Explaining deep learning for identifying structures and biases

A Talk at: AI.SG summerschool 2020.

Joint work with J. Sun, W. Samek, S. Lapuschkin (nee Bach), G. Montavon, K.-R. Müller, and deserving others

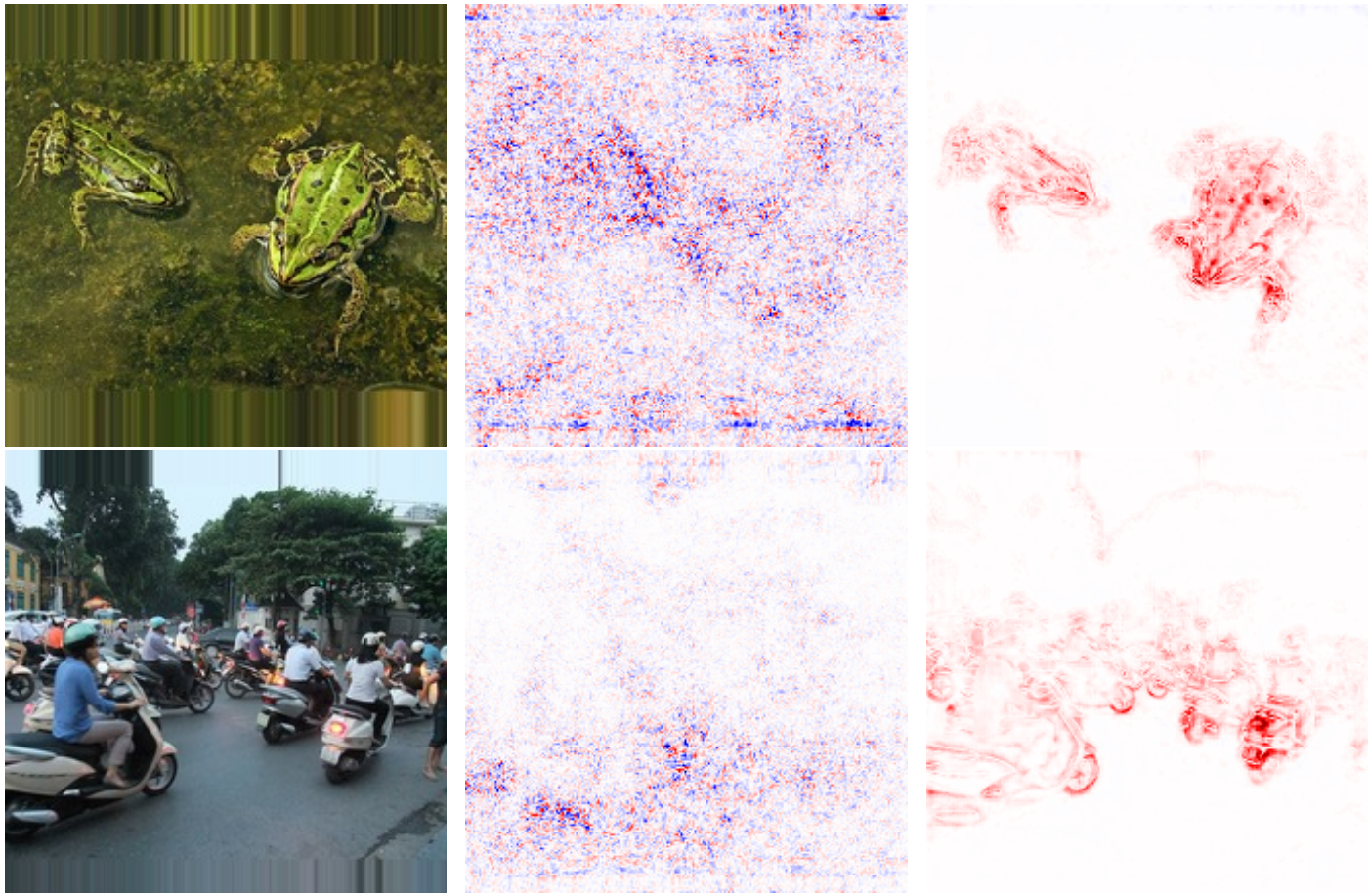
Alexander Binder

August 5, 2020

What is a possible explanation of a prediction? for images: (Densenet121, Keras+innvestigate, 2019)

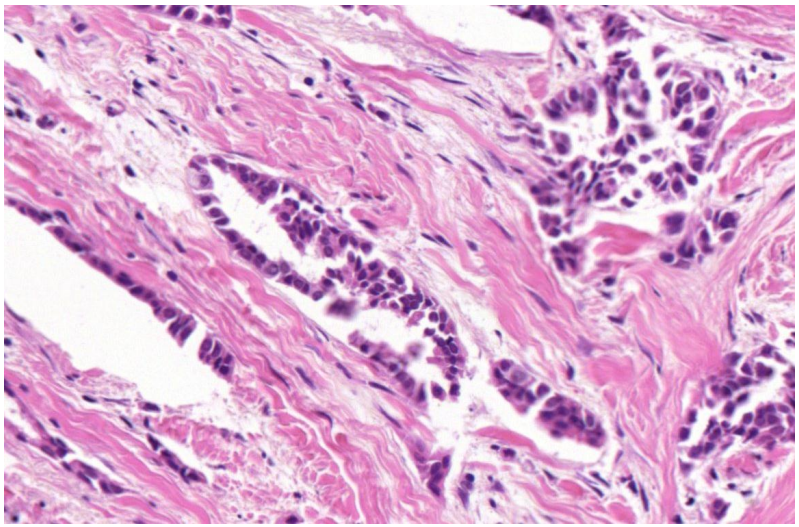
- case of images: compute a score for every pixel

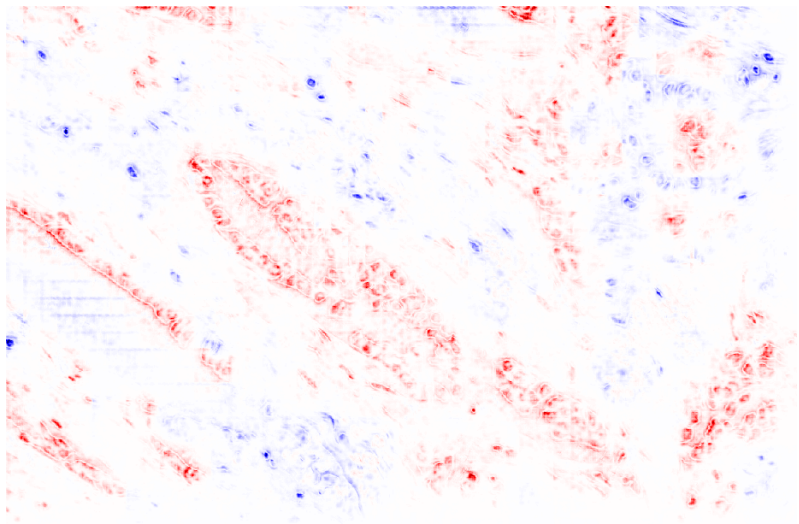
image | gradient | LRP-α-β



What is a possible explanation of a prediction?

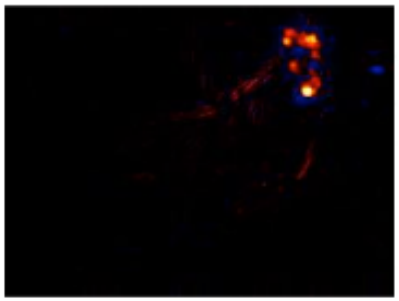
- case of images: compute a score for every pixel
  - patch-wise classification: label $= 1$ if patch contains breast cancer
  - pixel-wise explanation
- general case: score for every dim of an input sample
  $x = (x_1, \ldots, x_d, \ldots, x_D)$



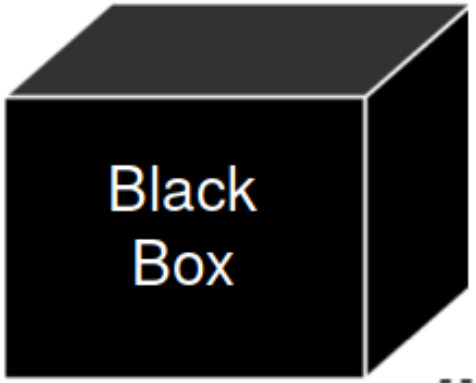


What is a possible explanation of a prediction?

input x



heatmap



Rooster

prediction $f(x)$

Idea: Decompose function

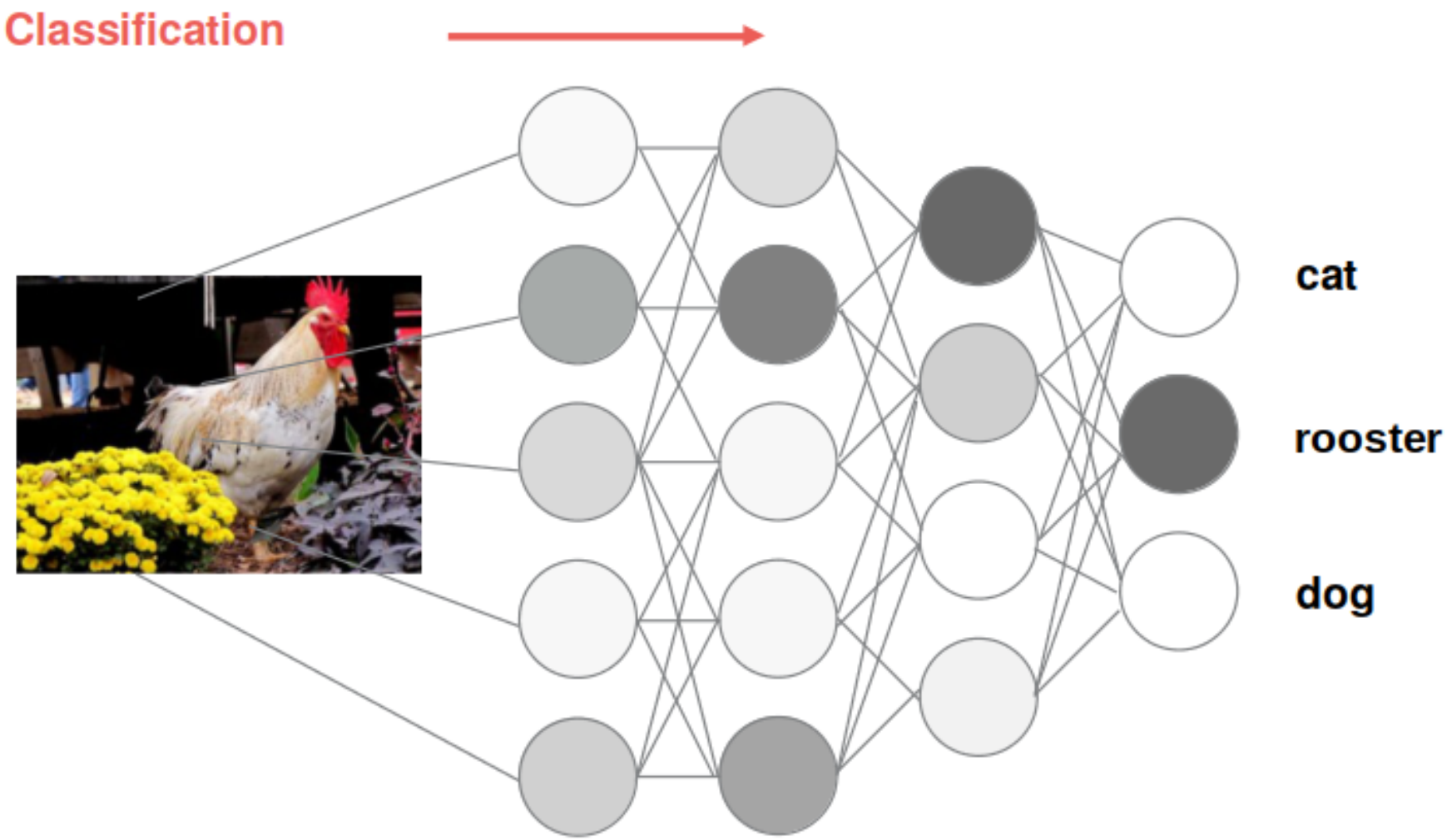
$$\sum_i R_i = f(x)$$

Explain prediction

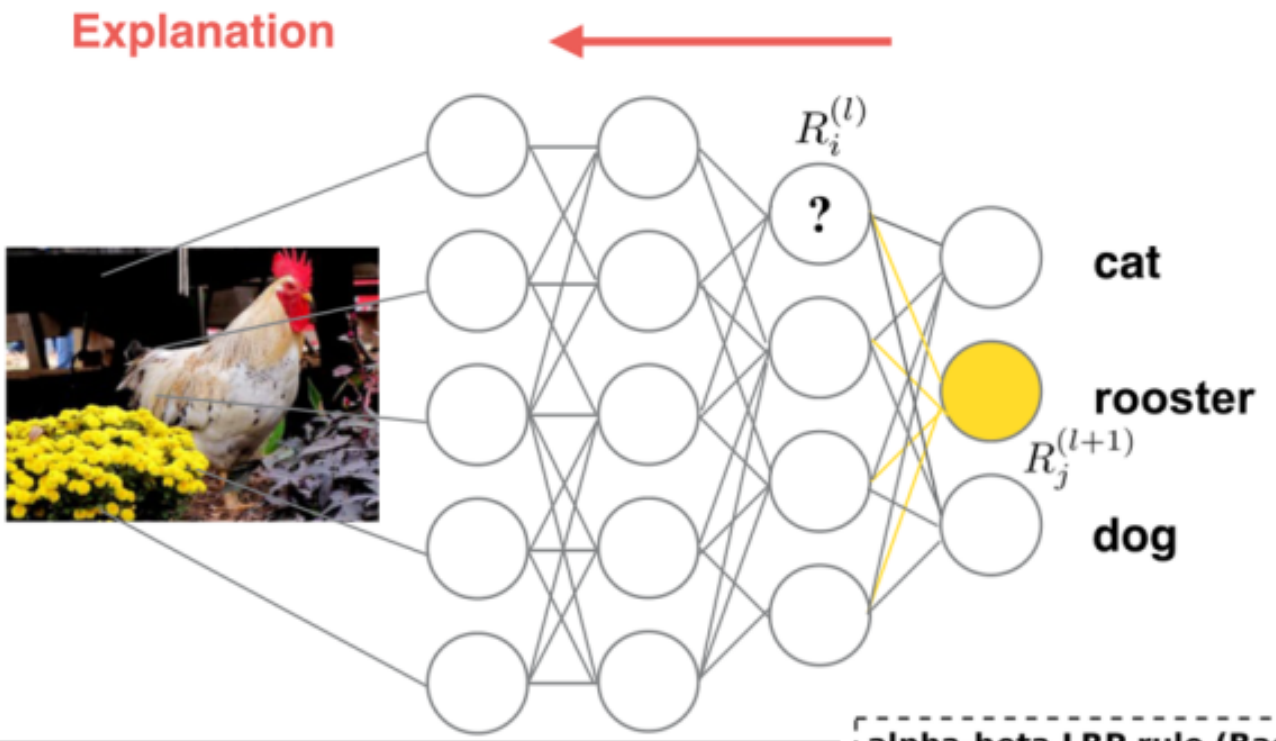
(how much each pixel contributes to prediction)

Layer-wise Relevance Propagation (LRP)
(Bach et al., PLOS ONE, 2015)

What is a possible explanation of a prediction?



What is a possible explanation of a prediction?



Theoretical interpretation
Deep Taylor Decomposition
(Montavon et al., 2017)

alpha-beta LRP rule (Bach et al. 2015)

$$R_i^{(l)} = \sum_j (\alpha \cdot \frac{(x_i \cdot w_{ij})^+}{\sum_{i'}(x_{i'} \cdot w_{i'j})^+} + \beta \cdot \frac{(x_i \cdot w_{ij})^-}{\sum_{i'}(x_{i'} \cdot w_{i'j})^-}) R_j^{(l+1)}$$

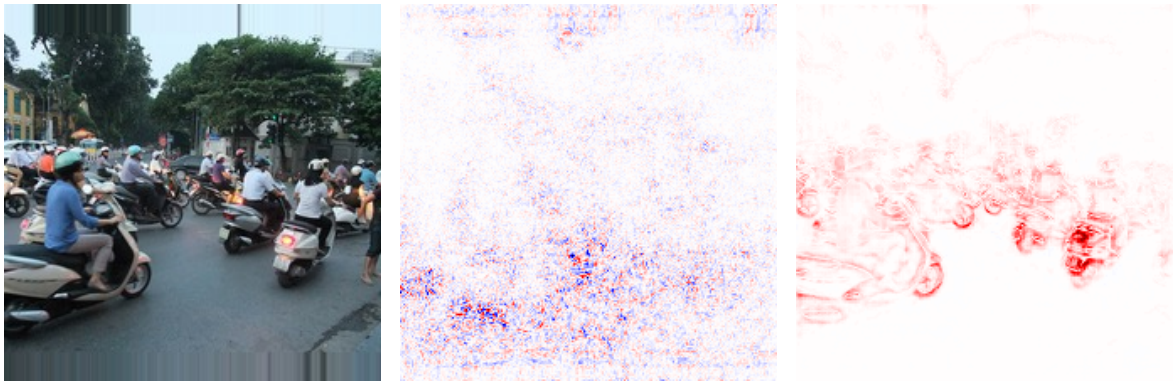
where $\alpha + \beta = 1$

What an explanation in the sense of LRP? (Densenet121, Keras+innvestigate, 2019)

- given: A. trained model f, B. a prediction $f(x)$ for input $x = (x_1, \ldots, x_d, \ldots, x_D)$.
- general case: To compute a relevance score $r_d(x)$ for every input dimension x_d of input x explaining the prediction $f(x)$, such that approximately:

$$f(x) \approx \sum_{d=1}^{D} r_d(x) \quad \leftarrow \text{ decomposition with constraints} \tag{1}$$

image gradient LRP-α-β



Trivial rules

Given $f(x)$, can obtain desired decomposition

$$f(\mathbf{x}) = \sum_{d=1}^{D} r_d(\mathbf{x}) \text{ by e.g.} \tag{2}$$

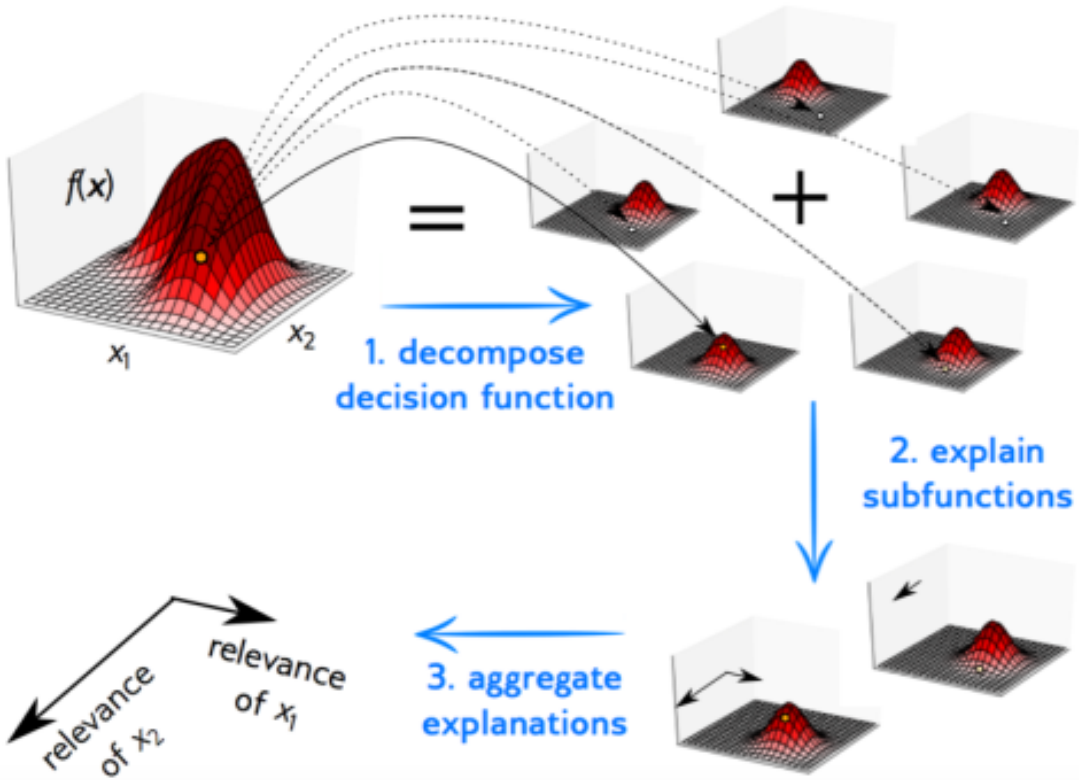
$$r_d(\mathbf{x}) = f(\mathbf{x})/D \tag{3}$$

$$r_d(\mathbf{x}) = \begin{cases} f(\mathbf{x}) & d = 1 \\ 0 & \text{else} \end{cases} \tag{4}$$

- underdetermined, many non-plausible decompositions
- need additional constraints
- theoretical foundation yielding constraints: Deep Taylor framework
  - Taylor decomposition of every single neuron with customized root points.

Deep Taylor Decomposition

LRP's idea: To robustly explain a model, leverage the neural network structure of the decision function.



Each explanation step:
- easy to find good root point
- no gradient shattering

(Montavon et al., 2017
Montavon et al. 2018)

Alternatives

No method is better than all others on all reasonable use cases.

- Gradient
- if the performance measure is sensitivity under occlusion of a single dimension/pixel, then you need only the gradient!

Alternatives

No method is better than all others on all reasonable use cases.

- Integrated Gradient, gradient times input
- Noisy heatmaps, suffers in ReLU networks from gradient shattering.
- noisy: poor consistency to smoothness measures based on keypoint detectors applied to heatmaps
- outperformed by LRP for larger blocks under evaluation measure: block-wise sequential randomized occlusion
- IG does get better with many roots used.

Alternatives

No method is better than all others on all reasonable use cases.

- LIME
- surely suitable if problem is low-dimensional
- local linear approximation. Any local linear approximation may suffer for ReLU networks due to gradient shattering, if the sampling radius is too small.
- need to evaluate sampling radius carefully (some eval measure)

Alternatives

No method is better than all others on all reasonable use cases.

- using trees - nothing wrong!
- moves the interpretation problem to (1) making sense of decision hierarchies, and (2) simplifying logical AND-clauses.

Alternatives

No method is better than all others on all reasonable use cases.

- learned occlusion masks! (Fong et al.)
- useful if you need segmentation type labels of what you need to remove for switching a class.

Alternatives

No method is better than all others on all reasonable use cases.

- Shapley-value methods. Can be better than LRP!
- Good whenever one has a clean notion of dropping a feature (tabular finance data). Fast if low-dimensional problem.
- Scale-choice needs to be evaluated: using Shap naively to drop single features may ignore in the measure correlations between feature blocks. e.g. when $x_2 > x_3 > 0$, then $x_3 = 0$ fakes a non-sensitivity.

$$x_1 + \max(x_2, x_3) \tag{5}$$

- dropping a feature is ill-defined in images, temporal sequence data, some graph data and NLP/Language. Not a fault of Shap*, its a problem of its assumptions.
  - force-dropping elements can create unplausible outliers, and result in explanations relative to outliers

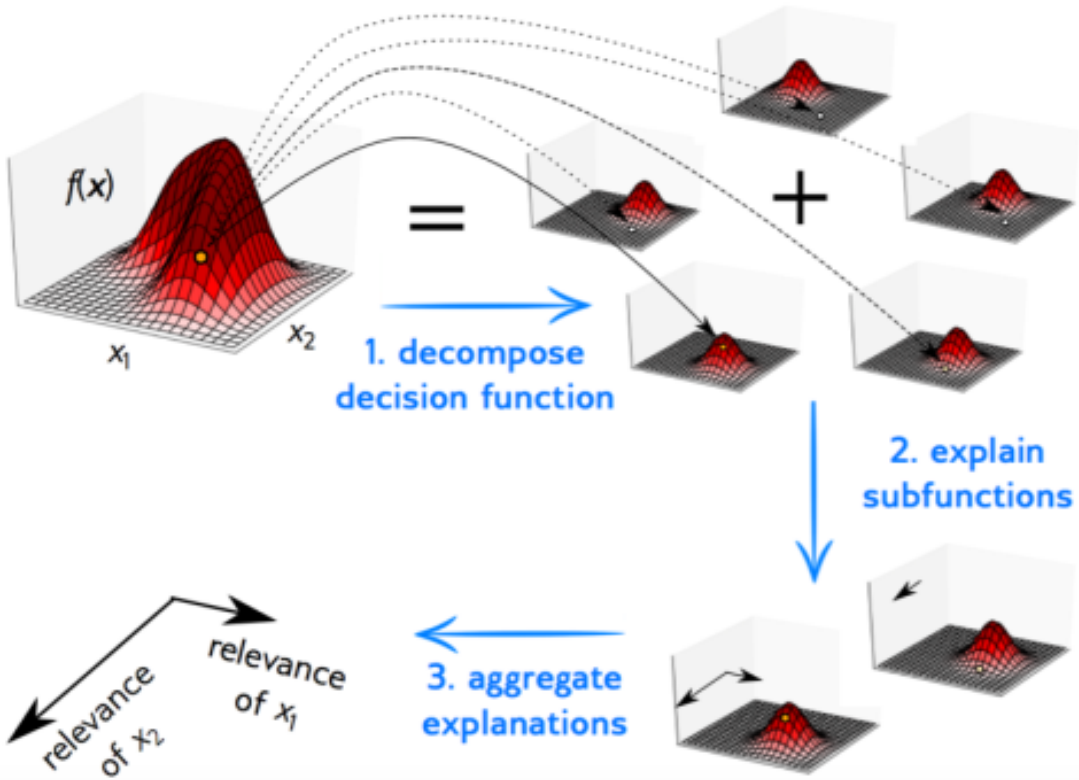
Alternatives

No method is better than all others on all reasonable use cases.

- LRP/Deep Taylor
- disadvantages: $(-)$ many parameters. technically one root per layer. Need to validate what to choose for each layer.
- $(-)$ using everywhere LRP-ϵ reduces to $\nabla \times \mathit{inp}$, can be unstable as $\nabla \times \mathit{inp}$
- $(+)$ works well for images with certain presets (ignore biases, use LRP-β, γ), RNNs such as LSTMs (!!), high dim data
- $(+)$ reasonably fast when properly implemented (Wressnegger et al.)

LRP: Deep Taylor Decomposition

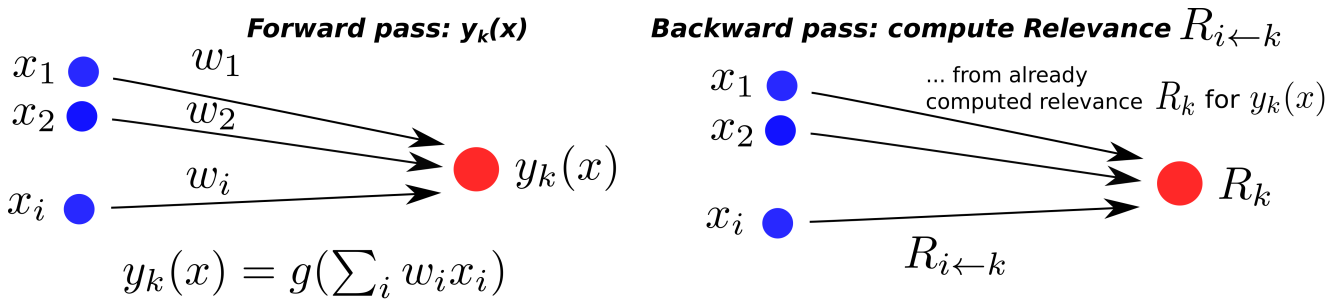
LRP's idea: To robustly explain a model, leverage the neural network structure of the decision function.



Each explanation step:
- easy to find good root point
- no gradient shattering

(Montavon et al., 2017
Montavon et al. 2018)

Relevance distribution for one neuron: example ϵ-rule

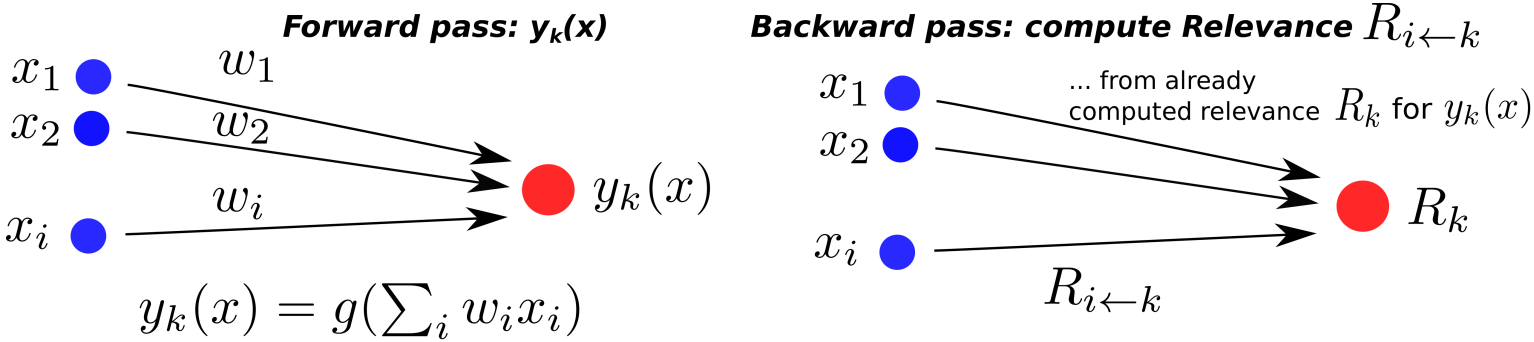


LRP-ϵ: given: have computed already R_k as relevance of neuron output $z_k = \sum_i w_{ik} x_i + b$,

$$
\begin{aligned}
R_{i\leftarrow k}(\mathbf{x}) &= R_k M_{i\leftarrow k} = R_k M_{i\leftarrow k}(w_{ik}, x_i) \\
R_{i\leftarrow k}(\mathbf{x}) &= R_k \left(\frac{w_{ik} x_i}{z_k + \epsilon \text{sign}(z_k)} \right) \qquad (6) \\
z_k &= \sum_{i'} w_{i'k} x_{i'} + b
\end{aligned}
$$

- may produce $R_{i\leftarrow k}$ with $|R_{i\leftarrow k}| \gg |R_k|$, no control over relevance scale!
- ϵ - dampens redistribution differences
- $\epsilon \to \infty$ convergence to flat redistribution
- not suitable for convolutions, works well for LSTMs (check out Leila Arras' works)

Relevance distribution for one neuron: example β-rule



LRP-β: given: have computed already R_k as relevance of neuron output $z_k = \sum_i w_{ik} x_i + b$,

$$R_{i\leftarrow k}(\mathbf{x}) = R_k M_{i\leftarrow k} = R_k M_{i\leftarrow k}(w_{ik}, x_i)$$

$$R_{i\leftarrow k}(\mathbf{x}) = R_k \left((1+\beta) \frac{(w_{ik}x_i)_+}{\sum_{i'} (w_{i'k}x_{i'})_+ + b_+} - \beta \frac{(w_{ik}x_i)_-}{\sum_{i'} (w_{i'k}x_{i'})_- + b_-} \right) \tag{7}$$

- β controls ratio of negative to positive evidence.
- bounded relevance scale: $|R_{i\leftarrow k}| \leq (1+\beta)|R_k|$
- negative to total evidence: $\frac{\beta}{1+2\beta} \overset{\beta\to\infty}{\to} 0.5$,
  It is fixed independent of network inputs(!).
- good for conv-layers

Relevance computation for one neuron

Got $R_{i \leftarrow k}$ from R_k. How to compute R_i ?

$$R_i := \sum_{k: i \text{ is input to } k} R_{i \leftarrow k} \tag{8}$$

which rule for which layer?

Name	Formula	layers
LRP-ϵ	$\sum_k R_k \left(\frac{x_i w_{ik}}{\sum_i x_i w_{ik} + b + \epsilon \text{sign}(z)} \right)$	fully connected
LRP-$\beta = 0$	$\sum_k R_k \left(\frac{(x_i w_{ik})_+}{\sum_i (x_i w_{ik})_+ (b)_+} \right)$	conv
LRP-γ	$\sum_k R_k \left(\frac{\gamma (x_i w_{ik})_+ + (x_i w_{ik})}{\sum_i \gamma (x_i w_{ik})_+ + \gamma (b)_+ + \sum_i (x_i w_{ik}) + b} \right)$	conv
LRP-z_β	$\sum_k R_k \left(\frac{x_i w_{ik} - l_i (w_{ij})_+ + h_i (w_{ij})_-}{\sum_i x_i w_{ik} + b - l_i (w_{ij})_+ + h_i (w_{ij})_-} \right)$	first conv layer
LRP-w^2	$\sum_k R_k \frac{w_{ik}^2}{\sum_i w_{ik}^2}$	same 1. conv

Its a mere serving suggestion. Define a loss and measure the quality of your explanations and choose by that!

Gradient $\times$ Input?

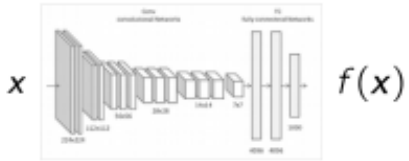
Motivation

- Compute an explanation in a single pass without having to optimize or search for a root point.

Gradient x Input

$$\forall_i : \; R_i = [\nabla f(\boldsymbol{x})]_i \cdot x_i$$

$$\boldsymbol{R} = \nabla f(\boldsymbol{x}) \odot \boldsymbol{x}$$



Gradient × Input?

Observation: Complex analyses reduce to gradient x input for simple cases.

Perturbation Analysis

$$f(\boldsymbol{x}) = \sum_{i=1}^{d} x_i w_i + b$$

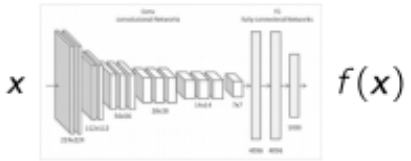
Taylor Expansions

$$\forall_{\boldsymbol{x}, t \geq 0} : \; f(t\boldsymbol{x}) = t f(\boldsymbol{x})$$

Gradient x Input

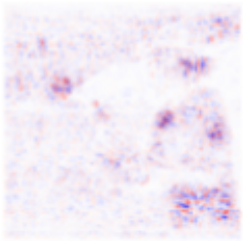
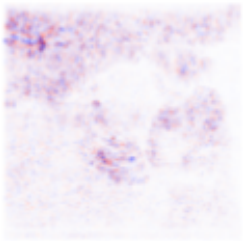
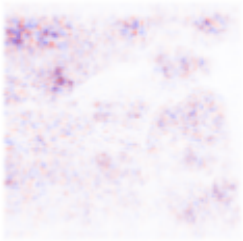
$$\forall_i : \; R_i = [\nabla f(\boldsymbol{x})]_i \cdot x_i$$

$$\boldsymbol{R} = \nabla f(\boldsymbol{x}) \odot \boldsymbol{x}$$



Question: Does it work in practice?

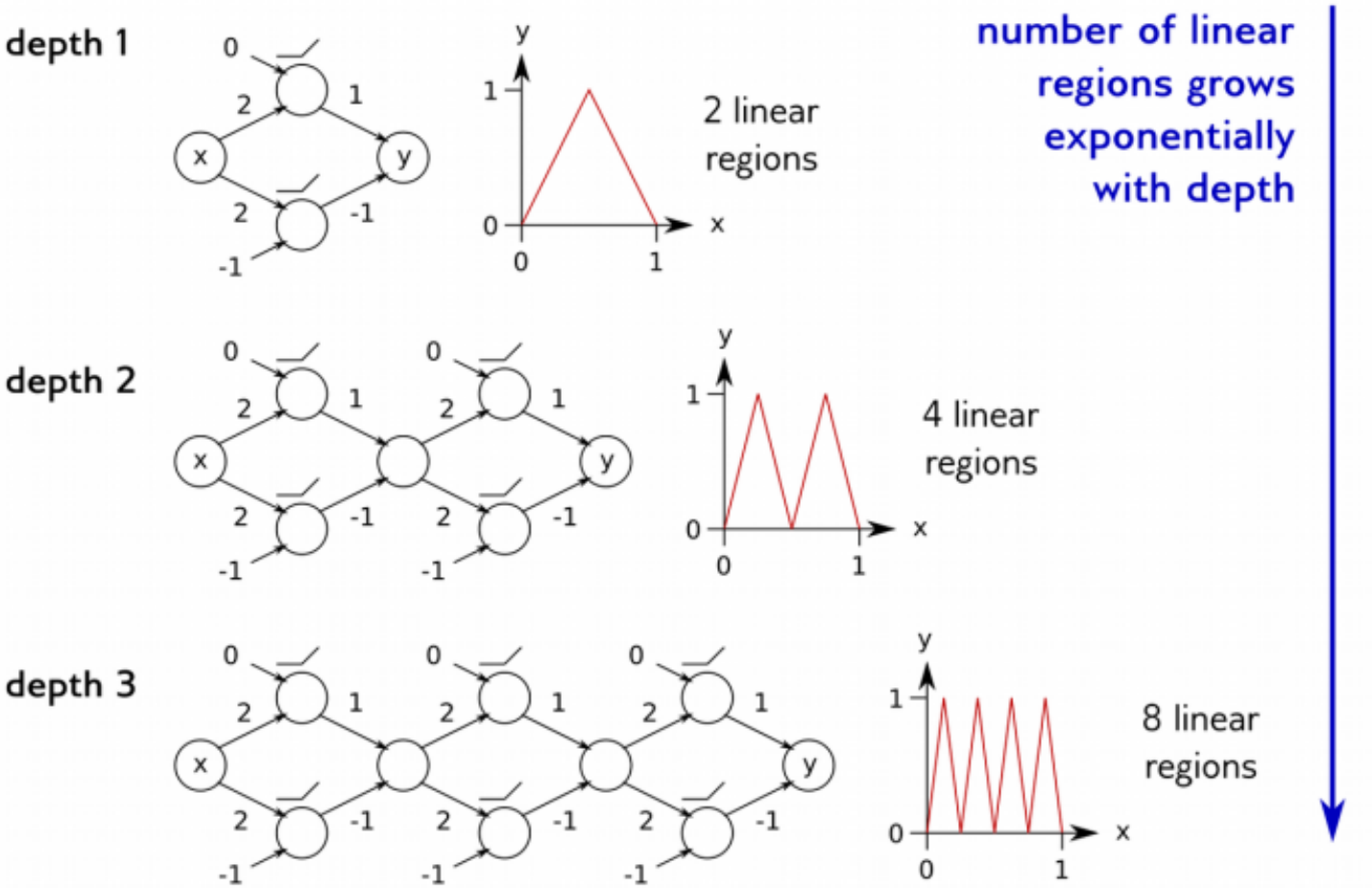
Gradient $\times$ Input?

Input	Model	Explanation
	VGG-16	
	Inception V3	
	ResNet 50	

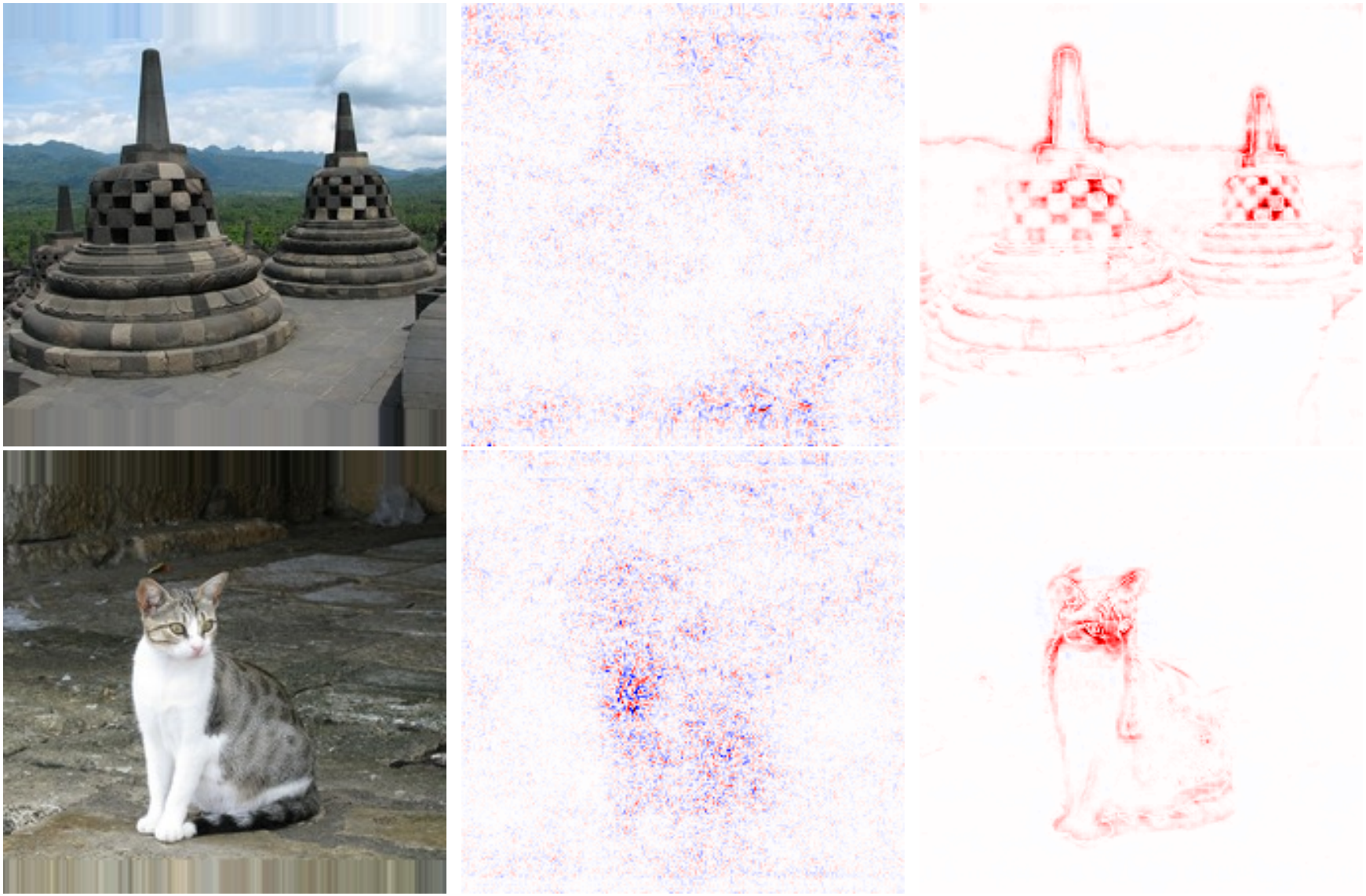
Observation:
Explanations are noisy.

Gradient $\times$ Input?

The Shattered gradients problem [Montufar'14, Balduzzi'17]

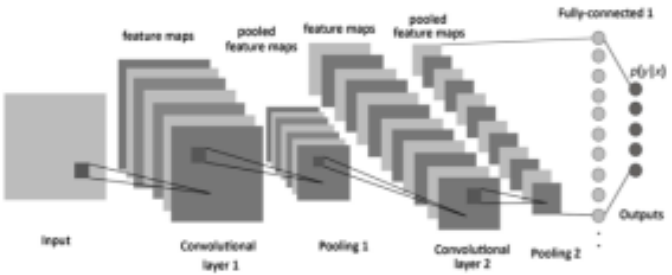


Examples (Densenet121, Keras, 2019)

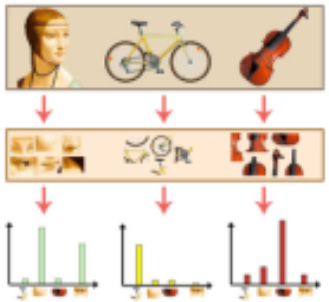


LRP Applied to Variety of Models

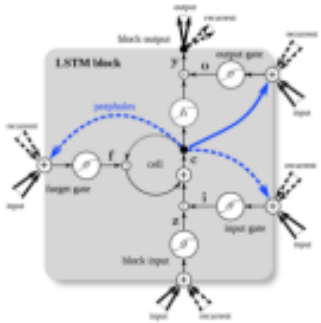
Convolutional NNs
(Bach'15, Binder'16, Arras'17 ...)



BoW models
(Bach'15, Lapuschkin'17 ...)

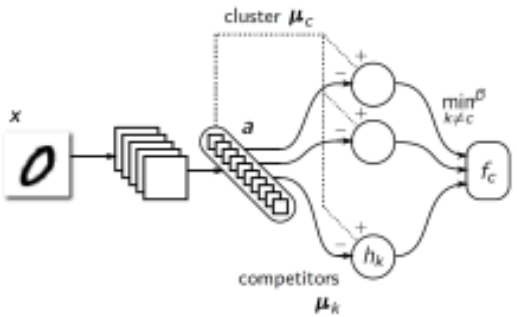


LSTM
(Arras'17, Arras ...
Hochreiter et al. 2019)



One-class SVM
(Kauffmann'18)

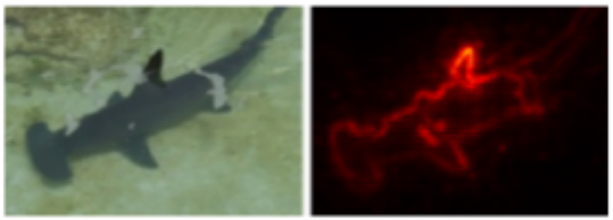
Clustering
(Kauffmann'19)



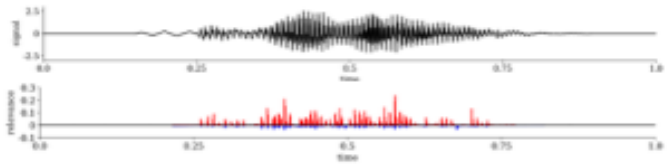
explaining unsupervised learning

LRP Applied to Variety of Tasks

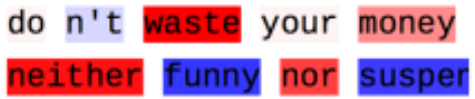
General Images (Bach' 15, Lapuschkin'16)



Speech (Becker'18)

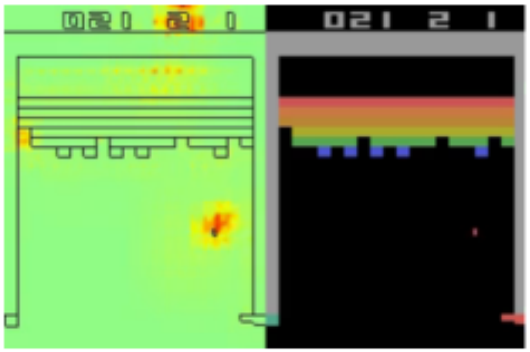


Text Analysis (Arras'16 &17)



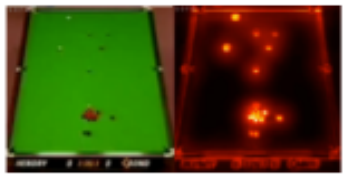
Morphing (Seibold'18)

Games (Lapuschkin'19)

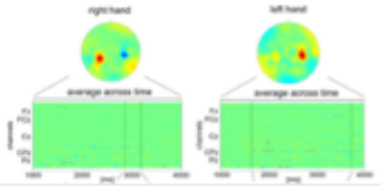


VQA (Samek'19)

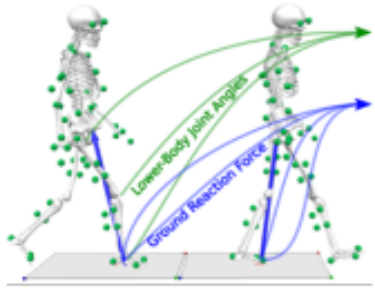
Video (Anders'18)



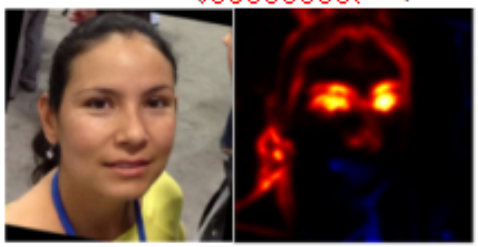
EEG (Sturm'16)



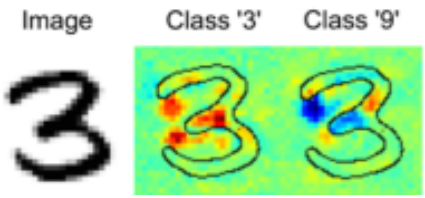
Gait Patterns (Horst'19)



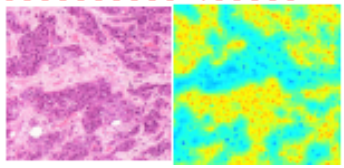
Faces (Lapuschkin'17)



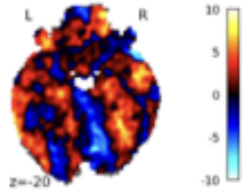
Digits (Bach' 15)



Histopathology (Hägele'19)



fMRI (Thomas'18)



Failing Axiomatic Requirements

e.g. Sanity Checks paper Adebayo et al. Neurips 2018 `https://papers.nips.cc/paper/8160-sanity-checks-for-saliency-maps.pdf`
Good paper. Personal opinion: *defining and evaluating a loss* in practice $>$ axiomatic requirements.

- Gradient passes sanity checks but estimates an often suboptimal measure: a single-pixel sensitivity instead of contributions which account for interactions between larger regions.
- Gradient: +high noise from gradient shattering in ReLU nets.
- For a **measurement-based comparison** of grad$*$ against guided backprop (also fails) in a medical context see eg. Eitel et.al. MICCAI 2019 `https://link.springer.com/chapter/10.1007/978-3-030-33850-3_1`
- in NLP: Poerner et al. ACL 2018, `https://www.aclweb.org/anthology/P18-1032.pdf`
- for RNNs: Arras et al. ACL 2019 BlackboxNLP Workshop, `https://arxiv.org/pdf/1904.11829.pdf`
- fail in parameter randomization test does not imply failure to explain current model at hand.

The value of explanations (not just LRP...)

A. Identifying action strategies in reinforcement learning predictors
B. Iterative Dataset Design (medical imaging): Identify what you need to label for the next round
C. Identifying and Removing biases: Spray for semi-automatic image debiasing
D. Sensing structure-induced habits – image captioning case: LRP detects words hallucinated from sentence structures/grammar/oversampling
E. improving model performance in small-sample size tasks: LRP-guided training to improve cross-domain few shot learning

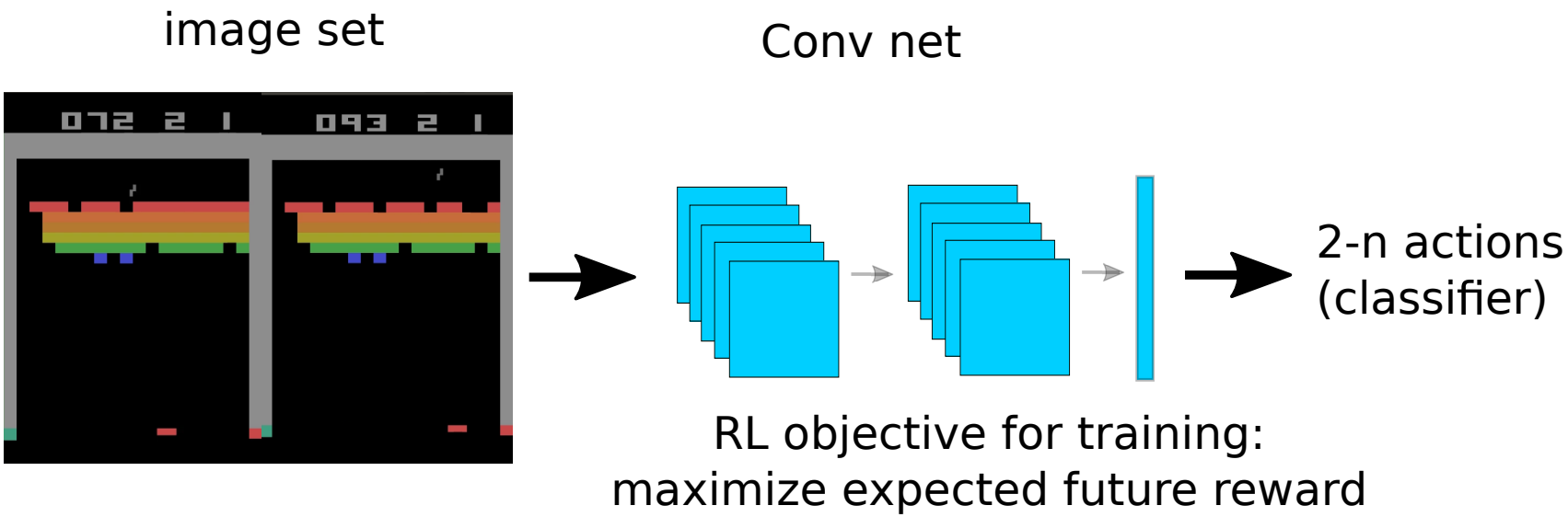
The value of explanations (not just LRP...)

Case A: Identifying action strategies in reinforcement learning predictors

LRP: DNN and Atari Breakout

A. application case: identify action strategies in reinforcement learning predictors

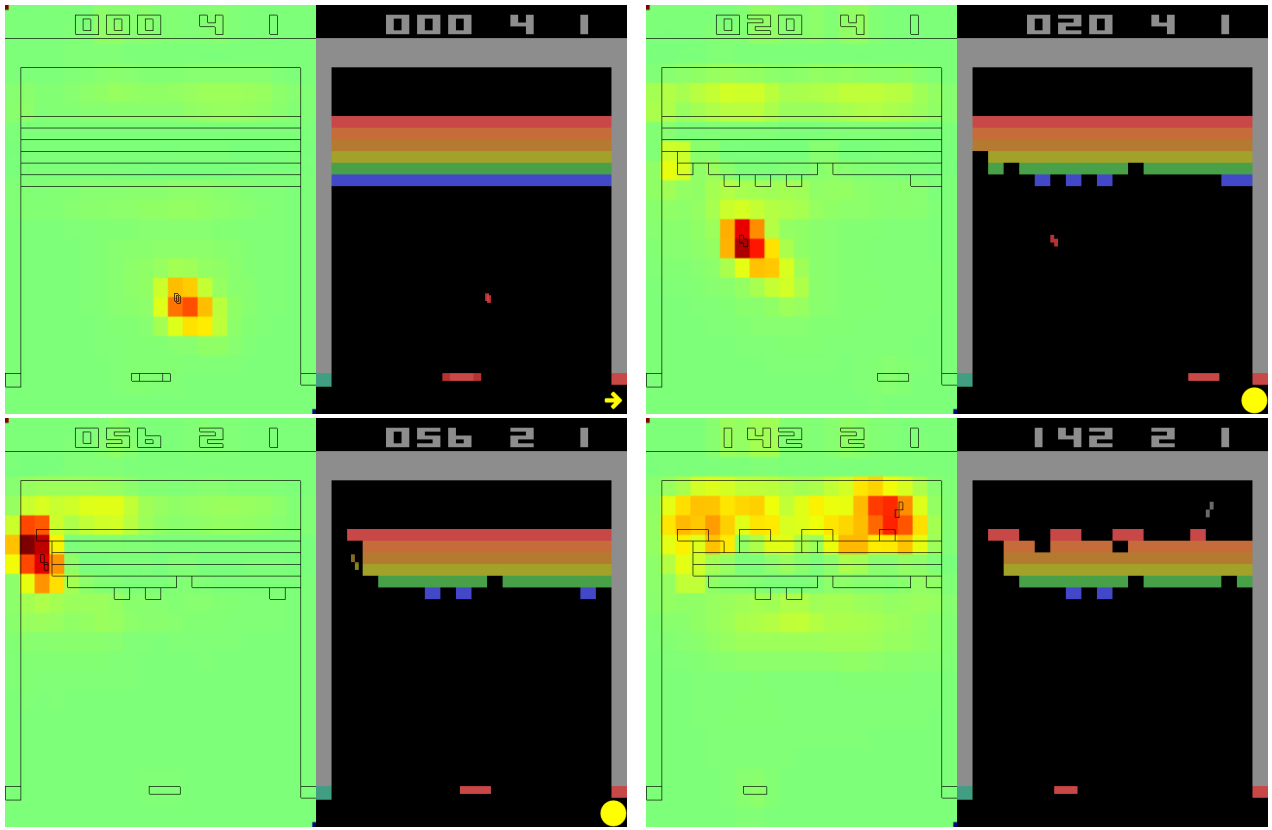
Trained a reinforcement learning classifier according to Mnih et al's Nature 2016 paper:
Volodymyr Mnih et al. Human-level control through deep reinforcement learning,
Nature 518, pages 529533, 2015



LRP: DNN and Atari Breakout

Trained a reinforcement learning classifier according to Mnih et al's Nature 2016 paper.

Explain a test game. LRP helps to discover strategies: building a tunnel.



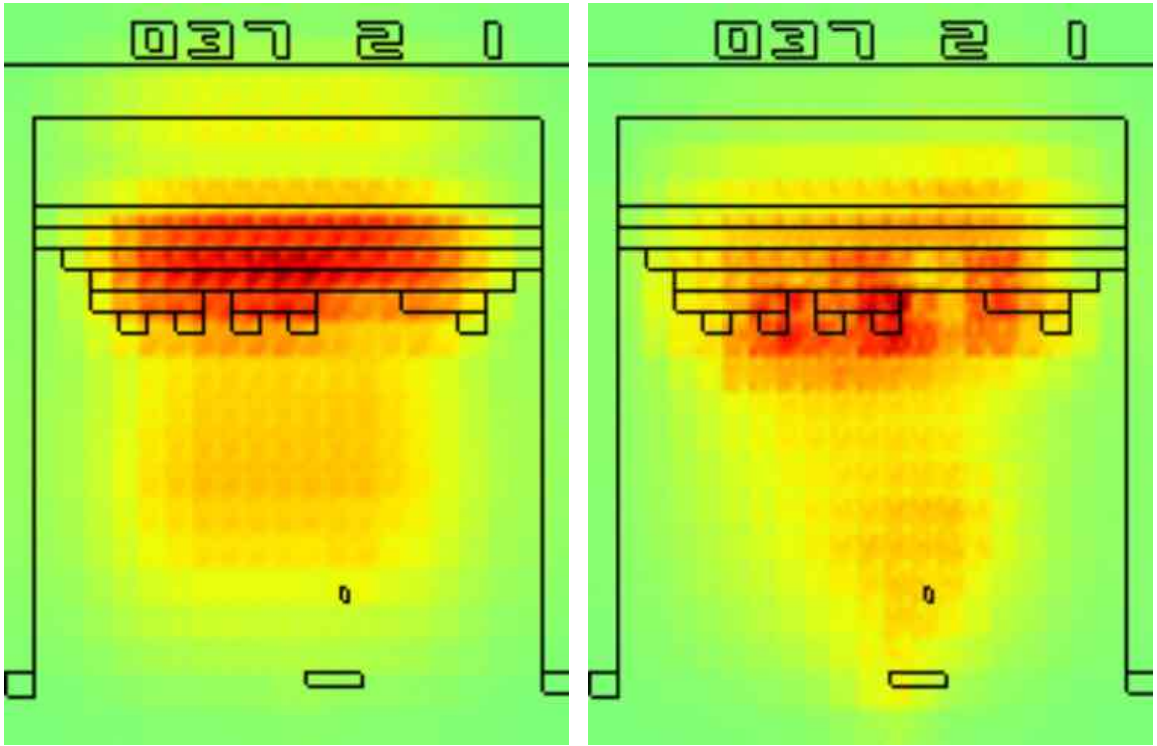
Lapuschkin et al., Unmasking Clever Hans predictors and assessing what machines really learn,

Nature Communications, 2019

LRP: DNN and Atari Breakout

Trained a reinforcement learning classifier according to Mnih et al's Nature 2016 paper.

LRP can help to discover strategies: building a tunnel - evolution of focus during training



epoch 0 and 6

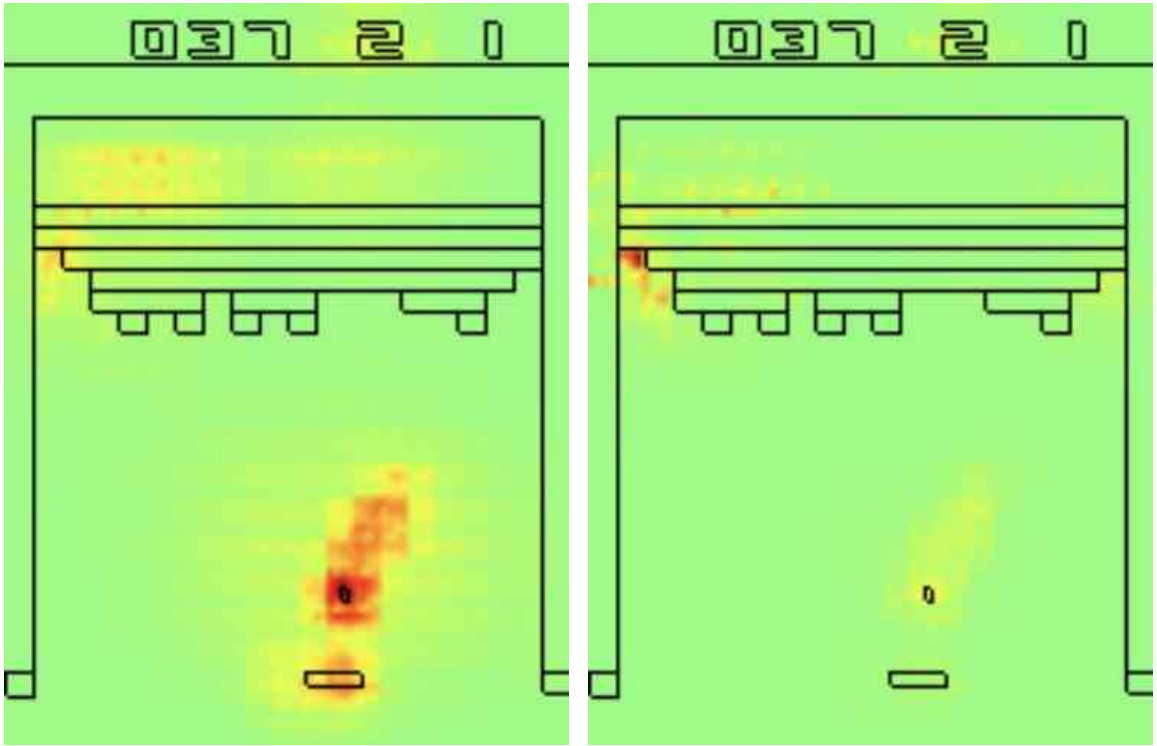
Lapuschkin et al., Unmasking Clever Hans predictors and assessing what machines really learn,

Nature Communications, 2019

LRP: DNN and Atari Breakout

Trained a reinforcement learning classifier according to Mnih et al's Nature 2016 paper.

LRP can help to discover strategies: building a tunnel - evolution during training



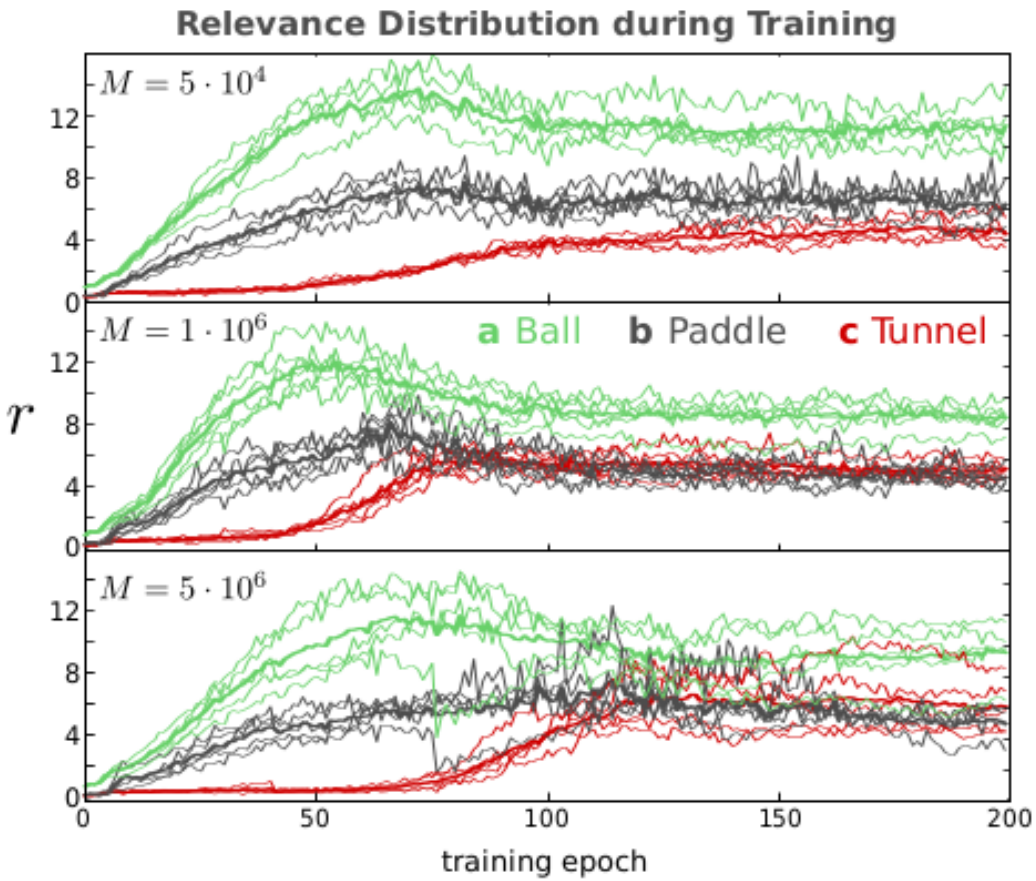
epoch 50 and 100

Lapuschkin et al., Unmasking Clever Hans predictors and assessing what machines really learn,

Nature Communications, 2019

LRP: DNN and Atari Breakout

LRP can help to find parameters for fast learning of known strategies. Here: impact of M = replay memory size



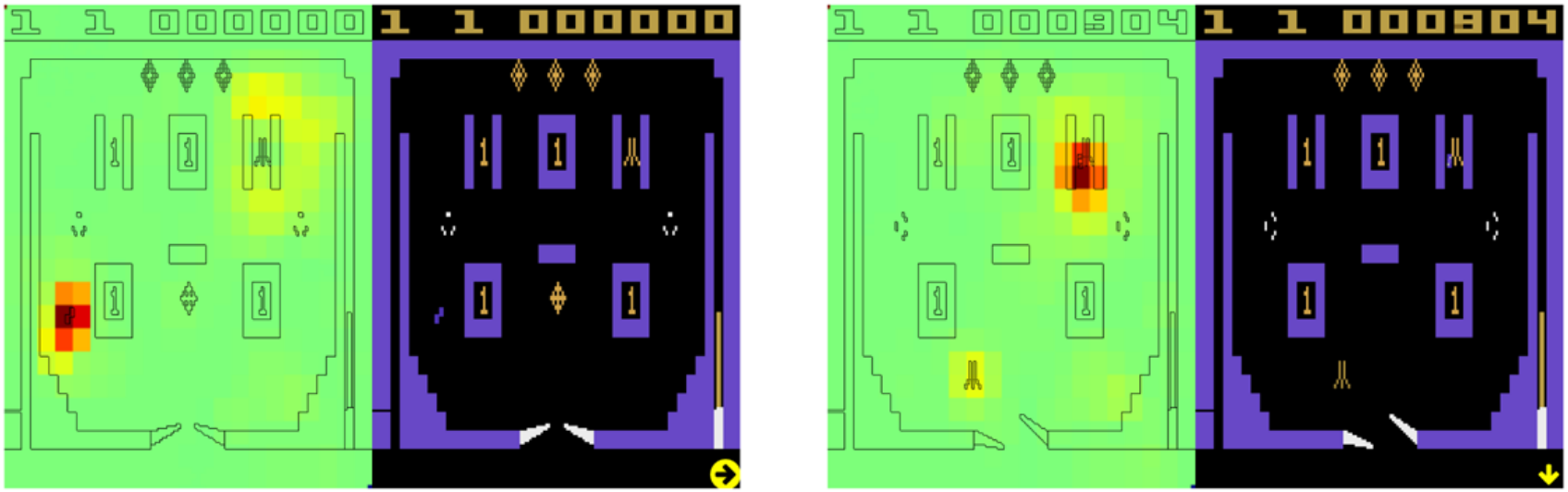
Lapuschkin et al., Unmasking Clever Hans predictors and assessing what machines really learn,

Nature Communications, 2019

LRP in reinforcement learning

Interpretability methods (here: LRP) can uncover complex relationships

Atari Pinball:



move ball 4 times over switch to activate a score multiplier.

.. if there are any

Lapuschkin et al., Unmasking Clever Hans predictors and assessing what machines really learn,

Nature Communications, 2019

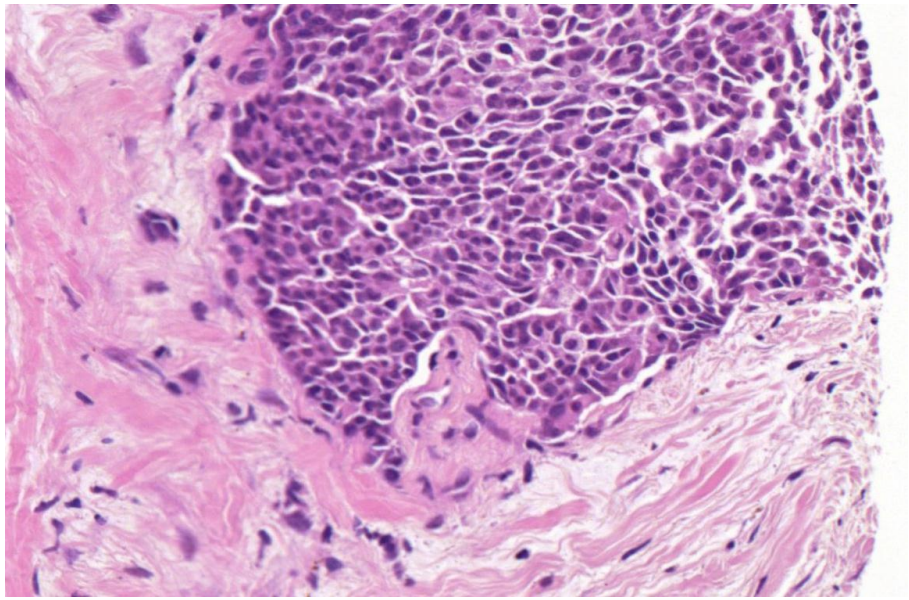
The value of explanations (not just LRP...)

Case B: Iterative Dataset Design (medical imaging):
Identify what you need to label for the next round

Iterative Dataset Design (Medical Data)

Why not just using test error ?

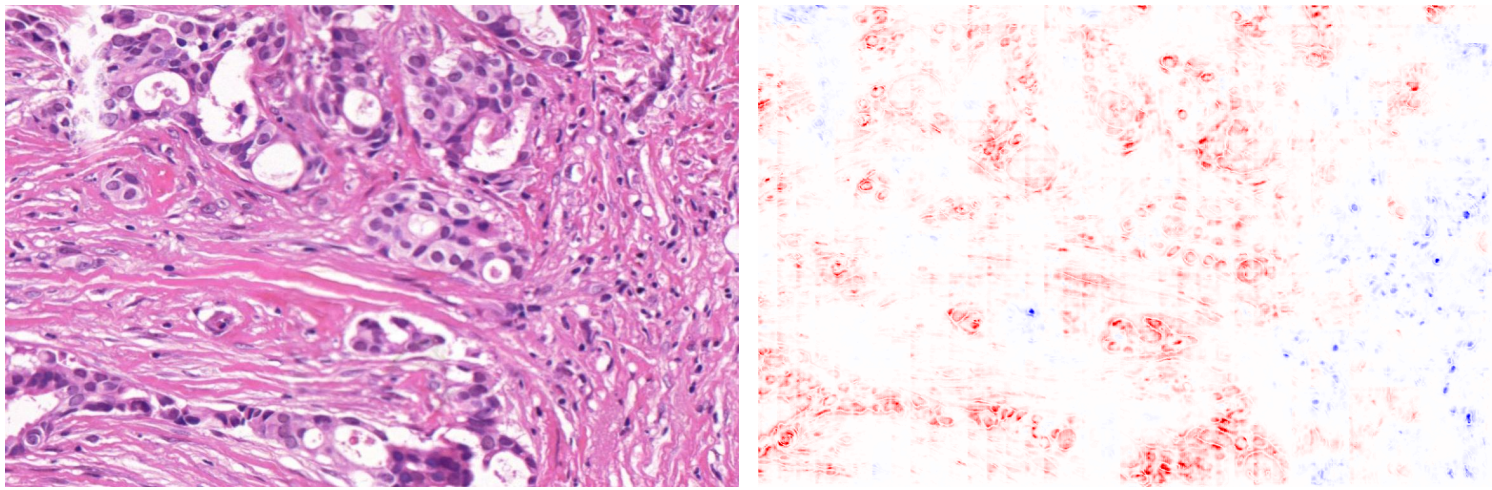
- some problems: labels very costly, unlabeled data abundant



Iterative Dataset Design (Medical Data)

More Importantly:

- decide what unlabeled data to add into next iteration of train and test set – precursor to labelling.

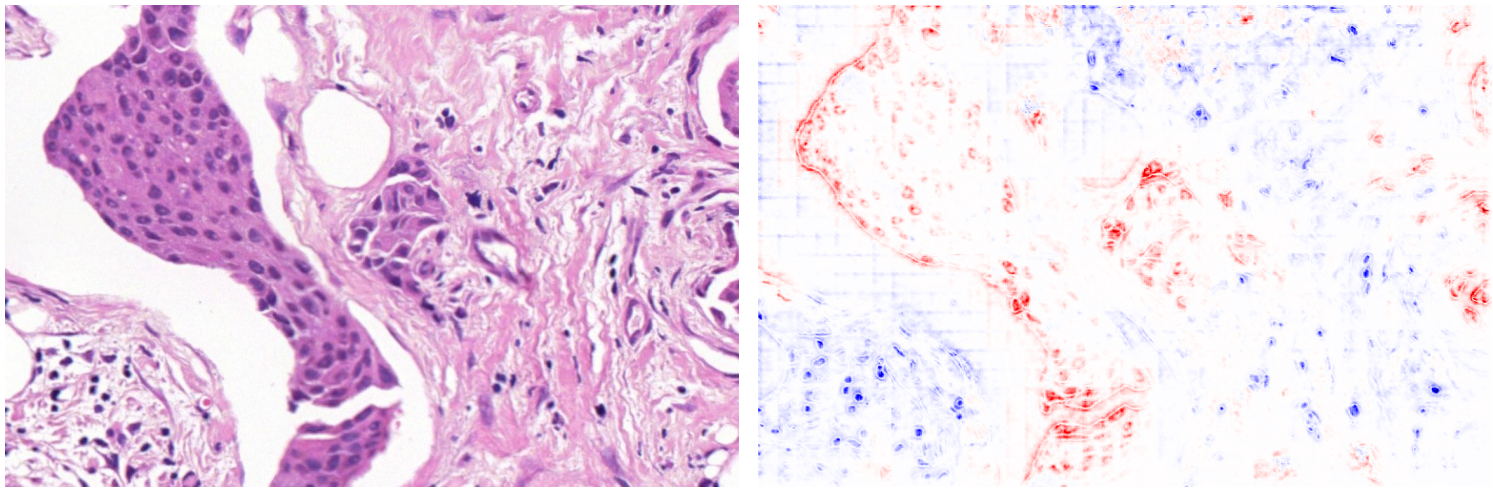


- Interpretability for efficiency in the selection step before labelling!

Iterative Dataset Design (Medical Data)

More Importantly:

- decide what unlabeled data to add into next iteration of train and test set – precursor to labelling.



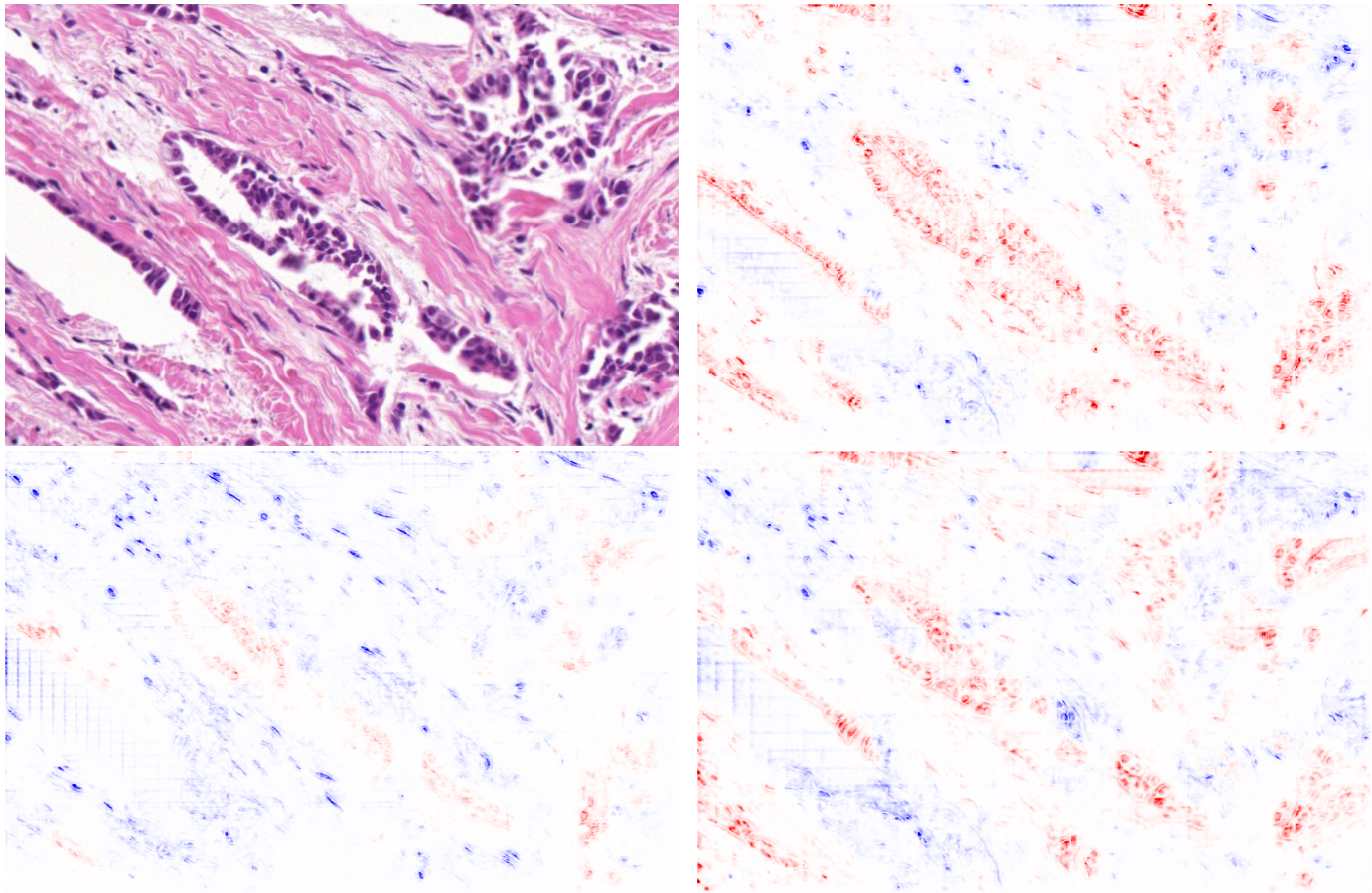
- Interpretability for efficiency in the selection step before labelling!

Iterative Dataset Design (Medical Data): Evaluate Impact of data augmentation

Image scaling ?

orig

80%

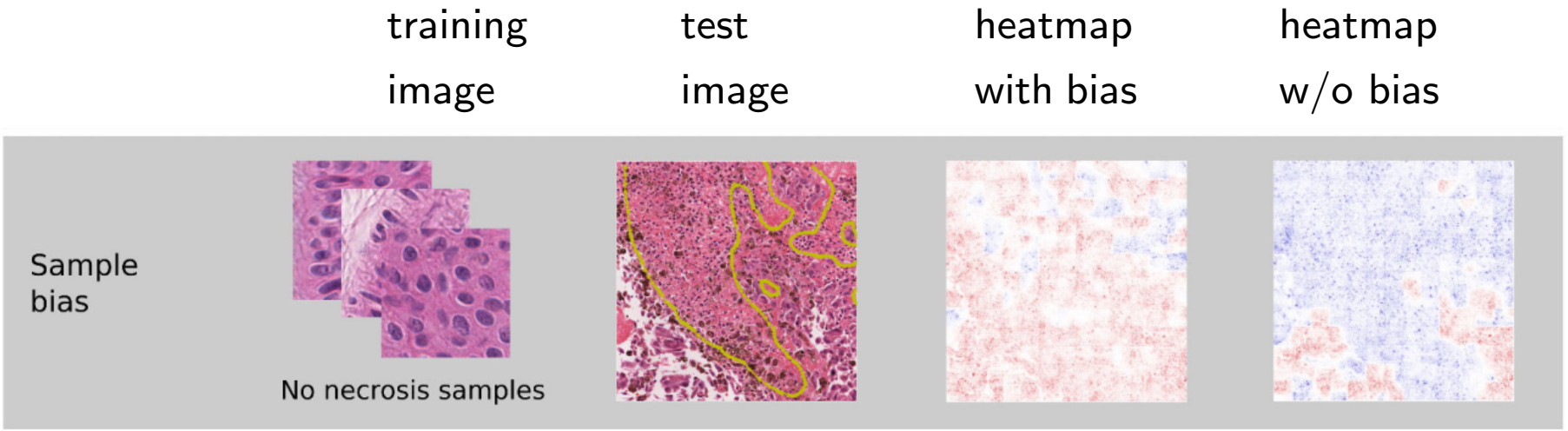


100%

66%

Iterative Dataset Design (Medical Data): what to label?

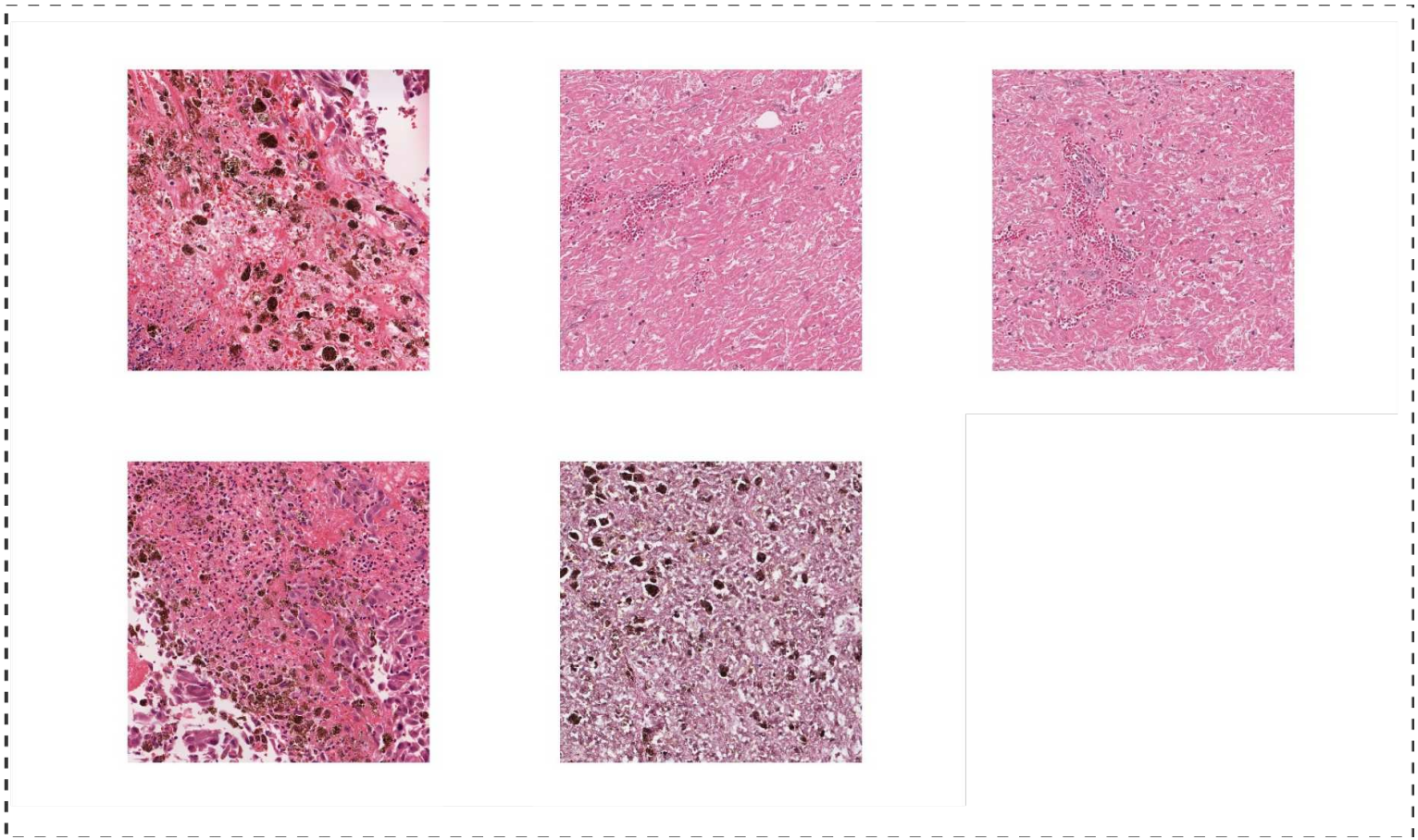
- left heatmap: false positive scores on unlabeled subclass.
- right heatmap: after augmenting training dataset with necrosis samples (labeled as negative)



Haegele et al., Resolving challenges in deep learning-based analyses of histopathological images using explanation methods, Nat Sci Rep 2020

Iterative Dataset Design (Medical Data): what to label?

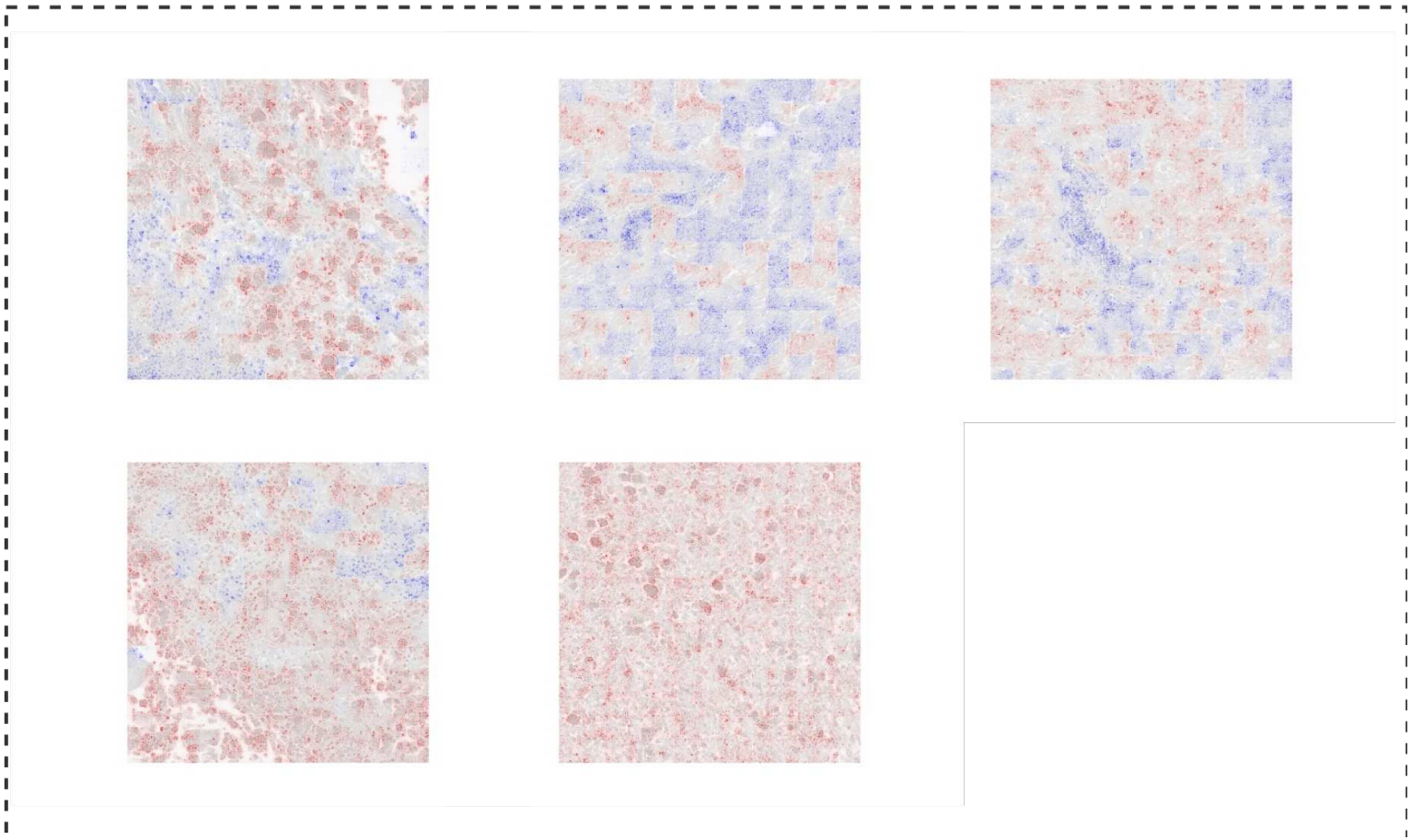
Original HE images:



Haegele et al., Resolving challenges in deep learning-based analyses of histopathological images using explanation methods, Nat Sci Rep 2020

Iterative Dataset Design (Medical Data): what to label?

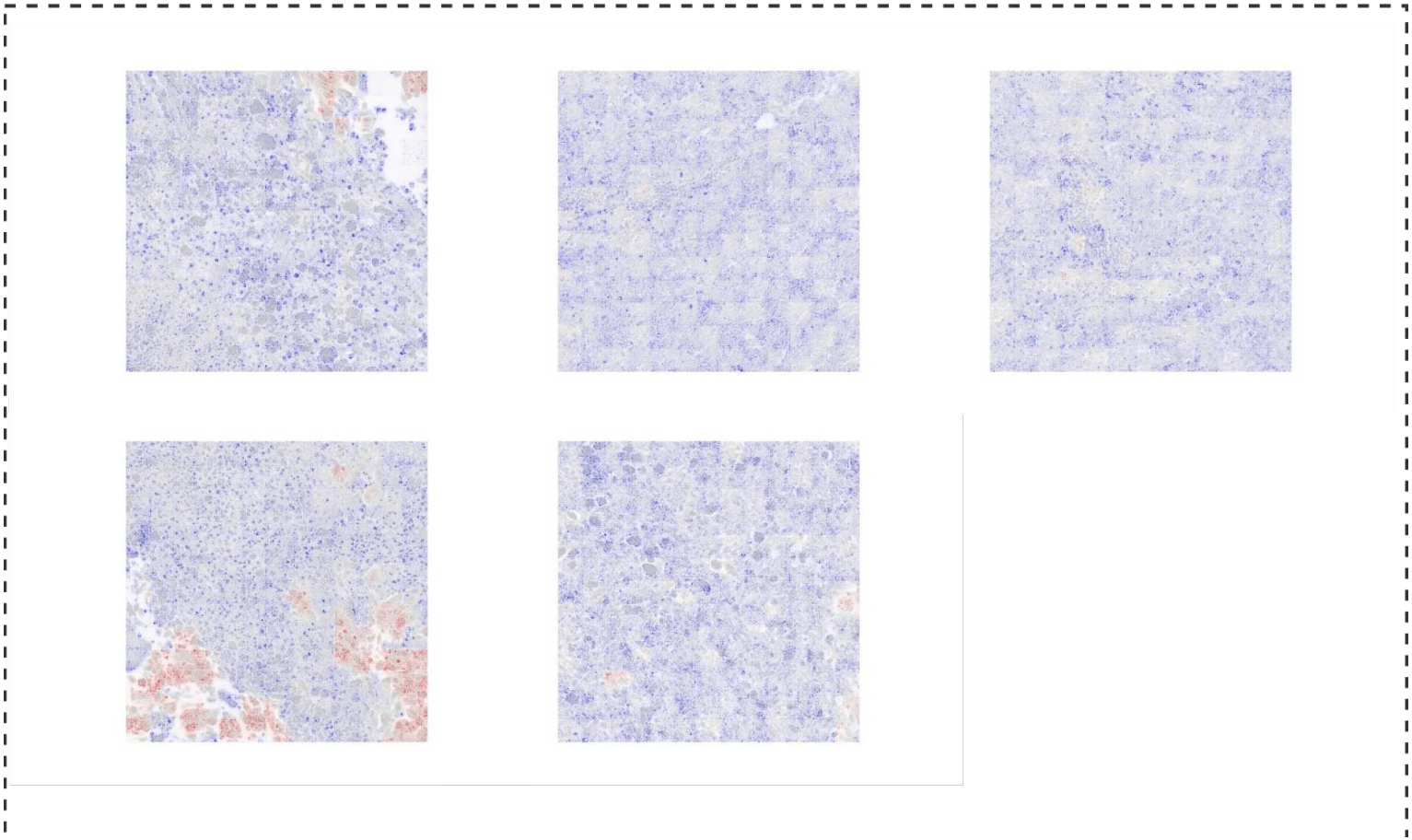
Training **without** necrosis samples.



Haegele et al., Resolving challenges in deep learning-based analyses of histopathological images using explanation methods, Nat Sci Rep 2020

Iterative Dataset Design (Medical Data): what to label?

Training **with** necrosis samples.



your version1 labels and test set error cannot discover it

Haegele et al., Resolving challenges in deep learning-based analyses of histopathological images using explanation methods, Nat Sci Rep 2020

The value of explanations (not just LRP...)

Case C: Identifying and Removing biases: Spray for semi-automatic image debiasing

`https://arxiv.org/pdf/1912.11425.pdf`
Analyzing ImageNet with Spectral Relevance Analysis:Towards ImageNet un-Hans'ed
CJ Anders, T Marinc, D Neumann, W Samek,K-R Müller, S Lapuschkin
See slides from Sebastian Lapuschkin.

Spray for semi-automatic image debiasing

Idea:

- find a way for efficient fishing within a large number of classes (imagenet classes) for those classes which have a strange subset of heatmaps / explanation strategies
- analyze every imagenet class separately
- goal: find/rank those imagenet classes which have an isolated cluster of heatmaps, indicating an outlier explanation strategy

Spray for semi-automatic image debiasing

approach:

- select a classifier, compute heatmaps for every image (size: $(224, 224)$)
- optional step: pool those heatmaps down to $(20, 20)$
- compute pairwise distance between two heatmaps – for all pairs, but only among those of the one class you are analyzing
  - euclidean distance
  - gromov-wasserstein distance `https://optimaltransport.github.io/slides-peyre/GromovWasserstein.pdf` slide 5
- result: distance matrix e_{ij}, $x_i, x_j \in$ one fixed class
- **discretize distance matrix:**
- compute binarized similarity w_{ij} between heatmaps of samples i and j using $k = \log$ sample size

$$w_{ij} = \begin{cases} 1 & \text{if } i \text{ is among the } k\text{-nearest neighbors of } j \text{ according to } e_{ij} \end{cases} \quad (9)$$

- symmetrize A: $A_{ij} = 0.5(W_{ij} + W_{ij})$

Spray for semi-automatic image debiasing

approach:

- obtained so far: turned heatmaps $h_i \in \mathbb{R}^{224 \times 224}$ into a symmetric discretized (values $0, 0.5, 1$)similarity matrix A_{ij} for all pairs of samples from within one class

Spray for semi-automatic image debiasing

next steps:

- have symmetric similarity A_{ij}, $x_i, x_j \in$ one fixed class
- compute a low dimensional embedding ϕ_i for x_i such that $\|\phi_i - \phi_j\|$ small when A_{ij} large.
- cluster ϕ_i into $k = 2, \ldots, 30$ clusters
- compute a separability score from ϕ_i, $x_i \in$ one fixed class, using the cluster assignments
- rank classes according to separability score, inspect classes with highest score
- for classes with high score manually inspect the clusters assignments

Spray for semi-automatic image debiasing

approach for: compute a low dimensional embedding ϕ_i for x_i such that $\|\phi_i - \phi_j\|$ small when A_{ij} large.

- compute graph laplacian: $L = D - A$, $D_{ii} = \sum_j A_{ij}$
- symmetrize graph laplacian: $L_{sym} = D^{-1/2}(D - A)D^{-1/2} = I - D^{-1/2}AD^{-1/2}$
- compute the $q = 32$ eigenvectors of L_{sym} for the 32 smallest eigenvalues of L_{sym}
- concat the 32 eigenvectors, results in a matrix $\phi \in \mathbb{R}^{n \times 32}$, choose $\phi_i = \phi[i, :] \in \mathbb{R}^{32}$
- what is that good for ??

Spray for semi-automatic image debiasing

understand the meaning of an eigenvector v for a small eigenvalue α of L_{sym} and L

1. $v^\top Lv = \alpha \|v\|_2^2 = \alpha$ by definition of eigenvalues and the fact that eigenvectors have unit length $\|v\|_2^2 = 1$
2. small eigenvalue means that $v^\top Lv$ is small
3. have theorem for graph laplacians:

$$v^\top Lv = \sum_{ij} A_{ij}(v_i - v_j)^2 = \text{ something small} \tag{10}$$

4. $v^\top Lv$ is small means that if samples i and j are close according to high value of A_{ij}, then $(v_i - v_j)^2$ are small values for most pairs (i, j) and therefore v_i tends to be close to v_j.
   $\Rightarrow$ if samples i and j are close according to $A_{ij} > 0$, then v_i tends to be close to v_j for most pairs (i, j) with $A_{ij} > 0$

- *v is an 1-dimensional embedding which attempts to obtain similar values v_i, v_j for samples (i, j) which are close according to A_{ij}*

Spray for semi-automatic image debiasing

understand the meaning of an eigenvector v for a small eigenvalue α of L_{sym} and L

- eigenvector v for a small eigenvalue α means that if samples i and j are close according to high value of $A_{ij} > 0$, then v_i tends to be close to v_j for most pairs (i, j) with $A_{ij} > 0$
- *v is an 1-dimensional embedding which attempts to obtain similar values v_i, v_j for samples (i, j) which are close according to A_{ij}*
- now take 32 v with the smallest eigenvalues and stack them. Result is: $\phi \in \mathbb{R}^{n \times q}$ where n is the number of samples.
- Taking the slice $\phi[i, :]$ provides for the i-th sample an 32-dimensional embedding. This has one property: if samples i and j are close according to $A_{ij} > 0$, then likely $\phi[i, :]$ and $\phi[j, :]$ are close to each other and have low euclidean distance to each other.
- $\phi[i, :]$ is a 32-dimensional embedding which tries to preserve the similarities according to A_{ij}.
  meaning: A_{ij} large, then $\phi[i, :]$ and $\phi[j, :]$ have likely a low euclidean distance

Spray for semi-automatic image debiasing

approach:

- cluster $\phi_i, x_i \in$same class, into $k = 2, \ldots, 30$ clusters. Get cluster assignments for each sample.
- compute a separability score τ for this class from the features $\phi_i \in \mathbb{R}^{32}$. Idea: high τ means there exists a cluster which is very isolated according to the features ϕ_i.
- how ? Label sample with its cluster index. Run Fisher discriminant analysis (FDA) for $k = 2, \ldots, 30$ cluster-defined synthetic classes. FDA tries to find directions, such then when data is projected onto them, then the within class variance is minimized, and the between class variance is maximized.

$$(S_b) = \sum_{l=1}^{k} (\mu_l - \mu_k)(\mu_l - \mu_k)^\top \in \mathbb{R}^{32 \times 32}$$

$$(S_w) = \sum_{l=1}^{k} \sum_{x_i \in C_l} (x_i - \mu_l)(x_i - \mu_l)^\top \in \mathbb{R}^{32 \times 32} \tag{11}$$

- Use the FDA criterion as value (its projection onto the eigenvector for the largest eigenvalue). Average this for $k = 2, ..., 30$ – This will be τ.
- Rank classes according to τ. Inspect clustering solutions for highest ranked ones.

Spray for semi-automatic image debiasing

approach:

- rank classes according to τ. Inspect them.
- Figure 6 in `https://arxiv.org/pdf/1912.11425.pdf` shows some clusters within top-τ-ranked classes:

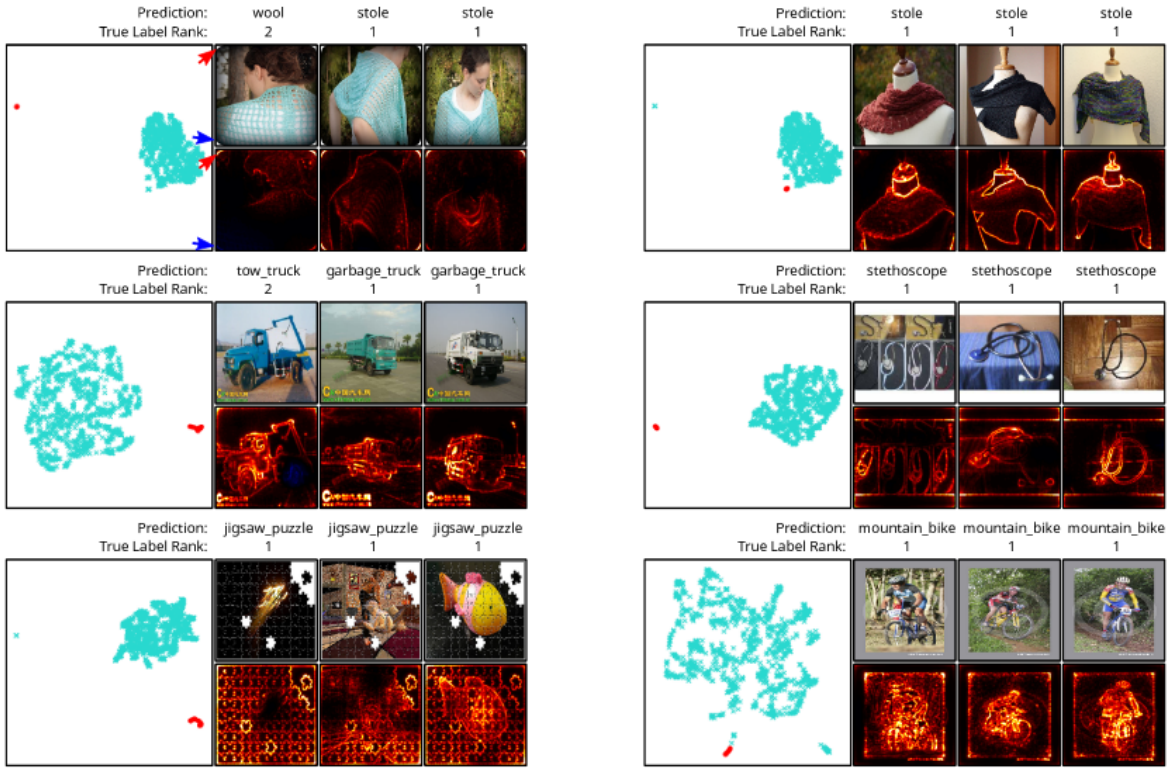
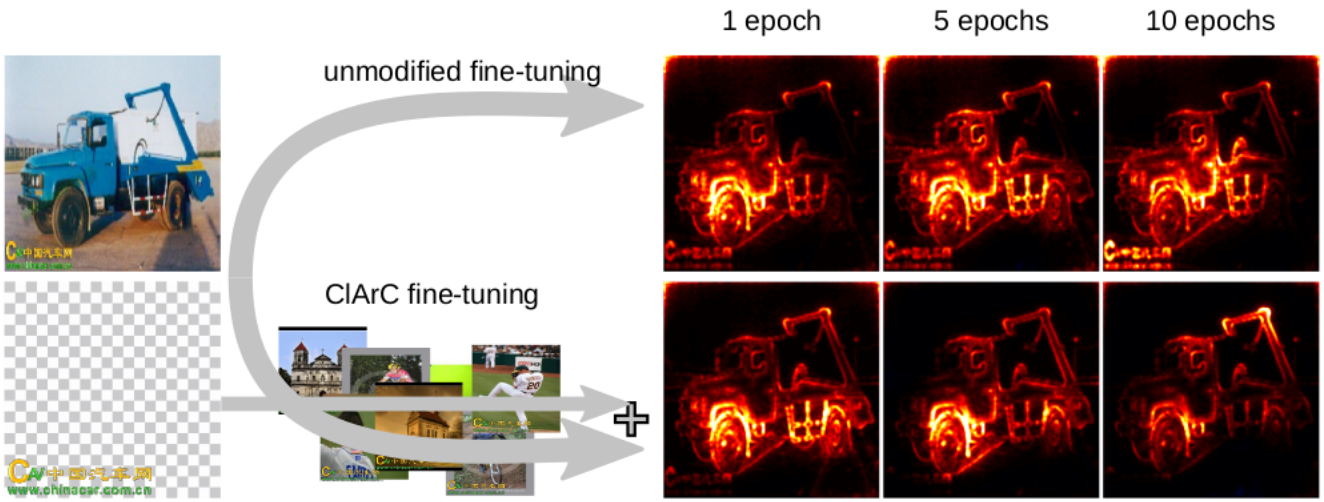


Figure 6. UMAP with samples and heatmaps of significant clusters for classes classes "stole" (top), "garbage truck" and "stethoscope" (mid) and "jigsaw puzzle" and "mountain bike" (bottom). All significant clusters are highly separated from the rest of the samples. For each class, some images and their respective attributions from the identified cluster are shown.

Spray for semi-automatic image debiasing

approach:

- bias removal: can remove the artifact by finetuning of the net on images, where all classes have the artefact added!



Isolate artefact, add to *other/all* classes, re-train model.

from Sebastian Lapuschkins talk XXAI workshop ICML2020

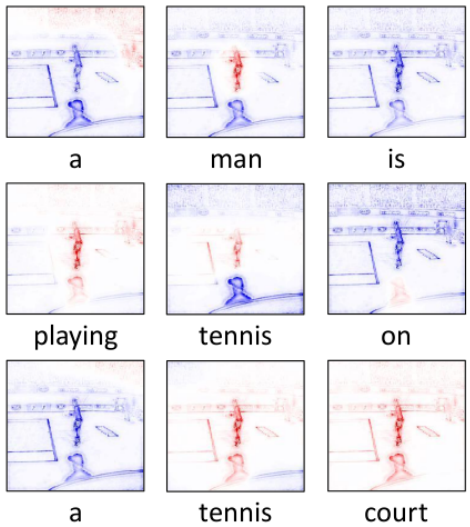
The value of explanations (not just LRP...)

Case D: Sensing structure-induced habits – image captioning case: LRP detects words hallucinated from sentence structures/grammar/oversampling

See slides from Sun Jiamei.

Sensing structure-induced habits – image captioning case

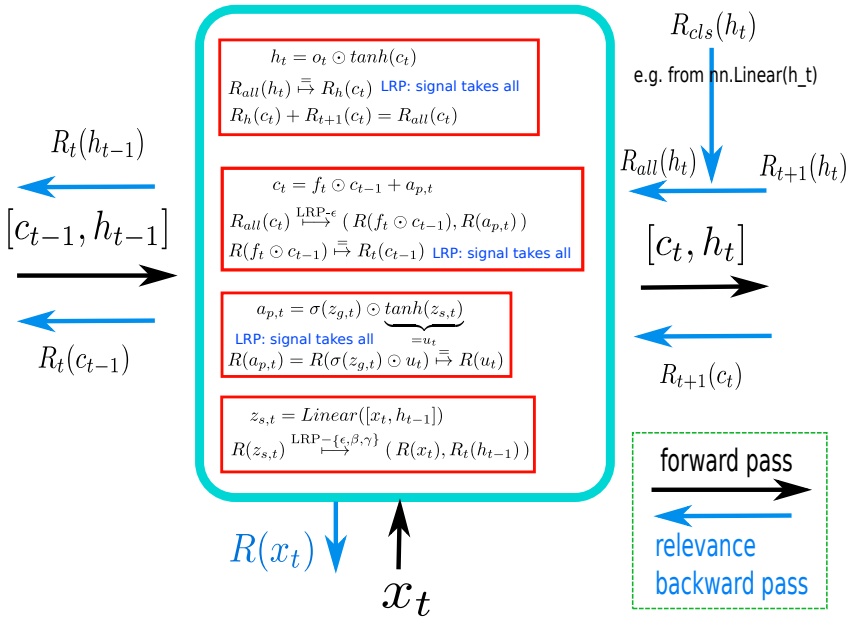
- Words are generated often by recurrent neural networks:
  $\text{word}_{n+1} = f(\text{Image}, \text{word}_1, \text{word}_2, \ldots, \text{word}_n)$
- LRP can be applied to RNNs such as LSTM.



2.Man: a
3.Is: a man
4.Playing: a man is
5.Tennis: a man is playing
6.On: a man is playing tennis
7.A: a man is playing tennis on
8.Tennis: a man is playing tennis on a
9.Court: a man is playing tennis on a tennis

0.75,1	0.5,0.75	0.25,0.5	0,0.25
-0.25,0	-0.5,-0.25	-0.75,-0.5	-1,-0.75

Sensing structure-induced habits – image captioning case



- Words are generated often by recurrent neural networks:
  $\text{word}_{n+1} = f(\text{Image}, \text{word}_1, \text{word}_2, \ldots, \text{word}_n)$
- LRP can be applied to RNNs such as LSTM.
- Three principles for LRP for RNNs:

- signal takes all in terms like $w = \sigma(z_{g,t}) \odot \tanh(z_{s,t})$ do not distribute relevance on gates $z_{g,t}$. Only onto signal terms $z_{s,t}$:

$$R(w) \mapsto (R(z_{g,t}), R(z_{s,t})) = (0, R(z_{s,t})) \tag{12}$$

- $+$: use LRP-ϵ
- Linear operations: use LRP-ϵ, β, γ up to evaluation results.

Sensing structure-induced habits – image captioning case

LRP for Attention-weighted features?

$$f = \sum_i w_i(v) v_i \tag{13}$$

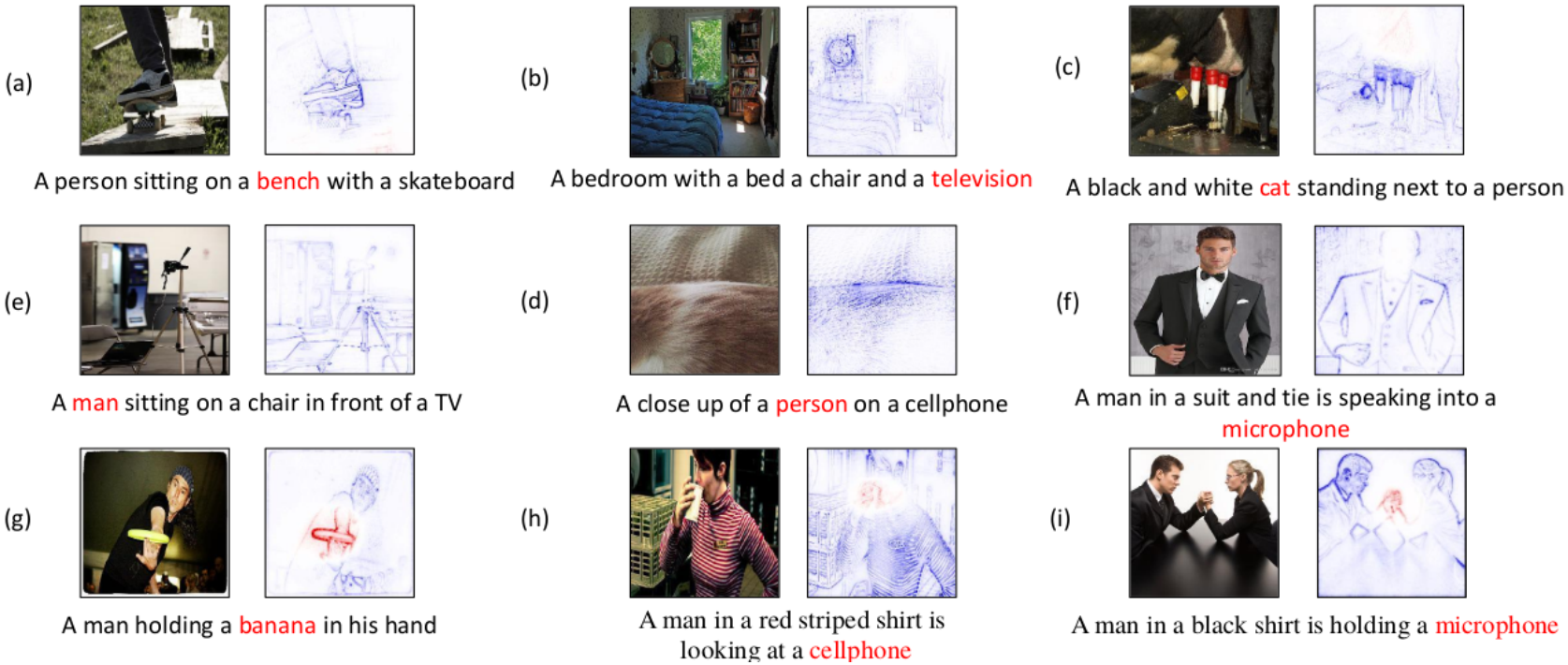
apply signal takes all idea:

- do not propagate relevance through weights to v
- propagate relevance only to v_i in above sum:

$$R(f) \overset{LRP-\epsilon}{\mapsto} \{R(v_i)\} \tag{14}$$

Combine LSTM-explanation idea and this idea – have explanations for image captioning

Sensing structure-induced habits – image captioning case



You can quantify how good is the detection using various methods (see slides).

The value of explanations (not just LRP...)

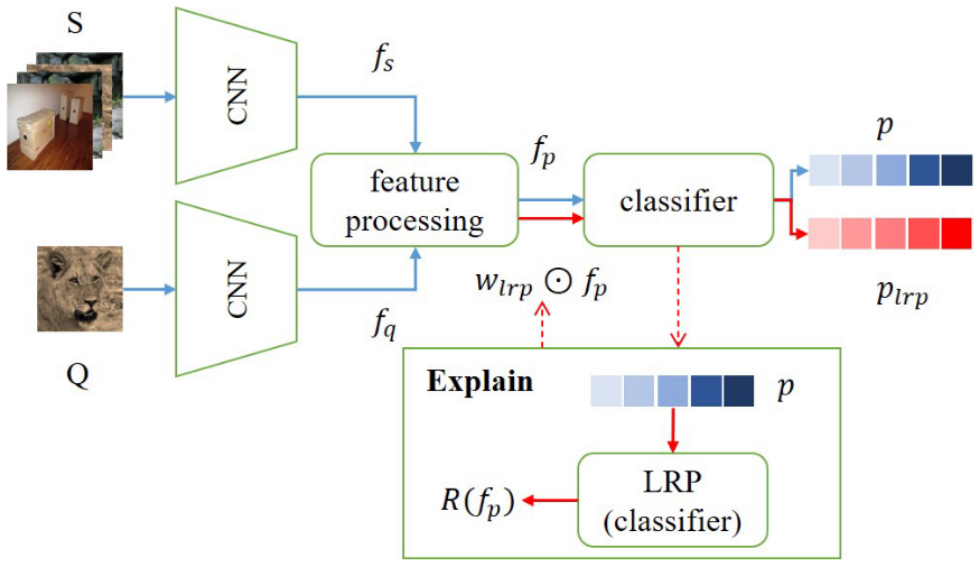
Case E: improving model performance in small-sample size tasks:

Explanation-Guided Training for Cross-Domain Few-Shot Classification

J Sun, S Lapuschkin, W Samek, Y Zhao, NM Cheung, A Binder
arXiv preprint arXiv:2007.08790
`https://arxiv.org/abs/2007.08790`

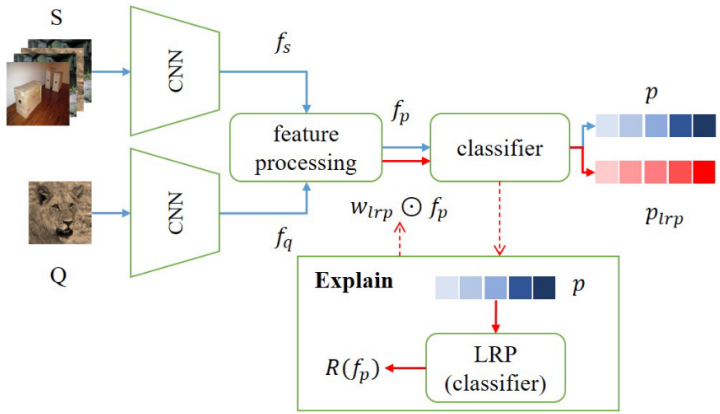
Explanation-Guided Training for Cross-Domain Few-Shot Classification

- improve prediction performance for few-shot classification in cross-domain tasks. **Steps:**
  - compute prediction
  - compute explanation scores for selected feature maps
  - re-weight selected feature maps
  - train: optimize sum of two losses: original features and reweighted features



- observation: consistent improvement (3 models, several datasets)
- combined with feature transform (HY Tseng, HY Lee, JB Huang, MH Yang, Cross-domain few-shot classification via learned feature-wise transformation, ICLR 2020), it improves synergistically

Explanation-Guided Training for Cross-Domain Few-Shot Classification



observation:

consistent improvement

(3 models, several datasets)

TABLE II: Evaluation of explanation-guided training on cross-domain datasets using RN and CAN. We report the average accuracy of over 2000 episodes with 95% confidence intervals. The models are trained on the miniImagenet training set and tested on the test set of various domains. **LRP-** means explanation-guided training using LRP. **T** indicates transductive inference.

miniImagenet	1-shot	1-shot-T	5-shot	5-shot-T
RN	58.31±0.47%	61.52±0.58%	72.72±0.37%	73.64±0.40%
LRP-RN	**60.06±0.47%**	**62.65±0.56%**	**73.63±0.37%**	**74.67±0.39%**
CAN	**64.66±0.48%**	67.74±0.54%	79.61±0.33%	80.34±0.35%
LRP-CAN	64.65±0.46%	**69.10±0.53%**	**80.89±0.32%**	**82.56±0.33%**
mini-CUB	1-shot	1-shot-T	5-shot	5-shot-T
RN	41.98±0.41%	42.52±0.48%	58.75±0.36%	59.10±0.42%
LRP-RN	**42.44±0.41%**	**42.88±0.48%**	**59.30±0.40%**	**59.22±0.42%**
CAN	44.91±0.41%	46.63±0.50%	63.09±0.39%	62.09±0.43%
LRP-CAN	**46.23±0.42%**	**48.35±0.52%**	**66.58±0.39%**	**66.57±0.43%**
mini-Cars	1-shot	1-shot-T	5-shot	5-shot-T
RN	29.32±0.34%	28.56±0.37%	38.91±0.38%	37.45±0.40%
LRP-RN	**29.65±0.33%**	**29.61±0.37%**	**39.19±0.38%**	**38.31±0.39%**
CAN	31.44±0.35%	30.06±0.42%	41.46±0.37%	40.17±0.40%
LRP-CAN	**32.66±0.46%**	**32.35±0.42%**	**43.86±0.38%**	**42.57±0.42%**
mini-Places	1-shot	1-shot-T	5-shot	5-shot-T
RN	**50.87±0.48%**	**53.63±0.58%**	66.47±0.41%	67.43±0.43%
LRP-RN	50.59±0.46%	53.07±0.57%	**66.90±0.40%**	**68.25±0.43%**
CAN	56.90±0.49%	60.70±0.58%	72.94±0.38%	74.44±0.41%
LRP-CAN	**56.96±0.48%**	**61.60±0.58%**	**74.91±0.37%**	**76.90±0.39%**
mini-Plantae	1-shot	1-shot-T	5-shot	5-shot-T
RN	33.53±0.36%	33.69±0.42%	47.40±0.36%	46.51±0.40%
LRP-RN	**34.80±0.37%**	**34.54±0.42%**	**48.09±0.35%**	**47.67±0.39%**
CAN	36.57±0.37%	36.69±0.42%	50.45±0.36%	48.67±0.40%
LRP-CAN	**38.23±0.45%**	**38.48±0.43%**	**53.25±0.36%**	**51.63±0.41%**

TABLE III: Evaluation of explanation-guided training on cross-domain datasets using GNN. We report the average accuracy of over 2000 episodes with 95% confidence intervals. The models are trained on the miniImagenet training set and tested on the test set of various domains. **LRP-** means explanation-guided training using LRP.

5-way 1-shot	miniImagenet	Cars	Places	CUB	Plantae
GNN	64.47±0.55%	30.97±0.37%	54.64±0.56%	46.76±0.50%	37.39±0.43%
LRP-GNN	**65.03±0.54%**	**32.78±0.39%**	**54.83±0.56%**	**48.29±0.51%**	**37.49±0.43%**
5-way 5-shot	miniImagenet	Cars	Places	CUB	Plantae
GNN	80.74±0.41%	42.59±0.42%	72.14±0.45%	63.91±0.47%	**54.52±0.44%**
LRP-GNN	**82.03±0.40%**	**46.20±0.46%**	**74.45±0.47%**	**64.44±0.48%**	54.46±0.46%

References

Opinion Paper

S Lapuschkin, S Wäldchen, A Binder, G Montavon, W Samek, KR Müller. Unmasking Clever Hans Predictors and Assessing What Machines Really Learn. *Nature Communications*, 10:1096, 2019.

Tutorial / Overview Papers

G Montavon, W Samek, KR Müller. Methods for Interpreting and Understanding Deep Neural Networks. *Digital Signal Processing, 73:1-15*, 2018.

W Samek, T Wiegand, and KR Müller, Explainable Artificial Intelligence: Understanding, Visualizing and Interpreting Deep Learning Models, ITU Journal: ICT Discoveries - Special Issue 1 - The Impact of Artificial Intelligence (AI) on Communication Networks and Services, 1(1):39-48, 2018.

Methods Papers

S Bach, A Binder, G Montavon, F Klauschen, KR Müller, W Samek. On Pixel-wise Explanations for Non-Linear Classifier Decisions by Layer-wise Relevance Propagation. *PLOS ONE*, 10(7):e0130140, 2015.

G Montavon, S Bach, A Binder, W Samek, KR Müller. Explaining NonLinear Classification Decisions with Deep Taylor Decomposition. *Pattern Recognition*, 65:211–222, 2017

L Arras, G Montavon, K-R Müller, W Samek. Explaining Recurrent Neural Network Predictions in Sentiment Analysis. *EMNLP'17 Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA)*, 159-168, 2017.

A Binder, G Montavon, S Lapuschkin, KR Müller, W Samek. Layer-wise Relevance Propagation for Neural Networks with Local Renormalization Layers. *Artificial Neural Networks and Machine Learning – ICANN 2016*, Part II, Lecture Notes in Computer Science, Springer-Verlag, 9887:63-71, 2016.

J Kauffmann, M Esders, G Montavon, W Samek, KR Müller. From Clustering to Cluster Explanations via Neural Networks. arXiv:1906.07633, 2019.

References

Application to Text

L Arras, F Horn, G Montavon, KR Müller, W Samek. Explaining Predictions of Non-Linear Classifiers in NLP. *Workshop on Representation Learning for NLP*, Association for Computational Linguistics, 1-7, 2016.

L Arras, F Horn, G Montavon, KR Müller, W Samek. "What is Relevant in a Text Document?": An Interpretable Machine Learning Approach. *PLOS ONE, 12(8):e0181142*, 2017.

L Arras, G Montavon, K-R Müller, W Samek. Explaining Recurrent Neural Network Predictions in Sentiment Analysis. *EMNLP'17 Workshop on Computational Approaches to Subjectivity, Sentiment & Social Media Analysis (WASSA)*, 159-168, 2017.

Application to Images & Faces

S Lapuschkin, A Binder, G Montavon, KR Müller, Wojciech Samek. Analyzing Classifiers: Fisher Vectors and Deep Neural Networks. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2912-20, 2016.

S Bach, A Binder, KR Müller, W Samek. Controlling Explanatory Heatmap Resolution and Semantics via Decomposition Depth. *IEEE International Conference on Image Processing (ICIP)*, 2271-75, 2016.

F Arbabzadeh, G Montavon, KR Müller, W Samek. Identifying Individual Facial Expressions by Deconstructing a Neural Network. *Pattern Recognition - 38th German Conference, GCPR 2016*, Lecture Notes in Computer Science, 9796:344-54, Springer International Publishing, 2016.

S Lapuschkin, A Binder, KR Müller, W Samek. Understanding and Comparing Deep Neural Networks for Age and Gender Classification. *IIEEE International Conference on Computer Vision Workshops (ICCVW)*, 1629-38, 2017.

C Seibold, W Samek, A Hilsmann, P Eisert. Accurate and Robust Neural Networks for Security Related Applications Exampled by Face Morphing Attacks. arXiv:1806.04265, 2018.

References

Application to Video

C Anders, G Montavon, W Samek, KR Müller. Understanding Patch-Based Learning by Explaining Predictions. *arXiv:1806.06926*, 2018.

V Srinivasan, S Lapuschkin, C Hellge, KR Müller, W Samek. Interpretable Human Action Recognition in Compressed Domain. *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1692-96, 2017.

Application to Speech

S Becker, M Ackermann, S Lapuschkin, KR Müller, W Samek. Interpreting and Explaining Deep Neural Networks for Classification of Audio Signals. *arXiv:1807.03418*, 2018.

Application to the Sciences

F Horst, S Lapuschkin, W Samek, KR Müller, WI Schöllhorn. Explaining the Unique Nature of Individual Gait Patterns with Deep Learning, *Scientific Reports*, 9:2391, 2019.

I Sturm, S Lapuschkin, W Samek, KR Müller. Interpretable Deep Neural Networks for Single-Trial EEG Classification. *Journal of Neuroscience Methods*, 274:141–145, 2016.

A Thomas, H Heekeren, KR Müller, W Samek. Interpretable LSTMs For Whole-Brain Neuroimaging Analyses. *arXiv:1810.09945*, 2018.

A Binder, M Bockmayr, M Hägele and others. Towards computational fluorescence microscopy: Machine learning-based integrated prediction of morphological and molecular tumor profiles. *arXiv:1805.11178*, 2018

References

Evaluation Explanations

W Samek, A Binder, G Montavon, S Lapuschkin, KR Müller. Evaluating the visualization of what a Deep Neural Network has learned. *IEEE Transactions on Neural Networks and Learning Systems*, 28(11):2660-2673, 2017.

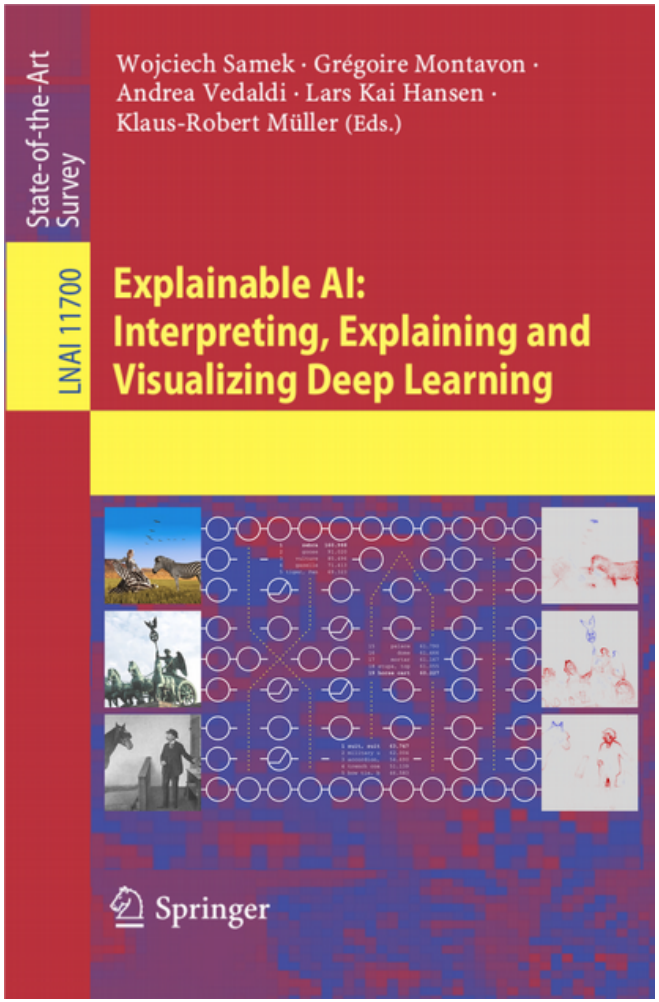
L Arras, A Osman, KR Müller, W Samek. Evaluating Recurrent Neural Network Explanations. *Proceedings of the ACL'19 Workshop on BlackboxNLP*, Association for Computational Linguistics, 113-126, 2019.

Software

M Alber, S Lapuschkin, P Seegerer, M Hägele, KT Schütt, G Montavon, W Samek, KR Müller, S Dähne, PJ Kindermans. iNNvestigate neural networks!. *Journal of Machine Learning Research*, 20:1-8, 2019.

S Lapuschkin, A Binder, G Montavon, KR Müller, W Samek. The Layer-wise Relevance Propagation Toolbox for Artificial Neural Networks. *Journal of Machine Learning Research*, 17(114):1-5, 2016.

New book out



Organization of the book:

- Part I Towards AI Transparency
- Part II Methods for Interpreting AI Systems
- Part III Explaining the Decisions of AI Systems
- Part IV Evaluating Interpretability and Explanations
- Part V Applications of Explainable AI
- 22 Chapters

Tutorial Paper
Montavon et al., "Methods for interpreting and understanding deep neural networks", Digital Signal Processing, 73:1-5, 2018

Keras Explanation Toolbox
`https://github.com/albermax/innvestigate`

link to the book:

`https://www.springer.com/gp/book/9783030289539`

papers, demos, ice cream at: `www.explain-ai.org`

Questions?!