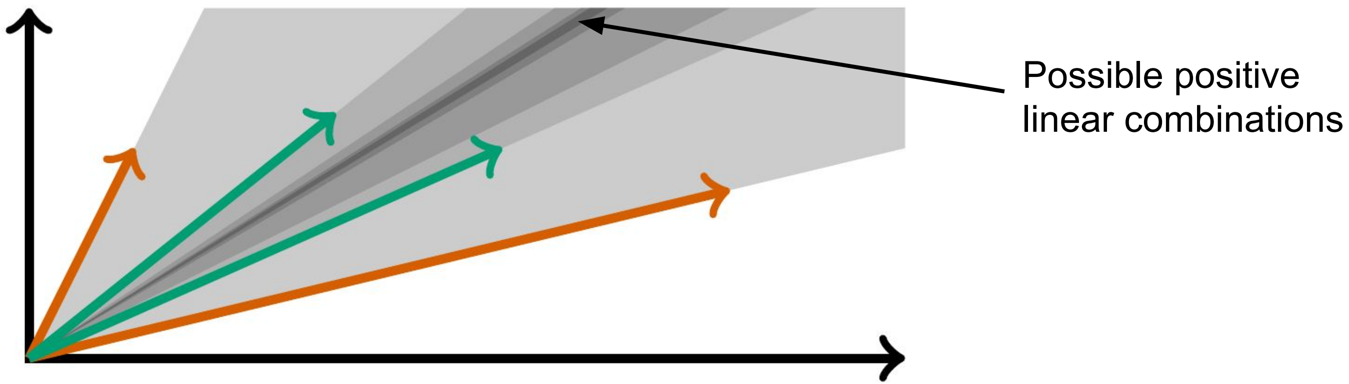
4th Layer



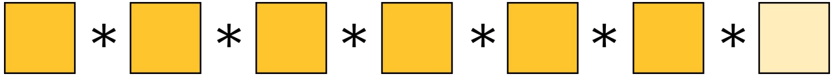
Geometric Intuition



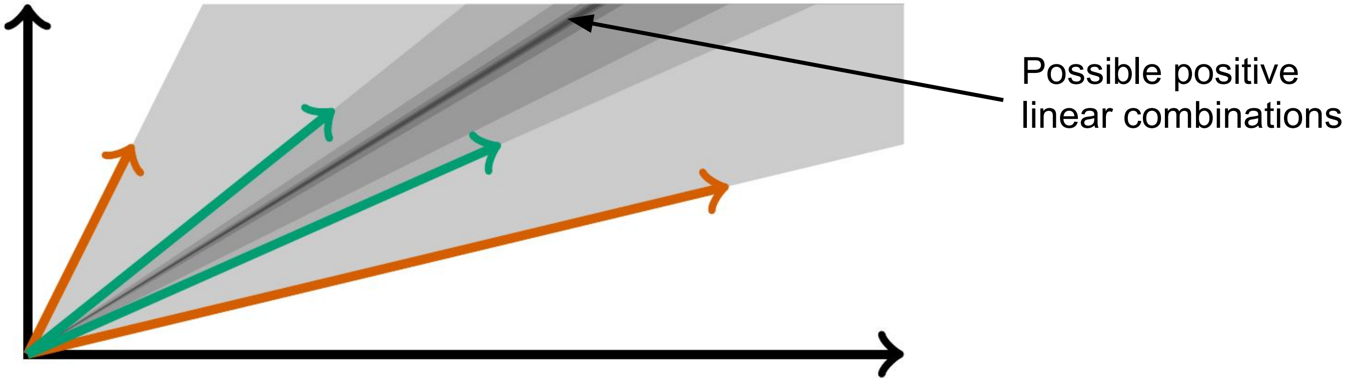
5th Layer



Geometric Intuition



6th Layer



- Output space shrink enormously!
- The saliency map is determined by early layers!

(see our paper for a rigorous proof)

LRP-αβ

- What happens if we add a few negative values?
- Weight positive **α** and negative **β** weights differently:

$$\left(\alpha Z_l^{+} - \beta Z_l^{-}\right)$$

- Restriction on α, β**:** $\alpha \geq 1 \text{ and } \alpha - \beta = 1$
- Most common α=1, β=0 and α=2, β=1

More Attribution Methods

See our paper for more methods:

- RectGrad, GuidedBP, Deconv
- LRP-z (non-converging, corresponds to *grad x input*)
- PatternAttribution: also ignores the network prediction
- DeepLIFT: takes later layers into account

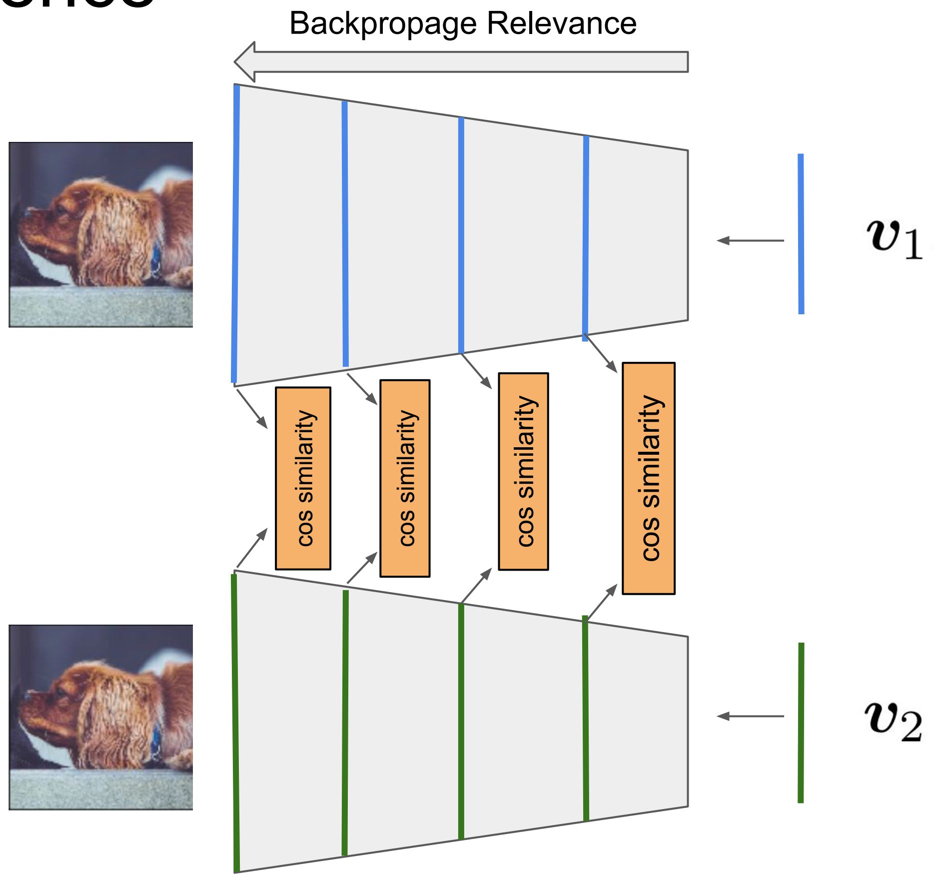
Cosine Similarity Convergence

Method to measure convergence

1. Sample two random vectors:

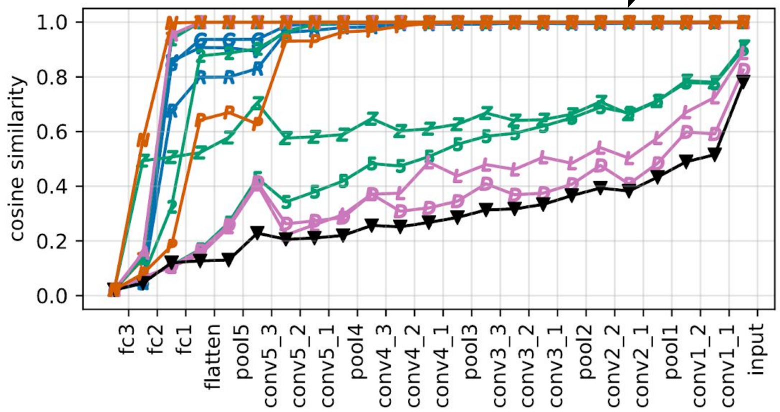
$$v_1, v_2 \sim \mathcal{N}(0, I)$$

2. Backpropagate random relevance vectors
3. Per layer, measure how well they align.

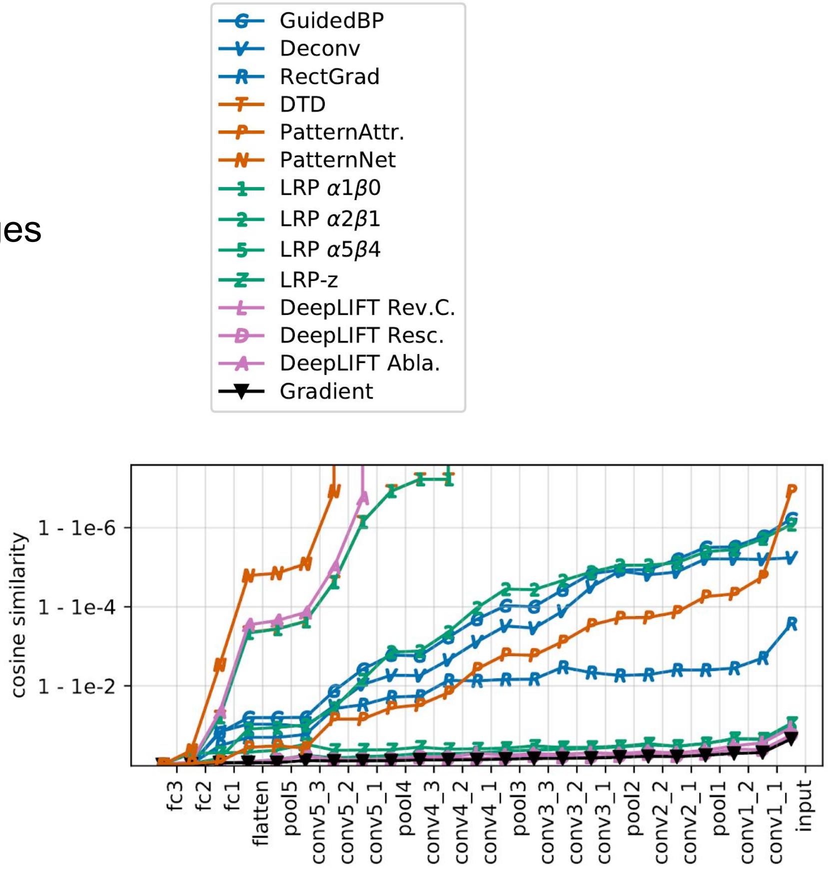


CSC: VGG-16

Median over many images
and random vectors

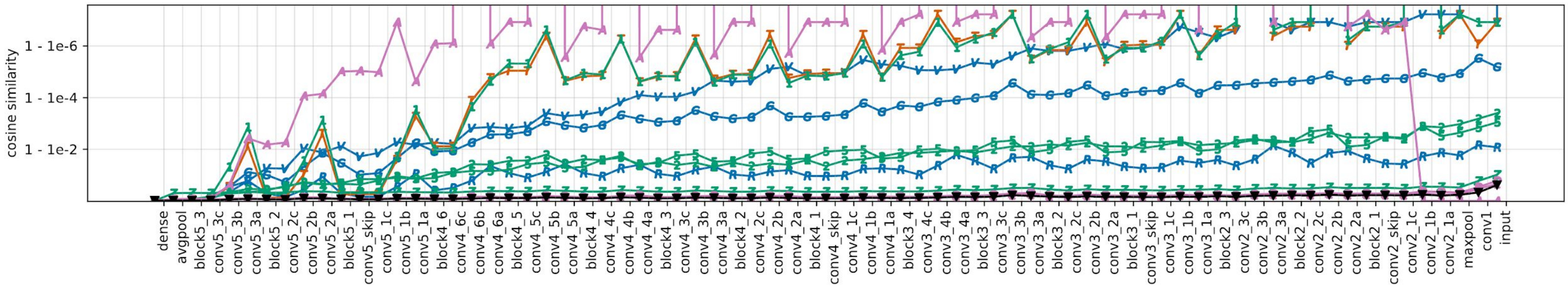


(b) VGG-16 (linear)

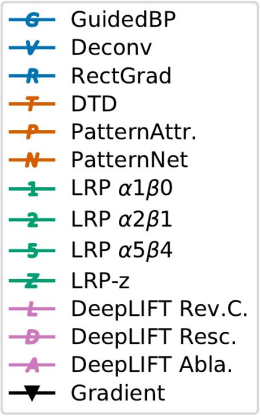


(c) VGG-16 (logarithmic)

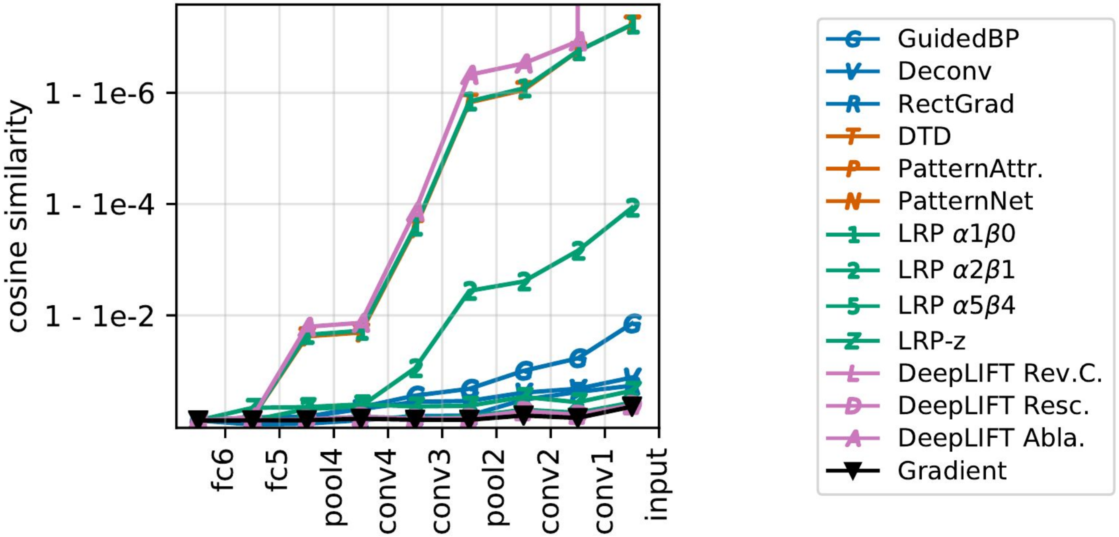
CSC: ResNet-50



(a) ResNet-50



CSC: Small CIFAR-10 Network



(d) CIFAR-10

Summary Attribution Methods

Insensitive to deeper layers

- PatternAttribution
- Deep Taylor Decomposition
- LRP-αβ
- ExcitationBP
- RectGrad
- Deconv
- GuidedBP

Sensitive to deeper layers

- DeepLIFT *(Shrikumar et al., 2017)*
- Gradient
- LRP-z
- Occlusion
- TCAV *(Kim et al., 2017)*
- Integrated Gradients, SmoothGrad
- IBA *(Schulz et al., 2020)*

Outlook to the paper

- More modified BP methods:
  - RectGrad, GuidedBP, Deconv
  - LRP-z
  - PatternAttribution: also ignores the network prediction
  - DeepLIFT: does not converge
- We discuss ways to improve class sensitivity
  - LRP-Composite *(Kohlbrenner et al., 2019)*
  - Contrastive LRP *(Gu et al., 2018)*
  - Contrastive Excitation BP *(Zhang et al., 2018)*

⇨ Do not resolve the convergence problem

Take away points

- Many modified BP methods ignore important parts of the network
- Check: If the parameter change, do the saliency maps change too?

Thank you!