When Explanations Lie: Why Modified BP Attribution Fails

Leon Sixt
Dahlem Center of
Machine Learning and Robotics
Freie Universität Berlin
leon.sixt@fu-berlin.de

Maximilian Granz
Dahlem Center of
Machine Learning and Robotics
Freie Universität Berlin
maximilian.granz@fu-berlin.de

Tim Landgraf
Dahlem Center of
Machine Learning and Robotics
Freie Universität Berlin
tim.landgraf@fu-berlin.de

Abstract

Modified backpropagation methods are a popular group of attribution methods. We analyse the most prominent methods: Deep Taylor Decomposition, Layer-wise Relevance Propagation, Excitation BP, PatternAttribution, Deconv, and Guided BP. We found empirically that the explanations of the mentioned modified BP methods are independent of the parameters of later layers and show that the z^+ rule used by multiple methods converges to a rank-1 matrix. This can explain well why the actual network's decision is ignored. We also develop a new metric cosine similarity convergence (CSC) to directly quantify the convergence of the modified BP methods to a rank-1 matrix. Our conclusion is that many modified BP methods do not explain the predictions of deep neural networks faithfully.

1 Introduction

Due to the large numbers of parameters and operations modern deep neural networks use to map an input to an output, it is difficult to interpret a network's decision. Attribution methods help understanding neural networks by assigning each input variable a score reflecting how relevant that variable was for the output. For images, the results can be visualised in so-called saliency maps. For a photo of a cat and a dog, and the network's classification result "dog", you would expect the attribution to highlight the image regions corresponding to only the canine. Many different attribution methods have been proposed and they can be categorized into three groups: black-box, gradient-based, and modified backpropagation methods.

Black-box methods, such as *Occlusion*, measure the sensitivity of the network when patches of the input are set to zero (Zeiler and Fergus, 2014). Gradient based methods compute the gradient of a given output class w.r.t. the input. An exemplary gradient based method is *SmoothGrad*(Smilkov et al., 2017). It averages the gradient within a local neighbourhood of the input to remove noise.

Modified backpropagation (BP)methods use custom definitions of relevance and propagate these back to the input. An example for how they differ from conventional gradient backpropagation is the ReLU operation. The gradient is only backpropagated through active neurons (those with input > 0); but most modified BP methods also assign a relevance to non-active neurons.

In this work, we analysed the following modified BP methods: *Layer-wise Relevance Propagation (LRP)*, *Deep Taylor Decompositon (DTD)*, *Deconv*, *Excitation BP*, *Guided BP*, and *PatternAttribution*

(Bach et al., 2015; Montavon et al., 2017; Zeiler and Fergus, 2014; Zhang et al., 2018; Springenberg et al., 2014; Kindermans et al., 2018).[1]

Modified BP methods are popular with practitioners. For example, LRP is used by Böhle et al. (2019) for the task of explaining MRT scans of Alzheimer patients and by Yang et al. (2018) to explain clinical decisions regarding breast cancer. Schiller et al. (2019) analyses the classification of whale sounds with DTD.

We found that the analysed modified BP methods do not explain the decisions of deep neural networks faithfully. Randomizing the parameters of later layers (Adebayo et al., 2018) did not change the saliency map. However, as the final layer is responsible for the prediction, this result directly questions the faithfulness of the analysed methods.

We investigate why the explanations are independent from the network's decision theoretically and empirically. Modified BP methods propagate a measure of relevance back to the input, effectively yielding a sequence of matrix multiplications. If the matrix chain converges to a rank-1 matrix, the resulting saliency map will always be the same, irrespective of the network's output. We show this theoretically for the often used z^+-rule as it corresponds to a chain of non-negative matrices. Using our novel cosine similarity convergence (CSC) metric, we measure the convergence of the modified BP methods to a rank-1 matrix empirically. CSC allows to retrace, layer by layer, how modified BP methods lose information about previous layers. Using CSC, we observe that all analysed modified BP methods converge to a rank-1 matrix on a ResNet-50 and VGG-16.

Our results shed light on the limitations of modified BP rules and allow practitioners to choose the right method for their task and model at hand.

2 Theoretical Analysis

Notation For our theoretical analysis, we consider feed forward neural networks with a ReLU activation function $[\boldsymbol{x}]^+ = \max(0, \boldsymbol{x})$. The neural network $f(\boldsymbol{x})$ contains n layers each with weight matrices W_l. The output of the l-th layer is denoted by $\boldsymbol{h}_l$. We use $[ij]$ to index the i, j element in W_l as in $W_{l_{[ij]}}$. To simplify notation, the bias terms can be absorbed into the weight matrix and we omit the final softmax layer. We refer to the input with $\boldsymbol{h}_0 = x$ and to the output with $\boldsymbol{h}_n = f(x)$. The output of the l-layer is given by:

$$\boldsymbol{h}_l = [W_l \boldsymbol{h}_{l-1}]^+ \tag{1}$$

All the results apply to convolutional neural networks as convolution can be expressed as matrix multiplication.

Gradient The gradient of the k-th output of the neural network w.r.t. the input $\boldsymbol{x}$ is given by:

$$\frac{\partial f_k(\boldsymbol{x})}{\partial \boldsymbol{x}} = W_1^T I_{(h_1>0)} \frac{\partial f_k(\boldsymbol{x})}{\partial \boldsymbol{h}_1} \overset{(a)}{=} \prod_l^n (G_l) \cdot \boldsymbol{v}_k, \tag{2}$$

where $I_{(h_1>0)} = \operatorname{diag}(\mathbb{1}_{h_1>0})$ denotes the gradient mask of the ReLU operation. Using $G_l = W_l^T I_{h_l}$, (a) follows from recursive expansion. The vector $\boldsymbol{v}_k$ is a one hot vector to select the k-th output.

The following methods modify the gradient definition and to distinguish the rules, we introduce the following notation: $r_l^{\nabla}(\boldsymbol{x}) = \frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{h}_l}$ which denotes the relevance at layer l for an input $\boldsymbol{x}$. For the gradient, the final saliency map is usually obtained by summing the absolute channel values of the relevance vector $r_0^{\nabla}(\boldsymbol{x})$ of the input layer.

z^+-Rule is used by DTD (Montavon et al., 2017), Excitation BP (Zhang et al., 2018) and also corresponds to the $\text{LRP}_{\alpha 1 \beta 0}$ rule (Bach et al., 2015). It only backpropagates positive relevance values. Let w_{ij} be an entry in the weight matrix W_l:

$$r_l^{z^+}(\boldsymbol{x}) = Z_l^+ \cdot r_{l+1}^{z^+}(\boldsymbol{x}), \quad \text{where} \quad {Z_l^+}^T = \left(\frac{[w_{ij} \boldsymbol{h}_{l_{[j]}}]^+}{\sum_k [w_{ik} \boldsymbol{h}_{l_{[k]}}]^+} \right)_{[ij]}. \tag{3}$$

[1] We did not evaluate *Constrastive Excitation BP* and *DeepLift* (Shrikumar et al., 2017), but plan to include them in an updated version of this manuscript.

The relevance function $r_l^{z^+} : \mathbb{R}^n \mapsto \mathbb{R}^m$ maps input $\boldsymbol{x}$ to a relevance vector of layer l. For the final layer the relevance is set to the value of the explained logit value, i.e. $r_n^{z^+}(\boldsymbol{x}) = f_k(\boldsymbol{x})$. Each entry in the derivation matrix Z_l^+ is obtained by measuring the positive contribution of the input neuron i to the output neuron j and normalizing by the total contributions to neuron j. The relevance from the previous layer $r_{l+1}^{z^+}$ is then distributed according to Z_l^+.

In DTD, Excitation BP, and LRP, the derivation rule for ReLU is the identity. This is different to the gradient which uses a mask to only backpropagate to active neurons. To compute the relevance of multiple layers, the z^+-rule is applied recursively and yields a product of matrices, similar as before with the gradient: $C_k = \prod_l^k Z_l^+$. As the matrices are non-negative, the product converges to a rank-1 matrix, i.e. the column vectors of the converged matrix are linear dependent $C = \boldsymbol{c}\boldsymbol{\gamma}^T$. If converged, the Z_{k+1}^+ matrix has no influence on the relevance other than scaling as $CZ_{k+1}^+ = \boldsymbol{c}\boldsymbol{\gamma}^T Z_{k+1}^+ = \boldsymbol{c}\boldsymbol{\lambda}^T$ and for any vector $\boldsymbol{v}$: $CZ_{k+1}^+\boldsymbol{v} = \boldsymbol{c}\boldsymbol{\lambda}^T\boldsymbol{v} = t\boldsymbol{c}$ where $t \in \mathbb{R}$. For attribution methods, this means that the relevance vectors r_l of later layers $l > k$ do not contribute to the final result other than the scaling. However, the final decision of the network is made in the last layer and therefore a method converged to a rank-1 matrix *cannot* explain the network's true decision process.

The convergence of squared irreducible non-negative matrices to a rank-1 matrix was proven in Hajnal (1976).[2] A matrix is irreducible if no permutation matrix P exists such that:

$$PAP^T = \begin{pmatrix} A_{m\times m} & A_{m\times n} \\ 0 & A_{n\times n} \end{pmatrix}, \tag{4}$$

where $A_{n\times n}$ is a block matrix.

The geometric intuition behind the proof is: The column vectors of the first matrix are all non-negative and therefore in the positive quadrant. Each matrix multiplication A_i with a non-negative vector is a non-negative linear combination of the column vectors and the result will stay in the positive span of the column vectors of A_i. The irreducibility ensures that the span shrinks with every iteration until it converged to a single vector. The convergence is exponentially fast.

The Z^+ matrices of dense layers fulfill the conditions of the theorem. For convolutional layers, the weight matrices are irreducible locally but not globally. Locally, convolutions behave like full matrix multiplications, e.g. a 1x1 convolution can be seen as a matrix multiplication applied to each feature map location separately. This implies that the z^+-rule will converge for convolutions only locally.

LRP$_z$ The LRP$_z$ rule of Layer-wise Relevance Propagation modifies the back-propagation rule as follows:

$$r_l^{z-\text{LRP}}(\boldsymbol{x}) = Z_l \cdot r_{l+1}^{z-\text{LRP}}(\boldsymbol{x}),, \quad \text{where} \quad Z_l = \left(\frac{w_{ij}\boldsymbol{h}_{l_{[j]}}}{\sum_k w_{ik}\boldsymbol{h}_{l_{[k]}}} \right)^T_{[ij]}. \tag{5}$$

For neural networks with only max-pooling and ReLU, it was shown that LRP$_z$ corresponds to Grad$\odot$Input, i.e. $r_0^{z-\text{LRP}}(\boldsymbol{x}) = \boldsymbol{x} \odot \frac{\partial f(\boldsymbol{x})}{\partial \boldsymbol{x}}$ (Ancona et al., 2017). This also means that LRP$_z$ could be considered rather a gradient based and not a modified BP method.

LRP$_{\alpha\beta}$ separates the positive and negative influences:

$$r_l^{\alpha\beta}(\boldsymbol{x}) = \left(\alpha Z_l^+ - \beta Z_l^-\right) r_{l+1}^{\alpha\beta}(\boldsymbol{x}), \tag{6}$$

where Z_l^+ and Z_l^- correspond to the positive and negative entries of the matrix Z from LRP$_z$. With $\alpha - \beta = 1$, $\alpha > 0$, and $\beta \geq 0$, it is ensured that the total amount relevance is conserved. For LRP$_{\alpha 1 \beta 0}$, this rule corresponds to the z^+-rule which converges. For $\alpha > 1$ and $\beta > 0$, the matrix $Z_l = \alpha Z_l^+ - \beta Z_l^-$ can contain negative entries.

Deep Taylor Decomposition uses the z^+-rule if the input to a convolutional or dense layer is in $[0, \infty]$, e.g. if the layers follow a ReLU activation. For inputs in $\mathbb{R}$, DTD also proposed the w^2-rule and the so-call $w^{\mathcal{B}}$ rule for bounded inputs. Both rules were specifically designed to produce non-negative outputs. DTD will necessarily converge to a rank-1 matrix for a sufficiently deep network.

[2] We plan to include a rigours proof independent of matrix size in an updated version of this preprint.

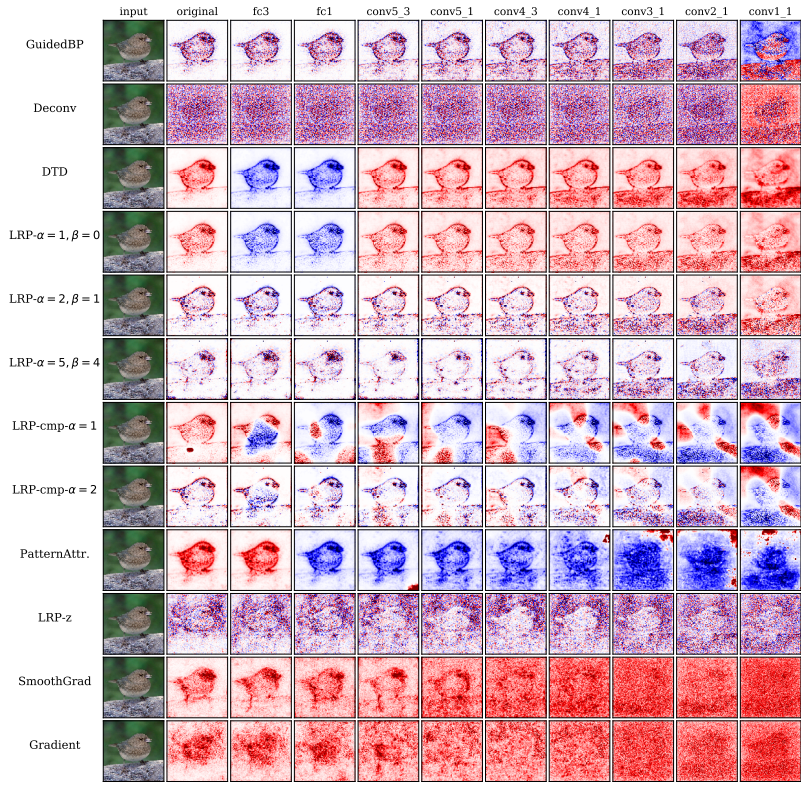


Figure 1: Saliency maps for all evaluated methods. The parameters of the VGG-16 are randomized starting from the last to the first layer.

Guided Backpropagation & Deconv apply an additional ReLU to the gradient and it was shown to be invariant to the randomization of later layers previously in Adebayo et al. (2018) and analysed theoretically in Nie et al. (2018):

$$r_l^{GBP}(\boldsymbol{x}) = W_l^T \left[I_{h_l} r_{l+1}^{GBP}(\boldsymbol{x}) \right]^+ . \tag{7}$$

$I_{h_1} = \mathrm{diag}(\mathbb{1}_{h_1>0})$ denotes the gradient mask of the ReLU operation. For Deconv, the mask of the forward ReLU is ignored and the gradients are rectified directly. As both apply a ReLU operation, the backpropagation is no longer a linear function.The ReLU operation results in non-negative outputs but it does not necessarily have to converge to a rank-1 matix.

PatternAttribution takes into account that the input h_l contains noise. They assume that $\boldsymbol{h}_l = \boldsymbol{s}+\boldsymbol{d}$ where $\boldsymbol{s}$ corresponds to the signal in the data and $\boldsymbol{d}$ to the noise. To assign the relevance towards the signal direction, they estimate for all weight vectors $\boldsymbol{w}$ a corresponding signal vector $\boldsymbol{a}$ from data. Let A_l be the corresponding signal matrix to a weight matrix W_l:

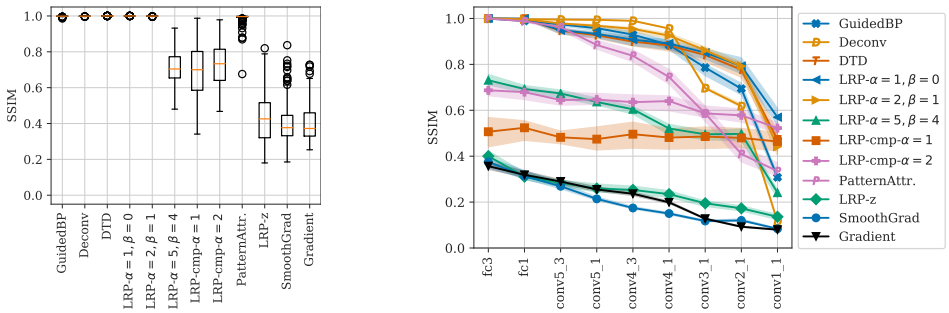
$$r_l^{\text{PA}}(\boldsymbol{x}) = (W_l \odot A_l)^T \cdot r_{l+1}^{\text{PA}}(\boldsymbol{x}), \tag{8}$$

As both the weight matrices and the signal matrices can contain negative values, they don't converge necessarily.

Excitation BP uses the z^+-rule similar to DTD and is actually equivalent to $\text{LRP}_{\alpha 1 \beta 0}$.

3 Evaluation

Setup We report our results on a VGG-16 (Simonyan and Zisserman, 2014) and a ResNet-50 (He et al., 2016) trained on the ImageNet dataset. All results were computed on the exemplary bird image and 199 randomly picked images from the validation set. We show bootstrap confidence intervals in figure 2b that justify this sample size. We used the implementation from the innvestigate package (Alber et al., 2019).



(a) Random Logits

(b) Cascading Parameter Randomization

Figure 2: (a) SSIM between saliency maps from the true and a random logits (b) Median SSIM between original saliency map and saliency maps from cascading layer randomization. The results were obtain on a VGG-16. Intervals show 99% bootstrap confidences.

Random Logit We display the difference of saliency maps for the right logit and a random logit in figure 2a. As the logit value is responsible for the predicted class, we would expect the saliency maps to change. We use the SSIM metric (Wang et al., 2004) to quantify the difference as in Adebayo et al. (2018). Except of $\text{LRP}_{\alpha 5 \beta 4}$ and LRP_{CMP}, all modified BP methods produce saliency maps independent of the explained class for the VGG-16. The ResNet-50 are show in the appendix A and the main difference for ResNet-5o is that the saliency maps of $\text{LRP}_{\alpha 2 \beta 1}$ change a bit more.

Sanity Check: Randomization of Parameters We follow Adebayo et al. (2018) and randomized the parameters starting from the last layer to the first layer. For DTD and $\text{LRP}_{\alpha 1 \beta 0}$, the randomization of the last layer flips the sign of the saliency map sometimes. We therefore compute the the SSIM also between the inverted saliency map and report the maximum.

In figure 1, we display the effect of random parameters on the saliency map for an exemplary input. In figure 2b, we report the SSIM between the saliceny maps obtained from the original model and from a model with partial random parameters. While the produced saliency maps of SmoothGrad and LRP_z drop already when the last fully connected layer is randomized, the saliency maps of $\text{LRP}_{\alpha\beta}$, DTD, PatternAttribution and GuidedBP [3]remain similar. The results for LRP_{CMP} are discussed below in detail after the next paragraph.

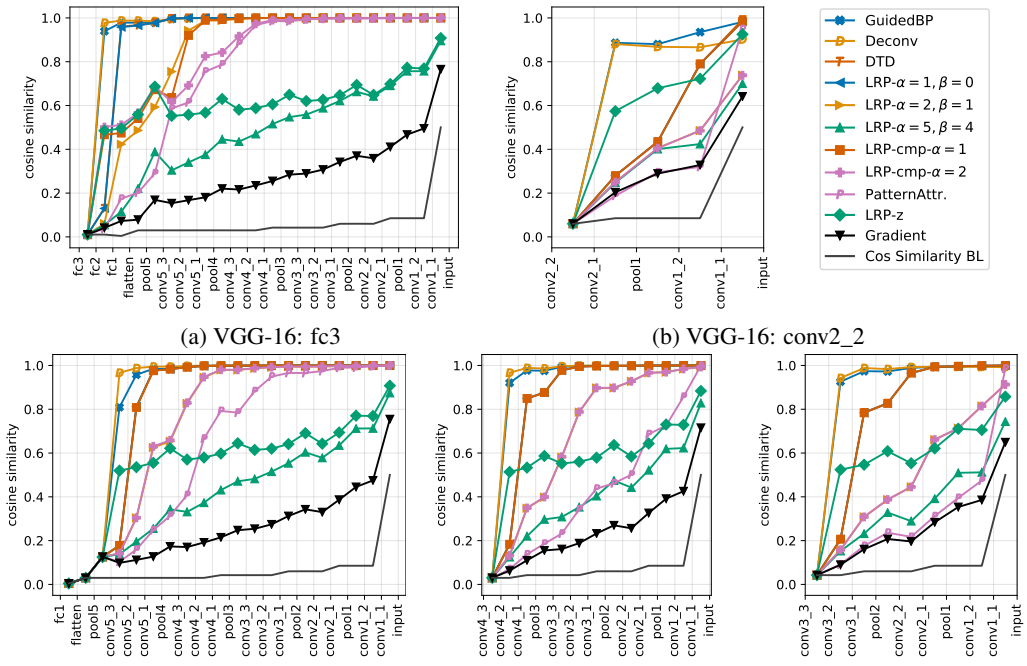
Cosine Similarity Convergence Metric (CSC) Instead of randomizing the parameters, we randomize the backpropagated relevance vectors directly. We select layer k and set the corresponding relevance to $r_k(\boldsymbol{x}) := \boldsymbol{v}_1$ where $\boldsymbol{v}_1 \sim \mathcal{N}(0, I)$ and than backpropagate it as before. For example, for the gradient we would do: $\frac{\partial h_k}{\partial h_1} \frac{\partial f(x)}{\partial h_k} := \frac{\partial h_k}{\partial h_1} \boldsymbol{v}_1$. We use the notation $r_l(\boldsymbol{x}|r_k := \boldsymbol{v}_1)$ to describe the relevance r_l at layer l when the relevance of layer k is set to $\boldsymbol{v}_1$.

Using two random relevance vectors $\boldsymbol{v}_1, \boldsymbol{v}_2 \sim \mathcal{N}(0, I)$, we can measure the convergence using the cosine similarity. If the relevance matrices converged to $C = \prod_l Z_l$, the columns of C are linear dependent $C = [\gamma_1 \boldsymbol{c}, \ldots, \gamma_k \boldsymbol{c}]$ and the result $C\boldsymbol{v} = \lambda \boldsymbol{c}$ is only a scaling of the column vector $\boldsymbol{c}$. The backpropagated relevance vectors of $\boldsymbol{v}_1, \boldsymbol{v}_2$ will align more and more as the matrix chain converges. We quantify their alignment using the cosine similarity:

$$s_{\cos}(r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_1), r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_2))) = \frac{r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_1)^T r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_2)}{\|r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_1)\| \cdot \|r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_2)\|}. \tag{9}$$

If the relevance matrix chain converged, we have for both $\boldsymbol{v}_1, \boldsymbol{v}_2$: $r_l(\boldsymbol{x}|r_k{=}\boldsymbol{v}_i) = C\boldsymbol{v}_i = c\boldsymbol{\gamma}^T \boldsymbol{v}_i = \lambda_i \boldsymbol{c}$ where $\lambda_i = \boldsymbol{\gamma}^T \boldsymbol{v}_i$ and their cosine similarity will be one. The opposite direction is also true. If C has shape $n \times m$ with $n \leq m$ and if for n linear independent vectors $\boldsymbol{v}_i$, the cosine similarity $s_{\cos}(C\boldsymbol{v}_i, C\boldsymbol{v}_j) = 1$, then C is a rank-1 matrix.

[3] For GuidedBP, we report different saliency maps than shown in figure 2 of Adebayo et al. (2018). We were able to confirm a bug in their code that resulted in saliency maps of GuidedBP images to remain identical even for earlier layers.



(a) VGG-16: fc3

(b) VGG-16: conv2_2

(c) VGG-16: fc1

(d) VGG-16: conv4_3

(e) VGG-16: conv3_3

Figure 3: Median of the cosine similarity convergence (CSC) per layer between relevance vectors obtained from randomizing the relevance vectors of different layers. A cosine similarity of 1 means the method converged to a rank-1 matrix. *Cos Similarity BL* shows the mean cosine similarity between two random vectors, i.e. $\mathbb{E}[s_{\cos}(\boldsymbol{u}_1, \boldsymbol{u}_2)]$ where $\boldsymbol{u}_1, \boldsymbol{u}_2 \sim \mathcal{N}(0, I)$.

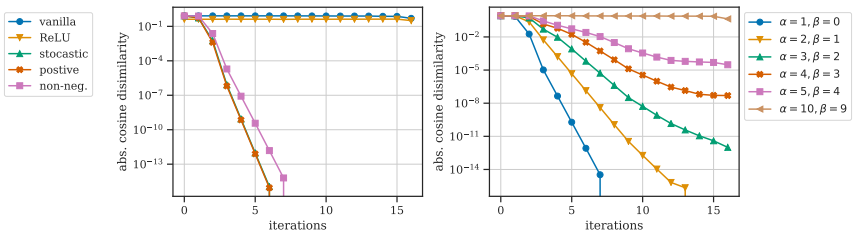
In figure 3, we plot the results and find that all investigated modified backpropagation rules converge. For convolution layers, we compute the cosine similarity per feature map location, i.e. for a shape of (h, w, c) we obtain $h \cdot w$ values. The jump in cosine similarity for the input, is a result of the input's low dimension of 3 channels. The convergence behaviour on a ResNet-50 is similar but a bit less pronounced (see appendix A). In particular, LRP$_{\alpha 2 \beta 1}$ does not fully converge on a ResNet-50.

LRP$_{CMP}$ A common practise is to apply LRP$_z$ to the final fully connected layers and LRP$_{\alpha\beta}$ to the convolutional layer (Kohlbrenner et al., 2019; Lapuschkin et al., 2017). This composition of LRP rules is called LRP$_{CMP}$ and we report results for $\alpha = 1, 2$ as in (Kohlbrenner et al., 2019). In figure 1, the saliency maps of LRP$_{CMP}$ are visualised and they do change when the network parameters are randomized. However, structurally, the underlying image seems to be scaled only locally (even switching signs).

Inspecting the cosine similarity path of the two LRP$_{CMP}$ variants in figure 3a, we can see why. Both do not converge for the fully connected layers where LRP$_z$ was applied but they quickly convergence after 3-5 convolutional layers when LRP$_{\alpha\beta}$ is applied. The explanations from the fully connected layer can change the coarse local scaling of the relevance vectors but they cannot alter the relevance vector's direction to highlight different details. LRP$_{CMP}$ is good for highlighting relevant image areas but its backpropagated relevance vectors contain no information about the networks decision.

Simulation of matrix convergences In figure 4, we show the converging behaviour for a matrix chains with similiar shapes as a VGG-16. The convolutional kernels are considered to be 1x1, e.g. for a kernel of size (3, 3, 256, 128) we would use a matrix of size (256, 128).

In figure 4a, we test out the effect of different matrix properties. For *vanilla*, we sample the matrix entries from an normal distribution. In the next setting, we apply a *ReLU* operation after each multiplication. We generate *non-negative* matrices containing 50% zeros by clipping them to $[0, \infty]$. And *positive* matrices by taking the absolute value. We report the cosine similarity between the column vectors of the matrix. The positive, stochastic, and non-negative matrices converge



(a) different matrix properties (b) $\alpha W^+ + \beta W^-$

Figure 4: Simulated convergence for a matrix chain.

exponentially fast to a rank-1 matrix. The 50% zeros in the non-negative matrices only result in a bit lower convergence slope. After 7 iterations, they converged to floating point imprecision.

We also investigated how a slightly negative matrix influence the convergence. In figure 4b, we show the converges of matrices: $\alpha W^+ + \beta W^-$ where $W^+ = \max(0, W), W^- = \min(0, W)$ and $W \sim \mathcal{N}(0, I)$. We find that for small enough $\beta < 4$ values the matrix chains still converge. This simulation motivated us to include $\text{LRP}_{\alpha 5 \beta 4}$ in our evaluation which show less convergence but its saliency maps also contain more noise.

4 Related Work

We did not run our evaluation on DeepLift (Shrikumar et al., 2017) and *Contrastive Excitation BP* (Zhang et al., 2018), as there was no ready to use implementation in innvestigate package.

Besides the modified BP attribution methods discussed here, there also exist gradient averaging methods such as SmoothGrad (Smilkov et al., 2017) and Integrated Gradients (Sundararajan et al., 2017). *CAM* (Zhou et al., 2016) and *Grad-CAM* (Selvaraju et al., 2017) use the activation of the last convolutional layer to determine important areas. *Occlusion* (Zeiler and Fergus, 2014) measures the sensitivity of the neural network when image patches are set to zero. Schulz et al. (2020) applies noise to an intermediate feature map to determine which areas do not contribute to the network output. All the mentioned attribution methods do not converge, as they either rely only on the gradient or measure the sensitivity directly as Occlusion. SpRay and TCAV move beyond pixel-wise attribution by using extracted concepts to explain the network's decision (Lapuschkin et al., 2019; Kim et al., 2018).

Evaluating attribution methods is inherently difficult. The results of an attribution method depends on the used model and dataset and it is hard to tell apart if an error is made by the attribution method or the neural network. A commonly used benchmark is to degrade input images according to the attribution heatmap and measure the impact on the model output (Kindermans et al., 2018; Samek et al., 2016; Ancona et al., 2017).

Nie et al. (2018) found the saliency maps of GuidedBP to be invariant when a random logit is explained. They also analyse GuidedBP theoretically and show that it has a tendency to rather reconstruct the input than to explain the network's decision. We used the parameter randomization sanity check (Adebayo et al., 2018) which showed that GuidedBP is invariant to the changes in the later layers. To our best knowledge, we are the first to show that many modified backpropagation attribution methods fail to faithfully explain the network's decision.

Another branch of related work is the analysis of neural networks. Balduzzi et al. (2017) investigated the scattering of gradients in ResNets.

5 Conclusion

While motivated well for linear models, we provide evidence that many modified backpropagation methods do not and can not explain decisions of deep neural networks. Specifically, we found PatternAttribution, DTD, $\text{LRP}_{\alpha\beta}$, Excitation BP, Guided BP, and Deconv to ignore large parts of the network's computation. The saliency maps of the mentioned methods stay almost identical when

explaining a random logit or when randomizing later layers. Thus, the layers responsible for the final decision have no influence on the explanations.

We analysed theoretically why the decisions of later layers are ignored and found that for the z^+-rule the corresponding matrices converge to a rank-1 matrix. We also analysed the convergences empirically and found that all analysed methods converged to a rank-1 matrix on VGG-16 and all except $\text{LRP}_{\alpha 2 \beta 1}$ converged on a ResNet-50.

Our theoretical findings and the CSC-metric could contribute to improving modified backpropagation methods. Our analysis suggests that negative entries in the derivation matrices play a vital role in keeping the matrix chain from converging. For $\text{LRP}_{\alpha\beta}$ this suggests testing higher values for α, confirmed by our result for $\text{LRP}_{\alpha 5 \beta 4}$ which exhibits lower CSC values but also produces more noise in the saliency maps.

6 Acknowledgements

We would like to thank Benjamin Wild, Karl Schulz, Julian Stastny, and David Dormagen for stimulating discussions and helpful feedback. LS was supported by the Elsa-Neumann-Scholarship by the state of Berlin. We are also grateful to Nvidia for providing us with a Titan Xp and to ZEDAT for granting us access to their HPC system.

References

Julius Adebayo, Justin Gilmer, Michael Muelly, Ian Goodfellow, Moritz Hardt, and Been Kim. Sanity checks for saliency maps. In *Advances in Neural Information Processing Systems*, pages 9505–9515, 2018.

Maximilian Alber, Sebastian Lapuschkin, Philipp Seegerer, Miriam Hägele, Kristof T Schütt, Grégoire Montavon, Wojciech Samek, Klaus-Robert Müller, Sven Dähne, and Pieter-Jan Kindermans. innvestigate neural networks! *Journal of Machine Learning Research*, 20(93):1–8, 2019.

Marco Ancona, Enea Ceolini, Cengiz Öztireli, and Markus Gross. A unified view of gradient-based attribution methods for deep neural networks. In *NIPS 2017-Workshop on Interpreting, Explaining and Visualizing Deep Learning*. ETH Zurich, 2017.

Sebastian Bach, Alexander Binder, Grégoire Montavon, Frederick Klauschen, Klaus-Robert Müller, and Wojciech Samek. On pixel-wise explanations for non-linear classifier decisions by layer-wise relevance propagation. *PLoS ONE*, 10(7), 2015.

David Balduzzi, Marcus Frean, Lennox Leary, JP Lewis, Kurt Wan-Duo Ma, and Brian McWilliams. The shattered gradients problem: If resnets are the answer, then what is the question? In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 342–350. JMLR.org, 2017.

Moritz Böhle, Fabian Eitel, Martin Weygandt, and Kerstin Ritter. Layer-wise relevance propagation for explaining deep neural network decisions in mri-based alzheimer's disease classification. *Frontiers in Aging Neuroscience*, 11:194, 2019. ISSN 1663-4365. doi: 10.3389/fnagi.2019.00194.

J. Hajnal. On products of non-negative matrices. *Mathematical Proceedings of the Cambridge Philosophical Society*, 79(3):521–530, May 1976. ISSN 0305-0041, 1469-8064. doi: 10.1017/S030500410005252X.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

Been Kim, Martin Wattenberg, Justin Gilmer, Carrie Cai, James Wexler, Fernanda Viegas, et al. Interpretability beyond feature attribution: Quantitative testing with concept activation vectors (tcav). In *International Conference on Machine Learning*, pages 2673–2682, 2018.

Pieter-Jan Kindermans, Kristof T. Schütt, Maximilian Alber, Klaus-Robert Müller, Dumitru Erhan, Been Kim, and Sven Dähne. Learning how to explain neural networks: Patternnet and patternattribution. In *International Conference on Learning Representations*, 2018.

Maximilian Kohlbrenner, Alexander Bauer, Shinichi Nakajima, Alexander Binder, Wojciech Samek, and Sebastian Lapuschkin. Towards best practice in explaining neural network decisions with lrp, 2019.

Sebastian Lapuschkin, Alexander Binder, Klaus-Robert Muller, and Wojciech Samek. Understanding and comparing deep neural networks for age and gender classification. In *The IEEE International Conference on Computer Vision (ICCV) Workshops*, Oct 2017.

Sebastian Lapuschkin, Stephan Wäldchen, Alexander Binder, Grégoire Montavon, Wojciech Samek, and Klaus-Robert Müller. Unmasking clever hans predictors and assessing what machines really learn. *Nature communications*, 10(1):1096, 2019.

Grégoire Montavon, Sebastian Lapuschkin, Alexander Binder, Wojciech Samek, and Klaus-Robert Müller. Explaining nonlinear classification decisions with deep taylor decomposition. *Pattern Recognition*, 65:211–222, 2017.

Weili Nie, Yang Zhang, and Ankit Patel. A theoretical explanation for perplexing behaviors of backpropagation-based visualizations. In *International Conference on Machine Learning*, pages 3806–3815, 2018.

Wojciech Samek, Alexander Binder, Grégoire Montavon, Sebastian Lapuschkin, and Klaus-Robert Müller. Evaluating the visualization of what a deep neural network has learned. *IEEE transactions on neural networks and learning systems*, 28(11):2660–2673, 2016.

Dominik Schiller, Tobias Huber, Florian Lingenfelser, Michael Dietz, Andreas Seiderer, and Elisabeth André. Relevance-Based Feature Masking: Improving Neural Network Based Whale Classification Through Explainable Artificial Intelligence. In *Proc. Interspeech 2019*, pages 2423–2427, 2019. doi: 10.21437/Interspeech.2019-2707. URL `http://dx.doi.org/10.21437/Interspeech.2019-2707`.

Karl Schulz, Leon Sixt, Federico Tombari, and Tim Landgraf. Restricting the flow: Information bottlenecks for attribution. In *International Conference on Learning Representations*, 2020.

Ramprasaath R Selvaraju, Michael Cogswell, Abhishek Das, Ramakrishna Vedantam, Devi Parikh, and Dhruv Batra. Grad-cam: Visual explanations from deep networks via gradient-based localization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 618–626, 2017.

Avanti Shrikumar, Peyton Greenside, and Anshul Kundaje. Learning important features through propagating activation differences. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3145–3153. JMLR.org, 2017.

Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014.

Daniel Smilkov, Nikhil Thorat, Been Kim, Fernanda Viégas, and Martin Wattenberg. Smoothgrad: removing noise by adding noise. *arXiv:1706.03825 [cs, stat]*, 2017.

Jost Tobias Springenberg, Alexey Dosovitskiy, Thomas Brox, and Martin Riedmiller. Striving for Simplicity: The All Convolutional Net. *arXiv e-prints*, 2014.

Mukund Sundararajan, Ankur Taly, and Qiqi Yan. Axiomatic attribution for deep networks. In *Proceedings of the 34th International Conference on Machine Learning-Volume 70*, pages 3319–3328. JMLR. org, 2017.

Zhou Wang, Alan C Bovik, Hamid R Sheikh, Eero P Simoncelli, et al. Image quality assessment: from error visibility to structural similarity. *IEEE transactions on image processing*, 13(4):600–612, 2004.

Y. Yang, V. Tresp, M. Wunderle, and P. A. Fasching. Explaining therapy predictions with layer-wise relevance propagation in neural networks. In *2018 IEEE International Conference on Healthcare Informatics (ICHI)*, pages 152–162, June 2018.

Matthew D Zeiler and Rob Fergus. Visualizing and understanding convolutional networks. In *European conference on computer vision*, pages 818–833. Springer, 2014.

Jianming Zhang, Sarah Adel Bargal, Zhe Lin, Jonathan Brandt, Xiaohui Shen, and Stan Sclaroff. Top-down neural attention by excitation backprop. *International Journal of Computer Vision*, 126 (10):1084–1102, 2018.

B. Zhou, A. Khosla, A. Lapedriza, A. Oliva, and A. Torralba. Learning deep features for discriminative localization. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2921–2929, 2016.

A Results on ResNet-50

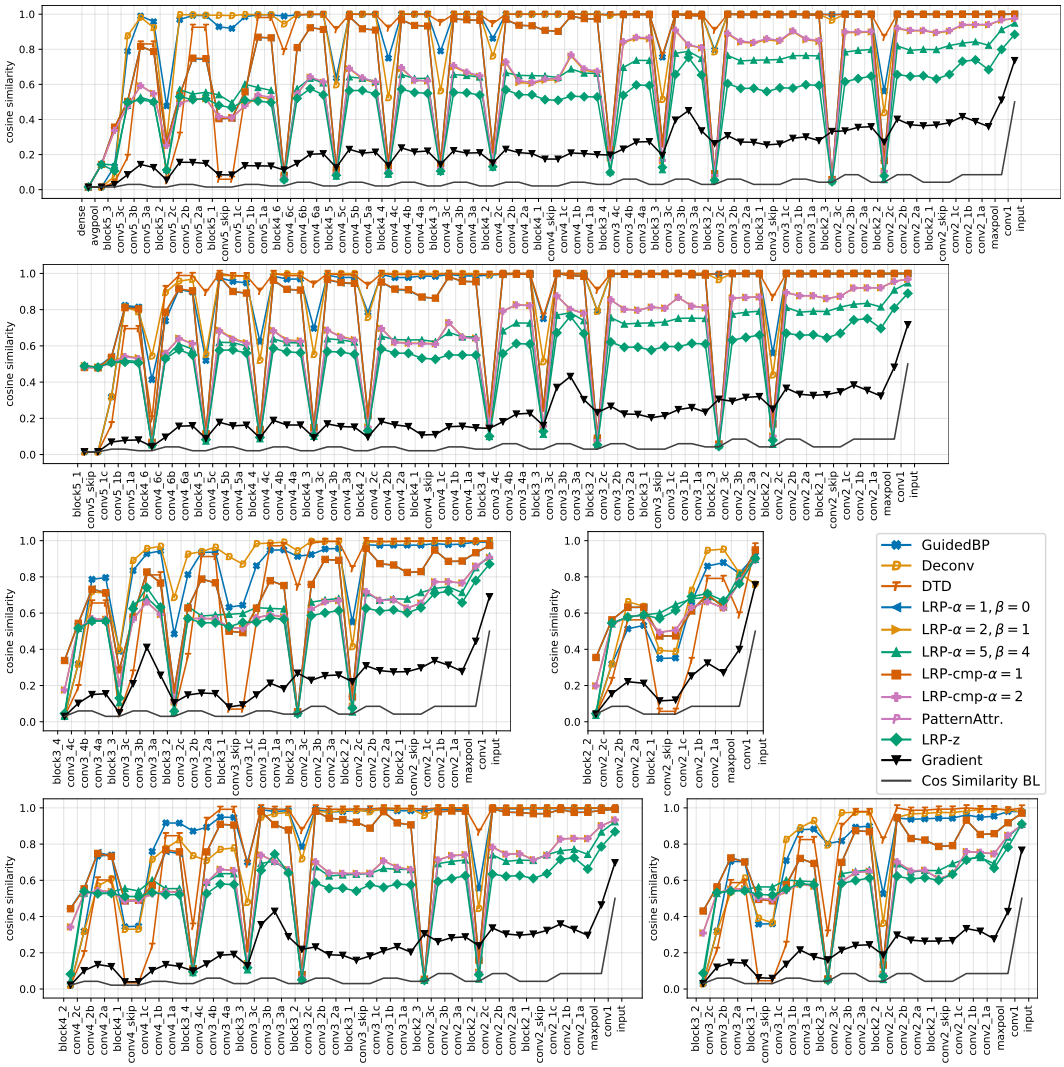


Figure 5: Convergence measured using the CSC for different starting layers in a ResNet-50. The drops happen for skip connections.

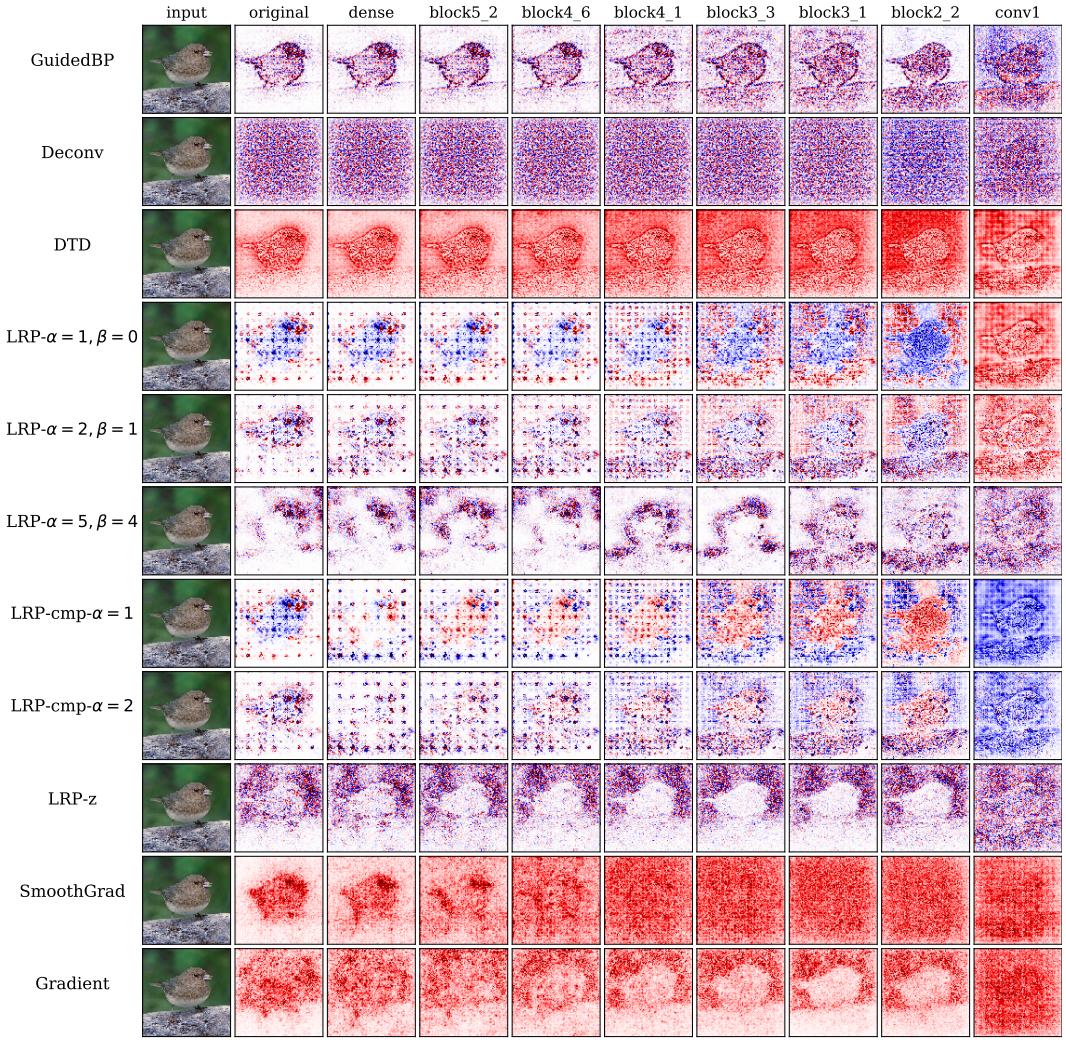
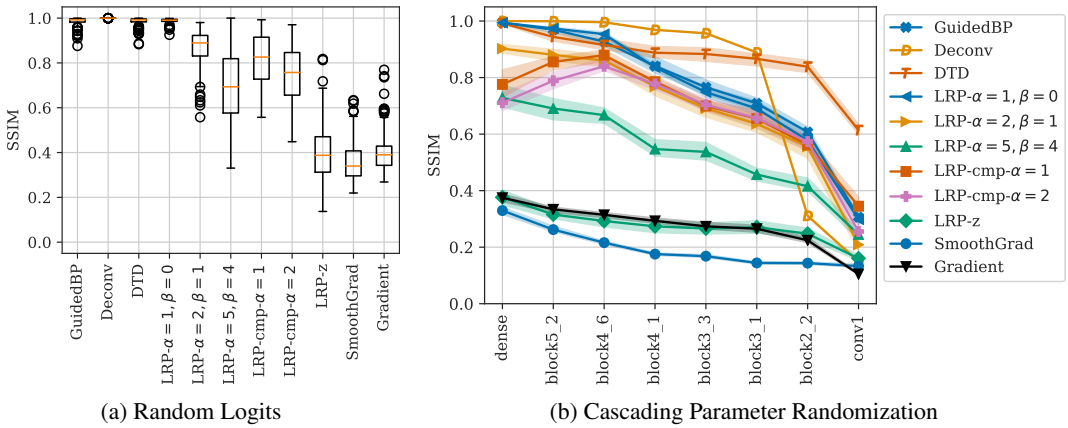


Figure 6: Saliency maps on a ResNet-50.



(a) Random Logits

(b) Cascading Parameter Randomization

Figure 7: Effect of (a) randomizing the logits or (b) the parameters on a ResNet-50.